Linear and Interface Circuit Applications

Volume 3

Data Acquisition, Peripheral Drivers and Hall-Effect Devices

D. E. Pippenger and E. J. Tobaben
Linear Applications

Contributors
C. L. McCollum
Field Applications Engineering
Linear Product Engineering

European Edition
Revised and Edited by David P. E. Dale of the
European Linear Applications Staff

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instuments reserves the right to make changes at any time in order to improve design and to supply the best product possible.

Texas Instruments assumes no responsibility for infringement of patents or rights of others based on Texas Instruments applications assistance or product specifications, since TI does not possess full access to data concerning the use or applications of customer's products. TI also assumes no responsibility for customer product designs.

1st European Edition Vol 3

Contents

Section 12
Peripheral Drivers

Section 13
Hall-Effect Devices

Preface

This is the third volume in a three-volume series of Linear and Interface Circuit Applications books. To maintain overall continuity in the series, the section numbers in this book follow those in Volume 2. Volume 1 presented information on operational amplifiers, voltage regulators and timers. Volume 2 provides information on display drivers and data transmission line circuits. Volume 3 provides information on data acquisition, peripheral drivers and Hall-effect devices.

The purpose of this series of books is to present linear and interface circuit applications in a manner that will give the reader a basic understanding of the products and provide simple but practical examples for typical applications. Care has been taken to choose illustrations which are of interest, at least by analogy, to a wide class of readers. This material is written for not only the design engineer but also for engineering managers, engineering technicians, systems designers, and marketing or sales people with some technical background. The authors have attempted to avoid lengthy mathematical analyses so that the important points may be clearly emphasized and not obscured by distracting derivations. In cases where a rigorous derivation has been omitted, an attempt has been made to state the results precisely and to emphasize limitations that are practically significant.

To facilitate their use, the sections have been made basically independent. The primary goal of the books is to assist the user in selecting the proper device for a particular application. To accomplish this, key features of devices are presented along with discussions of device or system theory and requirements.

Potential uses of the devices are demonstrated in circuit applications. These applications are not intended to be a how-to for specific circuits but to be examples of how the device might be used to solve your specific design requirements. In each case, a data book or data sheet should be referred to for complete device chracteristics and operating limits. The circuit examples selected for this book have accrued from numerous customer inquiries and related laboratory simulations.

Section 11

Data Acquisition

INTRODUCTION

This data acquisition section, after a general introduction of basic ideas, will concentrate on single-chip data acquisition devices and give applications information on how they may be interfaced to popular microprocessors. This text will assume that sensors and transducers have already converted physical stimuli into analog electrical signals.

TYPICAL SYSTEMS

Data acquisition is the process of transforming analog electrical signals into digital information for: storage, display, processing, further transmission or control. A data acquisition system is an electronic system used to perform this task and comprises sensors, transducers, signal conditioning and sample and hold circuits, analog multiplexers, and analog to digital (A/D) converters. Recovery of a digital signal into analog form is sometimes required. The components, such as digital to analog converters (DAC) and filters, used to perform this function are also covered under the heading of data acquisition.

Data acquisition systems encompass a wide range of applications and implementations. Examples range from simple digital voltmeters, that display an input voltage measured as a numerical output, to large automatic process control systems that incorporate a data acquisition system and a mainframe computer. Data acquisition systems fulfil the need of computing devices of all sizes to retrieve information from the outside world. These

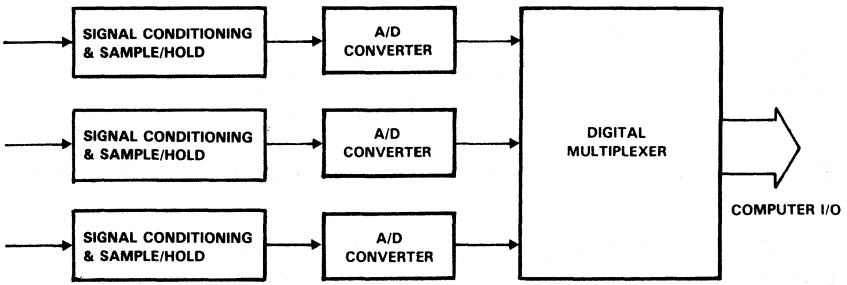


Fig. 11.1 Digital multiplexing of a multi-channel data acquisition system

can then be used to monitor or control processes of many types either locally or remotely.

Typical data acquistion system multi-input configurations are shown in Figs. 11.1 to 11.3. These may be used with any level of computing power, called a host processor in this text, either mainframe, mini, microprocessor or microcomputer, depending on the application.

Flexibility and versatility are features of the digitally multiplexed multi-channel system of Fig. 11.1 which has become increasingly feasible with the reducing cost of A/D devices. A/D converters with 3-state outputs and I/O port interfaces make this scheme easy to implement and allows the optimum component choice for each channel. If a serial interface to the digital multiplexer is used, low level analog signals can be digitised at source. This removes problems of interference and allows the possibility of including isolation by, for example, optocouplers. Larger systems can use data reduction, by manipulation of data before transmission, so that the work load of the communication channel is reduced, with only changes and new information being received by the host processor.

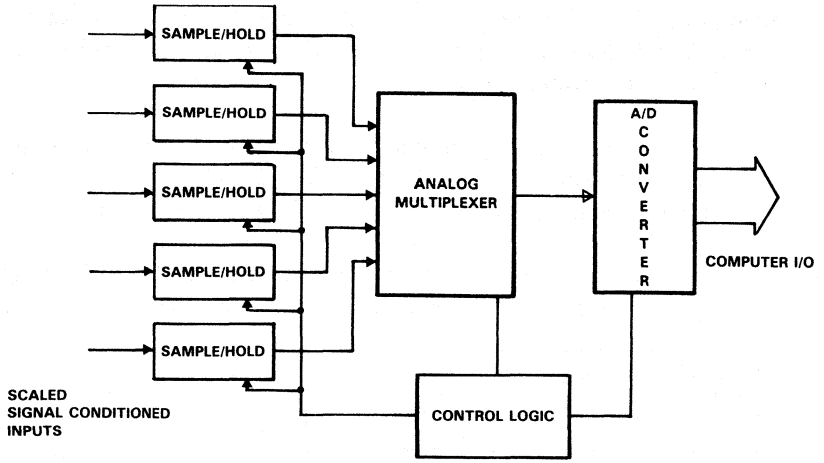


Fig. 11.2 Sample and hold outputs multiplexed to one A/D converter

Multiplexing the outputs of many sample and hold circuits, Fig. 11.2, produces a high speed, multi-channel, one-shot data capture capability. This configuration is useful in test applications where many items of data need to be collected at the same instant, but where there is time for a single

A/D converter to be sequenced through all the sample and hold's outputs. The hold and droop specifications of the sample and hold circuits need to be sufficient for the error performance required.

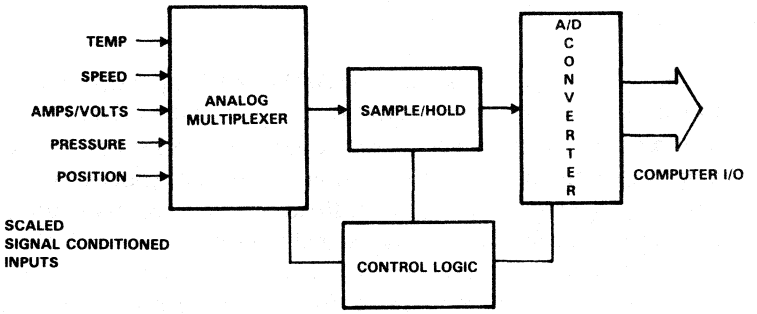


Fig. 11.3 Inputs multiplexed to one sample and hold and A/D converter

Multiplexing the analog signals to the input of one sample and hold circuit and A/D converter, Fig. 11.3, uses the minimum number of components to implement a multi-channel data acquisition system. The overall system is slower, but random input channel access allows the channels with the fastest changing input signal to be converted more frequently. This scheme is most often favoured for single chip data acquisition system implementations, especially those with a switched capacitor A/D converter which have an inherent sample and hold function.

Key parts of the Data Acquisition Systems of Figs. 11.1 to 11.3 include:

Signal Conditioning – This usually preceeds the multiplexer and sample and hold and is used to process the input signal ready for conversion. The conditioning might provide differential inputs, scaling, programmable gain, linear or logarithmic amplification, filtering, isolation, linearisation, etc.

Analog Multiplexer – This, by selecting between several analog input signals, allows the use of only one A/D converter.

Sample and Hold – This acquires and tracks the input signal, ready to sample at the instant required and then hold its output at the sampled value for the duration of the A/D conversion.

Analog-to-Digital Converter (A/D)–This converts an analog input signal into a digital output. A steady input signal, by comparison with a reference, is quantised into discrete levels which are then encoded into a digital form, usually binary. The analog resolution depends on the number of bits in the output code and the size of the reference voltage.

Computer I/O–This describes the A/D Input/Output data and control function. It usually has a serial or parallel 3-state data output and uses a Chip Select, a Write/Read control and an End of Conversion output to implement A/D operation under host processor control. Write is used to start conversion, Read is used to receive output data and these controls may be either signal lines or bits set in A/D control registers. End of Conversion output, when available, can be used in either interrupt or polling schemes with the host processor.

Typical Data Acquisition Applications

A data acquisition system (DAS) is a self contained subsytem which can communicate with many other systems. On some occasions the DAS is the major part of the system, for example when the system function is data acquisition, as with data logging. The DAS may include a feedback control path, Fig. 11.4, for local process control. On other occasions the DAS is a relatively minor part of another major system and is ancillary to the main function e.g. monitoring a telecoms' line voltage.

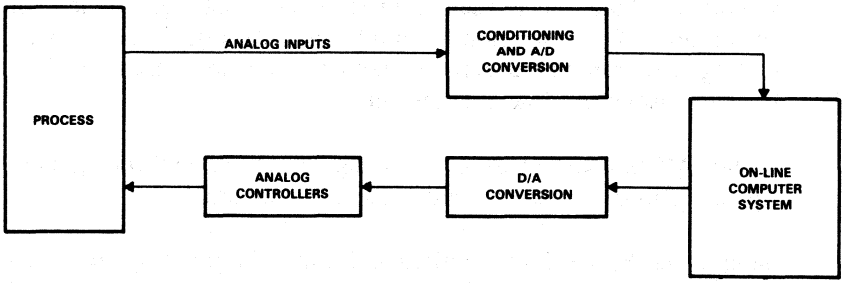


Fig. 11.4 A Data acquisition system with a feedback control path

Examples of the broad DAS application areas follow:

Data Logging:	Meteorology Environment-Structural-Safety Water Supply Medical Aviation Automatic testing
Process Control & Control Systems:	Refineries Food and Drink Processing Manufacturing: Steel to Semiconductors Energy Supply: Electricity, Gas, Oil, Nuclear Automotive: Engine Management, Suspension, Anti-lock brakes, Anti-slip Environmental: Heating, Air Conditioning Marine Mining
Communications:	Telecommunications Video Graphics Digital Television Railways

Data Logging

Data logging involves measuring the analog inputs and translating the results into digital signals which are stored for further processing or analysis.

Process Control and Control Systems

This is differentiated from data logging by the real time control that is exercised on the basis of the gathered data. The feedback may be either included locally as part of a data acquisition subsytem, which may be the entire system, or globally from a larger central host processor or both.

Communications

Communications in a very broad sense can be included as a category of data acquisition, electrical signals are digitised, stored with an emphasis on transmission.

BASIC CONCEPTS

Signals and Data Acquisition Sampling Systems

Signal characteristics are an important consideration when choosing data acquisition components and sampling rates. Account should be taken of whether the signal being digitised is either dc, a one-shot transient event or a continuous time varying wafeform (i.e. composed of sinusoidal components). In conjunction with these, such parameters as maximum frequency, signal to noise ratio and maximum amplitude also need consideration.

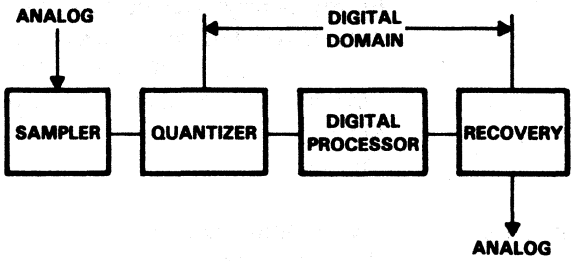


Fig. 11.5 Data acquisition sampling system

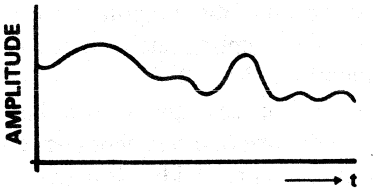
A block diagram is shown in Fig. 11.5 of a complete data acquisition process from analog-to-digital conversion through to recovery including digital-to-analog conversion. The signal sampling process and other functions are further discussed below.

Sampling Rate and Aliasing

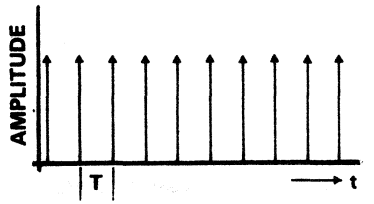
Continuous Time Signals and Nyquist's Criterion

Sampling is the first step in representing a continuous time signal in a discrete time, sampled data or digital signal, form. Sampling is achieved by instantaneously measuring a continuous waveform at regular discrete intervals called the sampling period, the inverse of which is the sampling rate. Ideally sampling should be performed by impulses, infinitesimally narrow pulses of unit area, on bandlimited continuous time input signals. Under these conditions, the Nyquist criterion for sampling rate states that all the information in a signal can be extracted if it is sampled at a frequency, f_s, greater than twice the highest input frequency. This is called the Nyquist rate.

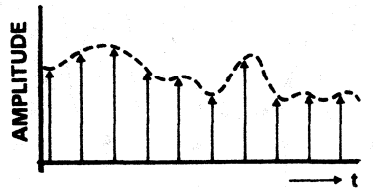
Ideal Sampled Frequency Spectrum



(a) Band-limited continuous-time signal



(b) Sampling impulses spaced at sampling period T



(c) Amplitude modulated sampling impulses

Fig. 11.6 Sampling in the time domain

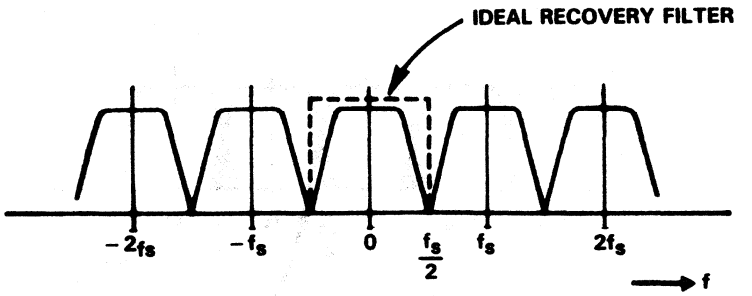


Fig. 11.7 Frequency spectrum of time sampled continuous signals at the Nyquist rate

Sampling in the time domain produces an impulse train that is amplitude modulated by the bandlimited input signal, Fig. 11.6. In the frequency domain the frequency spectrum of the sampling process consists of discrete frequencies at multiples of the sampling rate with double sidebands produced by the amplitude modulation, Fig. 11.7.

This describes the ideal process and shows that sampling adds new frequency components to the original frequency spectrum. Ideal recovery of the original signal would require an ideal low pass filter, of half Nyquist rate bandwith, to remove all but the original signal spectrum, Fig. 11.7.

Aliasing

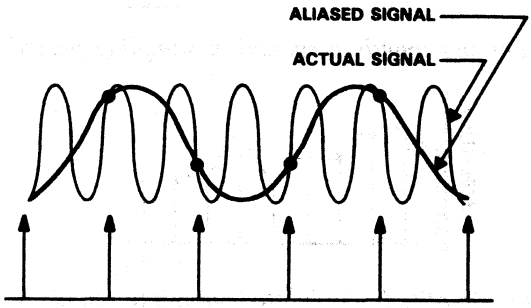


Fig. 11.8 Aliasing in the time domain due to sampling below the Nyquist rate

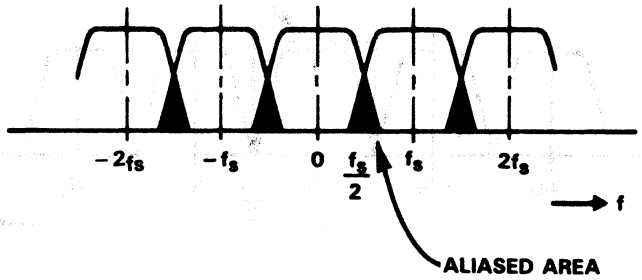


Fig. 11.9 Aliasing in the frequency domain

If sampling is lower than the Nyquist rate then the separation between the double sidebands of Fig. 11.7 disappear and they overlap. Higher frequency components fold back into the frequency range of the input signal bandwith and take on the identity of lower frequencies. This effect is called Aliasing, Figs. 11.8 and 11.9. This form of distortion is usually impossible to remove once it has occurred.

Practical Constraints

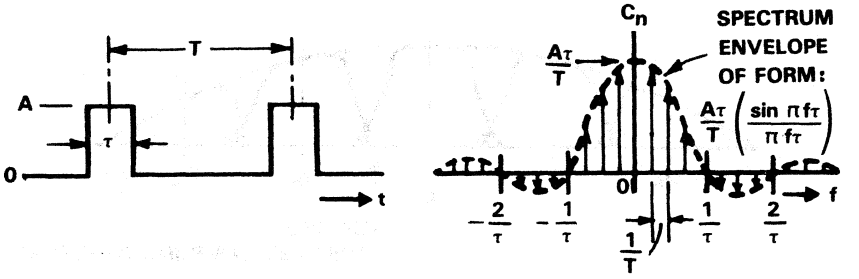


Fig. 11.10 Rectangular pulse train with frequency spectrum

Practically, the sampling waveform will not be an impulse but a pulse of finite width. The wider the sampling pulses become the greater the effect on the sampled signals spectrum. The sampling pulses, Fig. 11.10, have a sinx/x frequency spectrum which causes an extension in effective input signal bandwidth and high frequency attenuation after sampling, Fig. 11.11. This can aggravate problems with aliasing and is avoided by keeping sampling pulse widths small in comparison with the sampling period. From the expression for frequency domain amplitude response, of Fig. 11.10, it can be shown that sampling pulses τ of one-fifth the sampling period T give less than 0.1 dB attenuation at a frequency of 0.4 f_s. There are also practical constraints which limit the roll-off rates of real filters that band-limit the input frequency.

The constraints described above will increase the input bandwidth and modify the sampled input frequency spectrum. To account for this the sampling rate should be raised from the minimum given by the Nyquist criterion. Increasing the sample rates from the Nyquist minimum, to 10 samples per input cycle at maximum frequency, significantly increases the accuracy of a recovered signal. Without the use of reconstruction filters, the accuracy improves from approximately 30% to over 90%.

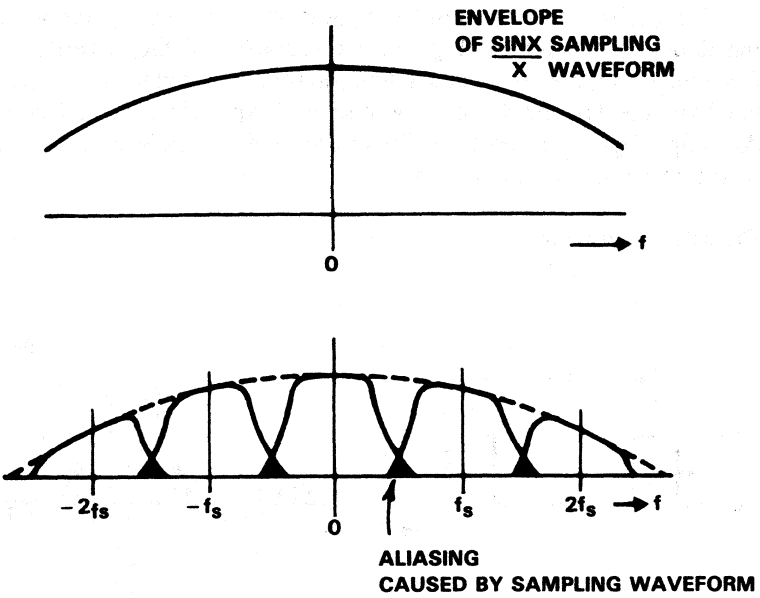


Fig. 11.11 Finite Sampling Pulse Width Effect on Sampled Frequency Spectrum of Fig. 11.7

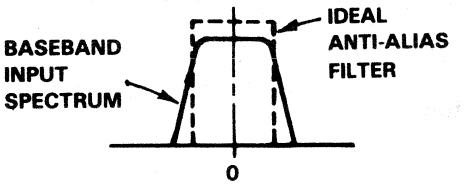
Sampling Transient Signals

Transient or one-shot input signals demand a different approach to sampling than continuous signals. Sample rate has to be determined from the slew rate of the transient and resolution requirements. Pre-knowledge of the signal characteristics will ease this decision. For example, information that a transient has passed through a bandpass filter will define its maximum slew rate. Specified interpolation requirements can also be used to determine sampling rates.

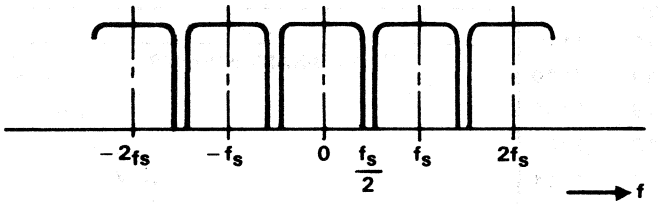
Anti-Aliasing Filter

Noise and out-of-band input signals can be prevented from folding over into the signal bandwidth of interest by employing some form of pre-sampling filter, most often called an anti-aliasing filter. An ideal filter response shape would be the "brickwall" filter shown in Fig. 11.12, which stops signal sidebands overlapping. Such filters are impossible to construct and complex filters that attempt the task introduce phase and amplitude

distortion of their own. The solution lies in a compromise between sample rate and anti-aliasing filter complexity.



(a) Baseband spectrum



(b) Sampled spectrum

Fig. 11.12 Removal of aliasing components with pre-sample or "anti-aliasing" filter

Acquisition

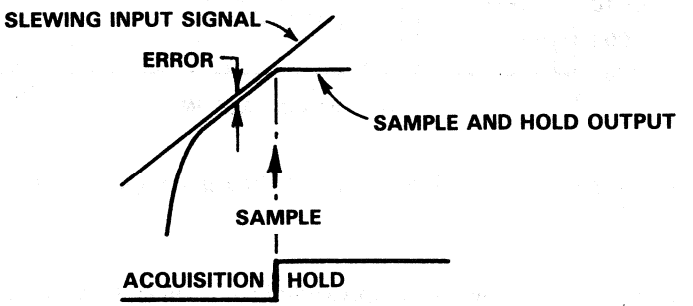


Fig. 11.13 Acquisition action of a sample and hold circuit

Acquisition is a term that applies to sample and hold circuits including those incorporated into A/D converters and recognises that before a signal

can be sampled it has to be acquired. These circuits, prior to the sampling instant, charge a capacitor to the input signal level and then track it within an acceptable error specification Fig. 11.13.

A/D Conversion

Quantisation

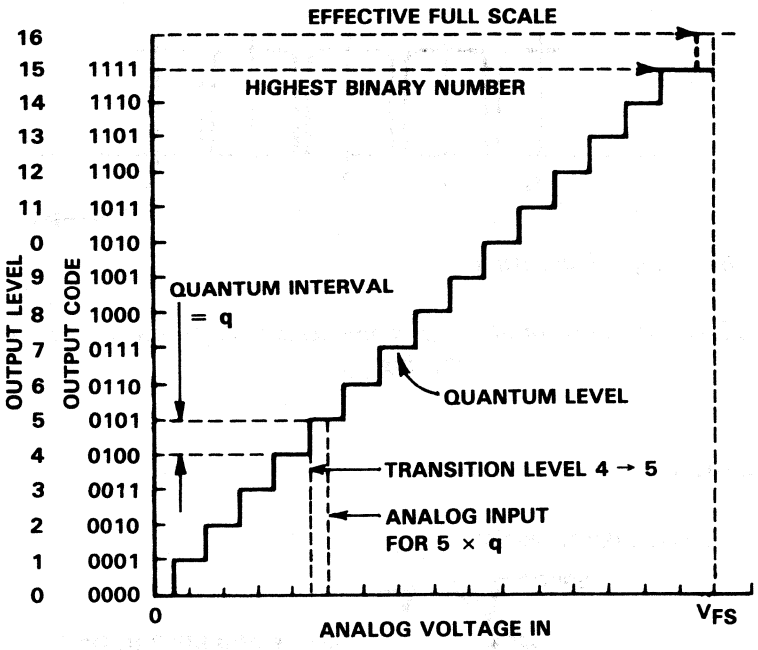


Fig. 11.14 Transfer Funtion for 4-Bit Binary Quantiser

When a continuous signal is represented by a binary sequence of some fixed length 'n', it limits the number of output signal levels that can be defined to 2^n. Each level is called a quantum and quantisation is the process of representing the input signal by one of these levels. Fig. 11.14 shows how each output quantum level corresponds to a range of input values, and the typical stair-case transfer characteristic of an A/D converter.

Table 11.1 Number of quantisation levels or resolution per binary 'n' bits

n	Levels	n	Levels
1	2	9	512
2	4	10	1,024
3	8	11	2,048
4	16	12	4,096
5	32	13	8,192
6	64	14	16,384
7	128	15	32,768
8	256	16	65,536

Table 11.1 shows the number of quantisation levels versus number of bits 'n'.

Subtracting a straight line transfer function, which represents an A/D with infinite resolution, from an ideal stair-case A/D transfer function of n-bit resolution, shows the theoretical minimum quantisation error of value ± 1/2 LSB (Least Significant Bit), or ± q/2 where q is one quantum interval. A 3-bit example is shown in Fig. 11.15.

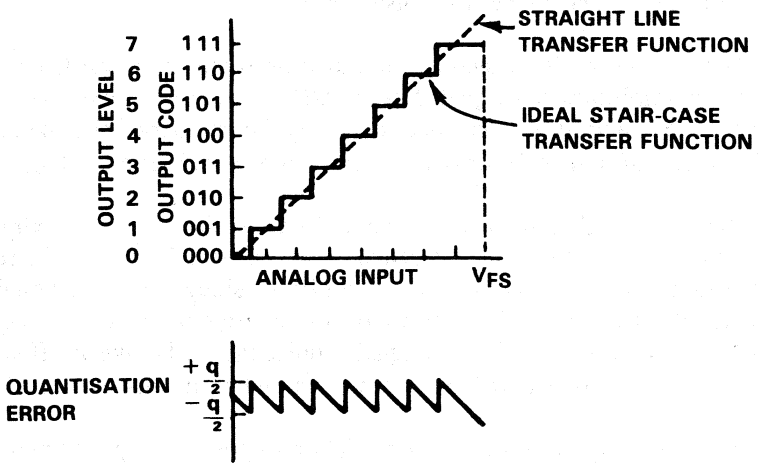


Fig. 11.15 A/D quantisation error, 3-bit example

Quantisation noise

Quantisation effectively adds noise to an A/D converter's output signal, in a similar way to many other signal conditioning functions such as amplification. This quantisation noise is usually considered as as uniformly random sequence that has a rms value = $q/\sqrt{12}$ where q is the value of the quantum level. The differential-nonlinearity of practical A/D converters can increase this noise level.

Encoding

To complete an A/D conversion process discrete quantised levels, which may be identified as a position on a multi-tap potential divider, have to be translated into some form of digital output code. Some A/D converter architectures internally produce a binary code as part of the quantisation process, successive approximation for example, however the output code may be required in offset binary, 2's complement, BCD or some other format that will need encoding.

Resolution

A/D converter resolution is the weight assigned to the least-significant bit (LSB) and is inherent in the output code represented by a number of binary bits 'n'. The weight then equals the full scale reference value divided by 2^n.

Dynamic Range and Signal to Noise Ratio

Dynamic range is defined as the range between the noise floor and maximum amplitude of a signal. For an A/D converter this extends from the effective quantisation noise, determined by the number of bits and error performance of a specific device, to the value of reference voltage.

The result of dividing a signal's amplitude, which may be maximum amplitude, by its noise voltage is called the signal to noise ratio. The ratio of root-mean-square (rms) maximum amplitude sinewave input signal to ideal quantisation rms noise, for an A/D converter of 'n' bits, is equal to: $S/N = 6{\cdot}02n + 1{\cdot}76$ dB. If a signal to noise ratio is known an effective value of bits 'n' for an A/D can be determined from this expression.

Dynamic range is a factor when deciding upon the required number of A/D converter bits for a particular application. Using many more bits than is required will result in digitising input noise. An optimum choice

occurs when the noise of the A/D converter quantisation equals that of the incoming signal.

Signal recovery

Digital-to-Analog converters are used to produce an analog output from a digital signal processing system. Recovering a continuous time signal from a sample data signal is achieved by a process of interpolation between sample amplitudes in the time domain, Fig. 11.16. This process is equivalent in the frequency domain to using an ideal filter to remove the extra unwanted signals introduced by sampling (see earlier Fig. 11.7).

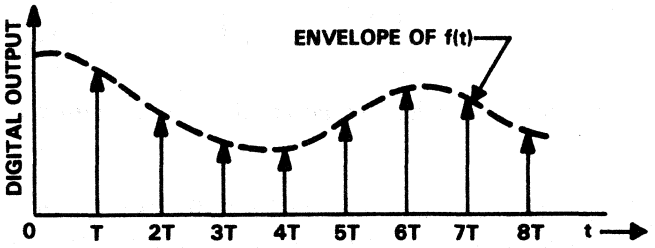


Fig. 11.16 Ideal amplitude modulated sample pulse train from DAC.

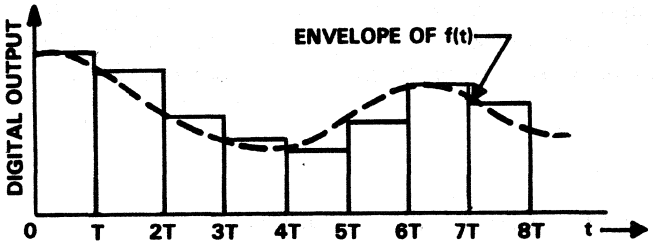


Fig. 11.17 DAC "zero order hold" recovery

Output impulses of very short duration, Fig. 11.16, do not contain significant power and so after interpolation a very low output amplitude would be obtained. To overcome this, the output of a DAC usually comprises sample-period duration pulses of an amplitude depending on input code. Fig. 11.17 shows the characteristic step interpolator output which has the form of the so-called 'zero-order hold'. Its sinx/x function frequency spectrum and phase characteristic are shown in Fig. 11.18.

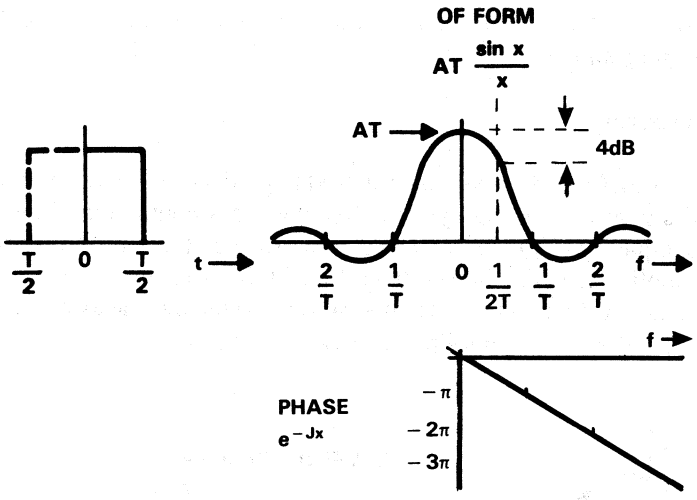


Fig. 11.18 Sinx/x frequency spectrum and phase characterisistics

The frequency spectrum of Fig. 11.18 performs a filtering action on the DAC output signal with the first zero in the frequency spectrum occuring at the sampling frequency. If sampling is at the Nyquist rate an amplitude reduction of 4 dB at the maximum input frequency will result. This can be compensated by filtering either in the digital or analog domain. The digital approach will more easily achieve high accuracy. If sampling is very much greater (100:1) than the Nyquist rate, it will allow the use of a simpler post-DAC reconstruction or interpolation filter.

DATA ACQUISITION TERMINOLOGY

Introduction

Terminology referring to data acquisition component parameters and performance, that commonly appear in data sheets and applications information, is described below. It should be noted that on many occasions, there are subtle differences in the meaning of terms when they describe parameters of both analog-to-digital (A/D) and digital-to-analog (DAC) converters. This arises because both have their performance measured and specified with respect to the analog domain.

A particular digital code input to a DAC produces a discrete analog output whereas a range of analog values input to an A/D determine a digital output code. For example: Quantisation, referring to an A/D, describes the input analog range for a particular output code but, for a DAC, describes the ideal difference in analog output between adjacent input codes. A DAC analog output, for a given code, has a discrete value modified only by error specifications.

Nominal Midstep and Step values

Error terms in general are expressed as deviations from an ideal, or theoretical value, usually in fractions of a least- significant-bit (LSB), but % of full-scale and ppm are sometimes used. For an A/D the nominal midstep value is the error free value corresponding to a particular code which lies on a line between zero and full-scale. Similarly for a DAC an error free discrete analog output for a particular code is called the nominal step value.

Absolute Accuracy

Absolute accuracy defines data acquisition component measurements that are made with voltage sources and measuring instruments traceable to internationally accepted standards.

Absolute Accuracy Error or Total Error

Absolute accuracy error, or total error, of a DAC for a given digital input code, is the difference between the actual step in analog output and the nominal step value.

A/D converter absolute accuracy error, or total error, is the maximum difference between actual input values and the nominal midstep value within any step to produce a given code. Absolute accuracy error includes contributions from offset error, gain error, linearity error and inherent quantisation error.

Relative Accuracy Error

Relative accuracy error is the deviation in analog value from its ideal value relative to the full-scale-range (FSR). This is expressed either as a fraction of an LSB, % FSR or ppm.

Quantisation Error

Quantisation Error is the inherent error of a converter due to the finite resolution defined by the length of an equivalent digital binary output code, Fig. 11.19.

Code Width

For an A/D converter, code width is the range of analog input values over which the same output code will occur. An ideal code width is 1 LSB, that is $FSR/2^n$ where n is the number of bits for a binary output code, actual code widths vary due to linearity errors.

Linearity Error (LE) or Integral Nonlinearity (INL)

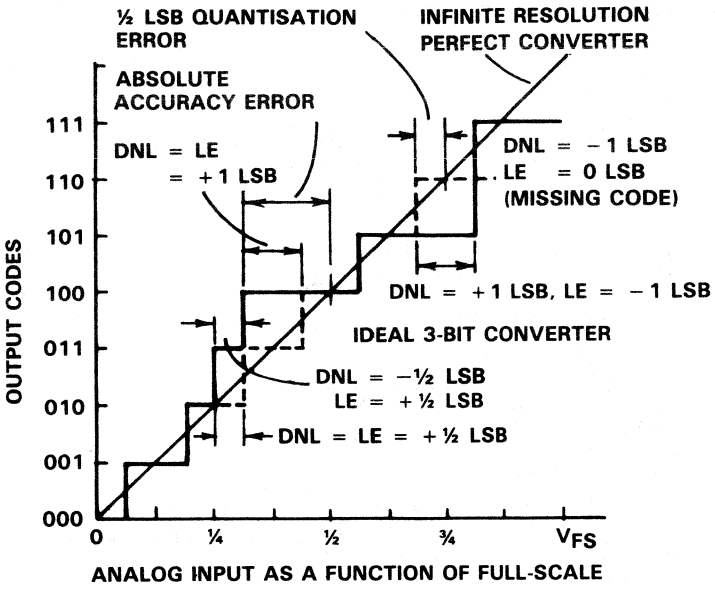
Linearity error or integral nonlinearity is the maximum difference in analog value at a transition between any two adjacent steps from its ideal value. The ideal value can be determined from either a best straight line fit, without adjustment of full-scale or offset error, or a line passing through the end points of an ideal transfer characteristic, which implies adjustment of full-scale and offset errors to zero, see Fig. 11.19.

Differential Nonlinearity (DNL)

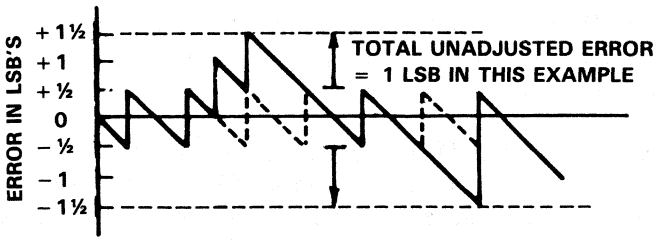
Differential nonlinearity for a DAC is the difference in analog output value for adjacent codes minus 1 LSB, this can have a positive or negative value. For a DAC, if DNL is more negative than −1 LSB a non-monotonic response will result where the analog output decreases for an increasing input code value.

Differential nonlinearity for an A/D is the difference between actual code width and an ideal code width of 1 LSB, see Fig. 11.19. Corresponding to the non-monotonic behaviour of a DAC with DNL more negative than −1 LSB, an A/D will exhibit missing codes in its output if DNL equals −1 LSB. A linearity error specification of ± ½ LSB infers a DNL of not greater than −1 LSB.

Some manufacturers quote A/D DNL as the actual difference in adjacent code midpoints minus 1 LSB, instead of actual code width minus 1 LSB. The missing code condition remains the same, at DNL equal to −1 LSB, but other values of DNL will apparently be different between the two specification methods.



(a) 3-bit A/D converter ideal and hypothetical error transfer characteristics



(b) Example error curve (actual−perfect transfer characteristic)

NOTE: EXAMPLES OF THE FOLLOWING ERRORS ARE SHOWN:
DIFFERENTIAL NONLINEARITY (DNL)
LINEARITY ERROR (LE) (INTEGRAL NONLINEARITY)
ABSOLUTE ACCURACY ERROR
QUANTISATION ERROR
TOTAL UNADJUSTED ERROR (TUE)
(WITHOUT ZERO OR FULL-SCALE ERRORS INCLUDED)

Fig. 11.19 Examples of nonlinearity errors

Zero-Scale Error or Offset Error

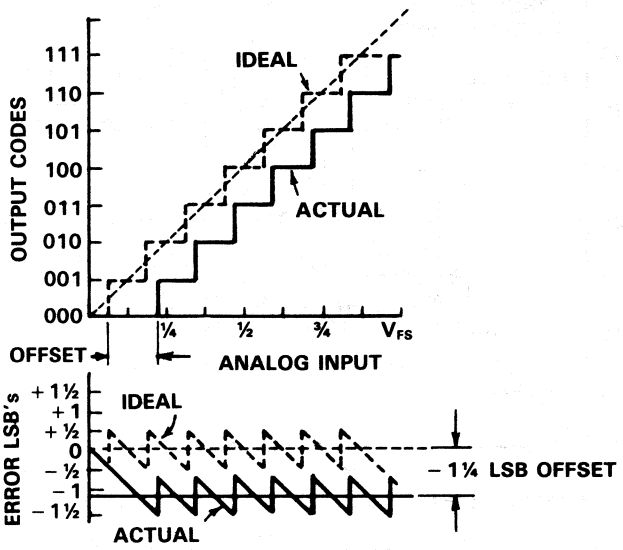


Fig. 11.20 3-bit A/D zero or offset error example

Zero-scale error is the difference between the actual and nominal step or midstep value at the specified zero scale point for both unipolar and bipolar devices. Devices are called unipolar when they allow an input voltage range from either $+V_{FS}$ or $-V_{FS}$ to zero, or bipolar when they allow an input voltage range of $-V_{FS}$ to $+V_{FS}$.

Offset error, for an A/D, is a deviation of equal value from ideal transition voltages for all codes, as shown in Fig. 11.20. Offset error is specified at the first LSB transition, as zero error, because other errors are not significant at this transition. For a DAC offset is a deviation of equal value to the nominal step value. The term zero-scale error is usually applied to converters without provision for external adjustment of offset error.

Full-Scale or Gain Error

Full-scale error is the difference between the actual and nomimal step or midstep values at full-scale. Gain error, applies when offset error has been adjusted to zero and, appears as a change in the slope of a converter

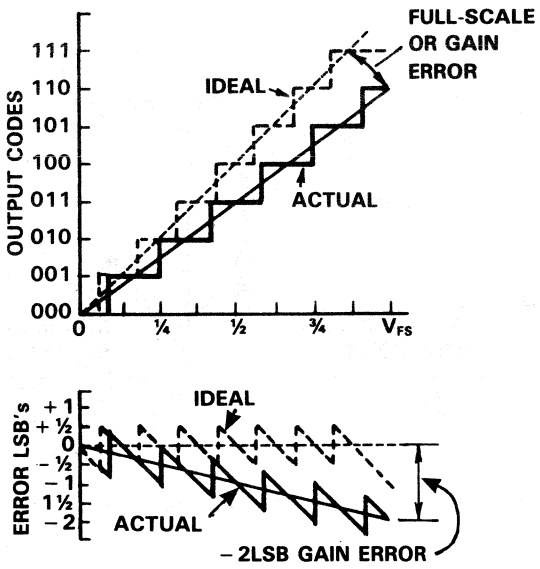


Fig. 11.21 3-bit A/D full-scale or gain error example

transfer function by effecting each code in equal ratio, see Fig. 11.21. The term full-scale error is usually applied to converters without provision for external adjustment of offset and gain error.

Adjusted Error

Adjusted error refers to linearity errors that converters, with the provision of exernal gain and offset adjustment, have after these errors are reduced to zero.

Total Unadjusted Error (TUE)

Total unadjusted error, applies to converters without the provision for external adjustment of errors and is the sum of offset, gain and linearity errors at any point over the full-scale range. Although the error curve of Fig. 11.19 (b) shows only linearity errors, it illustrates a total unadjusted error boundary, ranging from the quantisation error. A complete error curve would include the offset and gain errors of Fig. 11.20 and 11.21 respectively.

Monotonicity

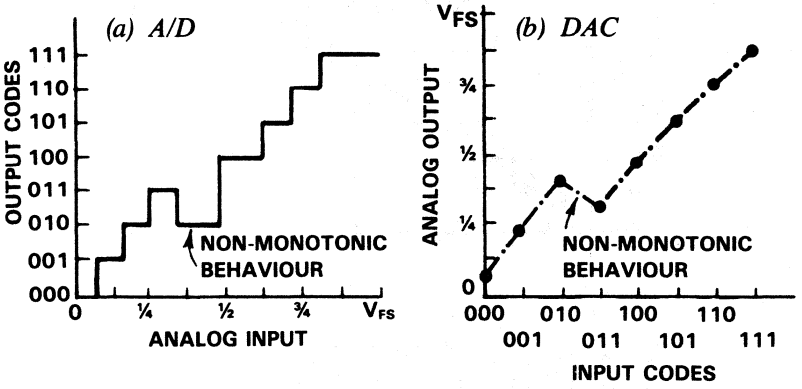


Fig. 11.22 Non-monotonic behaviour of an A/D and a DAC

Monotonicity is a converter transfer function behaviour that ensures a consistent increase or decrease in the analog output of a DAC or digital output of an A/D in response to a consistent increase or decrease in the digital or analog input respectively. Fig. 11.22 illustrates non-monotonic conversion.

Missing Codes

Missing codes are intermediate digital output codes of an A/D that, with a changing analog input, fail to occur and cause multiple code output steps, see Fig. 11.19. Differential nonlinearity of −1 LSB is synonymous with missing codes.

Acquisition Time

Acquisition time refers to sample and hold circuits, and A/D converters that include this function, and defines the time needed to acquire a signal to within a specified accuracy.

Aperture Time

Aperture time is the width of the sampling window for an A/D or sample and hold circuit. A/D converters that do not include a sample and hold function require an analog input to be held to within ½ LSB over

the conversion period to prevent errors. This can severely restrict the input frequency and slew rate ranges that are converted accurately.

Aperture Delay Time and Uncertainty

For sample and hold functions the above aperture time is the transition time from sample to hold. The sampled value held is the average input value over this transition and is effectively delayed from the equivalent input signal value. This is called aperture delay time and is usually unimportant as a sampling command signal can be advanced in time to compensate. In this situation the sampling window width is the uncertainty or jitter in the sampling instant which will be caused by noise on the digital hold command signal or analog input signal.

Effective Aperture Delay Time

Effective aperture delay time is a parameter that takes account of both the analog and hold command signals' propagation delays. For example, if in a device the propagation delay of the analog signal is longer than the hold command signal, then the effective analog signal sample instant will be before the hold command was issued, that is, a negative effective aperture delay time.

Hold Time, Droop Rate

Hold time and droop rate defines how long a sample and hold circuit will hold the stored values to within a given accuracy.

Settling Time

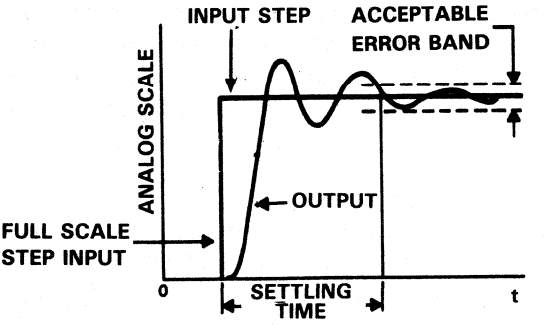


Fig. 11.23 Example of settling time

Settling time applies to DAC and sample and hold outputs and some A/D inputs, and is the time between the application of a full-scale input step and when the analog signal enters a specified error band for the last time, see Fig. 11.23.

Access and Conversion Time

Access and conversion time is, for a single-chip data acquisition system, the total time taken to address and select a multiplexer input, perform A/D conversion and output the digital code.

INPUT AND OUTPUT CODES

Introduction

Data acquisition systems can use a variety of input/output codes for operation of DACs and A/Ds respectively. Binary codes cater for most applications, natural binary is usually used for unipolar operation, but there are many binary bipolar codes that provide both polarity and amplitude information, the most popular of these are perhaps offset binary and two's complement. Other codes used for more specialised applications are Binary Coded Decimal (BCD) and Gray Codes.

Bipolar Codes

Natural Binary

Natural binary is the best known code and is most familiar for representing integers where, for example, an 8-bit byte most-significant-bit (MSB) would be 2^7 and equal to 128 in decimal. It is more useful to consider the binary digits of a code in data acquisition as representing the weighted value of a full scale reference, the MSB then becomes $2^{-1} \times V_{FS}$ = ½ of the full scale reference. Table 11.2 and Fig. 11.19 illustrate the natural binary code used for unipolar operation.

Offset Binary

Offset binary is a bipolar code, derived from natural binary by offsetting the bipolar zero to the half scale point and setting the MSB to '1'. The all zero's code equals negative full-scale range $-V_{FS}$ and all one's code equals the positive full-scale range, $+V_{FS}$, minus 1 LSB in a similar manner to unipolar natural binary. There is an inherent asymmetry between

the highest and lowest codes for bipolar operation see Fig. 11.24 and Table 11.2.

Table 11.2 Natural and offset binary

	Bit Position				Level	
	2^3	2^2	2^1	2^0	Natural Binary	Offset Binary
Natural Binary Zero →	0	0	0	0	0	−8
	0	0	0	1	1	−7
	0	0	1	0	2	−6
	0	0	1	1	3	−5
	0	1	0	0	4	−4
	0	1	0	1	5	−3
	0	1	1	0	6	−2
	0	1	1	1	7	−1
Offset Binary Zero →	1	0	0	0	8	+0
	1	0	0	0	9	+1
	1	0	1	0	10	+2
	1	0	1	1	11	+3
	1	1	0	0	12	+4
	1	1	0	1	13	+5
	1	1	1	0	14	+6
	1	1	1	1	15	+7

SIGN + MAGNITUDE	TWO'S COMPLEMENT (100)	ONE'S COMPLEMENT	OFFSET BINARY (1000)
011	010	011	111
010	010	010	110
001	001	001	101
000 100	000	000 111	100
101	111	110	011
110	111	101	010
111	101	100	001
– –	100	– –	000

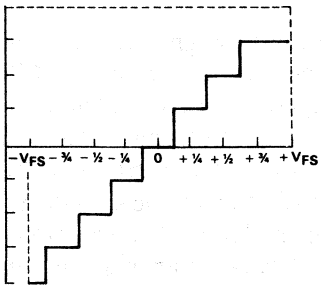


Fig. 11.24 Bipolar codes shown with a 3-bit A/D transfer function

One's and Two's Complement Codes

Table 11.3 One's and two's complement binary codes

	Bit Position				Level	
	2^3	2^2	2^4	2^0	Two's	One's
	1	0	0	0	−8	−7
	1	0	0	1	−7	−6
	1	0	1	0	−6	−5
	1	0	1	1	−5	−4
	1	1	0	0	−4	−3
	1	1	0	1	−3	−2
	1	1	1	0	−2	−1
	1	1	1	1	−1	−0
One's Zero →						
Two's Zero →	0	0	0	0	0	+0
	0	0	0	1	+1	+1
	0	0	1	0	+2	+2
	0	0	1	1	+3	+3
	0	1	0	0	+4	+4
	0	1	0	1	+5	+5
	0	1	1	0	+6	+6
	0	1	1	1	+7	+7

One's and two's complement are widely used codes because they are more compatible with digital computing. Both codes are bipolar, one's complement has two codes for zero and is symmetrical between highest and lowest codes whereas two's complement has a single code for zero and is asymmetrical between highest and lowest codes, see Fig. 11.24 and Table 11.3. Two's complement is arguably the most satisfactory bipolar code and is used by popular digital signal processing devices such as the TMS320 family.

Sign-plus-Magnitude Code

Sign-plus-magnitude code is a bipolar code which uses natural binary to indicate magnitude and the MSB to indicate polarity, 0 for positive, 1 for negative. There are two codes for bipolar zero, see Fig. 11.24 and Table 11.4. Where other codes which extend a binary code length over a bipolar range reduce resolution by one-half, the sign-plus-magnitude code with the addition of one bit allows an easy bipolar implementation without reducing resolution. This can carry over into hardware where, for example, the output range of an unipolar DAC can be extended to bipolar operation with the use of analog polarity switches controlled by the sign bit, see Fig. 11.25.

Table 11.4 Sign-plus-magnitude code

Bit Position 2^3	2^2	2^1	2^0	Level
0	1	1	1	−7
0	1	1	0	−6
0	1	0	1	−5
0	1	0	0	−4
0	0	1	1	−3
0	0	1	0	−2
0	0	0	1	−1
0	0	0	0	−0
Zero →				
1	0	0	0	+0
1	0	0	1	+1
1	0	1	0	+2
1	0	1	1	+3
1	1	0	0	+4
1	1	0	1	+5
1	1	1	0	+6
1	1	1	1	+7

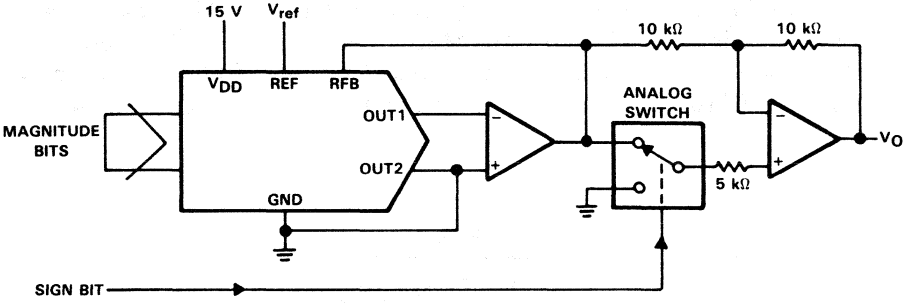


Fig. 11.25 Sign-plus-magnitude use with unipolar DAC

Gray Code

The Gray code has the unique feature of only one bit changing as the code moves to adjacent levels, bits of weight below the MSB reflect about the midpoint, see Table 11.5. The bit weights do not readily combine to give a binary magnitude, but the Gray code can be converted to natural binary by a simple algorithm as follows: the binary MSB is the same as the Gray code MSB, reading from MSB to LSB of the Gray code, if the next bit is 1 then the corresponding binary bit is the complement of the previous binary bit, if the next bit is 0 the corresponding binary bit is the

Table 11.5 The grey code

Bit Position 2^3	2^2	2^1	2^0	Level
0	0	0	0	0
0	0	0	1	1
0	0	1	1	2
0	0	1	0	3
0	1	1	0	4
0	1	1	1	5
0	1	0	1	6
0	1	0	0	7
Midpoint →				
1	1	0	0	8
1	1	0	1	9
1	1	1	1	10
1	1	1	0	11
1	0	1	0	12
1	0	1	1	13
1	0	0	1	14
1	0	0	0	15

same as the previous binary bit. For example Gray 1010 becomes binary 1100, that is from MSB to LSB, MSB 1 becomes 1, 1 to 0 becomes 1, 0 to 1 becomes 0, 1 to 0 becomes 0, thus Gray code 01110101 becomes binary 01011001. This is implemented with hardware such as shown in Fig. 11.26.

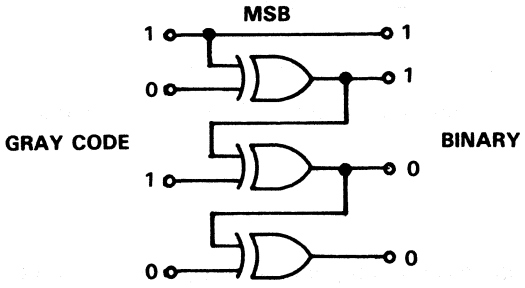


Fig. 11.26 Gray code to binary conversion

The Gray code is used for most types of electromechanical quantisers such as shaft encoders where the 1-bit change per level prevents the false intermediate codes that would occur if natural binary were used.

Binary-Coded-Decimal (BCD) Code

Table 11.6 The 8421 Code

8	4	2	1	Level
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

The BCD code with bit weights of 8421 is shown in Table 11.6, other bit weights are possible such as 2421. Although in binary form, the codes only exist in decades, 0–9, when the decade limit is reached the code carries to the next higher decade which has a 10 × multiplier associated with it. BCD codes are used when direct decimal readouts are required for measuring instruments such a digital thermometers or multimeters. The term "½ digit" is often used in connection with these refering to a decimal digit limited to the figures of "0" or "1" in the highest positional value. This additional digit doubles the range covered by the other "n" digits.

ANALOG-TO-DIGITAL CONVERSION TECHNIQUES

Introduction

Several types of analog-to-digital conversion techniques are available. Some feature speed of conversion while others give high resolution. The four most widely used methods are as follows:

1. Single Slope
2. Dual Slope
3. Successive Approximation
4. Flash and Semi-flash

Each of these methods have advantages and disadvantages, some of the primary features are shown in Table 11.7.

Table 11.7 Comparison of Conversion Techniques

TYPE	SPEED	ERROR	RESOLUTION
Single-slope	Slow (ms)	High	Medium-High (7 – 14 Bits)
Dual-slope	Slow (ms)	Low	High (10 – 18 Bits)
Successive-Approximation	Medium (10's μs)	Medium (0.5 – 1 LSB)	Medium-High (8 – 16 Bits)
Flash	Fast (ns – μs)	Medium (0.5 – 1 LSB)	Low-Medium (4 – 10 Bits)

Methods of Conversion

Single-Slope A/D Converters

A single-slope A/D converter arrives at it's digital output by comparing an unknown analog input signal with a ramp voltage. A digital value is obtained by counting the number of clock pulses needed to build a ramp from 0 V to the value of the unknown analog signal. The ramp must be precisely controlled so that it is very linear and will take 2^n clock pulses to reach the full scale reference voltage value. Fig. 11.27 shows a 6-bit example with 64 increments to the full scale 5 V reference, an example analog input voltage is shown to be 34/64 × 5 V = 2.6 V.

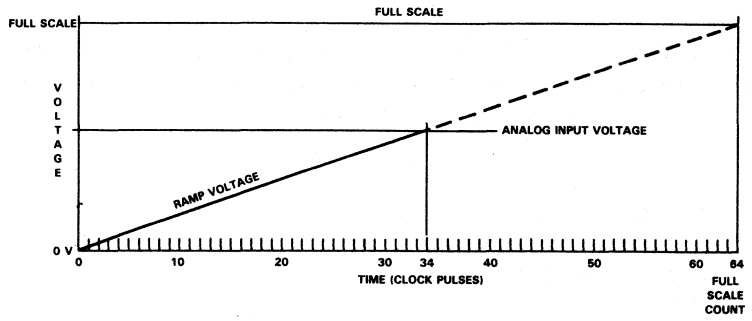


Fig. 11.27 Single-slope ramp with analog input voltage comparison

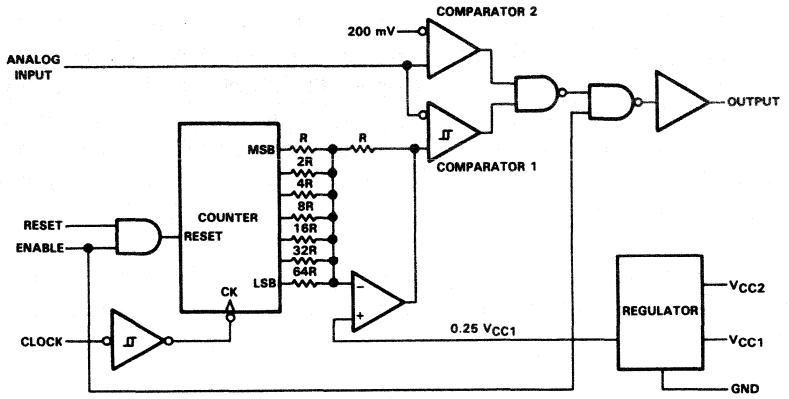


Fig. 11.28 Single-slope A/D converter block diagram (TL507)

The requirements for a good single-slope converter are a stable: reference voltage, clock and ramp generator. Fig. 11.28 shows a block diagram of a single-slope converter (TL507) with a ramp generator comprising a counter, resistive ladder DAC, and op-amp, a voltage reference and a comparator(1).

Dual-Slope A/D Converters

Dual-slope A/D conversion is an indirect method that uses a counter and an integrator to convert an unknown analog input voltage into a ratio of time periods multiplied by a reference voltage. A first integration is over a fixed period of time (T), see Fig. 11.29, and uses an unknown analog voltage as input.

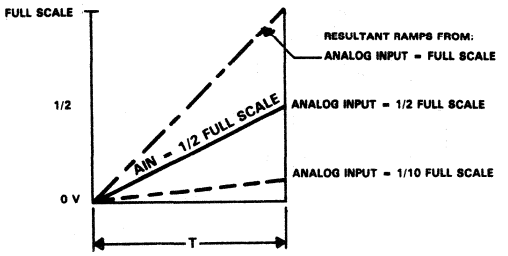


Fig. 11.29 Dual-slope A/D – first integration period

At the end of the first integration period a peak value proportional to the input voltage is held on the integrator output. A second integration period uses a reference voltage of opposite polarity as the input to the integrator whose output ramps down at a rate only dependent on the reference value, see Fig. 11.30. The resultant value of the first period is integrated down to 0 V in a variable time (t) proportional to the amplitude of the input voltage.

Putting these two ramps together, Fig. 11.31, using the same clock to count time periods T and t the unknown input voltage can be determined as a count over time t:

$$t = \frac{V_{in}}{V_{ref}} \times T$$

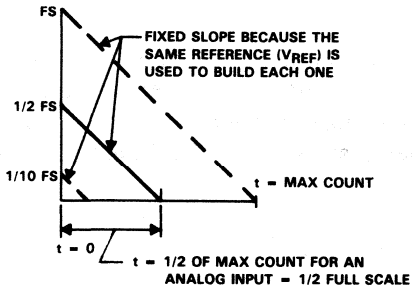


Fig. 11.30 Dual-slope A/D − second integration period

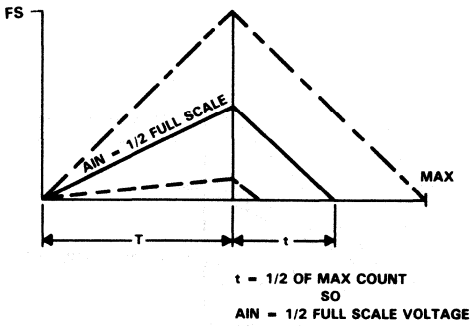


Fig. 11.31 Dual-slope A/D − both integration periods

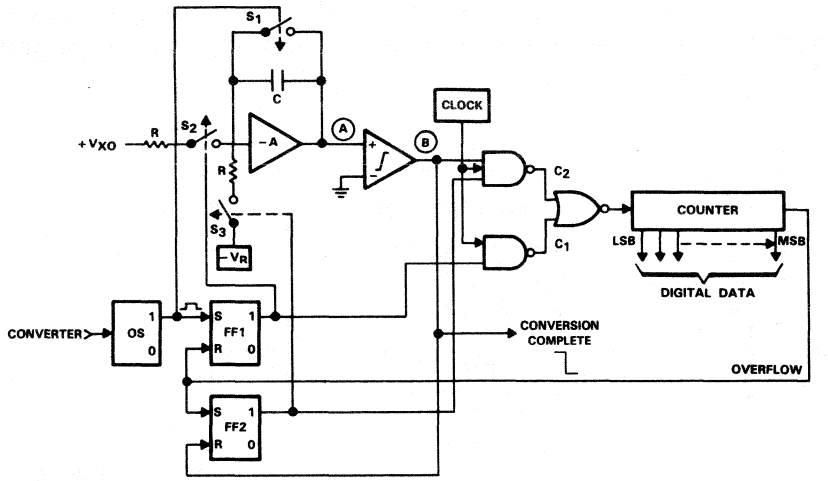


Fig. 11.32 Dual-slope A/D converter functional diagram

Using the same integrating network cancels errors due to comparator offset, capacitor tolerances, long term counter clock drift and integrator nonlinearities. Fig. 11.32 shows a functional block diagram of a dual-slope converter.

While the conversion speed of a dual-slope converter is slow, high resolution is possible. Resolution is determined from the ratio of counts in the integration periods, for example 1 in 20,000 counts is better than 14-bits of resolution, with the basic error as ±1 count plus the reference voltage error.

Successive Approximation A/D Converters

Successive approximation A/D Converters continue to be the most popular type and have conversion times in the low 10's of microseconds and resolution that ranges from 8-bits to beyond 12-bits. Successive comparison of an unknown analog input with binary weighted values of a reference give this method its name of "successive approximation". A converter of n-bit resolution takes "n" steps to achieve a digital output.

One input of the comparator, shown in the block diagram of Fig. 11.33, is driven by an unknown analog input signal while the output

of the reference DAC drives the other. The successive approximation register provides the input to the DAC and responds to the output from the comparator.

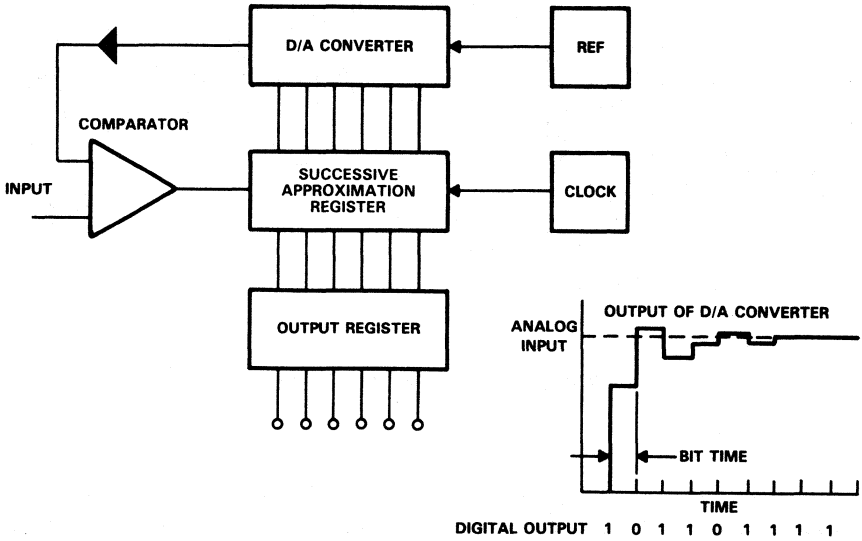


Fig. 11.33 Successive approximation A/D converter block diagram

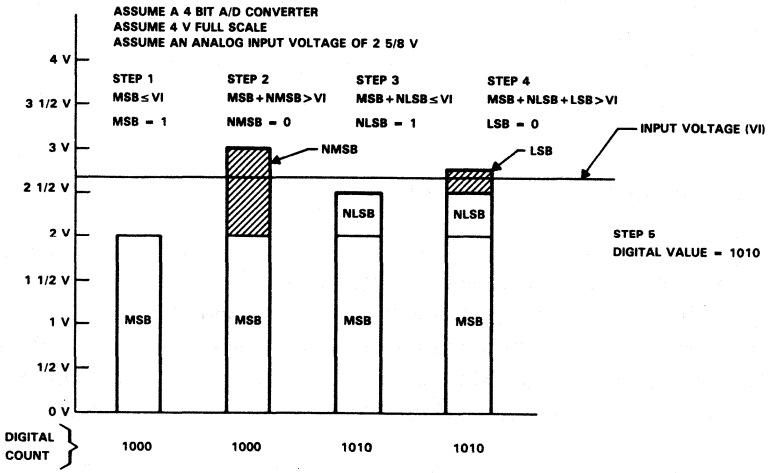


Fig. 11.34 4-Bit Successive approximation example

When the DAC of Fig. 11.33 has its MSB set to 1 (with all other bits zero), by the successive approximation register (SAR), it will produce a voltage output of ½ the reference and analog input full-scale range. The comparator then determines if the DAC output is above or below the unknown input signal. In Fig. 11.34 the input is above the DAC output value and the MSB is retained in the SAR while the next weight of ¼ the reference is compared. This process continues until all bits are tested and the nearest approximation to the input signal is obtained. The result is then passed to the output register.

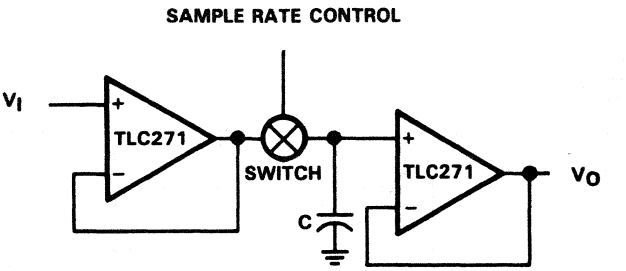


Fig. 11.35 Basic sample and hold circuit

While the successive approximation process continues the input signal must be held constant. This is accomplished with a sample and hold circuit such as the circuit of Fig. 11.35 which is placed ahead of the comparator of Fig. 11.33. Between conversions the sample and hold circuit acquires the input signal and just before the conversion starts it is placed into hold mode and remains there until conversion is complete.

More recent successive approximation converter designs, using switched capacitor networks utilizing charge redistribution, are replacing older designs using resistive ladder DACs. This is due to the switched capacitor techniques smaller chip area, higher speed and inherent sample and hold function.

Flash A/D Converters

The flash A/D converter derives its name from its ability to do a very fast conversion. This is accomplished by providing a comparator for every quantisation level (2^n—1 comparators). These comparators all examine the input in parallel and make an immediate conversion. The block diagram

of Fig. 11.36 shows that each comparator has one input connected to the input signal and the other to a tap on a reference potential divider. The comparator outputs are decoded and output through a latch.

Presently there are a wide range of flash converter speeds and resolutions on the market, the range covers 4 to 10-bit with maximum sampling rates in excess of 250 M samples/second. The most popular applications are in the video field where 8-bit 20 M sample/second devices are standard.

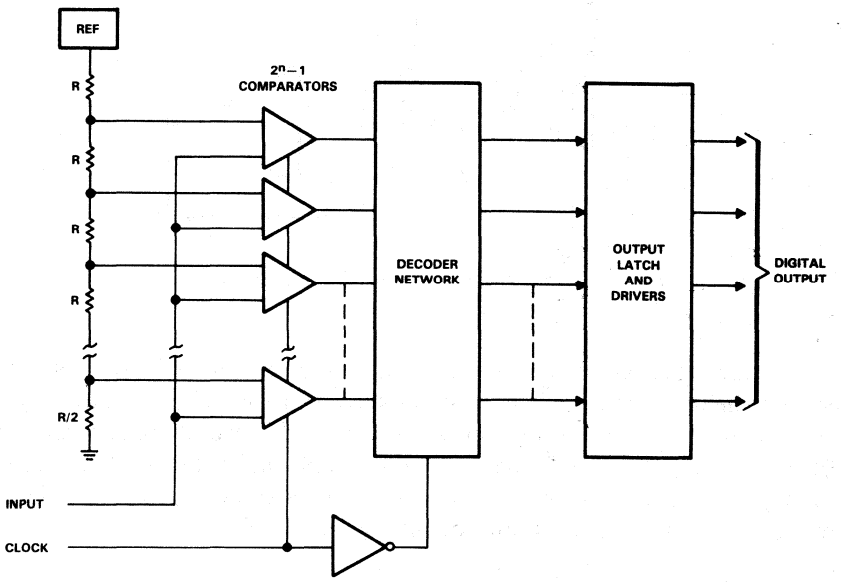


Fig. 11.36 Flash A/D converter block diagram

KEY SELECTION CRITERIA FOR DATA ACQUISITION COMPONENTS

With the overwhelming choice of available data acquisition components selection of the right device for a given application is an involved task. This should be tackled with full attention to detail so that the likelihood of problems at a later stage are much reduced.

Manufacture data sheets should always be consulted for the latest information and read using a full understanding of the specifications. When comparing alternative devices special attention should be paid to the differences that may exist between various manufacturers' specifications, it should not be assumed that terms used always have the same meaning.

Before considering a device for an application the first step is to clearly define the performance required trying to leave scope for unknowns that will appear at a later stage. The following is a list which gives an idea of items that should be considered when choosing a data acquisition component:

Accuracy, Signal and Noise Levels: System absolute accuracy or dynamic range requirements will determine the basic error or noise levels allowable. This can be interpreted as a resolution requirement, in terms of numbers of bits, modified by error terms. These include linearity errors, reference accuracy, temperature coefficients, monotonic or missing code performance importance, etc.

Throughput and Conversion Speed: This is determined by the sampling rate needs of the application, ensure that the converter has enough analog bandwidth and slew rate capability to handle the incoming signal.

Control Interface: Items to consider are output code format, logic level compatibility, parallel or serial data exchange, control line details, 3-state outputs, etc.

Analog Interface: Input bandwidth, slew rate, settling time, unipolar or bipolar range, maximum input range, absolute reference or ratiometric operation, source, output and load impedances, leakage currents, switch charge injection, cross talk, all these may need to be considered. Input channel numbers and configuration have to be fixed for multiplexer use. Reference voltage value is determined from input signal maximum value, care should be exercised with error specifications if device operating reference range is much reduced from specified values.

Environmental and Packaging: Supply voltage ranges, operating temperature ranges and their effect on performance and how a device can be integrated into a manufacturable product should be considered.

Budgetry Limitations: One of the first items to be tackled at the feasibility stage.

DATA ACQUISITION DESIGN EXAMPLE

Introduction

An example is given below to help tie together data acquisition concepts previously discussed. A simple block diagram, Fig. 11.37, shows the acquisition of analog signals into the digital domain, for some unspecified processing, and their subsequent recovery into analog signals.

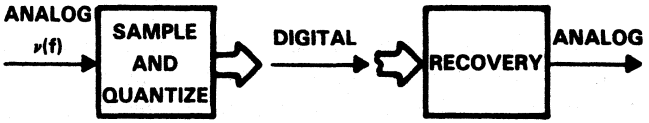


Fig. 11.37 Data acquisition and recovery block diagram

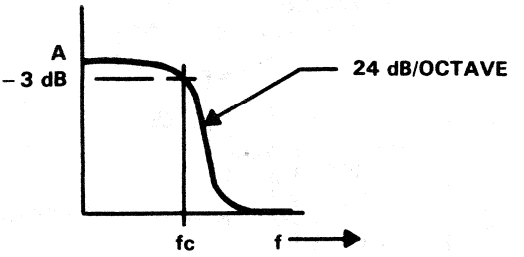
Application of data acquisition and recovery components will be determined by the characteristics of the signal being processed. Communication systems employ continuous time signals whose spectral properties are usually known and whose recovery is most concerned with the fidelity of its frequency content. Measurement and control systems more often process signals of a non-continuous nature and require sample rates that recover a measured amplitude to a specified resolution or accuracy.

From an overall fidelity or error specification, careful choices about error contributions from the separate signal processing operations and data acquisition components should be made to obtain a balance between performance and cost. The signal processing operations that should be considered are signal-to-noise, dynamic range, or accuracy, aliasing error, aperture error, and recovery error. These will determine sampling rates and the necessary performance of such components as the: anti-aliasing filter, sample and hold, A/D converter, D/A converter and output recovery filter or interpolator.

Example Specification

Assume for this example that the input signal has the characteristics of a fourth-order Butterworth filter, Fig. 11.38, and that the input range of interest extends to 2 kHz. The four requirements for this system are:

1. Dynamic range > 52 dB
2. Alias error < +3 dB above quantisation noise.
3. Recovery error < 0.1% of max rms input signal.
4. Aperture error < 0.1% of max rms input signal.



f	%A	Response
f_c	70.7	– 3 dB
2 f_c	6.25	– 24 dB
4 f_c	0.4	– 48 dB
8 f_c	0.024	– 72 dB
12 f_c	0.0048	– 86 dB
16 f_c	0.0015	– 96 dB

Fig. 11.38 Fourth order butterworth filter frequency response

The parameters to be determined are:

1. Number of bits of resolution
2. Sample rate, from aliasing or recovery requirements
3. Sample aperture time requirements

Bits of Resolution

Dynamic range is specified as greater than 52 dB, this is equal to the signal-to-noise ratio (S/N) for full-scale signals. The ratio of full-scale rms sinewave input to ideal quantisation (q) noise of $q/\sqrt{12}$, this is given by: S/N = 6.02n + 1.76 dB, where n = number of bits. A/D linearity errors increase quantisation noise. In the worst case code widths are doubled for a linearity error of ±½ LSB, See Fig. 11.39, and this increases the quantisation noise to $q/\sqrt{3}$. The worst case S/N is decreased by 6 dB giving S/N = 6.02n −4.24 dB. Substituting n = 10 gives a S/N of 56 dB which satisfies the specification with tolerance for other system errors.

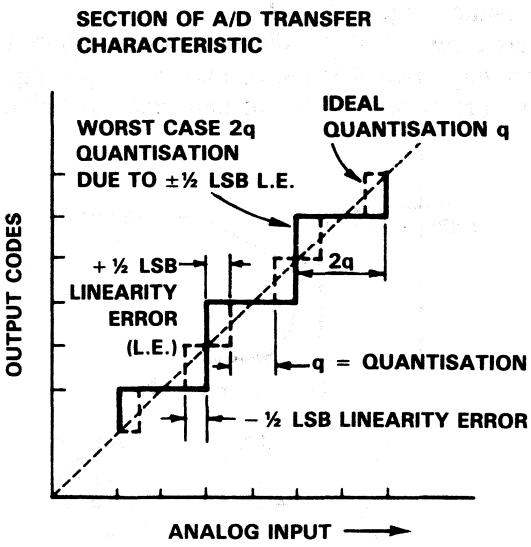


Fig. 11.39 Worst case quantisation due to ± ½ LSB linearity error

Sampling Frequency-Determined from Aliasing Error

With a wide-band input spectrum, which could be noise, there is the possibility of folding-back interfering signals across the entire input spectrum, see Fig. 11.40.

To guarantee that error due to aliasing is not more than +3 dB, the total rms voltage folded-back over the 2 kHz input frequency range of interest should be equal to or less than the quantisation noise $q/\sqrt{3}$. The attenuation of the input signal by the input filter to achieve this condition is determined as follows:

$$\text{Quantisation } q = \frac{2V_p}{2^n} = \frac{V_p}{2^9}$$

where: $n = 10$ bits,
V_p = peak input voltage

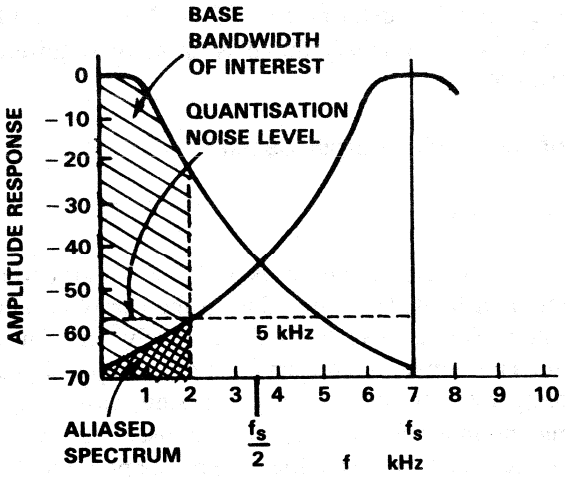


Fig. 11.40 Example of adjusting sampling frequency to reduce aliasing components to the quantisation noise level

$$\text{Max value folded-back} = \frac{V_p}{\sqrt{3}.2^9} \quad \text{Max rms input} = \frac{V_p}{\sqrt{2}}$$

Therefore attenuation A =

$$\frac{\text{Max rms input}}{\text{Max rms folded value}} = 2^9\sqrt{1.5} = 627 \text{ (56 dB)}$$

Attenuation A of a Butterworth filter $= [1 + (f/f_c)^{2n}]^{1/2}$

For $f > f_c$ $A \simeq (f/f_c)^n$ where f_c −3 dB filter cutoff frequency = 1 kHz
and n = 4 = filter order

$$\therefore f = f_c \sqrt[4]{A} = 1 \times \sqrt[4]{627} = 5 \text{ kHz}$$

This shows that the input Butterworth filter response attenuates a full amplitude signal to the level of the quantisation noise at a frequency of 5 kHz. This quantisation noise, is the maximum value specified to be folded-back at 2 kHz and occurs at a frequency of f_s − 5 kHz, Fig. 11.40. The

sampling frequency f_s is determined as $f_s = 2 + 5 = 7$ kHz. A 7 kHz sampling frequency will satisfy the aliasing error requirement with the criteria chosen above.

Recovery of the Analog output Signal

Stepped Output Error

When an A/D converter produces a digital output code representing an analog input, it holds this value during the sampling period until the next conversion result is available and effectively performs a sample and hold function. The step-hold representation of the input signal is preserved at the sampling rate through digital processing, assuming that a digital oversampling and interpolation process is not used. A DAC maintains this stepping in its recovery of the analog output, see previous Fig. 11.17. To improve the fidelity of recovery, some form of interpolation or filtering is required to remove the error contained in the steps. If the ideal output waveform is subtracted from this recovered output a sawtooth error waveform is the result. Fig. 11.41.

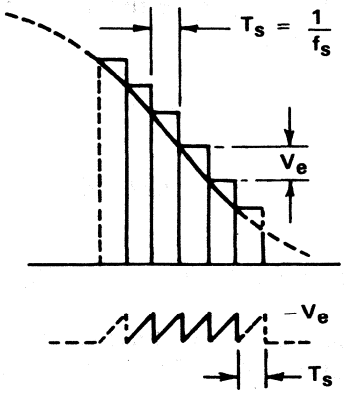


Fig. 11.41 Section of stepped DAC output and error waveform

The error shown in Fig. 11.41 is similar to that caused by either aperture time or aperture uncertainty when using A/D's or sample-and-

holds with slewing signals. The error voltage V_e can be determined in terms of a sinewave's rate of change as:

$$V_e = \frac{2\pi f_c V_p}{f_s} \quad (1)$$

Where f_c = maximum frequency
V_p = peak amplitude
$f_s = 1/T_s$ = sampling frequency

If the error, Fig. 11.41, is approximated as a sawtooth waveform of amplitude V_e a Fourier analysis can be performed to yield its frequency components.

$$f(t) = V_e \left[\tfrac{1}{2} + \frac{1}{\pi} \sin 2\pi f_s t + \frac{1}{2\pi} \sin 4\pi f_s t + \ldots.\right]$$

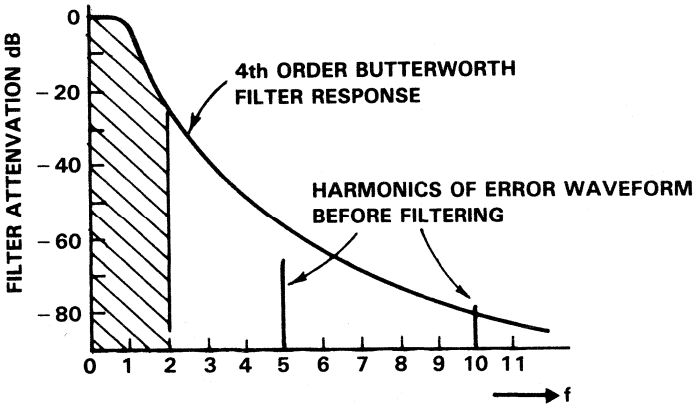


Fig. 11.42 Filter response and harmonic content of the error waveform

Fig. 11.42 represents the filter response and harmonic content of the error waveforms. From the Fourier expression the rms value of the first and second harmonic of f_s can be evaluated as:

$$\frac{V_e}{\sqrt{2}} \sqrt{\left(\frac{1}{\pi}\right)^2 + \left(\frac{1}{2\pi}\right)^2} = 0.251\ V_e$$

Higher frequency harmonics can be neglected as of small influence, in practice the average dc component will be zero as it cancels on alternate quarter cycles of the input waveform.

If f_s is chosen as $5 \times f_c$ then $V_e = \frac{2\pi V_p}{5}$ from (1)

If a fourth order Butterworth filter, of the same response as the input filter spectrum, is used to recover the output signal the attentuation at $5 \times f_c = 625$ and $10 \times f_c = 10^4$ can be applied to the 1st and 2nd harmonics. The error voltage then becomes $= 3.6 \times 10^{-4}\ V_e$.

The % error with respect to a full-amplitude signal is then:

$$3.6 \times 10^{-4} \times \frac{2\pi V_p}{5} \div \frac{V_p}{\sqrt{2}} \times 100 = 0.06\%$$

This fulfils the <0.1% recovery error requirement.

Sinx/x Error

The spectrum of the DAC stepped output has the envelope of the sinx/x or sinc function, Fig. 11.43.

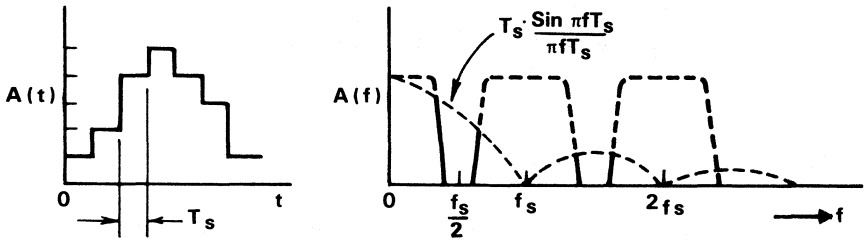


Fig. 11.43 DAC stepped output and frequency spectrum envelope

The error at any output frequency f equals:

$$\% \text{ error} = \left(1 - \frac{\sin \pi f T_s}{\pi f T_s}\right) \times 100$$

where $T_s = 1/f_s$ = sampling period

For the sampling rate, from the aliasing condition of $f_s = 7f_c$, $fT_s = 1/7$ and the % error is evaluated as 3.4%. To reduce this error either the sample rate needs to be increased, $f_s = 40f_c$ will achieve a 0.1% error, or the recovery filter response should be tailored to compensate for the sinx/x output. An ideal recovery filter response is shown in Fig. 11.44. Alternatively the compensation could be achieved in the digital domain before being output to the DAC.

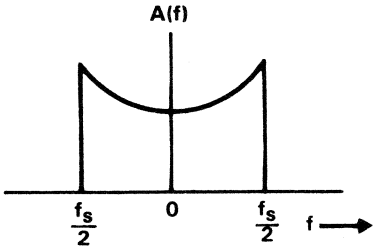


Fig. 11.44 Ideal recovery filter response shape

Aperture Error

When the sample rate is known, from either the requirements of aliasing or recovery, the maximum aperture time can be evaluated for a specified error. Fig. 11.45 shows the frequency response envelope of aperture error. This shows a similarity to the stepped DAC output spectrum but has a aperture time t_a not restricted to the sampling period T_s.

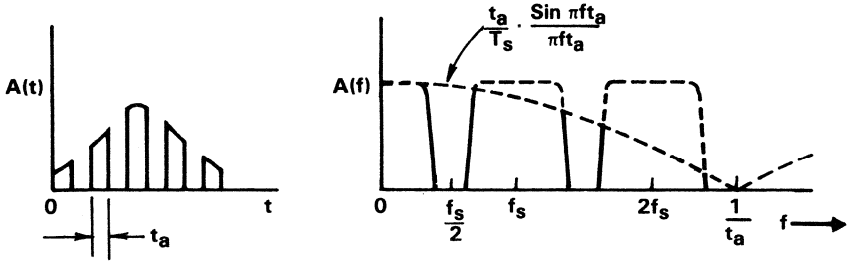


Fig. 11.45 Aperture error frequency response envelope

$$\% \text{ error} = \left(1 - \frac{\sin \pi f t_a}{\pi f t_a}\right) \times 100$$

For a 0.1% error $ft_a = 1/40$ and the maximum aperture time for $f = f_c = 1$ kHz is 25 μs, not a stringent condition.

In summary to fulfil the specification, a 10-bit A/D is required that can sample at 7 kHz with less than: ± ½ LSB linearity error and 25 μs aperture time. TLC1540 could be used in this application. This example has been presented to demonstrate the considerations that have to be made in the specification of performance and selection of system parameters and components.

ANALOG-TO-DIGITAL AND DIGITAL-TO-ANALOG CONVERTER DEVICES AND APPLICATIONS

Introduction

Digital and microcomputing techniques in measurement, communications and control systems have increased rapidly, their need to interface to the analog real world has been filled by a parallel growth in analog-to-digital (A/D) and digital-to-analog (DAC) device availability. This section concentrates on Texas Instruments' cost effective range of A/D and DAC devices and their application with popular microprocessors, microcontrollers and microcomputers.

Proprietary devices built with LinCMOS™ technology using the "TLC" prefix are discussed first. A/Ds covered are the multi-input TLC532A/33A with a parallel output and TLC540/1540 family of serial output devices, TLC0820 semi-flash and TLC32040/41 analog interface circuit DSP peripheral follow. The TLC549 stands out here, with its 8-pin SO package capability, as becoming the standard for cost effective 8-bit applications. Advanced LinCMOS™ DACs TLC7524 and TLC7528 are followed by the established bipolar TL500 series of dual slope and the TL507 single slope A/D converters. Popular multi-sourced CMOS devices ADC0803/4/5, ADC0808/9 and ADC0834/8 conclude the section.

TLC532A AND TLC533A LinCMOS 8-Bit ANALOG-TO-DIGITAL PERIPHERALS WITH PARALLEL INPUT/OUTPUT

The TLC532A and TLC533A are complete peripheral data acquisition systems on a single chip, that are designed to interface with a microprocessor, and are built with LinCMOS technology. Each contains a 12 channel analog multiplexer, an 8-bit ratiometric analog-to-digital (A/D) converter, a sample and hold, three 16-bit registers and microprocessor compatible control logic.

ADDRESS/CONTROL					DESCRIPTION
R/$\overline{W}$	RS	$\overline{CS}$	$\overline{R}$	CLK	
X	X	X	L†		Reset
L	H	L	H	↓	Write bus data to control register
H	L	L	H	↑	Read data from analog conversion register
H	H	L	H	↑	Read data from digital data register
X	X	H	H	X	No response

H = High-level, L = Low-level, X = Irrelevant
↓ = High-to-low transition, ↑ = Low-to-high transition
†For proper operation, Reset must be low for at least three clock cycles.

Fig. 11.46 Function table

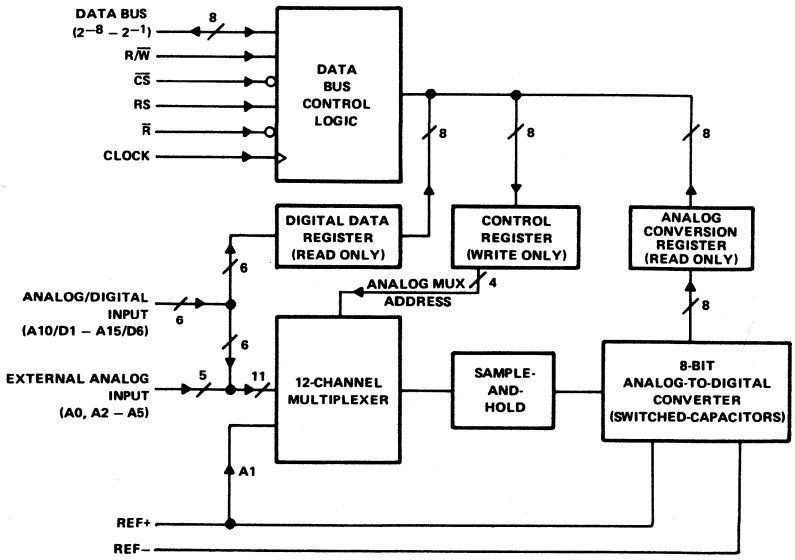


Fig. 11.47 Functional block diagram

Fig. 11.46 shows the function table and Fig. 11.47 is the functional block diagram. The 12 analog multiplexer inputs are used for: a built-in self test, six multipurpose analog or digital inputs and five analog only inputs. The three on-chip data registers store the control data, conversion results and the input digital data. These can be accessed via the 8-bit input/output (I/O) three-state bidirectional port as two 8-bit bytes, most significant byte first. In this manner a microprocessor can access the 11 external analog inputs or six digital signals and the positive reference voltage used for self test.

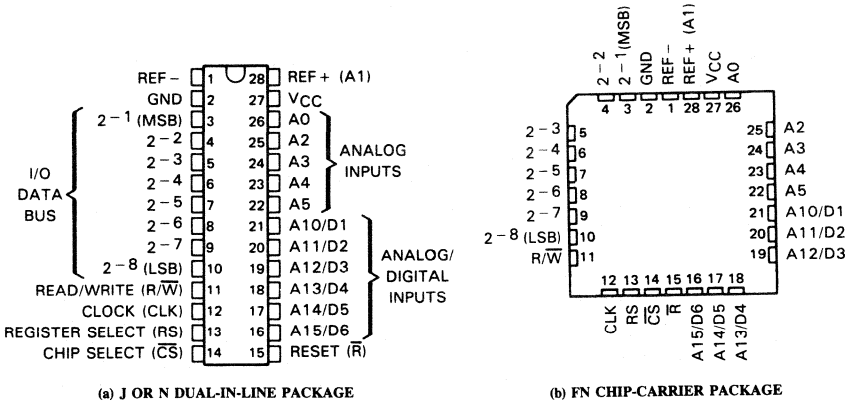
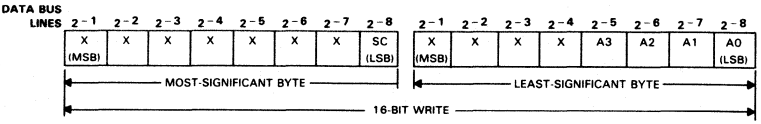


Fig. 11.48 TLC532A and TLC533A pinouts (top view)

The access plus conversion time is 15 μs maximum for the TLC532A and 30 μs for the TLC533A. Other features are 5 V single supply operation, with a low power consumption of typically 6.5 mW. Fig. 11.48(a) and (b) show the pinouts, the I/O data bus lines as well as the analog input lines are grouped together for efficient PC board layout.

Principles of Operation

The TLC532A and TC533A can be directly connected to a microprocessor-based system. Control of the TLC532A and TLC533A is handled via the 8-line TTL-compatible three-state data bus, the three control inputs (Read/Write, Register Select, and Chip Select), and the Clock input. Each device contains three 16-bit registers. These registers are the control register, the analog conversion data register, and the digital data register, Fig. 11.49.



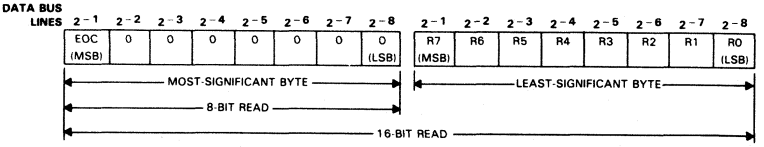
Unused Bits (X) – The MS byte bit 2^{-1} through 2^{-7} and LS byte bits 2^{-1} through 2^{-4} of the control register are not used internally.

Start Conversion (SC) – When the SC bit in the MS byte is set to a logical 1 (high level), analog-to-digital conversion of the specified analog channel will begin immediately after the completion of the control register write.

Analog Multiplex Address (A0-A3) – These four address bits are decoded by the analog multiplexer and used to select the appropriate analog channel as shown below:

Hexadecimal Address (A3 = MSB)	Channel Select
0	A0
1	REF+ (A1)
2-5	A2-A5
6-9 (not used)	
A-F	A10-A15

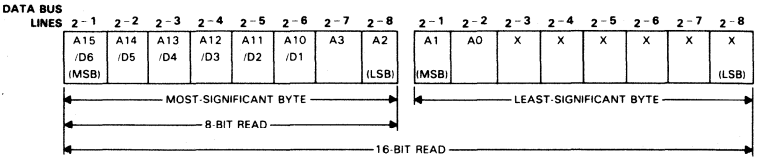
CONTROL REGISTER TWO-BYTE WRITE WORD FORMAT AND CONTENT



A/D Status (EOC) – The A/D status end-of-conversion (EOC) bit is set whenever an analog-to-digital conversion is successfully completed by the A/D converter. The status bit is cleared by a 16-bit write from the microprocessor to the control register. The remainder of the bits in the MS byte of the analog conversion data register are always reset to logical 0 to simplify microprocessor interrogation of the A/D converter status.

A/D Result (R0-R7) – The LS byte of the analog conversion data register contains the result of the analog-to-digital conversion. Result bit R7 is the MSB and the converter follows the standard convention of assigning a code of all ones (11111111) to a full scale analog voltage. There are no special overflow or underflow indications.

ANALOG CONVERSION DATA REGISTER ONE-BYTE AND TWO-BYTE READ WORD FORMAT AND CONTENT



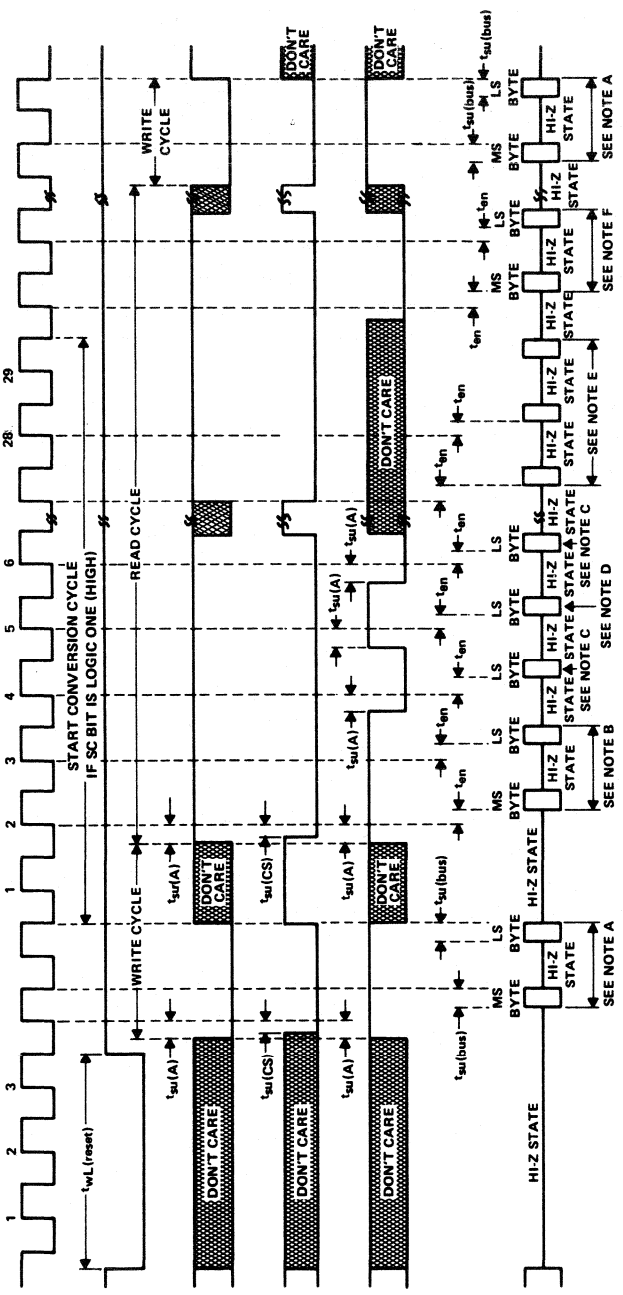
Shared Digital Port (A10/D1-A15/D6) – The voltage present on these pins is interpreted as a digital signal and the corresponding states are read from these bits. A digital value will be given for each pin even if some or all of these pins are being used as analog inputs.

Analog Multiplexer Address (A0-A3) – The address of the selected analog channel presently addressed is given by these bits.

Unused Bits (X) – LS byte bits 2^{-3} through 2^{-8} of the digital data register are not used.

DIGITAL DATA REGISTER ONE-BYTE AND TWO-BYTE READ WORD FORMAT AND CONTENT

Fig. 11.49 The three internal register details



NOTES:
A. This is a 16-bit input instruction from the microprocessor being sent to the control data register.
B. This is the 2-byte (16-bit) content of the digital data register being sent to the mircoprocessor.
C. This is the LS byte (8-bit) content of the analog conversion data register being sent to the microprocessor.
D. This is the LS byte (8-bit) content of the digital data register being sent to the microprocessor.
E. These are MS byte (8-bit), LS byte (8-bit), and LS byte (8-bit) content of the analog conversion data register or digital data register being sent to the microprocessor.
F. This is the 2-byte (16-bit) content of the analog conversion data register being sent to the microprocessor.

Fig. 11.50 Typical operating sequence

A typical operating sequence timing diagram is shown in Fig. 11.50.

A high level at the Read/Write input and a low level at the Chip Select input sets the device to output data on the 8-line data bus for the microprocessor. A low level at the Read/Write input and a low level at the Chip Select input sets the device to receive instructions into the internal control register on the 8-line data bus from the microprocessor. When the device is in the read mode and the Register Select input is at a low level, the processor will read the data contained in the analog conversion data register. However, when the Register Select input is at a high level, the processor reads the data contained in the digital data register.

The control register is a write-only register into which the microprocessor writes the command instructions for the device to start A/D conversion and to select the analog channel to be converted. The analog conversion data register is a read-only register that contains the current converter status and the most recent conversion results. The digital data register is also a read-only register that holds the digital input logic levels from the six dual-purpose inputs.

Internally each device contains a byte pointer that selects the appropriate byte during two cycles of the clock input in a normal 16-bit microprocessor instruction. The internal pointer will automatically point to the most significant byte any time that the Chip Select is at a high level for at least one clock cycle. This causes the device to treat the next signal on the 8-line data bus as the most significant (MS) byte. A low level at the Chip Select input activates the inputs and outputs and an internal function decoder. However, no data is transferred until the Clock goes high. The internal byte pointer first points to the MS byte of the selected register during the first clock cycle. After the first clock cycle in which the MS byte is accessed, the internal pointer switches to the least significant (LS) byte and remains there for as long as Chip Select is low. The MS byte of any register may be accessed by either an 8-bit or 16-bit microprocessor instruction; however, the LS byte may only be accessed by a 16-bit microprocessor instruction.

Normally, a two-byte word is written into or read from the controlling microprocessor, but a single byte can be read by the microprocessor by proper manipulation of the Chip Select input. This can be used to read conversion status from the analog conversion data register or the digital multipurpose input levels from the digital data register.

A conversion cycle is started after a two-byte instruction is written

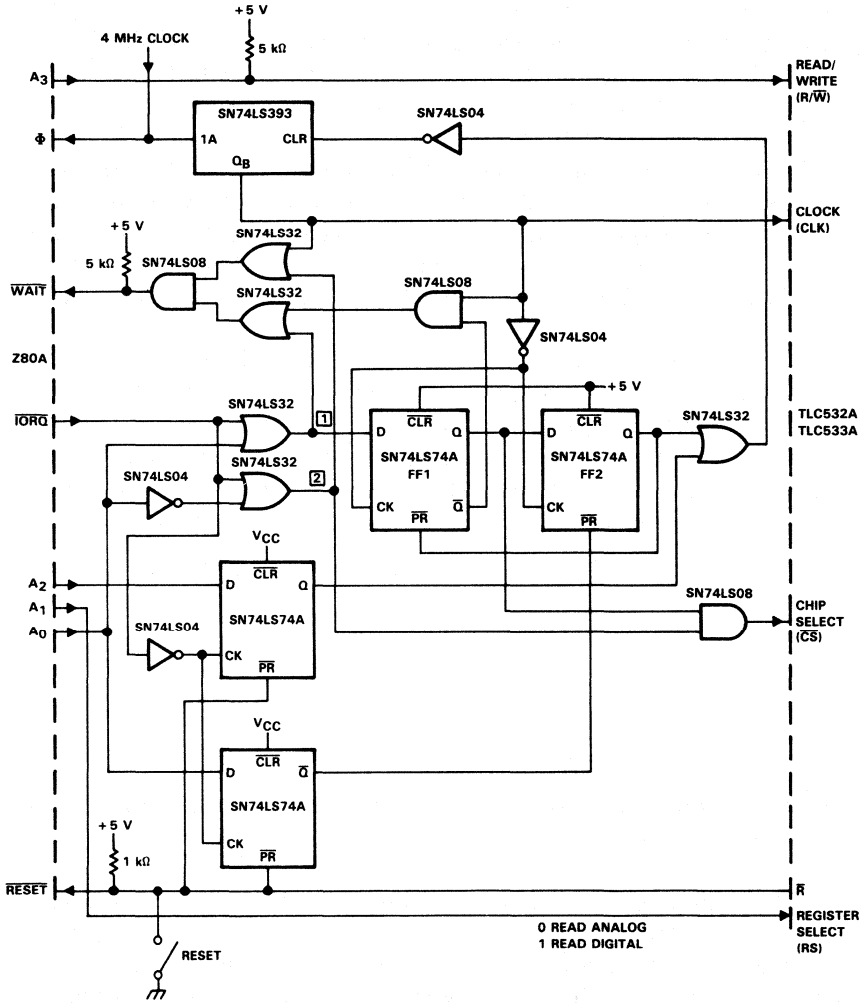
into the control register and the start conversion (SC) bit is a logic high. This two-byte instruction also selects the input analog channel to be converted. The end-of-conversion (EOC) status bit in the analog conversion data register is reset and it remains at that level until the conversion is completed, when the status bit is set again. After conversion, the results are loaded into the analog conversion data register. These results remain in the analog conversion data register until the next conversion cycle is completed. If a new conversion command is entered into the control register while the conversion cycle is in progress, the on-going conversion will be aborted and a new channel acquisition cycle will immediately begin.

The Reset input allows the device to be externally forced to a known state. When a low level is applied to the Reset input for a minimum of three clock periods, the start conversion bit of the control register is cleared. The A/D converter is then idled and all the outputs are placed in the high-impedance off-state. However, the content of the analog conversion data register is not affected by the Reset input going to a low level.

TLC532A and TLC533A A/D Converter Interface to Zilog Z80A and Z80 Microprocessors

The circuitry shown in Fig. 11.51 interfaces the Z80A microprocessor to the TLC532A A/D converter. This interface circuitry operates the TLC532A bus-oriented devices in an optimum fashion providing a quick read/write capability. It allows the microprocessor to use both one and two byte A/D converter read/write instructions. One byte A/D instructions can be used to quickly initiate conversion, by setting the SC bit, and to determine the end of conversion by monitoring the EOC bit.

The 4 MHz clock frequency is intended for use with a Z80A, a Z80 could also be used by reducing the clock frequency to 2 MHz. This change will double the A/D conversion time. The 4 MHz clock signal can be changed to any frequency with the proviso that the resulting A/D clock frequency, whose frequency is the microprocessor clock frequency divided by four, does not exceed the A/D upper clock frequency limit.



NOTE: Interface has been tested with TTL-LS parts only, however interface should work with HCMOS ICs.

Fig. 11.51 Z80A Microprocessor interface diagram, TLC532A, TLC533A

Input/Output Mapping

Table 11.8 TLC532A I/O map

		A_3	A_2	A_1	A_0	NO. OF BYTES COMMUNICATED
Control Register Write	MSB	0	0	1	0	2
	LSB	0	0	1	1	
	MSB	0	1	1	0	1
Analog Data Register Read	MSB	1	0	0	0	2
	LSB	1	0	0	1	
	MSB	1	1	0	0	1
Digital Data Register Read	MSB	1	0	1	0	2
	LSB	1	0	1	1	
	MSB	1	1	1	0	1

Table 11.8 shows the Input/Output (I/O) map which allows the microprocessor to write or read to the A/D converter in any of the possible modes. If other microcomputer I/O interfaces are required in addition to the A/D converter, the $\overline{\text{IORQ}}$ signal must be masked from the interface circuitry during non-A/D write or read operations. $\overline{\text{IORQ}}$ is the signal that drives the interface circuitry and causes a write or read to occur. Failure to mask the $\overline{\text{IORQ}}$ signal will allow the A/D converter and other I/O devices to access the microprocessor bus at the same time and cause a data collision. If a 2-byte write or read is desired, each of the two bytes must be written or read completely. Partial execution of a 2-byte write or read operation will leave chip select ($\overline{\text{CS}}$) low and prevent A/D conversion from taking place.

Although not tested, this interface circuitry might be memory mapped, rather than I/O mapped, by substituting MREQ in place of $\overline{\text{IORQ}}$.

Timing Diagram

Fig. 11.52 shows the timing diagram. The microprocessor's WAIT input is used to slow down the microprocessor to synchronize the A/D converter and microprocessor bus communications. Note that the A/D converter's chip select ($\overline{\text{CS}}$) is kept low during an entire 2-byte write or read cycle. If chip select ($\overline{\text{CS}}$) were allowed to go high between the 2-byte operations, the microprocessor would write or read the A/D converter's most significant byte twice.

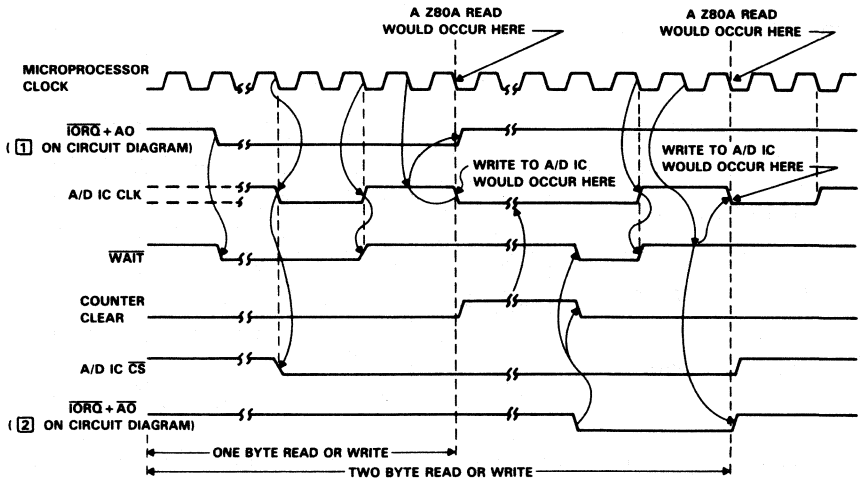


Fig. 11.52 Timing diagram for the Z80A/Z80 interface to the TLC532A, TLC533A

Software

The following software listings can be used for both A/D write and read operations. The first routine uses an initial 2-byte write to start the A/D conversion, to address analog input A0, to select a 1-byte read test loop or a delay to allow completion of the A/D conversion, and uses a 2-byte read to obtain the A/D conversion result in the least significant or second byte. Note that a 2-byte write or read is necessary to access the A/D converter's least significant byte.

```
                                ; Software for A/D Conversion Using the TLC532A,
                                  TLC533A
                                ;
0000                            ORG 00H             ;(used by cross-assembler)
0000    3E 01                   LD A,01H            ;01H into accumulator
0002    D3 02                   OUT (02H),A         ;write MSB (STRT CONV = 1)
0004    3E 00                   LD A,00H            ;00H into accumulator
0006    D3 03                   OUT (03H),A         ;write LSB (ADDR = 0)
                                ;
0008    DB 0C           TEST:   IN A,(OCH)          ;read MSB analog register
000A    F2 08 00                JP P,TEST           ;branch if not END OF CONVERSION = 0
                                ;
                                ;The above test loop can be substituted by any delay
                                ;which meets the A/D IC's time requirements for A/D
                                ;conversion.
                                ;
0000    DB 08                   IN A,(08H)          ;read analog register MSB
000F    DB 09                   IN A,(09H)          ;read analog register LSB,
                                                    ;conversion result is in accumulator
                                ;
                                ;
```

Two more listings present software routines that can be used to read the digital data registers in 2- and 1-byte formats, respectively. These read instructions do not have to be preceded by an A/D conversion delay period, since these registers provide digital information only.

```
                                ;
                                ; Software for Digital Data Register 2-Byte Read
                                ;
0100                            ORG 100H            ;(used by cross assembler)
0100    DB 0A                   IN A,(0AH)          ;fetch MSB
0102    47                      LD B,A              ;store MSB
0103    DB 0B                   IN A,(0BH)          ;fetch LSB, MSB is in B and
                                                    ;LSB is in A
                                ;
                                ;
-------------------------------------------------------------------------------
                                ;
                                ; Software for Digital Data Register 1-Byte Read
                                ;
0200                            ORG 200H            ;(used by cross assembler)
0200    DB 0E                   IN A,(0EH)          ;fetch MSB, MSB is in A
0202    47                      END                 ;(used by cross-assembler)
```

Table 11.9 A/D conversion times for the Z80A/Z80 interface to the TLC532A, TLC533A

	A/D IC CLOCK CYCLE'S REQUIRED FOR CONVESION (See Note 1)	CONVERSION TIME (μs) USING AN A/D IC CLOCK FREQUENCY OF 1 MHz (See Note 2)
TLC532A	29½	29.5
TLC533A	29½	29.5

NOTES: 1. Time 0 is the first rising edge of the first write cycle of a 2-byte write operation.
2. The interface circuitry will generate an A/D IC clock signal of 1 MHz when using a 4 MHz microprocessor clock signal.

Additional Comments

The time required for analog-to-digital conversion for the A/D converters in this application are presented in Table 11.9. It is necessary that the microprocessor allow these time periods for A/D conversion, either through a test loop, where the microprocessor reads the end of conversion (EOC) bit, or a delay period that may be obtained in any convenient manner by the microprocessor.

TLC532A and TLC533A A/D Converter Interface to Intel 8048, 8049, 8051 and 8052 Microcontrollers

This application covers all the possible interface combinations for these parts. These interfaces offer the following advantages:

1. Fast complete cycle times for loading the A/D address, performing conversion, and retrieving the conversion.
2. Flexible use of A/D converter pins for either A/D conversion or input of digital data
3. Low cost.

These Intel microcontroller families consist of the following:

INTEL 8048 and 8049 Family

8035AHL	8048AH	8748H
8039AHL	8049AH	8749H
8040AHL	8050AH	80C49

INTEL 8051 and 8052 Family

8031AH	8051AH	8751H
8032AH	8052AH	80C51

Because of differences in timing, two different circuit configurations are required for interfacing these A/D converters and microcontroller families. For reference, these two configurations are called Interface 1 and Interface 2.

Interface 1 is more desirable than Interface 2 since it does not constrain the microcontroller's data bus operations. It also allows the microcontroller to use the A/D converter software in external program memory. Unfortunately, different timing specifications prevent Interface 1 from being used for Interface 2 (8051 and 8052 to TLC533A interface).

Hardware–Interface 1

Fig. 11.53 shows the circuit configuration for Interface 1. This circuit can be used for the 8051 and 8052 devices interface to the TLC532A device, and the 8048 and 8049 devices interface to the TLC533A device.

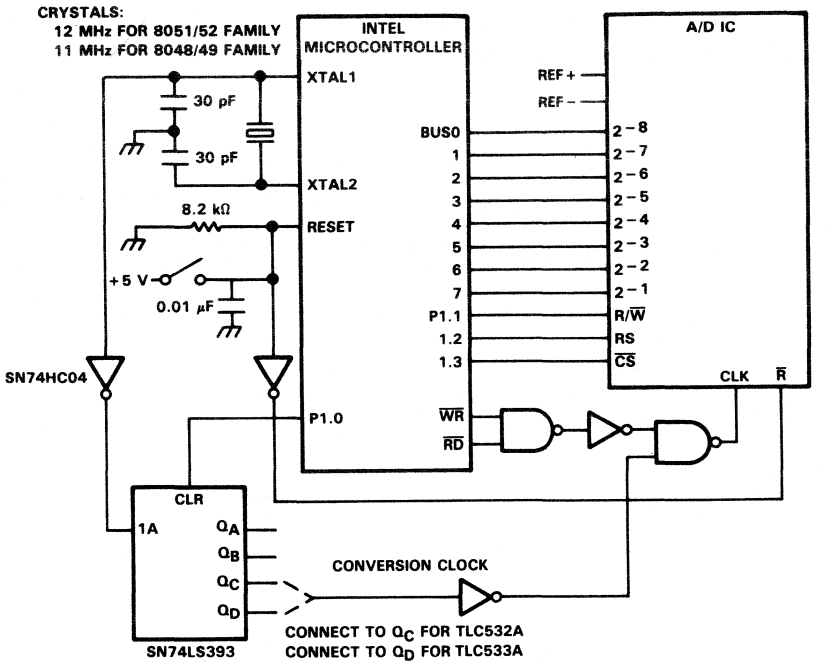


Fig. 11.53 Intel 8051/52 family to TLC532A interface 1 and intel 8048/49 family to TLC532A, and TLC533A interface 1

The system clock for the A/D converter is obtained from the microcontroller crystal oscillator. To assure proper operation of the crystal oscillator, a high impedance buffer must be used to prevent overloading the oscillator. An important design detail is that the low and high level input requirements of the buffer must lie within the range of the oscillator signal. This compatibility will prevent missing edge transitions in the A/D converter's system clock signal. The buffered crystal oscillator signal must be frequency divided to assure that the resulting system clock frequency does not exceed the upper frequency limit of the A/D converter. Any convenient divider circuitry may be used to accomplish this task.

Timing Diagram – Interface 1

The timing diagrams for the 8051, 8052 and 8048, 8049 devices to TLC532A converter interface are presented in Figs 11.54 and 11.55, respectively. The timing diagrams cover a complete conversion cycle and the reading of the digital data registers. In Fig. 11.55, the required number

of conversion clocks will increase for the TLC533A device due to a lower clock frequency operating range.

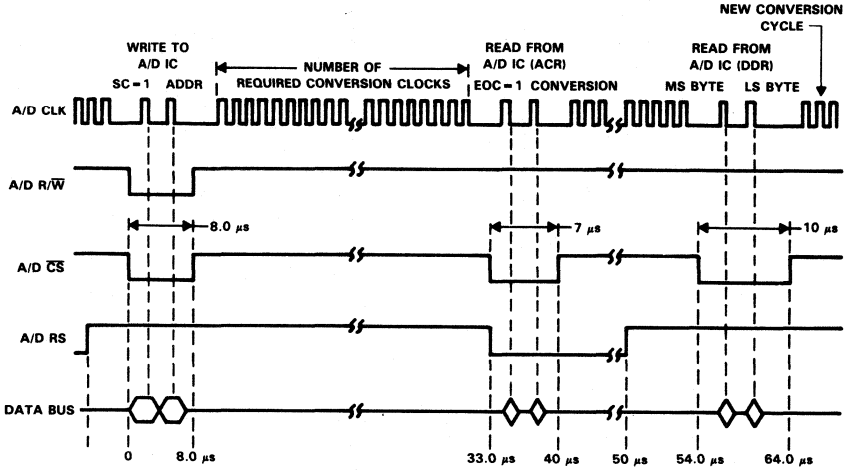


Fig. 11.54 Timing diagram of A/D conversion cycle for Intel 8051/52 family – TLC532A interface 1

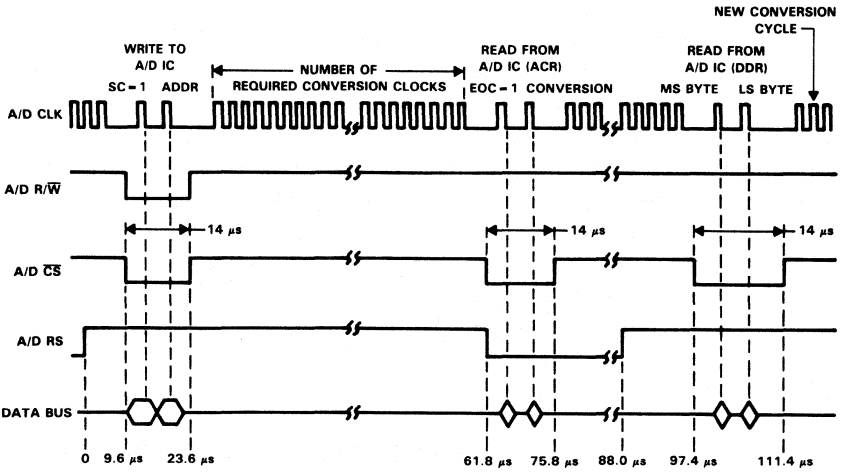


Fig. 11.55 Timing diagram of A/D conversion cycle for Intel 8048/49 family – TLC532A

Software – Interface 1

The software listing for the interface between the 8051 and 8052 devices and the TLC532A device follows. The software listing for the 8048 and 8049 to the TLC532A, and TLC533A interface is also included. The conversion software program or the digital data register software program can be incorporated into a subroutine so the designer can easily access the

```
                        ; Software for Intel 8051/52 Family to
                        ; TLC532A Interface 1
                        ;
                        ;               Software for Conversion
                        ;
0000                                    ORG 000H
0000    43 90 0C                        ORL P1,#0CH         ;CS, RS = 1
0003    C2 90                           CLR P1.0            ;Counter clear = 0
0005    7F 01                           MOV R7,#01H         ;Do delay to allow 3 system clocks
0007    DF FE           D1CSEQ1:        DJNZ R7,D1CSEQ1     ;to occur so A/D IC will
                                                            ;recognize CS = 1
0009    B2 90                           CPL P1.0            ;Counter clear = 1
000B    53 90 FS                        ANL P1,#F5H         ;CS, R/W(bar) = 0
000E    74 01                           MOV A,#01H          ;Write MSByte (Set SC = 1)
0010    F2                              MOVX @R0,A
0011    EA                              MOV A,R2            ;Get A/D IC address, which is
                                                            ;assumed to be in R2
0012    F2                              MOVX @R0,A          ;Write LSByte or A/D IC address
0013    43 90 0A                        ORL P1,#0AH         ;CS, R/W(bar) = 1
0016    C2 90                           CRL P1.0            ;Counter clear = 0
                        ;
                        ;               A delay must occur here to allow the A/D IC to
                        ;               complete conversion. The delay must allow at least
                        ;               29.5 A/D IC system clock cycles to occur.
                        ;
0018    B2 90                           CPL P1.0            ;Counter clear = 1
001A    53 90 F3                        ANL P1,#F3H         ;CS, RS = 0
001D    E2                              MOVX A,@R0          ;Read Analog Conversion Register MSByte
001E    E2                              MOVX A,@R0          ;Read Analog Conversion Register LSByte
                                                            ;or conversion result
001F    FA                              MOV R2,A            ;Store conversion result
0020    B2 93                           CPL P1.3            ;CS = 1
0022    C2 90                           CLR P1.0            ;Counter clear = 0
                        ;
-------------------------------------------------------------------------------------------------------
                        ; Software for Reading Digital Data Registers
                        ;
0100                                    ORG 0100H
0100    43 90 0E                        ORL P1,#0EH         ;CS, RS, R/W(bar) = 1
0103    C2 90                           CLR P1.0            ;Counter clear = 0
0105    7F 01                           MOV R7,#01          ;Do delay to allow 3 system clocks
0107    DF FE           D2CSEQ1:        DJNJ R7,D2CSEQ1     ;to occur so A/D IC will
                                                            ;recognize CS = 1
0109    B2 90                           CPL P1.0            ;Counter clear = 1
010B    C2 93                           CLR P1.3            ;CS = 0
010D    E2                              MOVX A,@R0          ;Read Digital Data Register MSByte
010E    FA                              MOV R2,A            ;Store Digital Data Register MSByte
010F    E2                              MOVX A,@R0          ;Read Digital Data Register LSByte
0110    FB                              MOV R3,A            ;Store Digital Data Register LSByte
0111    B2 93                           CPL P1.3            ;CS = 1
0113    C2 90                           CLR P1.0            ;Counter clear = 0
0115                                    END
```

software with a simple subroutine call. Also, the conversion software assumes that the A/D converter address has been placed in register R2 and, upon completion of A/D conversion, stores the conversion result in register R2. The digital data register software stores the most significant (MS) byte and the least significant (LS) byte in registers R2 and R3, respectively.

```
                        ; Software for Intel 8048/49 Family —
                        ; TLC532A, and TLC533A Interface 1
                        ;
;                               Software for Conversion
                        ;
                                ORG 000H
0000    89 0C                   ORL P1,#0CH         ;CS, RS = 1
0002    99 FE                   ANL P1,#FEH         ;Counter clear = 0
0004    BF 01                   MOV R7,#01H         ;Do delay to allow 3 system clocks
0006    EF 06           D1CSEQ1: DJNZ R7,D1CSEQ1    ;to occur so A/D IC will
                                                    ;recognize CS = 1
0008    89 01                   ORL P1,#01H         ;Counter clear = 1
000A    99 F5                   ANL P1,#F5H         ;CS, R/W(bar) = 0
000C    23 01                   MOV A,#01H          ;Write MSByte (Set SC = 1)
000E    90                      MOVX @R0,A
000F    FA                      MOV A,R2            ;Get A/D IC address, which is
                                                    ;assumed to be in R2
0010    90                      MOVX @R0,A          ;Write LSByte or A/D IC address
0011    89 0A                   ORL P1,#0AH         ;CS, R/W(bar) = 1
0013    99 FE                   ANL P1,#FEH         ;Counter clear = 0
                        ;
                        ;       A delay must occur here to allow the A/D IC to
                        ;       complete conversion. The delay must allow at least
                        ;       29.5 A/D IC system clock cycles to occur for the
                        ;       TLC532A and TLC533A.
                        ;
0015    89 01                   ORL P1.#01H         ;Counter clear = 1
0017    99 F3                   ANL P1,#F3H         ;CS, RS = 0
0018    80                      MOVX A,@R0          ;Read Analog Conversion Register MSByte
0019    80                      MOVX A,@R0          ;Read Analog Conversion Register LSByte
                                                    ;or conversion result
001A    AA                      MOV R2,A            ;Store conversion result
001B    89 08                   ORL P1,#08H         ;CS = 1
001D    99 FE                   ANL P1,#FEH         ;Counter clear = 0
                        ;
-------------------------------------------------------------------------------------------
                        ;       Software for Reading Digital Data Registers
                        ;
                                ORG 0100H
0100    89 0E                   ORL P1,#0EH         ;CS, RS, R/W(bar) = 1
0102    99 FE                   ANL P1,#FEH         ;Counter clear = 0
0104    BF 01                   MOV R7,#01          ;Do delay to allow 3 system clocks
0106    EF 06           D2CSEQ1: DJNZ R7,D2CSEQ1    ;to occur so A/D IC will
                                                    ;recognize CS = 1
0108    89 01                   ORL P1,#01H         ;Counter clear = 1
010A    99 F7                   ANL P1,#F7H         ;CS = 0
010C    80                      MOVX A,@R0          ;Read Digital Data Register MSByte
010D    AA                      MOV R2,A            ;Store Digital Data Register MSByte
010E    80                      MOVX A,@R0          ;Read Digital Data Register LSByte
010F    AB                      MOV R3,A            ;Store Digital Data Register LSByte
0110    89 08                   ORL P1,#08H         ;CS = 1
0112    99 FE                   ANL P1,#FEH         ;Counter clear = 0
                                END
```

Hardware – Interface 2

Fig. 11.56 presents the circuit configuration for Interface 2. This circuit can be used for the 8051 and 8052 devices interface to TLC533A. The A/D converter's data port is shown connected to port 3 of the 8051/2, the appropriate data books should be consulted if it is required to interface with other ports.

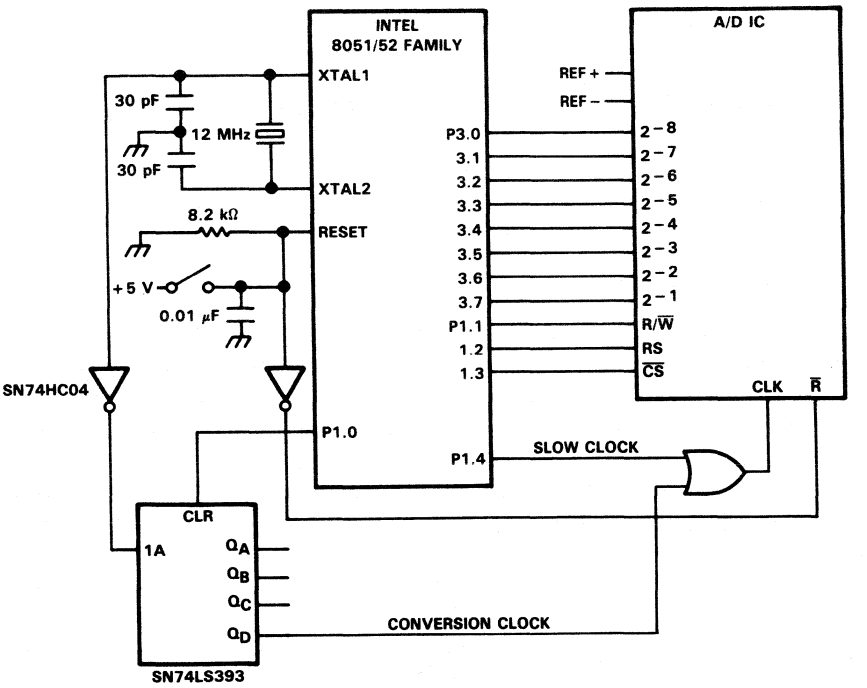


Fig. 11.56 Intel 8051/52 family – TLC533A interface 2

Timing Diagram – Interface 2

The timing diagram for the 8051 and 8052 devices interface to the TLC533A device is presented in Fig. 11.57. The timing diagram covers a complete conversion cycle and the reading of the digital data registers.

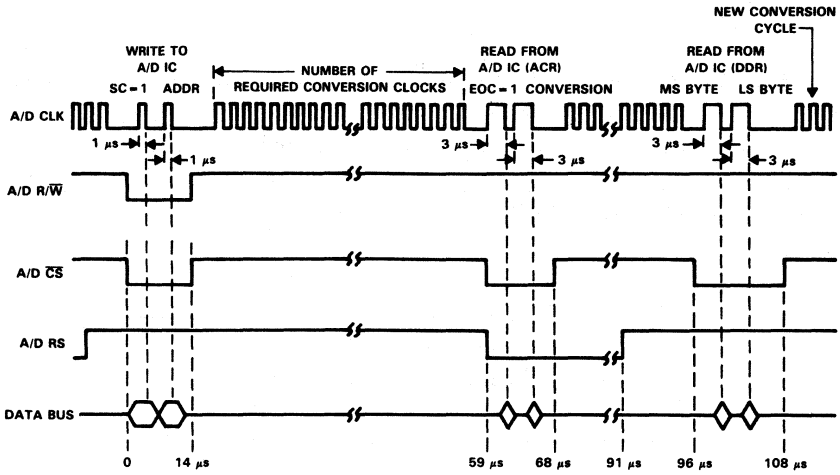


Fig. 11.57 Timing diagram of A/D conversion cycle for Intel 8051/52 family to TLC533A interface 2

Software–Interface 2

The software listings for Interface 2 follow. Unlike Interface 1, the microcontroller write (WR) and read (RD) pins are not used to generate the A/D converter clock during write and read operations. Instead, the clock is generated by toggling tye P1·4 pin. This change between Interface 1 and Interface 2 was made to accommodate the different timing specifications of the 8051 and 8052 devices and the TLC533A converter. (See the Interface 1 paragraph for further description of the software listings.)

```
                        ; Software Conversion for Intel 8051/52
                        ; Family — TL530, TL531, TLC533A Interface 2
                        ;
0000                            ORG 000H
0000   43 90 0C                 ORL P1,#0CH      ;CS, RS = 1
0003   53 90 EE                 ANL P1,#EEH      ;Counter clear, Slow clock = 0
0006   7F 02                    MOV R7,#02H      ;Do delay to allow 3 system clocks
0008   DF FE            D1CSEQ1: DJNZ R7,D1CSEQ1  ;to occur so A/D IC will
                                                 ;recognize CS = 1
000A   B2 90                    CPL P1.0         ;Counter clear = 1
000C   53 90 F5                 ANL P1,#F5H      ;CS, R/W(bar) = 0
000F   75 B0 01                 MOV P3,#01H      ;Write MSByte (Set SC = 1) to Port 3
0012   B2 94                    CPL P1.4         ;Raise slow clock
0014   B2 94                    CPL P1.4         ;Lower slow clock
0016   8A B0                    MOV P3,R2        ;Write A/D IC address, which is
                                                 ;assumed to be in R2, to Port 3
0018   B2 94                    CPL P1.4         ;Raise slow clock
001A   B2 94                    CPL P1.4         ;Lower slow clock
001C   75 B0 FF                 MOV P3,#FFH      ;Put Port 3 pins to input mode
001F   43 90 0A                 ORL P1,#0AH      ;CS, R/W(bar) = 1
0022   C2 90                    CLR P1.0         ;Counter clear = 0
                        ;
```

```
                        ;               A delay must occur here to allow the A/D IC to
                        ;               complete conversion. The delay must allow at least
                        ;               29.5 A/D IC system clock cycles to occur for the
                        ;               TLC533A. Further the frequency of the clock signal
                        ;               must not exceed the specification for the A/D IC.
                        ;
0024    B2 90                           CPL P1.0        ;Counter clear = 1
0026    53 90 F3                        ANL P1,#F3H     ;CS, RS = 0
0029    B2 94                           CPL P1.4        ;Raise slow clock
002B    AA B0                           MOV R2,P3       ;Read MSByte of the analog
                                                        ;conversion register
002D    B2 94                           CPL P1.4        ;Lower slow clock
002F    B2 94                           CPL P1.4        ;Raise slow clock
0031    AA B0                           MOV R2,P3       ;Read and store LSByte of the analog
                                                        ;conversion register or conversion
0033    B2 94                           CPL P1.4        ;Lower slow clock
0035    B2 93                           CPL P1.3        ;CS = 1
0037    C2 90                           CLR P1.0        ;Counter clear = 0
                        ;
                        ;
-------------------------------------------------------------------------------------------------------------------------
                        ;
                        ;
                        ; Software for Reading Digital Data Registers
                        ;
0100                                    ORG 0100H
0100    43 90 0E                        ORL P1,#0EH     ;CS, RS, R/W(bar) = 1
0103    53 90 EE                        ANL P1,#EEH     ;Counter clear, Slow clock = 0
0106    7F 02                           MOV R7,#02      ;Do delay to allow 3 system clocks
0108    DF FE           D2CSEQ1:        DJNZ R7,D2CSEQ1 ;to occur so A/D IC will
                                                        ;recognize CS = 1
010A    B2 90                           CPL P1.0        ;Counter clear = 1
010C    75 B0 FF                        MOV P3,#FFH     ;Put Port 3 pins to input mode
010F    C2 93                           CLR P1.3        ;CS = 0
0111    B2 94                           CPL P1.4        ;Raise slow clock
0113    AA B0                           MOV R2,P3       ;Read and store MSByte of the digital
                                                        ;data register
0115    B2 94                           CPL P1.4        ;Lower slow clock
0117    B2 94                           CPL P1.4        ;Raise slow clock
0119    AB B0                           MOV R3,P3       ;Read and store LSByte of the digital
                                                        ;data register
011B    B2 94                           CPL P1.4        ;Lower slow clock
011D    B2 93                           CPL P1.3        ;CS = 1
011F    C2 90                           CLR P1.0        ;Counter clear = 0
0121                                    END
```

Additional Comments

The A/D converter clock signal should not be interrupted during A/D conversion because of the possible loss of electrical charge on the A/D converter's internal capacitors. Interruption of the clock signal during an A/D converter write or read operation presents no problem. All of the software offered for use with these interfaces complies with the above requirement since the conversion clock is not stopped until after the A/D conversion is completed.

TLC532A and TLC533A A/D Converter Interface to Rockwell 6502 Microprocessors

This application describes techniques for interfacing the TLC532A and TLC533A converter devices to the Rockwell 6502 microprocessor. The TLC532A family of A/D converters consists of chips offering combinations of conversion speeds and numbers of analog inputs. It is important that SN74S00 or SN74AS00 NAND gates be used to guarantee operation of this interface circuit.

Principles of Operation

The TLC532A device to 6502 device interface circuit and timing diagrams are shown in Figs 11.58 and 11.59, respectively. A listing of the control software necessary to operate the interface follows.

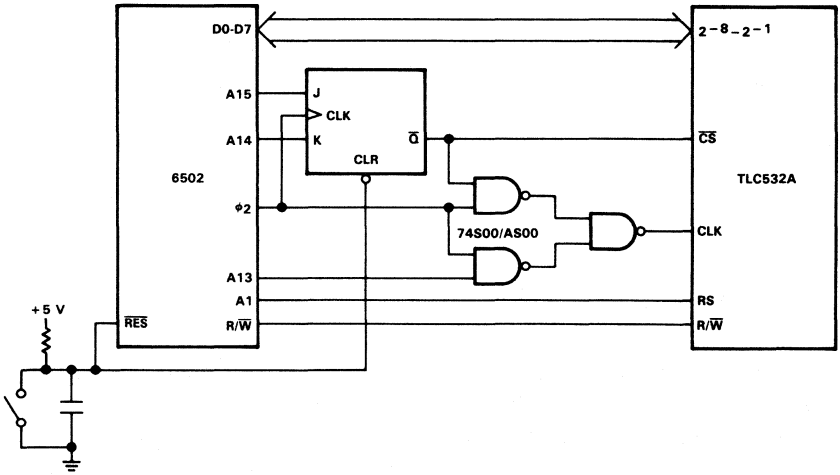


Fig. 11.58 6502 – TLC532A interface circuit diagram

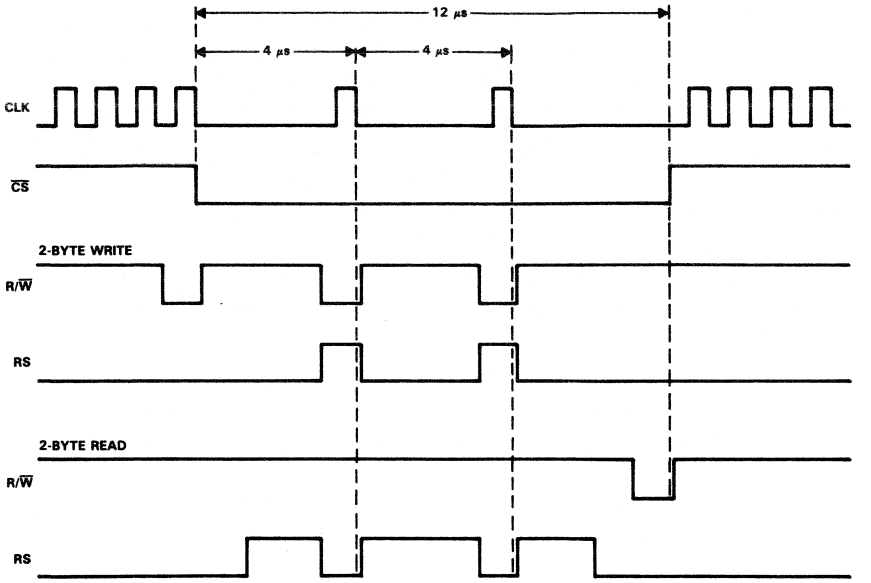


Fig. 11.59 6502 – TLC532A interface timing diagram

```
;
;ADDRESS ASSIGNMENTS
;
ANALYZER:   .EQU 0000H
ACMSB:      .EQU 2400H
ACLSB:      .EQU 2401H
CRMSB:      .EQU 2402H
CRLSB:      .EQU 2403H
DDMSB:      .EQU 2502H
DDLSB:      .EQU 2503H
SETCS:      .EQU 4400H
CLRCS:      .EQU 8400H
;
;
;
            LDX #$01        ; SET SC BIT
            LDA #$00        ; SELECT MUXADDRESS
            STA CLRCS       ; BRING CHIP SELECT LOW
            STX CRMSB       ; LOAD CONTROL REG MSB
            STA CRLSB       ; LOAD CONTROL REG LSB, START CONV.
            LDA SETCS       ; BRING CHIP SELECT HIGH
            LDY #$07        ;
LOOP:       DEY             ; DELAY UNTIL CONVERSION COMPLETE
            BNE LOOP        ;
            LDA CLRCS       ; BRING CHIP SELECT LOW
            LDA ACMSB       : READ CONV. REG MSB (EOC BIT)
            LDA ACLSB       ; READ CONV. REG LSB (RESULT)
            STA SETCS       ; BRING CHIP SELECT HIGH
```

A conversion cycle begins by bringing chip select ($\overline{CS}$) low. This is accomplished by clocking a high into the J input of the flip-flop on the trailing edge $\phi 2$, bringing $\overline{Q}$ low. A low on $\overline{Q}$ also inhibits the $\phi 2$ clock from reaching the TLC532A device. When the first byte of a 2-byte write cycle is being written, address bit 13 goes high. This allows one $\phi 2$ pulse to reach the TLC532AD device and clock in the byte. The write cycle for the second byte is exactly the same as the first. Chip select ($\overline{CS}$) is then brought high by clocking a high into the K input of the flip-flop to bring $\overline{Q}$ high again. Analog-to-digital conversion requires 29 clock cycles. Therefore, a delay loop is included in the software listing to allow for this conversion time.

The 2-byte read cycle operation is similar to that of the write cycle since it uses address bit A13 as a gating signal for the clock input to the TLC532A device. One- or 2-byte reads of either the analog conversion registers or the digital data registers may be performed with the proper register select signals. The software listing shows the necessary software for a 2-byte read of the analog conversion register, and leaves the A/D conversion result in the accumulator. One complete data acquisition cycle can be performed every 66 microseconds.

TLC532A and TLC533A A/D Converter Interface to Motorola 6800, 6802, 6809 and 6809E Microprocessors

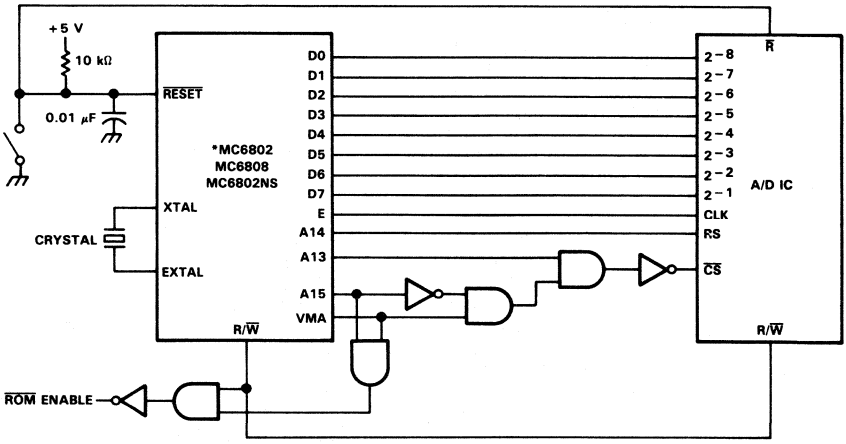
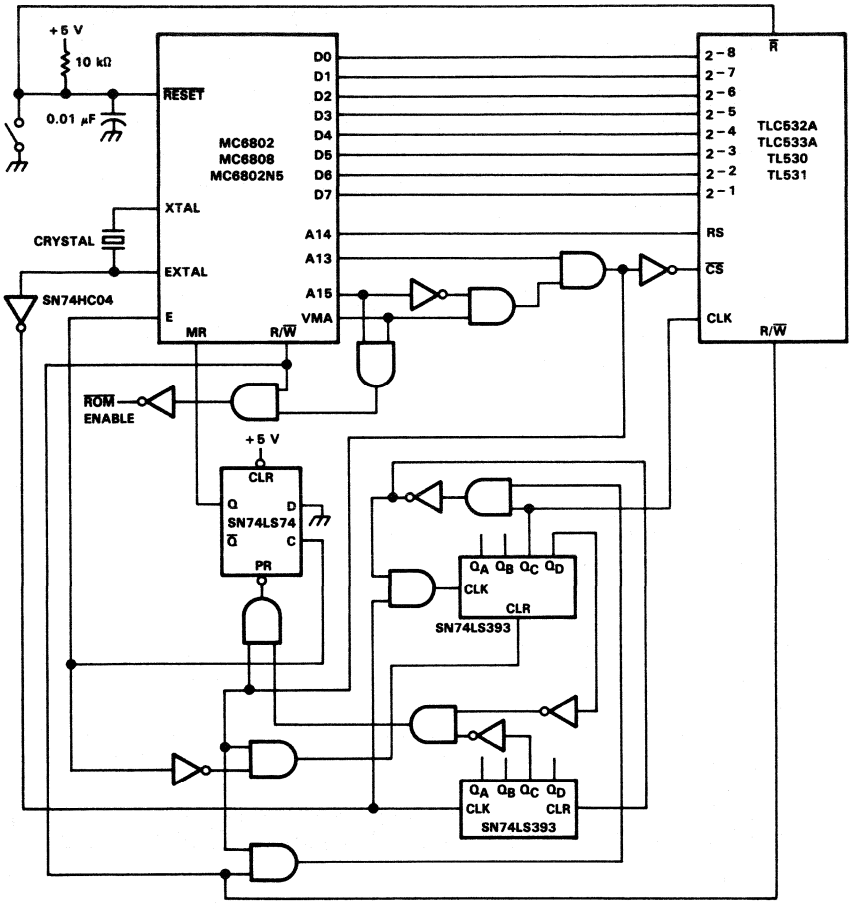


Fig. 11.60 Interface 1 hardware circuitry

Two interface circuits are presented: Interface 1 in Fig. 11.60 and Interface 2 in Fig. 11.61. Interface 1 is used with microprocessors with lower clock frequencies, both provide 1- and 2-byte write read communications capability between the microprocessor and the A/D.



*See Table 11-43 for information about other microprocesor configurations.

Fig. 11.61 Interface 2 hardware circuitry

Only the 2-byte capability is demonstrated in this application. To use the 1-byte capability of the A/D converter, 1-byte external memory write/read instructions can be used.

Input/Output Mapping

Table 11.10 I/O map

	A15	A14	A13*	R/$\overline{W}$
Control Register Write	0	1	1	0
Analog Data Register Read	0	0	1	1
Digital Data Register Read	0	1	1	1
ROM Program	1	X	X	X

*A13 = 1 to prevent the 6802 from reading its internal RAM and the A/D IC simultaneously.

Table 11.10 presents the I/O map that allows the microprocessor to write or read to the A/D converter. This I/O map is intended as a guide and does not preclude other mapping possibilities.

Timing Diagram

A timing diagram for Interface 1 with a 1 MHz microprocessor clock cycle is shown in Fig. 11.62.

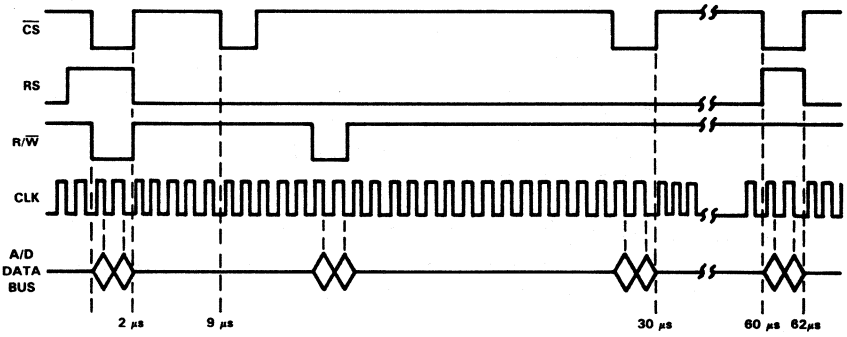


Fig. 11.62 Timing diagram (interface 1 and 6802 microprocessor with a 1 MHz clock cycle)

The microprocessor write and read timing diagrams for Interface 2 are shown in Figs 11.63 and 11.64.

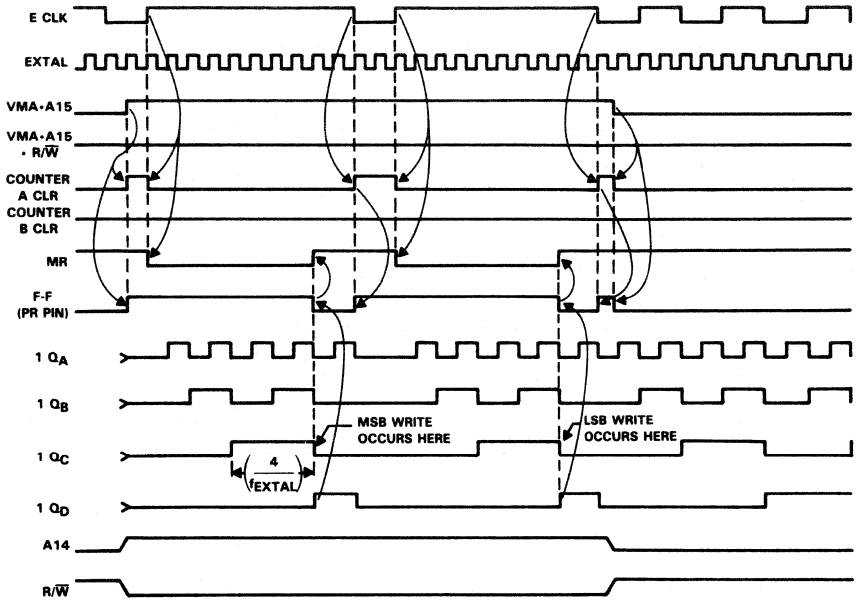


Fig. 11.63 Interface 2 microprocessor write timing diagram

Table 11.11 A/D conversion time

A/D IC	A/D IC CLOCK CYCLES REQUIRED FOR CONVERSION (See Note)
TLC532A	29½
TLC533A	29½

NOTE: Time 0 is the first rising edge of the first write cycle of a 2-byte write operation.

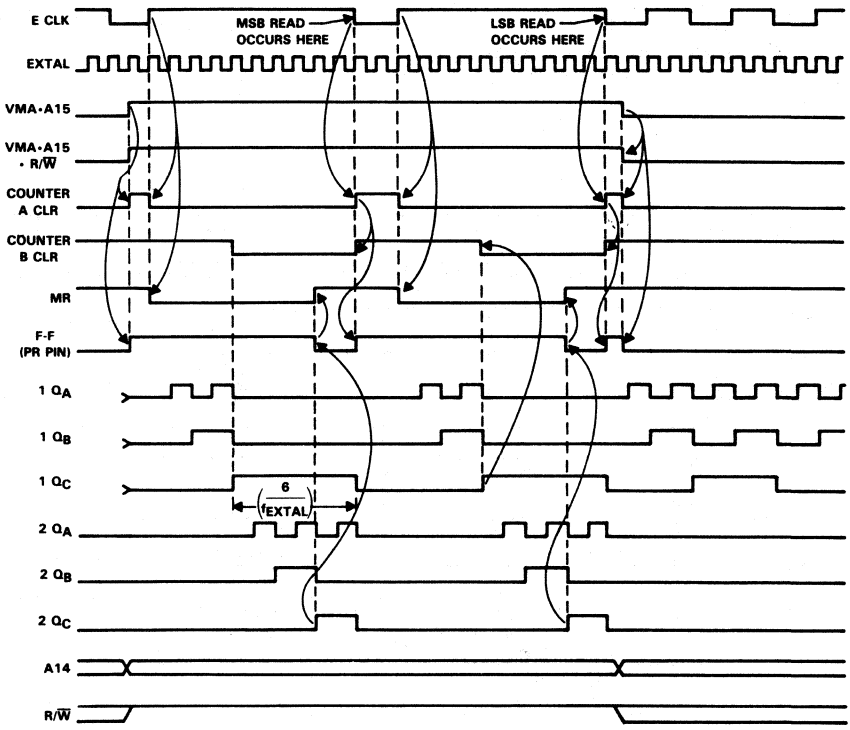


Fig. 11.64 Interface 2 microprocessor read timing diagram

Software

The following software listings can be used for both interface circuits. These listings show the software for conversion and for reading the digital data registers, respectively. For conversion, the software must determine the end of conversion by monitoring the end of conversion bit (EOC) in the most significant (MS) byte of the analog conversion register or by allowing a delay so the A/D converter can complete conversion. The length of the delay can be computed by referring to Table 11.11. This table shows the number of clock cycles required for each A/D converter to complete conversion.

```
                    ; Software for 6800 Family —
                    ; TLC532A, TLC533A Interface
                    ;
                    ;              Software for Conversion
                    ;
07FE  80                                        ;Microprocessor reset, branch
07FF  00                                        ;to 8000H. 2716 UVPROM was used
                                                ;for this test.
8000                        ORG 8000H           ;A15 = 1 enables ROM memory
8000  CE 01 05        LDX  #0105H               ;SC = 1 in MSB, Address analog
                                                ;channel #5 in LSB
8003  FF 60 00        STX $6000H                ;Write MSB and LSB to A/D IC
8006  01        STILLC: NOP                     ;Allow I A/D clock cycle to
8007  01              NOP                       ;occur so that A/D recognizes
                                                ;CS[5]1
8008  FE 20 00        LDX $2000H                ;Read MSB and LSB from Analog
                                                ;Conversion Register
800B  DF 00           STX $0000H                ;Store MSB & LSB in 0000H & 0001H
                                                ;RAM memory
800D  96 00           LDAA $0000H               ;Load A with MSB
800F  85 80           BITA $80H                 ;If EOC bit = 0; set microprocessor
                                                ;condition code Z bit
8011  27 F3           BEQ STILLC                ;If EOC = 0; branch and re-test EOC bit.
                                                ;If EOC = 1; conversion result is in
                                                ;the LSB of the X register.
                    ;
                    ;The loop within the statements STILLC: NOP and BEQ STILLC can
                    ;be replaced with a delay, which can be obtained in any way
                    ;which is convenient to the designer. After the delay, the
                    ;software must execute a LDX $2000H instruction and the con-
                    ;version result will lie in the LSB of the X register. Table 11.11
                    ;shows the delays which are required for each A/D IC.
                    ;              Software for Reading Digital Data Registers
                    ;
8013  FE 60 00        LDX $6000H                ;Read MSB and LSB from Digital Data
                                                ;Register. These bytes are now in the
                                                ;X register.
8016                        END
```

Considerations for Different Microprocessors

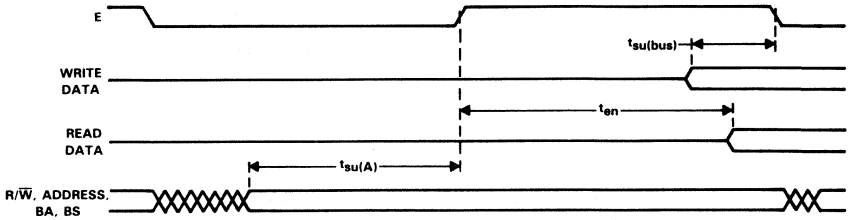
Table 11.12 Considerations for using other microprocessors

MICROPROCESSOR	CRYSTAL OSCILLATOR AND E PIN	SLOWING THE MICROPROCESSOR*
6800	ϕ_2 is equivalent to the 6802/6809 E pin	Need an external circuit to slow the microprocessor
6802/6808/6802	See Fig. 11.60	See Fig. 11.60
6809	See Fig. 11.60	See Fig. 11.60 Substitute MRDY for MR
6809E/68HC09E	See Fig. 11.60	Need an external circuit to slow the microprocessor

*Applies to Interface 2 only.

Table 11.12 lists the factors to be considered when using the Motorola microprocessors. The crystal oscillator and E pin considerations in column 1 apply to both Interface 1 and 2. Considerations such as slowing the microprocessor (in column 2) apply only to Interface 2. All of the microprocessors can be easily adapted to Interface 1. Table 11.12 and Fig. 11.63 enable the designer to readily adapt all of the microprocessors except the 6800 to Interface 2. For the 6800 device with a clock frequency greater than 1 MHz to the A/D converter, an external clock generator which can be slowed down must be used.

Interface Selection



	$t_{su(bus)}$ (ns–MIN)	t_{en} (ns–MAX)	$t_{wH(CLK)}$ (ns–MIN)	$t_{su(A)}$ (ns–MIN)	
TLC532A	140	250	230	100	A/D IC SPECIFICATIONS
TLC533A	185	335	440	145	
6800	225	300	450	180	COMPARABLE MICROPROCESSOR SPECIFICATIONS WITH THE ASSUMPTION THAT THE MICROPROCESSOR CLOCK CYCLE IS THE FASTEST POSSIBLE.
68A00	80	170	280	100	
68B00	60	140	220	85	
6802	225	350	450	160	
68A02	110	210	280	100	
68B02	50	150	220	50	
6809	475	370	450	265	
68A09	280	220	280	165	
68B09	220	180	220	120	

Fig. 11.65 Timing considerations

In general, microprocessors whose clock frequency is less than or equal to 1 MHz can use Interface 1. The TLC533A device specifications are very closely matched but not completely with 1 MHz microprocessor operation and Interface 1. The TLC532A device works satisfactorily with 1 MHz microprocessor operation and Interface 1, although even faster microprocessor clock speeds can be used. If the 68AOX and 68BOX microprocessors are run at full speed, Interface 2 must be used to assure matching between A/D converter and microprocessor specifications. Interface 2 will work with a microprocessor clock frequency that is greater than 2 MHz; however, 2 MHz is generally the upper limit for Motorola microprocessors. Fig. 11.65 shows the pertinent timing considerations.

Additional Comments

Although Interface 2 appears complicated, an examination of its write and read timing diagrams in Figs. 11.63 and 11.64 reveal the simple twofold strategy of the interface design. First, the lowering of the clock signal during an A/D write causes the microprocessor to leave its temporarily frozen or halted state. This action assures that the necessary data is written into the A/D converter before the microprocessor continues. Thus, a write is guaranteed. Second, the microprocessor's reading of the A/D converter must occur before the clock signal is lowered. Thus, a read is guaranteed. This inexpensive interface allows the designer to take full advantage of these flexible A/D devices.

TLC540 SERIES OF 8-BIT ANALOG-TO-DIGITAL PERIPHERALS WITH SERIAL INPUT/OUTPUT

The TLC540 series of LinCMOS A/D peripheral devices are a family which feature serial data communication and control. Each one is a complete data acquisition system on a single chip that includes a switched-capacitor successive-approximation A/D converter, a software controllable sample and hold, an analog multiplexer and a built-in self test. The 8-bit device types have a maximum total unadjusted error of ±0.5 LSB, with 11-analog inputs for TLC540 and TLC541, 19-analog inputs for TLC545 and TLC546 and a single analog input for the TLC548 and TLC549. The 10-bit devices TLC1540 and TLC1541 have 11-analog inputs and maximum total unadjusted errors of ±0.5 and 1 LSB respectively.

TLC540 and TLC541 8-Bit Analog to Digital Converters with 11-Inputs and Serial Input/Output.

The TLC540 and TLC541 are LinCMOS A/D peripherals built around an 8-bit switched capacitor successive-approximation A/D converter. They are designed for serial interface to a microprocessor or peripheral via a three-state, Data Out, output with four control lines: System Clock, I/O

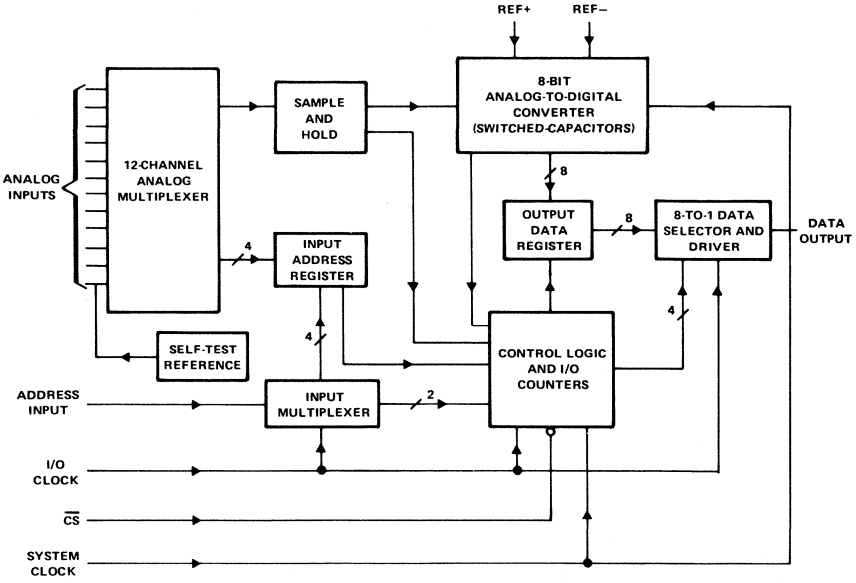


Fig. 11.66 TLC540/541 functional block diagram

Clock, Chip Select and Address Input. The system clock, which drives the A/D and sets conversion time, has maximum values of 4 MHz for the TLC540 and 2.1 MHz for the TLC541. Fig. 11.66 shows a functional block diagram.

Each device features high-speed data transfer and sample rates through the use of simultaneous read, write and analog sampling operations. In addition, the on-chip 12 channel analog multiplexer can be used to sample any one of the 11 inputs or an internal self test voltage and the sample and hold can operate automatically or under processor control. Maximum sample rates are 75,188 samples per second for the TLC540, and 40,000 samples per second for the TLC541.

The A/D converter incorporated in the TLC540 and TLC541 has differential high impedance reference inputs that facilitate ratiometric conversion, scaling and isolation of analog circuitry from logic supply and ground noise. The switched capacitor design allows a guaranteed low-error of ±0.5 LSB over the full operating temperature range. With TLC540 a conversion is completed in 9 μs and using careful interface timing complete input-conversion-output cycles can be repeated every 14 μs. For TLC541 conversion takes 17 μs, while complete input-conversion-output can be repeated in 25 μs, see Table 11.13.

Table 11.13

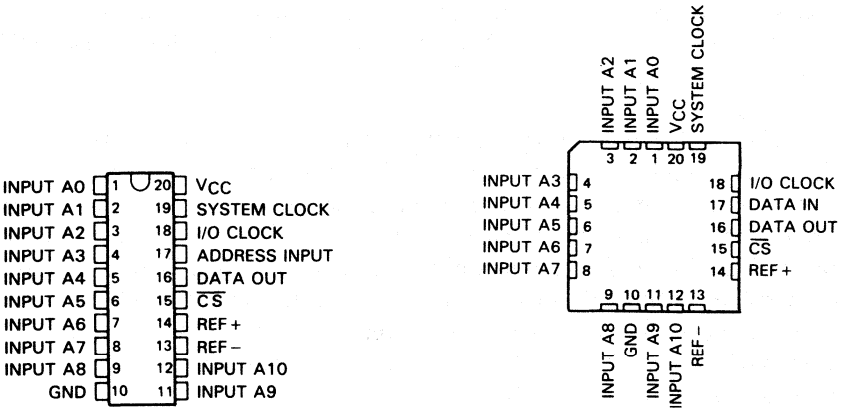
	TLC540	**TLC541**
Acquisition Time (max)	2 μs	3.6 μs
Conversion Time (max)	9 μs	17 μs
Sampling Rate (max)	75 × 10^3sps	40 × 10^3sps
Power Dissipation (typ)	6 mW	6 mW

The pin layout is shown in Fig. 11.67.

TLC540, TLC541 Operation

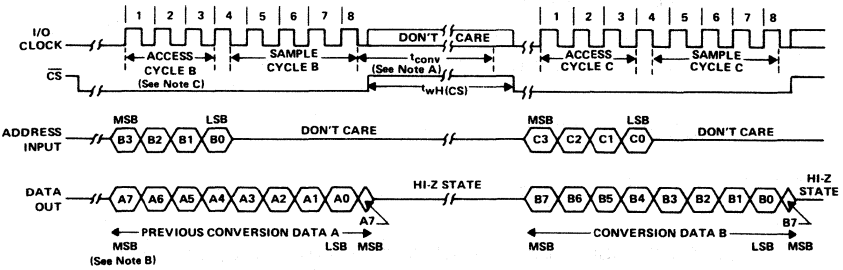
For flexibility and access speed the four control inputs (System Clock, I/O Clock, Chip Select ($\overline{CS}$) and Address Input) and a TTL compatible three-state, Data Out, output are intended for serial communication with a microprocessor or microcomputer. The System and I/O clocks are normally used independently and do not require any special speed or phase relationships between them. This independence simplifies the hardware and software control tasks for the device. Once a clock signal within the

specification range is applied to the system clock input, the control hardware and software need only be concerned with addressing the desired analog channel, reading the previous conversion result, and starting the conversion by using the I/O clock. The System clock will drive the "conversion crunching" circuitry so that the control hardware and software need not be concerned with this task. See Figure 11.68 for Operating Sequence Timing.



(a) N DUAL-IN-LINE PACKAGE (b) FN CHIP-CARRIER PACKAGE

Fig. 11.67 TLC540 and TLC541 pinouts (top view)



NOTES: A. The conversion cycle, which requires 36 internal system clock periods, is initiated with the 8th I/O clock pulse trailing edge after $\overline{CS}$ goes low for the channel whose address exists in memory at the time.

B. The most significant bit (MSB) will automatically be placed on the DATA OUT bus after $\overline{CS}$ is brought low. The remaining seven bits (A6-A0) will be clocked out on the first seven I/O clock falling edges.

C. To minimize errors caused by noise at the $\overline{CS}$ input, the internal circuitry waits for three internal system clock cycles (1.4 μs at 2 MHz) after a chip select transition before responding to control input signals. Therefore, no attempt should be fmade to clock-in address data until the minimum chip-select setup time has elapsed.

Fig. 11.68 TLC540/541 interface timing diagram

Control Sequence

When chip select ($\overline{CS}$) is high, the Data Out pin is in a three-state condition, and the Address Input and I/O Clock pins are disabled. This feature allows each of these pins, with the exception of $\overline{CS}$, to share a control logic point with their counterpart pins on additional A/D devices when additional TLC540 and TLC541 devices are used. This feature serves to minimize the required control logic pins when using multiple A/D devices.

The control sequence has ben designed to minimize the time and effort required to initiate conversion and obtain the conversion result. A normal control sequence is:

1. $\overline{CS}$ is brought low. To minimise errors by noise at the $\overline{CS}$ input internal circuitry waits for two rising edges and then a falling edge of the System clock after a $\overline{CS}$ high-to-low transition before the transition is recognised. This technique is used to protect the device from falsely enabling Data Out and to prevent possible output contention of bussed devices. The MSB of the previous conversion result will immediately appear on the Data Out pin when $\overline{CS}$ is recognised.
2. A new positive-logic multiplexer address is shifted in on the first 4 rising edges of the I/O clock. The negative edges of these four I/O clocks shift out the 2nd, 3rd, 4th, and 5th most significant bits of the previous conversion result. The on-chip sample-and-hold begins sampling the newly addressed analog input after the 4th falling edge. The sampling operation basically involves the charging of internal capacitors to the level of the analog input voltage.
3. Three clock cycles are then applied to the I/O pin and the 6th, 7th, and 8th conversion bits are shifted out on the negative edges of these clock cycles.
4. The final (8th) clock cycle is applied to the I/O clock pin. The falling edge of this clock cycle completes the analog sampling process and initiates the hold function. Conversion is then performed during the next 36 System clock cycles.

$\overline{CS}$ can be kept low during periods of multiple conversion. Also, if $\overline{CS}$ is taken high, it must remain high until the end of conversion. Otherwise, a valid falling edge of $\overline{CS}$ will cause a reset condition, which will abort the conversion in progress.

A new conversion may be started and the ongoing conversion simultaneously aborted by performing steps 1 through 4 before the 36

System clock cycles occur. Such action will yield the conversion result of the previous conversion and not the ongoing conversion.

Sample and Hold Control

For certain applications, such as strobing, conversion must be started at a specific point in time. These devices will accommodate these applications. Although the on-chip sample-and-hold begins sampling upon the negative edge of the 4th I/O clock cycle, the hold function is not initiated until the negative edge of the 8th I/O clock cycle. Thus, the control circuitry can leave the I/O clock signal in its high state during the 8th I/O clock cycle, until the moment at which the analog signal must be converted. The TLC540 and TLC541 will continue sampling the analog input until the 8th falling edge of the I/O clock. The control circuitry or software will then immediately lower the I/O clock signal and initiate the hold function to hold the analog signal at the desired point in time and start conversion.

Combined System and I/O Clock Operation

It is possible to connect the System and I/O clocks together in special situations where controlling circuitry must be minimized. In this case, the following special requirements must be taken into consideration in addition to the requirements of the normal control sequence, which was previously described.

1. When $\overline{CS}$ is recognized by the device to be at a low level, the common clock signal is used as an I/O clock. When the $\overline{CS}$ is recognized by the device to be at a high level, the common clock signal is used to drive the "conversion crunching" circuitry.
2. The device will recognize a $\overline{CS}$ transition only when the $\overline{CS}$ input changes and subsequently the System Clock pin receives two positive edges and then a negative edge. For this reason, after a $\overline{CS}$ negative falling edge, the first two clock cycles will not shift in the address because a low $\overline{CS}$ must be recognized before the I/O clock can shift in an analog channel address. Also, upon shifting in the address, $\overline{CS}$ must be raised after the 6th I/O clock, so that a $\overline{CS}$ low level will be recognized upon the lowering of the 8th I/O clock signal. Otherwise, additional common clock cycles will be recognized as I/O clocks and will shift in an erroneous address.

TLC540 A/D Converter Interface to Zilog Z80A and Z80 Microprocessors

Two different interfaces are shown in this application. While both are TLC540 to Z80 interfaces, the two differ as follows:

1. The control signals are generated by additional hardware for Interface 1, but the control signals for Interface 2 are generated through software
2. Interface 1 is about five times faster than Interface 2
3. Interface 1 has less complicated software than Interface 2

Interface 1

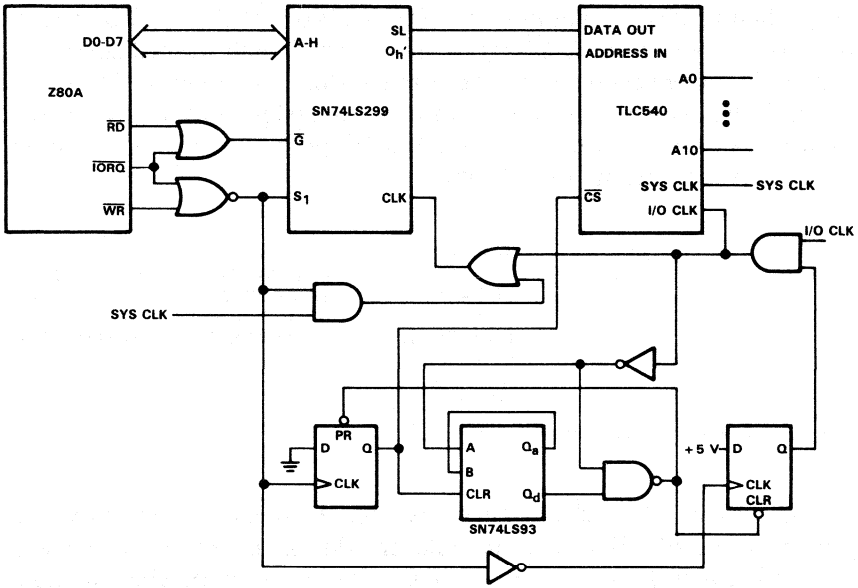


Fig. 11.69 Z80A to TLC540 interface 1 circuit diagram

The circuit shown in Fig. 11.69 initiates conversion with an OUT instruction. The timing diagram for this circuit is shown in Fig. 11.70. When $\overline{IORQ}$ and $\overline{WR}$ go low, $\overline{CS}$ is brought low while the universal shift register is placed in the load data mode and the system clock is enabled to the clock input to latch in the multiplexer address. The rising edge of

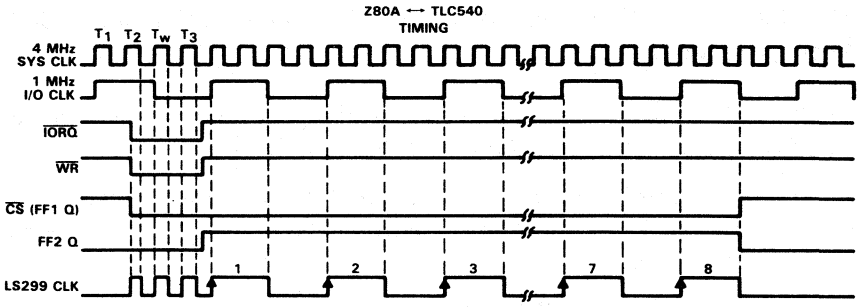


Fig. 11.70 Z80A to TLC540 interface timing diagram

$\overline{\text{IORQ}}$ enables the I/O clock to the shift register to shift the multiplexer address out while shifting the previous conversion result in. Sampling begins at the falling edge of the fourth I/O clock pulse and continues until the eighth falling edge. At this time the I/O clock is disabled and $\overline{\text{CS}}$ is brought high to ensure that the TLC540 device will remain undisturbed during the conversion. Conversion of the addressed analog input requires 36 system clock cycles. During this time, previous conversion results may be read by using an IN instruction. A typical interrupt service routine is shown in the software listing that follows.

```
ISR     PUSH AF
        IN A,(LS299)        ; READ PREVIOUS CONV. RESULT
        LD (DATA), A        ; STORE RESULT
        LD A,D              ; LOAD NEW MUX ADDRESS
        OUT (LS299),A       ; SAMPLE CHANNEL [ INITIATE CONV.
        POP AF
        EI
        RETI
```

NOTE: Register D contains the analog multiplexer address.

At least 36 system clock cycles must occur between interrupts to ensure proper operation. If a new multiplexer address is shifted in while a conversion is in progress, the ongoing conversion will be aborted and a new conversion will be initiated by the falling edge of the eighth I/O clock pulse.

Another software approach is to initiate a new conversion, wait in a delay loop until the conversion is complete and then read in the previous conversion results. A simple program segment using the delay loop method is shown in the software listing that follows. Using this method it is possible to complete a conversion cycle in 21.5 μs.

```
                LD A,D          ; LOAD MUX ADDRESS
                OUT (LS299),A   ; SAMPLE CHANNEL [ INITIATE CONV.
                LD C,03H        ; INITIALIZE COUNTER
WAIT:           DEC C           ; DECREMENT COUNTER
                JP NZ, WAIT     ; IF NOT ZERO KEEP WAITING
                OUT (LS299),A   ; CLOCK RESULTS INTO LS299
                IN A,(LS299)    ; READ CONV RESULTS
```

NOTE: A count of 03H will produce a delay of 10.50 microseconds, suitable for 4 MHz operation, while a count of 05H produces a delay of 17.50 microseconds, suitable for 2.5 MHz operation.

Interface 2

The circuit diagram for Interface 2 shows the software controlled TLC540 to Z80A interface and is shown in Fig. 11.71. Circuit timing is shown in Fig. 11.72.

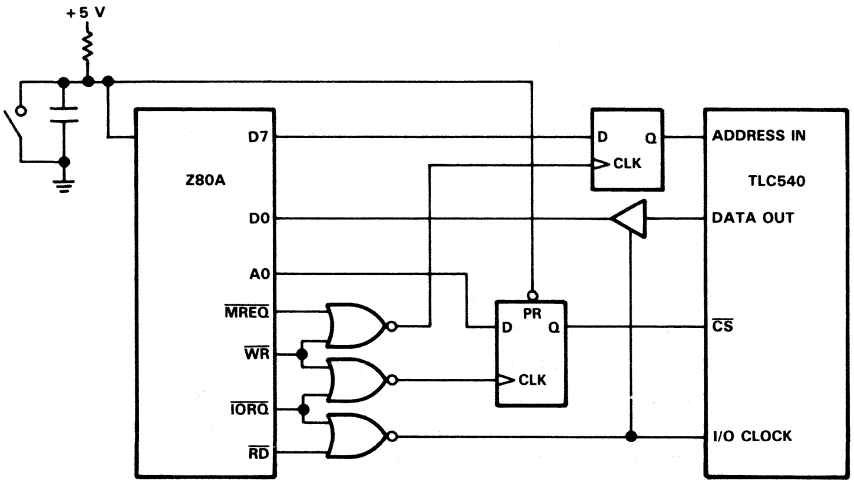


Fig. 11.71 Z80A to TLC540 interface 2 circuit diagram

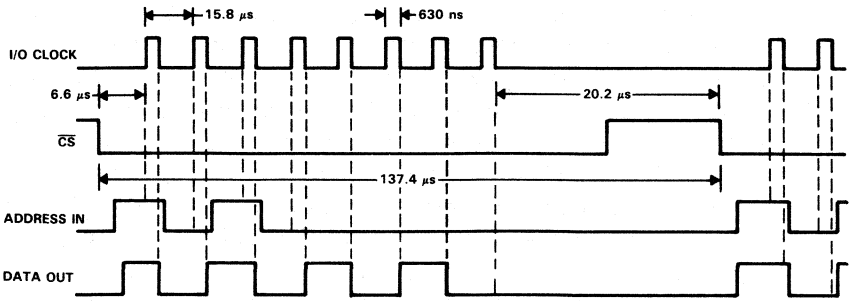


Fig. 11.72 Z80A to TLC540 interface 2 timing diagram

Execution of an IN instruction causes the $\overline{RD}$ line and the $\overline{IORQ}$ line to become active to shift an address bit into the TLC540 device and a data bit out of the TLC540 device. $\overline{CS}$ is brought low by latching in a low from address bit A0 on the positive going edge of the $\overline{WR}$ signal. A simple program segment that shifts out a new analog multiplexer address while also shifting in previous conversion results is shown in the software listing that follows.

```
        LD C,08          ; INITIALIZE BIT COUNTER
        LD B,00          ; CLEAR RESULT REGISTER
        OUT (CSLOW),A    ; BRING CHIP SELECT LOW
LOOP    RLC B            ; ROTATE RESULT LEFT
        LD (HL),D        ; LATCH ADDRESS BIT INTO D FF
        IN A. (BIT)      ; READ IN DATA BIT
        AND 01H          ; MASK OFF BIT 0
        OR B             ; OR NEW BIT INTO RESULT REG.
        LD B,A           ; STORE IN RESULT REGISTER
        RLC D            ; SHIFT ADDRESS LEFT
        DEC C            ; DECREMENT BIT COUNTER
        JP NZ,LOOP       ; GET ANOTHER BIT IF NOT ZERO
        OUT (CSHIGH),A   ; BRING CHIP SELECT HIGH
```

This program segment uses the B register to store the conversion result, the C register as a bit counter, and the D register to hold the analog multiplexer address. The analog multiplexer address is shifted left out of the D register; therefore, the 4-bit address must be placed in the most significant 4-bits of the byte. Conversion results are read in one bit at a time and then shifted left to the proper position in the B register. Sampling of the addressed input begins at the falling edge of the fourth I/O clock pulse and continues until the falling edge of the eighth I/O clock pulse starts the time conversion. Conversion requires 36 system clock cycles; therefore, an appropriate software delay must be included. The amount of the delay depends on the system clock frequency. If a new multiplexer address is shifted in before a conversion has been completed, the ongoing conversion will be aborted and a new conversion cycle will begin at the eighth falling edge of the I/O clock. $\overline{CS}$ is brought high after the eighth falling edge of the I/O clock to ensure that extraneous noise or glitches in the I/O clock line are not interpreted as the beginning of a new cycle. Using this program segment with the system clock at 4 MHz, it is possible to initiate a new conversion cycle and read the results of the previous conversion in 138 μs.

TLC540 A/D Converter Interface to Rockwell 6502 Microprocessors using the 6522 VIA

Interfacing the TLC540 A/D converter to the 6502 microprocessor can be accomplished by several methods. This application presents the design of an interface using the 6522 VIA. A method using TTL gates is described in the next application. Cost and performance are the basic trade-offs between the two designs. The 6522 device interface is faster, but the TTL method costs less.

Principles of Operation

The interface circuit diagram is shown in Fig. 11.73. Timing for a data read cycle and an address write cycle is shown in Figs 11.74 and 11.75, respectively. A software listing that initializes the 6522 device is shown below. Interface control software is also listed below.

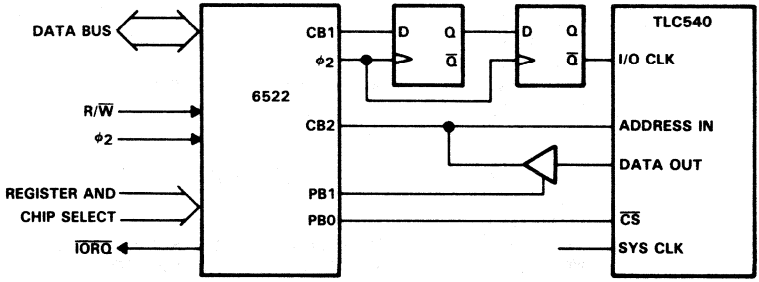


Fig. 11.73 6502 to TLC540 interface circuit diagram (6522 VIA)

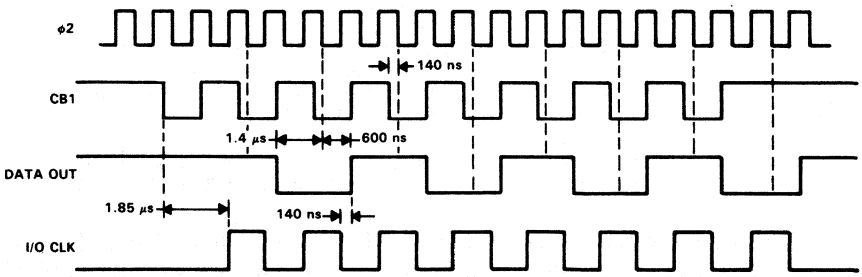


Fig. 11.74 6522 to TLC540 data read cycle timing diagram

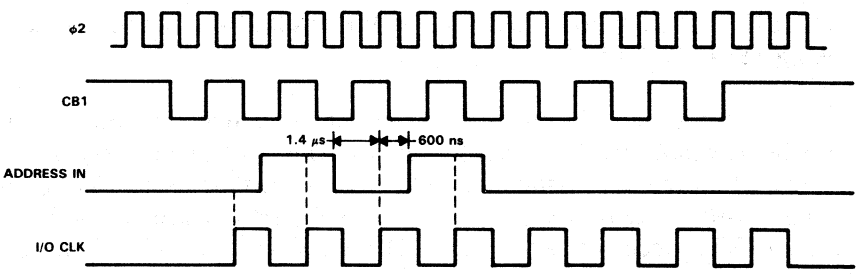


Fig. 11.75 6502 to TLC540 address write cycle timing diagram (6522 VIA)

```
;               REGISTER ASSIGNMENTS
;
ORB             .EQU 0000H
DDRB            .EQU 0002H
SR              .EQU 000AH
ACR             .EQU 000BH
PCR             .EQU 000CH
IFR             .EQU 000DH
IER             .EQU 000EH
;
;
                LDA  #$03           ;
                STA  DDRB           ; INITIALIZE PORT B I/O PINS
                LDA  #$01           ;
                STA  ORB            ; BRING CHIP SELECT HIGH
                LDX  #$00           ; INITIALIZE MUXADDRESS


                LDA  #$18           ; SHIFT OUT ON PHI 2
                STA  ACR            ;
                LDA  #$00           ;
                STA  ORB            ; DISABLE DATA OUT, BRING /CS LOW
                STX  R              ; SHIFT OUT MUXADDRESS TO 540
                LDY  #$02           ; LOAD DELAY LOOP COUNTER
DELAY1          DEY                 ; DECREMENT DELAY COUNTER
                BNE  DELAY1         ; BRANCH IF NOT ZERO
                NOP                 ;
                LDX  #$02           ;
                LDA  #$08           ;
                STA  ACR            ; SHIFT IN ON PHI 2
                STX  ORB            ; ENABLE OUTPUT OF 74LS126
                LDA  SR             ; DUMMY LOAD TO SHIFT RESULTS IN
                LDY  #$03           ; LOAD DELAY LOOP COUNTER
DELAY2          DEY                 ; DECREMENT DELAY LOOP COUNTER
                BNE  DELAY2         ; BRANCH IF NOT ZERO
                LDA  #$01           ;
                STA  ORB            ; DISABLE DATA OUT, BRING /CS HIGH
                LDA  SR             ; READ CONVERSION RESULTS INTO 6502
```

The interface makes use of the serial port available on port B pins CB1 and CB2. Since the serial port is not capable of full duplex communication, the port is configured to function as an output port for

the address write cycle and as an input port during the data read cycle. This requires the use of a SN74LS126 3-state buffer. The D-type flip-flops are used to delay the I/O clock to ensure that the set up and hold times for shifting data in and out are met. Port B pins PB0 and PB1 are used to generate $\overline{CS}$ and the output enable signal for the 3-state buffer.

A data conversion cycle begins by bringing $\overline{CS}$ low. This is accomplished by writing a low to PB0. The analog multiplexer address is shifted out by writing to the SR register of the 6522. A delay loop is inserted to wait until the multiplexer address has been shifted out. The serial port is then configured as an input port in order to shift in the A/D conversion results. The output of the 3-state buffer is enabled by writing a high to PB1, and data is shifted into the SR register of the 6522. Again, a delay loop is included to wait until the data is shifted in. $\overline{CS}$ is then brought high, and the 3-state buffer is disabled, completing one data acquisition cycle in 55 μs.

TLC540 A/D Converter Interface to Rockwell 6502 Microprocessors using TTL Gates

This application shows an interface between the 6502 microprocessor and the TLC540 A/D converter using TTL gates. The previous application showed an interface between these same two parts using the 6522 VIA. Cost and performance are the basic trade-offs between the two designs. The 6522 device interface is faster, but the TTL method costs less.

Principles of Operation

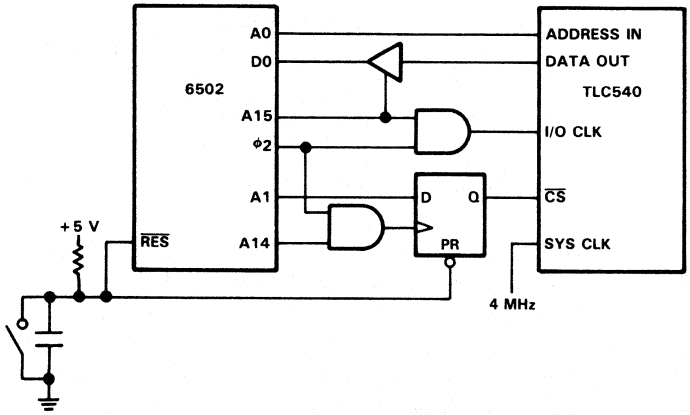


Fig. 11.76 6502 to TLC540 interface circuit diagram (TTL gates)

The basic premise of the interface circuit shown in Fig. 11.76 is that all timing control signals are generated under the control of software. Circuit timing is shown in Fig. 11.77 and interface control software is listed below.

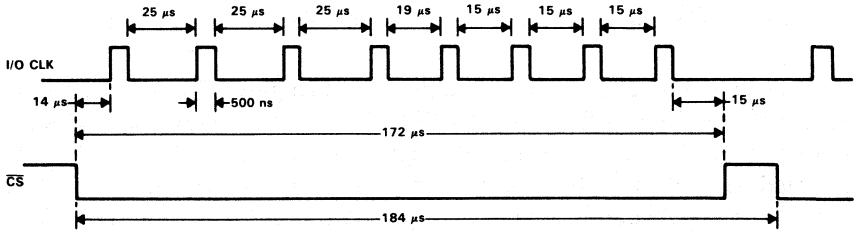


Fig. 11.77 6502 to TLC540 interface circuit timing diagram (TTL gates)

```
CSLOW         .EQU 4000H
CSHIGH        .EQU 4002H
ADDLOW        .EQU 8000H
ADDHIGH       .EQU 8001H
MUXADDRESS    .EQU 0000H
RESULT        .EQU 0001H
;
;
;
              STA  CSLOW        ; BRING /CS LOW
              LDX  #$04         ; SET BIT COUNTER FOR FIRST 4 BITS
LOOP1:        ROL  MUXADDRESS   ; ROTATE MUXADDRESS BIT INTO CARRY
              BCS  HIGH         ; BRANCH IF BIT IS SET
              LDA  ADDLOW       ; WRITE OUT A LOW ON A0, CLOCK DATA IN
              JMP  CONTINUE     ; SKIP NEXT INSTRUCTION
HIGH:         LDA  ADDHIGH      ; WRITE OUT A HIGH ON A0, CLOCK DATA IN
CONTINUE:     ROR  A            ; ROTATE DO INTO CARRY
              ROL  RESULT       ; ROTATE CARRY INTO RESULT
              DEX               ; DECREMENT BIT COUNTER
              BNE  LOOP1        ; GO BACK FOR ANOTHER BIT
              LDX  #$04         ; SET COUNTER FOR SECOND 4 BITS
LOOP2:        LDA  ADDLOW       ; READ IN DATA BIT
              ROR  A            ; ROTATE INTO CARRY
              ROL  RESULT       ; ROTATE CARRY INTO RESULT
              DEX               ; DECREMENT BIT COUNTER
              BNE  LOOP2        ; GET ANOTHER BIT
              STA  CSHIGH       ; BRING /CS HIGH
```

A data conversion cycle is initiated by bringing $\overline{CS}$ low. This is accomplished by latching a low into the D-type flip-flop from address line A1 on the positive edge of system clock $\phi 2$. Address bit A14 is used as a gating signal to prevent the TLC540 device from being inadvertently selected during normal program execution. After $\overline{CS}$ is brought low, I/O clock pulses shift in the multiplexer address while shifting out the previous conversion results. The I/O clock is enabled by gating the positive going

pulse of $\phi 2$ to the TLC540 I/O CLK input. This gating occurs by addressing a location so A15 is high. The high on A15 also enables the 3-state buffer output onto the data bus. Address bit A0 determines whether the multiplexer address bit being written is a high or a low. This multiplexer address bit is shifted into the TLC540 device on the positive edge of $\phi 2$.

Once the data is loaded into the accumulator, it is rotated into the carry bit and then rotated into memory. $\overline{CS}$ is brought high again by writing a high into the D-type flip-flop by placing a high on address bit A1. This cycle can be completed every 172 μs.

This interface circuit uses an address decoding scheme that requires a minimum of decoding hardware. Small modifications of the address decoding scheme may be necessary in order to fit the interface to a particular application.

TLC540 and TLC541 A/D Converter Interface to Motorola 6805 Microcomputers

This application describes techniques for using software controlled interfaces between the TLC540 and TLC541 devices and the 6805 microcomputer. Interfaces for the 6805 microcomputer to the TLC540 and TLC541 devices may be accomplished using either of two methods:

1. Generating all necessary control signals under software control by toggling the output port pins
2. Using the serial peripheral interface (SPI) to generate necessary control signals for data transfer.

The TLC540 and TLC541 devices are particularly well suited for use with the SPI, however, not all 6805 microcomputers include the SPI on the chip. Therefore, the software controlled interface has the advantage although it is less efficient.

Principles of Operation

The circuit diagram for the software controlled 6805 microprocessor to TLC540 and TLC541 device interface is shown in Fig. 11.78. Circuit timing is shown in Fig. 11.79. The software controlled interface makes use of four pins on port A of the 6805 microcomputer. Three of these pins are used as outputs to generate the $\overline{CS}$, I/O clock and, multiplexer address inputs. The remaining pin is used as an input to receive the conversion

results from the TLC540 and TLC541 devices. A short program segment which can be used to initialize the input/output pins is listed on the next page.

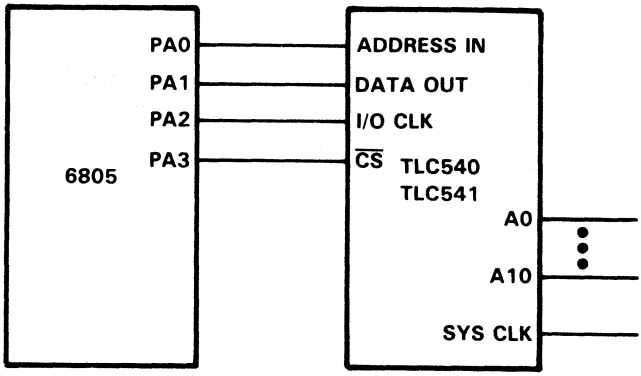


Fig. 11.78 6805 to TLC540 and TLC541 interface circuit diagram

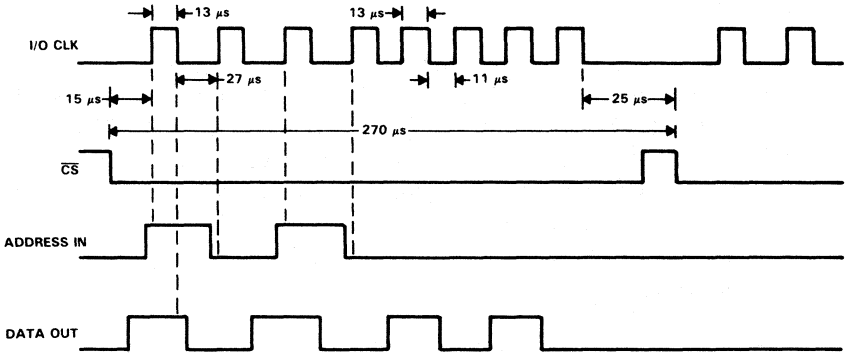


Fig. 11.79 6805 to TLC540 and TLC541 interface timing

A program listing which controls the actual transfer of data is also given on the next page. This program block sends out the 4-bit analog multiplexer address that is shifted into the TLC540 and TLC541 devices on the first four I/O clock rising edges. Sampling of the addressed input begins at the falling edge of the fourth I/O clock and continues until the falling edgte of the eighth I/O clock occurs. Conversion requires 36 clock cycles of the system clock, which can run at up to 4 MHz to allow conversion in only 9 µs. Conversion results are shifted out of the TLC540 and TLC541 devices on the negative edge of the I/O clock with the

```
PORTA       .EQU  0000H
DDRA        .EQU  0004H
MUXADDRESS  .EQU  0010H
            .ORG  0100H

START:      LDA   #0DH              ;
            STA   DDRA              ; INITIALIZE PORT A I/O PINS
            BSET  3,PORTA           ; BRING CHIP SELECT HIGH
            LDA   #0DH              ;
            STA   MUXADDRESS        ; INITIALIZE MULTIPLEXER ADDRESS


            LDX   #04               ; LOAD COUNTER FOR FIRST 4 BITS
            BCLR  3,PORTA           ; BRING CHIP SELECT LOW
;
LOOP1:      ROL   MUXADDRESS        ; ROTATE MUXADD BIT INTO CARRY BIT
            BCS   SET               ; GO TO SET IF MUXADD BIT IS 1
            BCLR  0,PORTA           ; WRITE OUT A 0 TO TLC540 ADDRESS IN
            JMP   SKIP              ; SKIP NEXT INSTRUCTION
SET:        BSET  0,PORTA           ; WRITE OUT A 1 TO TLC540 ADDRESS IN
SKIP:       BSET  2,PORTA           ; BRING I/O CLOCK HIGH
            BRSET 1,PORTA,LABEL1    ; READ DATA BIT INTO CARRY BIT
LABEL1:     ROLA                    ; ROTATE NEW DATA BIT INTO ACCUM
            BCLR  2,PORTA           ; BRING I/O CLOCK LOW
            DECX                    ; DECREMENT COUNTER
            BNE   LOOP1             ; CONTINUE IF COUNTER IS NOT ZERO
;
            LDX   #04               ; LOAD COUNTER FOR LAST 4 DATA BITS
;
LOOP2:      BSET  2,PORTA           ; BRING I/O CLOCK HIGH
            BRSET 1,PORTA,LABEL2    ; READ DATA BIT INTO CARRY BIT
LABEL2:     ROLA                    ; ROTATE NEW DATA BIT INTO RESULT
            BCLR  2,PORTA           ; BRING I/O CLOCK LOW
            DECX                    ; DECREMENT COUNTER
            BNE   LOOP2             ; CONTINUE IF COUNTER IS NOT ZERO
;
            BSET  3,PORTA           ; BRING CHIP SELECT HIGH
```

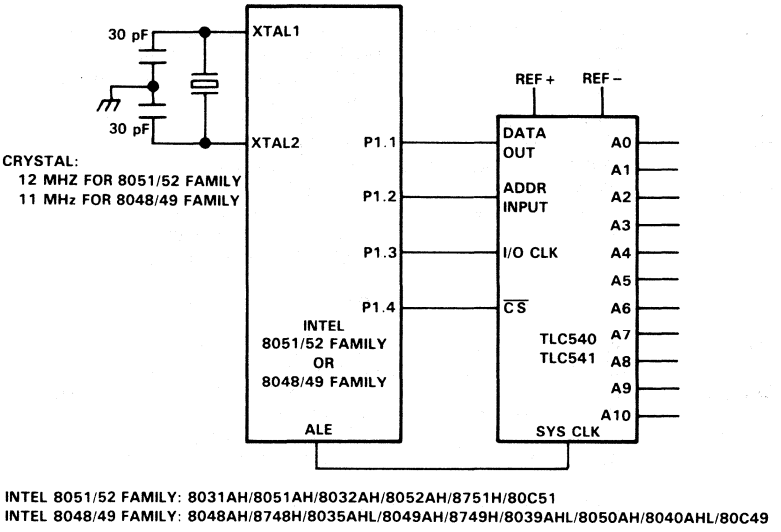
program block leaving these results in the accumulator. With the 6805 device running at 5 MHz and the TLC540 device system clock at 4 MHz, one conversion cycle can load an analog multiplexer address and read the results from the previous conversion. This cycle can be completed in 270 μs. All 11 analog inputs can be consecutively converted in 3.25 ms.

TLC540 and TLC541 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Parallel Ports

Two types of interface are presented. Interface 1 derives the A/D converter clock from the microcontroller ALE clock. In Interface 2, the A/D converter clock signal is derived from the microcontroller crystal oscillator.

These interfaces minimize the hardware and rely extensively on software techniques. Although the amount of hardware and associated cost is reduced, the use of more software increases the time required to load the address into and retrieve the conversion data from the A/D converter. However, the trade-off of minimum hardware versus longer conversion time may benefit many designs.

Interface 1 (ALE CLOCK)



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751H/80C51
INTEL 8048/49 FAMILY: 8048AH/8748H/8035AHL/8049AH/8749H/8039AHL/8050AH/8040AHL/80C49

Fig. 11.80 TLC540 and TLC541 to Intel 8051/52 microcontroller parallel port interface 1 diagram (clock signal from ALE)

The interface shown in Fig. 11.80 will always work with the TLC540 device, but will not work with the TLC541 device at the higher microcontroller clock frequencies. Before using the TLC541 device, the designer must verify that the high and low pulse widths of the ALE clock signal meet the requirements of the TLC541 device. These pulse widths are dependent on the microcontroller instruction cycle clock frequency.

Interface 2 (CRYSTAL CLOCK)

As shown in Fig. 11.81, the system clock for the A/D converter is obtained from the microcontroller crystal oscillator. To assure proper

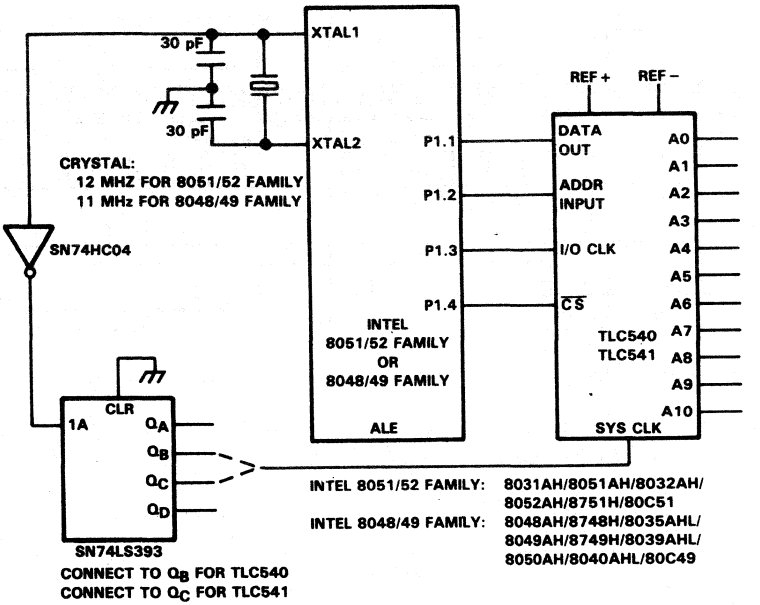
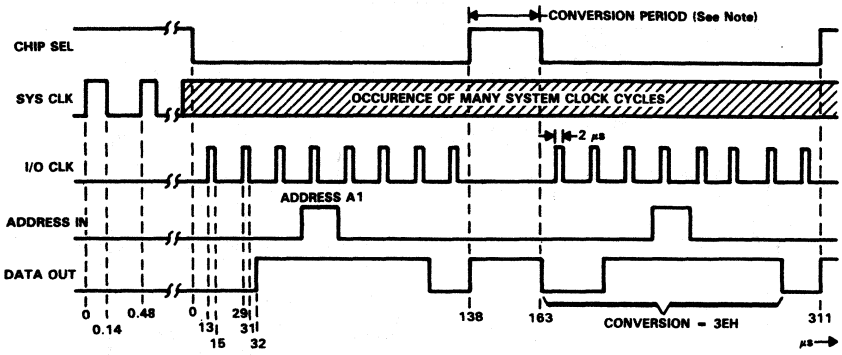


Fig. 11.81 TLC540 and TLC541 to Intel 8051/52 microcontroller parallel port interface 2 diagram (clock signal from crystal osc.)

operation of the crystal oscillator, a high impedance buffer must be used to tap the signal from the oscillator. An important detail is that the low and high level input requirements of the buffer must lie within the range of the oscillator signal. This compatibility will prevent missing edge transitions in the A/D converter's system clock signal. In addition, the buffered crystal oscillator signal must be frequency divided to assure that the resulting system clock signal does not exceed the upper frequency limit of the A/D converter. Any convenient divider circuitry may be used for this task.

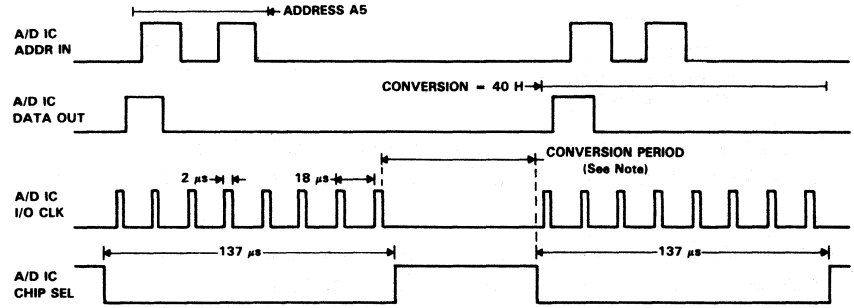
Timing Diagrams

Figs 11.82 and 11.83 show the timing diagrams for the A/D conversions of Interfaces 1 and 2, respectively. Loading the new address and retrieving the conversion for the previous address requires 137 μs. The conversion period requires 36 system clocks. This time period is longer for the TLC541 device since the microcontroller clock frequency (see software listing) or A/D clock frequency (see software listing) must be lower to satisfy the TLC541 device specifications.



NOTE: Conversion period requires 36 system clock cycles after 8th I/O clock goes low upon clocking in the address.

Fig. 11.82 Timing diagram of an A/D conversion cycle for TLC540 and TLC541 to Intel 8051/52 parallel port Interface 1



NOTE: Conversion period requires 36 system clock cycles after 8th I/O clock goes low upon clocking in the address.

Fig. 11.83 Timing diagram of an A/D conversion cycle for TLC540 and TLC541 to Intel 8051/52 parallel port interface 2

Software

The software listings for Interface 1 and Interface 2 follow. These software programs use a subroutine, S540D, that simultaneously loads the A/D converter with a new address and retrieves the conversion result for the previous address. This subroutine can be used any time the designer desires to load a new address and retrieve a conversion value. Simultaneous loading and retrieving makes this subroutine very effective for continuous monitoring of A/D converter analog inputs. The subroutine assumes that the new address has been previously placed in the R2 register. Upon completion of the subroutine, the conversion result for the previous address is left in the accumulator.

Software Listing for Intel 8051/52–TLC540/541 Parallel Interface 1

```
0000  C2 93          CLR P1.3          ;Lower I/O clock
0002  7A 10          MOV R2,#10H       ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
0004  11 0E          ACALL S540D       ;Load 540 address, assumes the address
                                        ;is currently in R2.
0006  7B 09          MOV R3,#09H       ;This software loop allows a
                                        ;conservative 40 A/D system clocks,
                                        ;since only 36
0008  DB FE  DELAY:  DJNZ R3,DELAY     ;clocks are required to perform
                                        ;conversion, to be emitted from the
                                        ;microprocessor ALE pin
              ;
000A  7A 10          MOV R2,#10H       ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
000C  11 0E          ACALL S540D       ;Load new 540 address, assumes this
                                        ;address is in R2; leaves the
                                        ;conversion result for the previous
                                        ;address in A
```

Software Listing for Intel 8051/52–TLC540/541 Parallel Interface 2

```
0000  C2 93          CLR P1.3          ;Lower I/O clock
0002  7A 10          MOV R2, #10H      ;Load A/D analog input address. Note
                                        ;that to send address= 1, R2=10h.
0004  11 0E          ACALL S540D       ;Load 540 address, assumes the address
                                        ;is currently in R2.

                     A delay must occur here to allow the A/D I.C. to complete
                     conversion. The delay must allow 36 A/D I.C. system clock
                     cycles to occur.

0006  7A 10          MOV R2, #10H      ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
0008  11 0E          ACALL S540D       ;Load new 540 address, assumes this
                                        ;address is in R2; leaves the conversion
                                        ;result for the previous address in A.
```

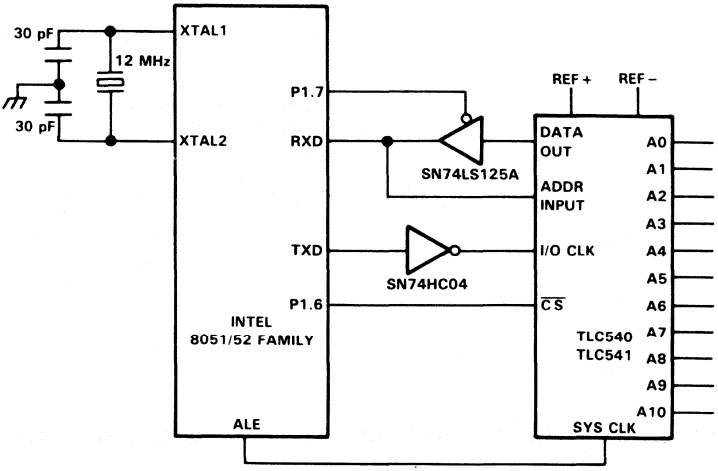
Subroutine S540D for Intel 8051/52–Parallel Interface 1 and 2

```
000E  7E 08  S540D:  MOV R6,#08H       ;Set bit counter to 8
0010  C2 94          CLR P1.4          ;Lower chip select
0012  C3     S540DL: CLR C             ;Initialize C=0
0013  30 91 01       JNB P1.1,S540DI   ;If 540 Data Out=0; branch
0016  B3             CPL C             ;540 Data Out=1; set C=1
0017  EA     S540DI: MOV A,R2          ;Get serial buffer
```

```
0018    33                      RLC A           ;Shift Data Out bit into serial
                                                ;buffer and shift 540 address
0019    FA                      MOV R2,A        ;Store serial buffer
001A    50 05                   JNC S540DWO     ;If 540 address bit = 0; branch
001C    43 90 04                ORL P1,#04H     ;Set 540 branch line to 1
001F    80 03                   SJMP S540DWE    ;Go and raise the I/O clock
0021    53 90 FB    S540DWO:    ANL P1,#FBH     ;Set 540 address line to 0
0024    00          S540DWE:    NOP             ;Allow address line to setup
0025    B2 93                   CPL P1.3        ;Raise I/O clock
0027    00                      NOP             ;Delay to slow I/O clock
0028    C2 93                   CLR P1.3        ;Lower I/O clock
002A    DE E6                   DJNZ R6,S540DL  ;Do all 8 bits
002C    B2 94                   CPL P1.4        ;Raise chip select
002E    EA                      MOV A,R2        ;Get serial buffer
002F    22              RET
0030                    END
```

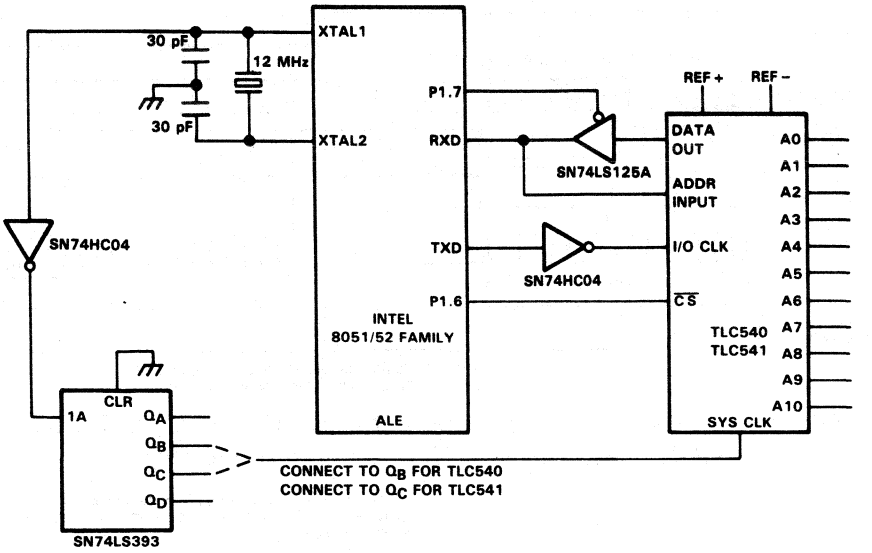
TLC540 and TLC541 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Serial Ports

Two types of interface are again presented. Interfaces 1 and 2 obtain the A/D converter's system clock respectively from the microcontroller's ALE clock and crystal oscillator.



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751H/80C51
CONNECT TO Q_B FOR TLC540
CONNECT TO Q_C FOR TLC541

Fig. 11.84 TLC540 and TLC541 to Intel 8051/52 microcontroller serial port interface 1 diagram (clock signal from ALE)



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751H/80C51

Fig. 11.85 TLC540 and TLC541 to Intel 8051/52 microcontroller serial port interface 2 diagram (clock signal from crystal oscillator)

The circuit for Interface 1, which obtains the clock signal from the ALE pin, is shown in Fig. 11.84. The circuit for Interface 2, which obtains the clock signal from the crystal oscillator, is illustrated in Figure 11.85. The signal at the microcontroller TXD pin must be inverted so the communication protocols for the microcontroller and the A/D converter are compatible. Use of a SN74LS125 3-state buffer allows the microcontroller serial port to be used for both transmission and reception. Although this method does not permit simultaneous loading of a new address while retrieving the conversion of a previously loaded address, the time loss is small. This is because the serial port can quickly load an address and retrieve the resulting conversion.

Timing Diagrams

Figs 11.86 and 11.87 show the timing diagrams for an A/D conversion for each of the two circuits. Loading the address, waiting for conversion, and retrieving the conversion result requires 56 μs for Interface 1 and 48 μs for Interface 2 when using the TLC540 device. This time period is longer for the TLC541 device because the system clock frequency must be lower for this device.

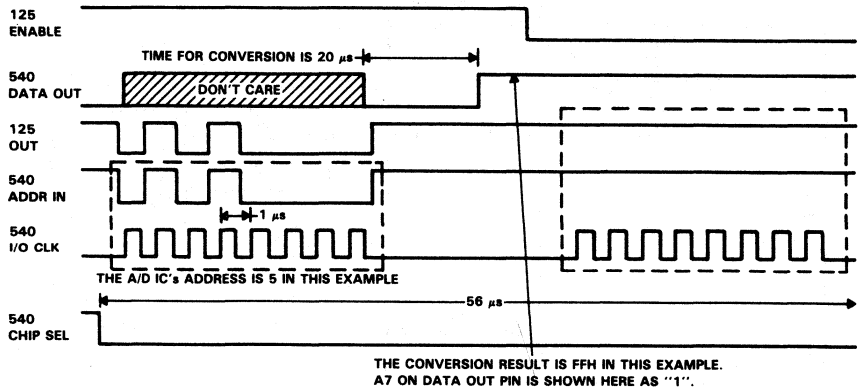


Fig. 11.86 Timing diagram of an A/D conversion cycle for TLC540 and TLC541 to Intel 8051/52 serial port interface 1

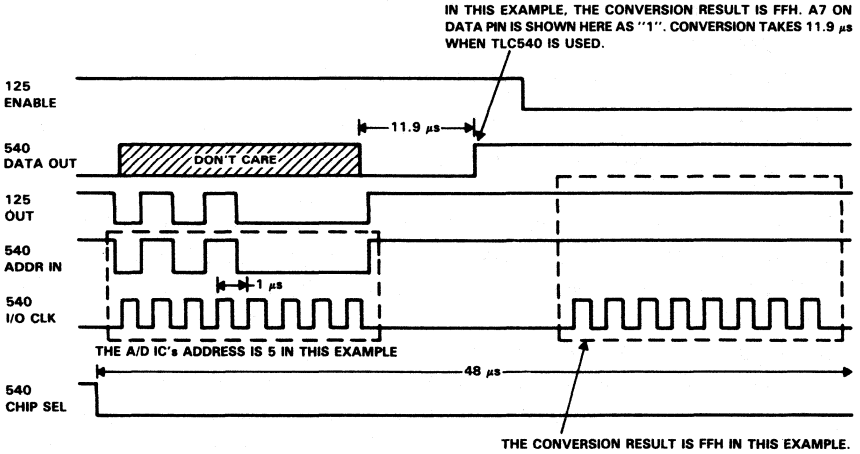


Fig. 11.87 Timing diagram of an A/D conversion cycle for TLC540 and TLC541 to Intel 8051/52 serial port interface 2

Software

The combined software routines for Interface 1 and Interface 2 follow. The serial port mode 0 is used. If desired, the software can be incorporated into a subroutine so the designer can address the software with a simple subroutine call. Careful attention must be exercised when placing the address bits in the serial buffer since the serial port sends the least significant bit first and the A/D converter accepts this bit as the most significant bit of the address. A similar process occurs when the serial port receives the conversion result.

Software Listing for Intel 8051/52–TLC540/541 Serial Interface 1 and 2

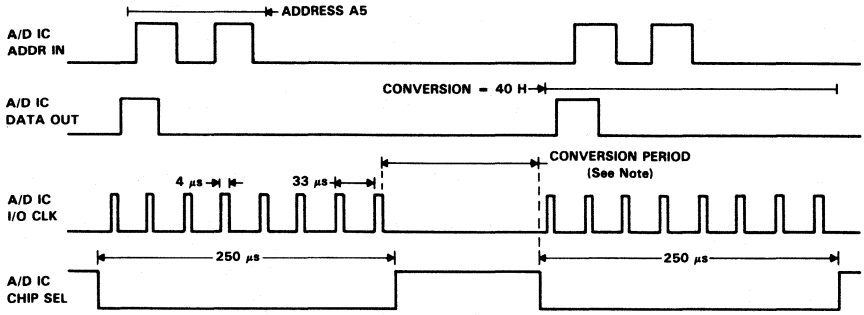
```
                        ;
0000  74 0A             SR540L:   MOV A, #0AH       ;A/D IC address of 5 in this example
                        ;
                        ;The serial port will send 0A(00001010) with the least
                        ;significant bit first. Therefore, the A/D IC will see
                        ;(01010000), which will load an address of 5 into the control
                        ;register.
                        ;
0002  C2 97                       CLR P1.7          ;Disable 125
0004  B2 97                       CPL P1.7
0006  C2 96                       CLR P1.6          ;Lower chip select
0008  53 98 ED                    ANL SCON, #EDH    ;Reset REN & TI flags
000B  F5 99                       MOV SBUF,A        ;Send 540 address (LSB FIRST)
000D  30 99 FD          SNDTST:   JNB SCON.1,SNDTST ;TI flag not set; branch
                                                    ;until transmission is complete
                        ;
                        ;A delay must occur here to allow the A/D IC to complete
                        ;conversion. The delay must llow 36 A/D IC system clock
                        ;cycles to occur.
                        ;
0010  7B 09                       MOV R3, #09H      ;This software loop allows a conservative
                                                    ;40 A/D system clocks, since only 36
0012  DB FE             DELAY:    DJNZ 3, DELAY     ;clocks are required to perform
                                                    ;conversion, to be emitted from the
                                                    ;microprocessor ALE pin
0014  B2 97                       CPL P1.7          :Enable 125
0016  43 98 10                    ORL SCON, #10H    :Set REN
0019  53 98 FE                    ANL SCON, #FEH    :Reset RI
0010  30 98 FD          RCVTST:   JNB SCON.0,RCVTST :RI FLAG not set; branch
                                                    :until reception is complete
001F  B2 96                       CPL P1.6          ;Raise chip select
0021  E5 99                       MOV A,SBUF        ;Get SBUF
                        ;
                        ;The serial port read reverses the data conversion bits coming
                        ;to the microprocessor so that they are in the following order:
                        ;b0(1sb),b1,b2,b3,b4,b5,b6,b7(msb). These bits (01234567) along
                        ;with the carry bit (C) in the following instruction comments
                        ;are presented so that the reader will understand the technique,
                        ;which is used to place the bits in their proper order.
                        ;
0023  33                          RLC A             ;6543210C 7; b7 is now in carry
0024  33                          RLC A             ;543210C7 6; b6 is now in carry
0025  92 E1                       MOV ACC.1,C       ;54321067 6; put b6 into ACC.1
0027  A2 E2                       MOV C,ACC.2       ;54321067 0; put b0 into C
0029  33                          RLC A             ;43210670 5; b5 is now in carry
002A  92 E3                       MOV ACC.3,C       ;43215670 5; put b5 into ACC.3
002C  A2 E4                       MOV C,ACC.4       ;43215670 1; put b1 into C
002E  33                          RLC A             ;32156701 4; b4 is now in carry
002F  92 E5                       MOV ACC.5,C       ;32456701 4; put b4 into ACC.5
0031  A2 E6                       MOV C,ACC.6       ;32456701 2; put b2 into C
0033  33                          RLC A             ;24567012 3; b3 is now in carry
0034  92 E7                       MOV ACC.7,C       ;34567012 3; put b3 into ACC.7
0036  23                          RL A              ;45670123 ; prepare for SWAP A
0037  C4                          SWAP A            ;01234567 ; bits are ordered
                                                    ;correctly
                                                    ;conversion result is in accumulator
0038                              END
```

TLC540 and TLC541 A/D Converter Interface to Intel 8048 and 8049 Microcontroller Parallel Ports

Hardware

Figs 11.80 and 11.81 show the circuit diagrams for this application. Also refer to the **TLC540 and TLC541 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers Parallel Ports** for more details about this application.

Timing Diagram



NOTE: Conversion period requires 36 system clock cycles after 8th I/O clock goes low upon clocking in the address.

Fig. 11.88 Timing diagram of A/D conversion cycle for TLC540 and TLC541 – Intel 8048/49 microcontroller interface 2

Fig. 11.88 of this application shows the timing of an A/D conversion. Loading the new address and retrieving the conversion for the previous address requires 250 µs. The conversion period requires 36 system clocks, however, this period is longer for the TLC541 device because the system clock frequency must be lower for this device.

Software

Software listings follow below.

Software listing for Intel 8048/49 – TLC540/541 Parallel Interface 1

```
0000    99 F7       ANL P1, #F7H        ;Lower I/O clock
0002    B8 10       MOV R0, #10H        ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
0004    14 0E       CALL S540D          ;Load 540 address, assumes the address
                                        ;is currently in R0. Note that to send
                                        ;address = 1, R0 = 10h.
                    ;
0006    BB 13       MOV R3, #13H        ;This software loop allows a
                                        ;conservative 40 A/D system clocks,
                                        ;since only 36
```

```
0008  EB 08    DELAY:   DJNZ R3,DELAY   ;clocks are required to perform
                                        ;conversion, to be emitted from the
                                        ;microprocessor ALE pin
                ;
000A  B8 10              MOV R0,#10H    ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
000C  14 0E              CALL S540D     ;Load new 540 address, assumes this
                                        ;address is in R0; leaves the
                                        ;conversion result for the previous
                                        ;address in A
                ;
```

Software listing for Intel 8048/49−TLC540/541 Parallel Interface 2

```
                ;
0000  99 F7              ANL P1,#F7H    ;Lower I/O clock
0002  B8 10              MOV R0,#10H    ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
0004  14 0E              CALL S540D     ;Load 540 address, assumes the address
                                        ;is currently in R0. Note that to send
                                        ;address = 1, R0 = 10h.
                ;
                ;A delay must occur here to allow the A/D IC to complete
                ;conversion. The delay must allow 40 A/D IC system clock
                ;cycles to occur.
                ;
0006  B8 10              MOV R0,#10H    ;Load A/D analog input address. Note
                                        ;that to send address = 1, R2 = 10h.
0008  14 0E              CALL S540D     ;Load new 540 address, assumes this
                                        ;address is in R0; leaves the
                                        ;conversion result for the previous
                                        ;address in A
```

Subroutine S540D for Intel 8048/49−Parallel Interface 1 and 2

```
                ;
000A  BA 08    S540D:    MOV R2,#08H    ;Set bit counter to 8
000C  99 EF              ANL P1,#EFH    ;Lower chip select
000E  97       S540DL:   CLR C          ;Initialize C = 0
000F  09                 IN A,P1        ;Get Port 1
0010  37                 CPL A          ;Complement accumulator
0011  32 14              JB1 S540DI     ;If 540 Data Out = 0; branch
0013  A7                 CPL C          ;540 Data Out = 1; set C = 1
0014  F8       S540DI:   MOV A,R0       ;Get serial buffer
0015  F7                 RLC A          ;Shift Data Out bit into serial
                                        ;buffer and shift 540 address
                                        ;bit into C
0016  A8                 MOV R0,A       ;Store serial buffer
0017  E6 1D              JNC S540DWO    ;If 540 address bit = 0; branch
0019  89 04              ORL P1,#04H    ;Set 540 branch line to 1
001B  04 1F              JMP S540DWE    ;Go and raise the I/O clock
001D  99 FB    S540DWO:  ANL P1,#FBH    ;Set 540 address line to 0
001F  00       S540DWE:  NOP            ;Allow address line to setup
0020  89 08              ORL P1,#08H    ;Raise I/O clock
0022  00                 NOP            ;Delay to slow I/O clock
0023  99 F7              ANL P1,#F7H    ;Lower I/O clock
0025  EA 0E              DJNZ R2,S540DL ;Do all 8 bits
0027  89 10              ORL P1,#10H    ;Raise chip select
0029  F8                 MOV A,R0       ;Get serial buffer
002A  83                 RET
```

TLC545 and TLC546 8-Bit Analog to Digital Converters with 19-Inputs and Serial Input/Output

The TLC545 and TLC546 have the same basic features as the TLC540 but have a 20 input analog multiplexer allowing sampling on any of 19 inputs or an internal self test voltage. Table 11.14 shows the performance and Fig. 11.89 the functional block diagram.

Table 11.14

	TLC545	TLC546
Acquisition Time (max)	1.5 μs	2.7 μs
Conversion time (max)	9 μs	17 μs
Sampling rate (max)	76 × 10^3sps	40 × 10^3sps
Power Dissipation (typ)	6 mW	6 mW

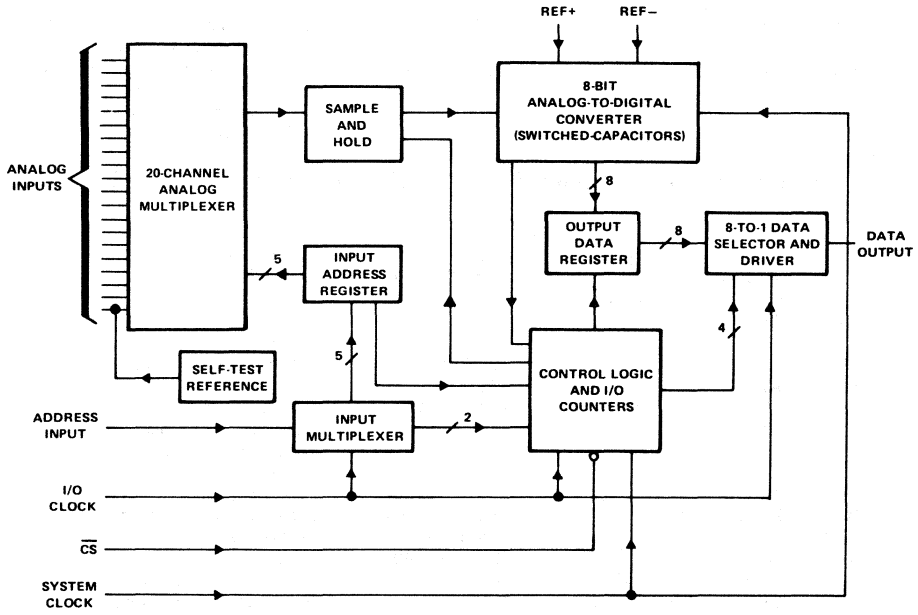


Fig. 11.89 TLC545/546 functional block diagram

The pin layout is shown in Fig. 11.90.

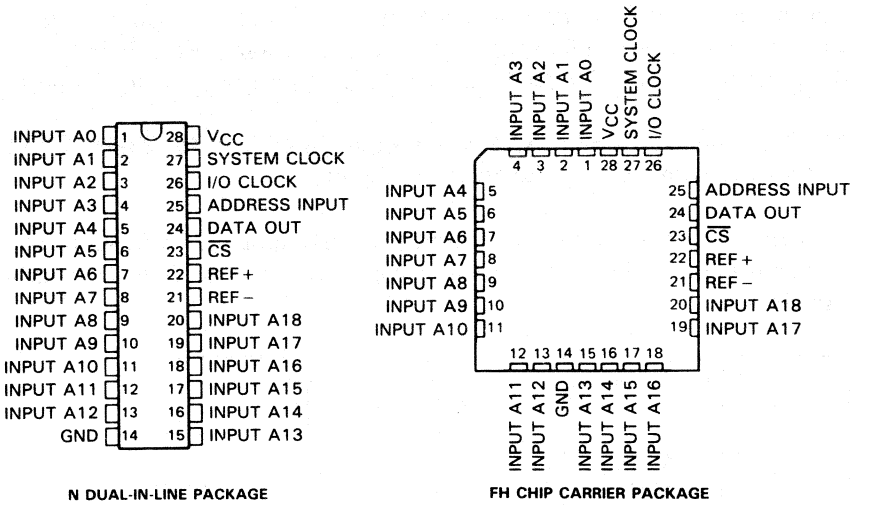
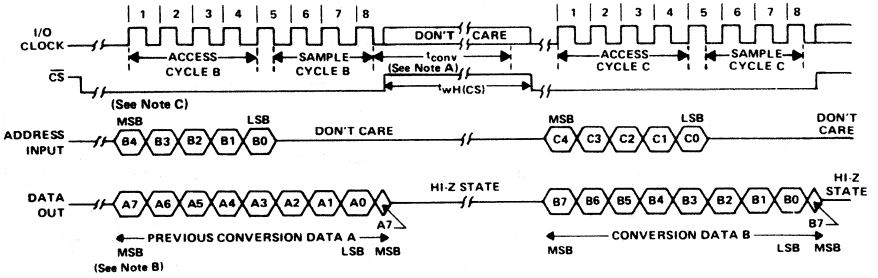


Fig. 11.90 TLC545 and TLC546 pinouts (top view)

TLC545, TC546 Operation



NOTES: A. The conversion cycle, which requires 36 system clock periods, is initiated with the 8th I/O clock↓ after CS↓ for the channel whose address exists in memory at that time.

B. The most significant bit (MSB) will automatically be placed on the DATA OUT bus after $\overline{CS}$ is brought low. The remaining seven bits (A6-A0) will be clocked out on the first seven I/O clock falling edges.

C. To minimize errors caused by noise at the $\overline{CS}$ input, the internal circuitry waits for three system clock cycles (or less) after a chip select transition before responding to control input signals. Therefore, no attempt should be made to clock-in address data until the minimum chip-select setup time has elapsed.

Fig. 11.91 TLC545/546 interface timing diagram

Operation of TLC545 and TLC546 is the same as TLC540 (see page 11−79 in all respects except for the need to accommodate the wider input analog channel addressing range. As shown by the timing diagram of Fig. 11.91, 5-bits are required instead of the 4-bits of the TLC540. This modifies the Control Sequence step 2 of page 11−80 only in the extent

that the multiplexer address is clocked in on the first five rising edges of the I/O clock instead of four. Similarly the on-chip sample and hold begins sampling the newly addressed input on the 5th falling edge instead of 4th.

TLC545 and TLC546 A/D Converter Interface to Zilog Z80A Microprocessors Using Hardware Control

This application describes a technique for interfacing the TLC545/6 A/D converter to the Z80A microprocessor using hardware to generate the control signals.

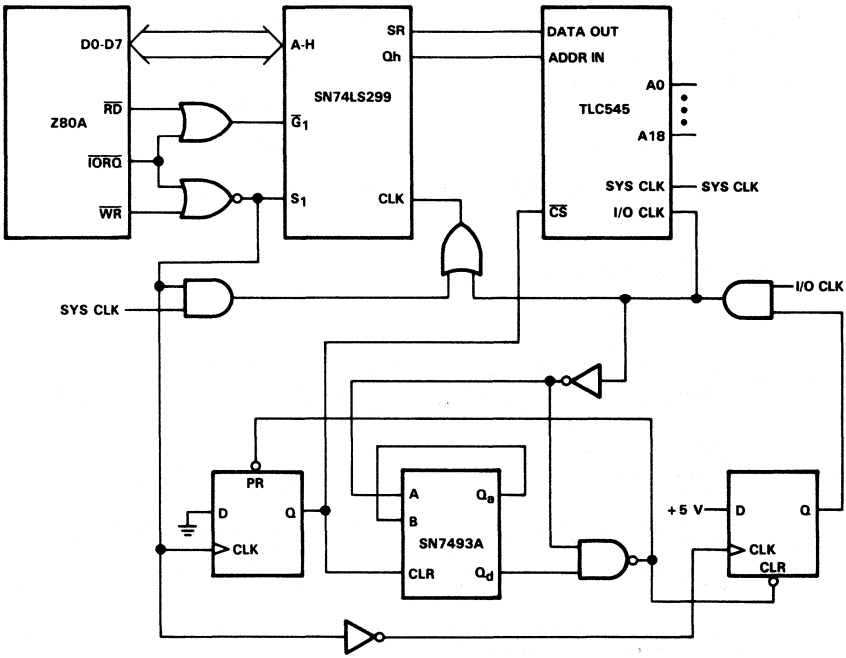


Fig. 11.92 Z80A to TLC545 interface circuit diagram

The circuit shown in Fig. 11.92 initiates conversion with an OUT instruction. The timing diagram for this circuit is shown in Fig. 11.93. When $\overline{IORQ}$ and $\overline{WR}$ go low, $\overline{CS}$ is brought low while the universal shift register is placed in the load data mode and the system clock is enabled to the clock input to latch in the multiplexer address. The multiplexer address should occupy the five most significant bits of the shift register since the data is shifted out to the left. The rising edge of $\overline{IORQ}$ enables the I/O

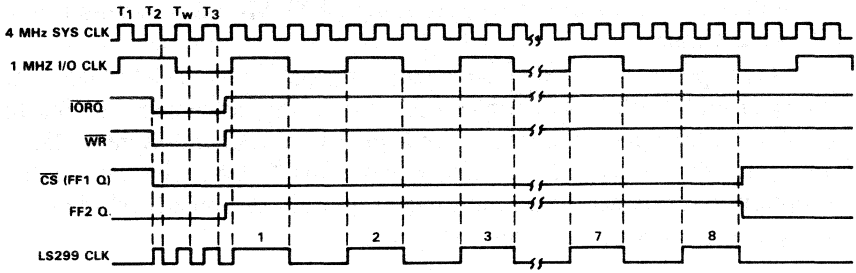


Fig. 11.93 Z80A to TLC545 interface timing diagram

clock to the shift register, which shifts the multiplexer address out while shifting in the previous conversion result. Sampling begins at the falling edge of the fifth I/O clock cycle and continues until the eighth falling edge occurs. At this time the I/O clock is disabled and $\overline{CS}$ is brought high to ensure that the TLC545 device will remain undisturbed during the conversion. Conversion of the addressed analog input requires 36 system clock cycles. During conversion time, previous conversion results may be read by using an IN instruction. A typical interrupt service routine is shown below.

```
ISR     EX AF,AF                ; Save accumulator
        IN A,(LS299)            ; Read previous conv. result
        LD (DATA), A            ; Store result
        LD A, (MUXADDRESS)      ; Load new mux address from RAM
        OUT (LS299),A           ; Initiate new conversion cycle
        EX AF,AF'               ; Restore accumulator
        EI
        RETI

        LD A, (MUXADDRESS)      ; Load mux address from RAM
        OUT (LS299),A           ; Sample channel & initiate conv.
        LD C,03H                ; Initialize counter
WAIT:   DEC C                   ; Decrement counter
        JP NZ, WAIT             ; If not zero keep waiting
        OUT (LS299),A           ; Shift results into LS299
        IN A,(LS299)            ; Read conv results
```

NOTE: A count of 03H will produce a delay of 10.50 μs, suitable for 4 MHz operation, while a count of 05H produces a delay of 17.50 μs, suitable for 2.5 MHz operation.

At least 36 system clock cycles must be used for conversion to ensure proper operation. If a new multiplexer address is shifted in while a conversion is in progress, the ongoing conversion will be aborted and a new conversion will be initiated at the falling edge of the eighth I/O clock cycle. Another software approach is to initiate a new conversion, wait in a delay loop until the conversion is complete, and then read in the previous conversion results. A sample program segment using the delay loop method precedes this discussion. Using this method, a conversion cycle can be completed in 21.5 μs.

TLC545 A/D Converter Interface to Zilog Z80A Microprocessors Using Software Control

This application describes a technique for operating the TLC545 A/D converter with the Z80A microprocessor using software generated control signals. These signals are $\overline{CS}$, address in, I/O clock, and system clock. Timing for the TLC545 device is identical to the TLC540 device with the exception that one additional multiplexer address bit must be shifted out to address the additional analog inputs.

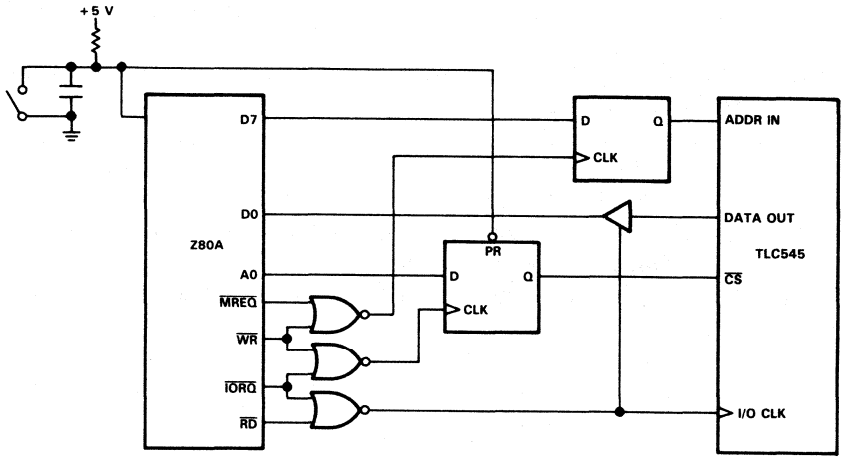


Fig. 11.94 Z80A to TLC545 software interface circuit diagram

The circuit diagram for the software controlled Z80A microprocessor to TLC545 device interface is shown in Fig. 11.94. Circuit timing is shown in Fig. 11.95.

Execution of an IN instruction causes the $\overline{RD}$ line and the $\overline{IORQ}$ line to become active and shift an address bit in and a data bit out of the TLC545 device. $\overline{CS}$ is brought low by latching in a low from address bit A0 on the positive going edge of the $\overline{WR}$ signal. A simple program segment listing that shifts out a new analog multiplexer address while also shifting in previous conversion results follows the discussion.

This program segment uses the B register to store the conversion result, the C register as a bit counter, and the D register to hold the analog multiplexer address. The analog multiplexer address is shifted left out of

the D register, therefore, the 5-bit address must be placed in the five most significant bits of the byte.

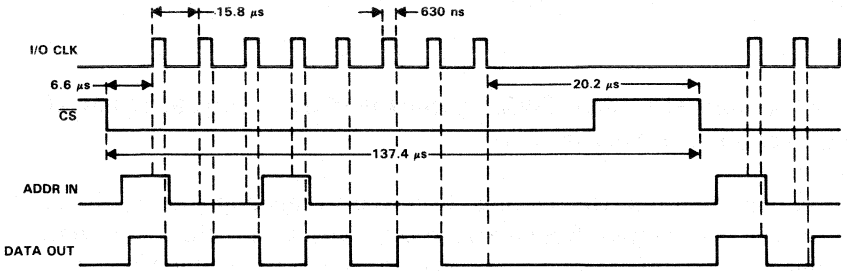


Fig. 11.95 Z80A to TLC545 software interface timing diagram

```
;          *** REGISTER ASSIGNMENTS ***
;
CSLOW      .EQU 0100H        ; A0=0
CSHIGH     .EQU 0101H        ; A0=1
BIT        .EQU 00FFH        ; Arbitrary address
;
;          *** MAIN PROGRAM ***
;
           LD C,08           ; Initialize bit counter
           LD B,00           ; Clear result register
           OUT (CSLOW),A     ; Bring chip select low
LOOP       RLC B             ; Rotate result left
           LD (HL),D         ; Latch address bit into D FF
           IN A, (BIT)       ; Read in data bit
           AND 01H           ; Mask off bit 0
           OR B              ; Or new bit into result reg.
           LD B,A            ; Store in result register
           RLC D             ; Shift address left
           DEC C             ; Decrement bit counter
           JP NZ,LOOP        ; Get another bit if not zero
           OUT (CSHIGH),A    ; Bring chip select high
```

Conversion results are read in one bit at a time and then shifted left to the proper position in the B register. Sampling of the addressed input begins at the falling edge of the fifth I/O clock and continues until the falling edge of the eighth I/O clock when conversion begins. Conversion requires 36 system clock cycles, therefore, an appropriate software delay that depends upon the system clock frequency must be included. If a new multiplexer address is shifted into the TLC545 device before a conversion has been completed, the ongoing conversion will be aborted and a new conversion cycle will begin at the eighth falling edge of the I/O clock. $\overline{CS}$ is brought high after the eighth falling edge of the I/O clock to ensure that extraneous noise or glitches on the I/O clock line are not interpreted as the beginning

of a new cycle. Using the program segment just presented with the system clock at 4 MHz, it is possible to initiate a new conversion cycle and read the results of the previous conversion in 138 μs as indicated in Fig 11.95.

TLC545 and TLC546 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Parallel Ports

For the two interface circuits discussed in this application, the A/D system clock is derived from the microcontroller's ALE clock for Interface 1 and from the microcontroller's crystal oscillator for Interface 2.

These interfaces minimize the amount of hardware and rely mainly upon software techniques. Although the amount of hardware and its cost are reduced, the use of more software increases the time required to load the address into, and retrieve the conversion data from, the A/D converter. However, the trade-off of minimum hardware versus longer conversion time may benefit many designs.

Interface 1 (ALE CLOCK)

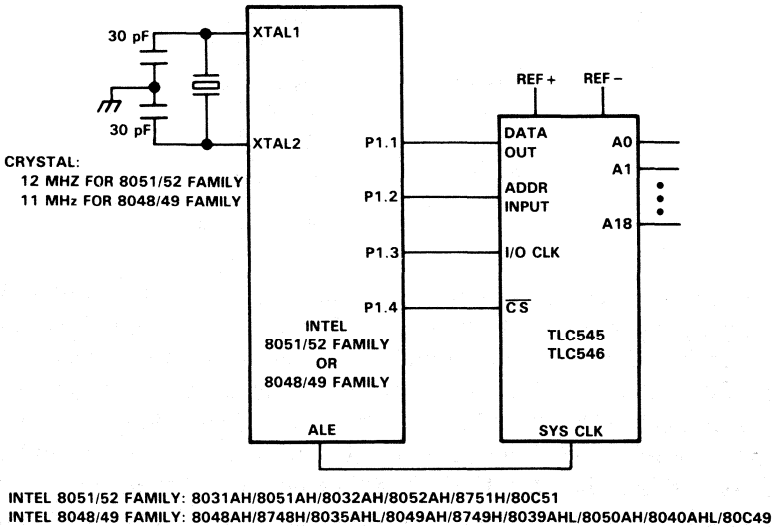


Fig. 11.96 TLC545 and TLC546 to Intel 8051/52 microcontroller parallel port Interface 1 diagram (clock signal from ALE)

The interface shown in Fig. 11.96 will always work with the TLC545 device, but will not work with the TLC546 device at the higher microcontroller instruction cycle frequencies. Before using the TLC546 device, the designer must verify that the high and low pulse widths of the ALE clock signal meet the specifications of the TLC546 device. These pulse widths are dependent upon the microcontroller instruction cycle frequency.

Interface 2 (CRYSTAL CLOCK)

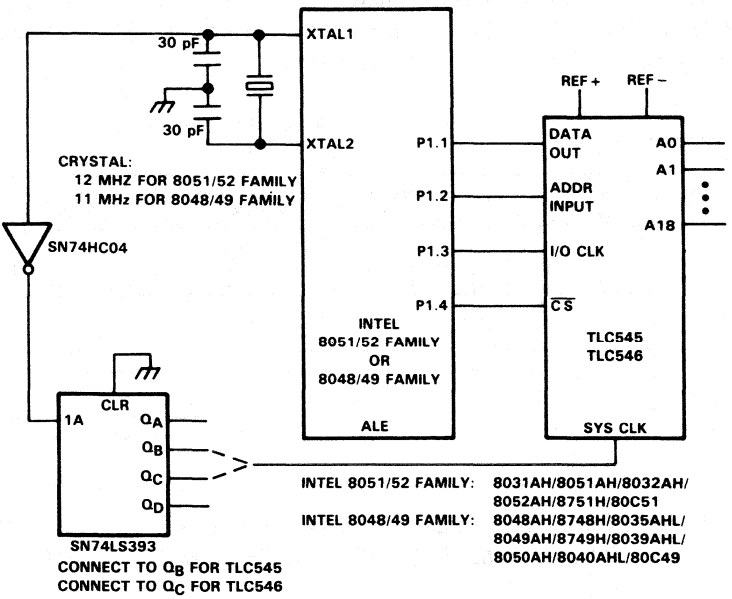


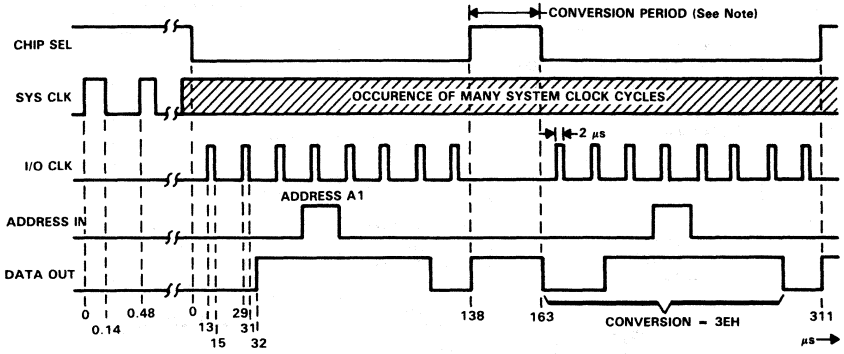
Fig. 11.97 TLC545 and TLC546 to Intel 8051/52 microcontroller parallel port Interface 2 diagram (clock signal from crystal osc.)

In the A/D converter of Fig. 11.97, the system clock for the A/D converter is obtained from the microcontroller crystal oscillator. To assure proper operation of the crystal oscillator, a high impedance buffer must be used to tap the signal from the oscillator. An important detail is that low and high level input buffer requirements must lie within the range of the oscillator signal. This compatibility will prevent missing edge transitions in the A/D converter's system clock signal. Subsequently, the buffered crystal oscillator signal must be frequency divided to assure that the resulting system clock signal does not exceed the upper frequency specification of

the A/D converter. Any convenient divider circuitry may be used to accomplish this task.

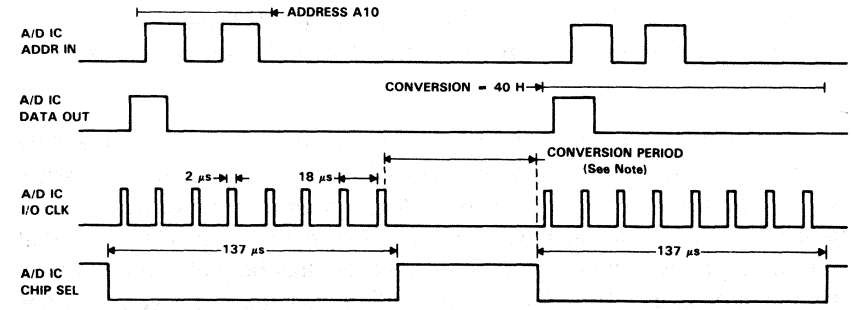
Timing Diagrams

Figs 11.98 and 11.99 show the timing diagrams for the A/D conversions of Interfaces 1 and 2, respectively. Loading the new address and retrieving the conversion for the previous address requires 137 μs maximum. The conversion period requires 36 system clocks. This period is longer for the TLC546 device since the microprocessor clock frequency (see Fig. 11.96) or the A/D clock frequency (see Fig. 11.97) must be lower to satisfy the TLC546 device specifications.



NOTE: Conversion period requires 36 system clock cycles after 8th I/O clock goes low upon clocking in the address.

Fig. 11.98 Timing diagram of an A/D conversion cycle for TLC545 and TLC546 to Intel 8051/52 parallel port Interface 1



NOTE: Conversion period requires 36 system clock cycles after 8th I/O clock goes low upon clocking in the address.

Fig. 11.99 Timing diagram of an A/D conversion cycle for TLC545 and TLC546 to Intel 8051/52 parallel port Interface 2

Software

Refer to the software listings for Interfaces 1 and 2 for the following discussion. The software uses a subroutine, S545D, that simultaneously loads the A/D converter with a new address and retrieves the conversion result for the previous address. This subroutine can be used any time the designer desires to load a new address and retrieve a conversion value. This simultaneous loading and retrieving makes the subroutine very appropriate for continuous monitoring of several A/D converter analog inputs. The subroutine assumes that the new address has been previously placed in the five most significant bits of register R2. Upon completion of the subroutine, the conversion result for the previous address is left in the accumulator.

Software Listing for Intel 8052/52-TLC545/546 Parallel Interface 1

```
                CLR P1.3          ;Lower I/O clock
                MOV R2,#10H       ;Load A/D analog input address. Note
                                  ;that to send address=A2, R2=10H.
                ACALL S545D       ;Load 545 address, assumes the address
                                  ;is currently in R2.
;
                MOV R3,#09H       ;This software loop allows a
                                  ;conserative 40 A/D system clocks,
                                  ;since only 36
DELAY:          DJNZ R3,DELAY     ;clocks are required to perform
                                  ;conversion, to be emitted from the
                                  ;microprocessor ALE pin
;
                MOV R2,#10H       ;Load A/D analog input address. Note
                                  ;that to send address=A2, R2=10H.
                ACALL S545D       ;Load new 545 address, assumes this
                                  ;address is in R2; leaves the
                                  ;conversion result for the previous
                                  ;address in A
```

Software Listing for Intel 8052/52-TLC545/546 Parallel Interface 2

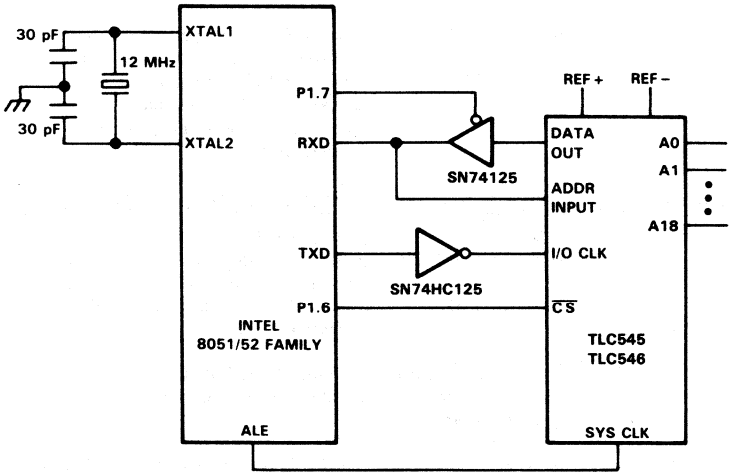
```
                CLR P1.3          ;Lower I/O clock
                MOV R2,#50H       ;Load A/D analog input address. Note
                                  ;that to send address=A10, R2=50H.
                ACALL S545D       ;Load 545 address, assumes the address
                                  ;is currently in R2.
;
;               A delay must occur here to allow the A/D IC to complete
;               conversion. The delay must allow 36 A/D IC system clock
;               cycles to occur.
;
                MOV R2,#50H       ;Load A/D analog input address. Note
                                  ;that to send address=10, R2=50H.
                ACALL S545D       ;Load new 545 address, assumes this
                                  ;address is in R2; leaves the
                                  ;conversion result for the previous
                                  ;address in A
```

Subroutine S545D for Intel 8051/52−Parallel Interface 1 and 2

```
S545D       MOV R6, #08H      ;Set bit counter to 8
            CLR P1.4          ;Lower chip select
S545DL:     CLR C             ;Initialize C = 0
            JNB P1.1,S545DI   ;If 545 Data Out = 0; branch
            CPL C             ;545 Data Out = 1; set C = 1
S545DI:     MOV A,R2          ;Get serial buffer
            RLC A             ;Shift Data Out bit into serial
                              ;buffer and shift 545 address
            MOV R2,A          ;Store serial buffer
            JNC S545DWO       ;If 545 address bit = 0; branch
            ORL P1, #04H      ;Set 545 address line to 1
            SJMP S545DWE      ;Go and raise the I/O clock
S545DWO:    ANL P1, #FBH      ;Set 545 address line to 0
S545DWE:    NOP               ;Allow address line to setup
            CPL P1.3          ;Raise I/O clock
            NOP               ;Delay to slow I/O clock
            CLR P1.3          ;Lower I/O clock
            DJNZ R6,S545DL    ;Do all 8 bits
            CPL P1.4          ;Raise chip select
            MOV A,R2          ;Get serial buffer
            RET
            END
```

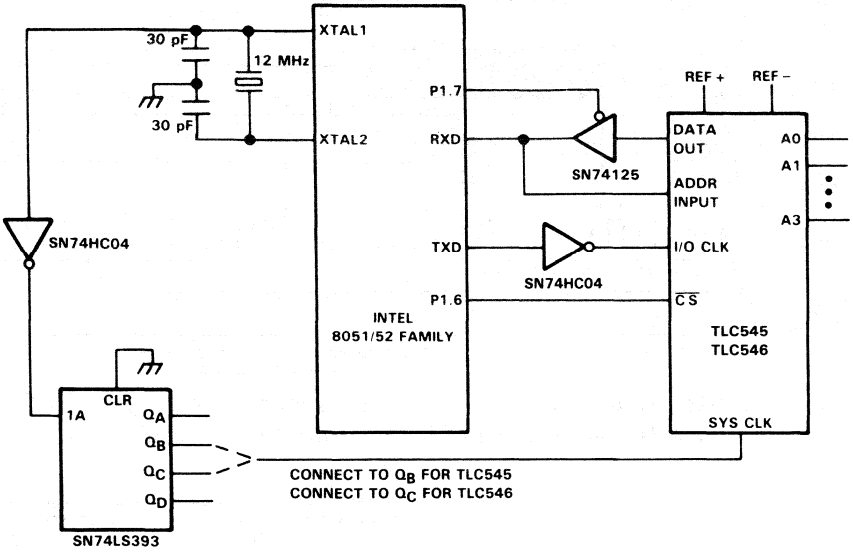
TLC545 and TLC546 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Serial Ports

A/D system clock for Interface 1 is derived from the microcontroller's ALE clock and for Interface 2 from the microcontroller's crystal oscillator.



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751AH/80C51

Fig. 11.100 TLC545 and TLC546 to Intel 8051/52 microcontroller serial port Interface 1 diagram (clock signal from ALE)



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751AH/80C51

Fig. 11.101 TLC545 and TLC546 to Intel 8051/52 microcontroller serial port Interface 2 diagram (clock signal from crystal oscillator)

Hardware

The circuit for Interface 1 is shown in Fig. 11.100 and that for Interface 2 in Fig. 11.101. The signal at the microcontroller's TXD pin must be inverted so the communication format for the microcontroller serial port and the A/D converter are compatible. Use of a 3-state buffer allows the microcontroller serial port to be used for both transmission and reception. Use of the 3-state buffer prevents the simultaneous loading of a new address while retrieving the conversion of a previously loaded address. However, this time loss is small because the serial port can quickly load an address and retrieve the resulting conversion.

Timing Diagram

Figs 11.102 and 11.103 are the timing diagrams for an A/D conversion for each of the two circuits. With the TLC545 device, loading the address, waiting for conversion, and retrieving the conversion result requires 56 μs for Interface 1 and 48 μs for Interface 2. This time period is longer for the TLC546 device since the system clock frequency must be lower for this device.

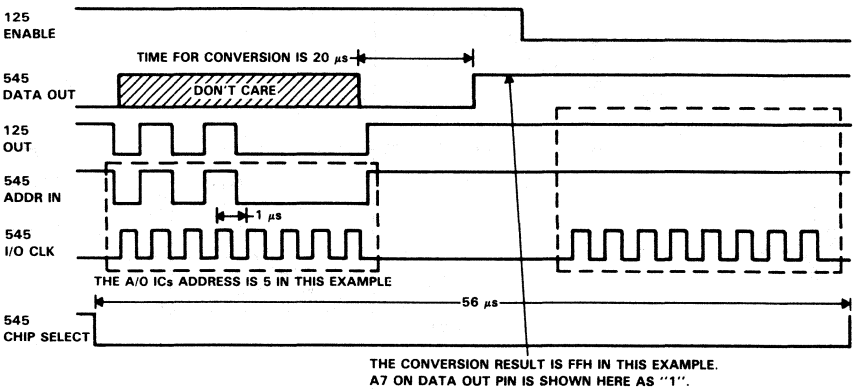


Fig. 11.102 Timing diagram of an A/D conversion cycle for TLC545 and TLC546 to Intel 8051/52 serial port Interface 1

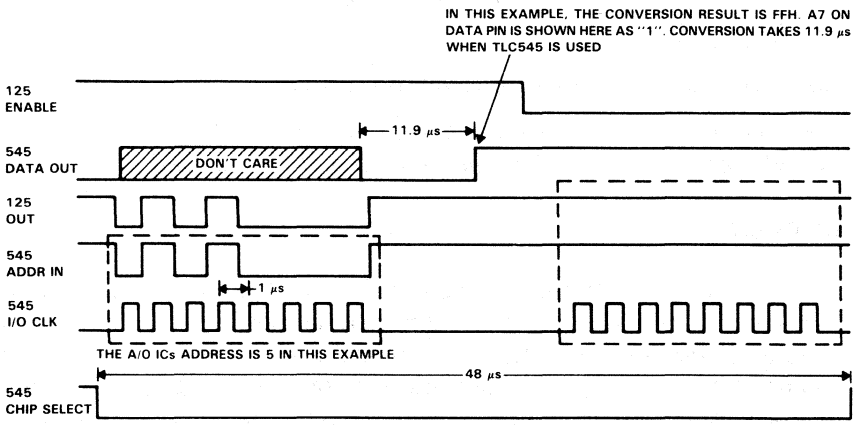


Fig. 11.103 Timing diagram of an A/D conversion cycle for TLC545 and TLC546 to Intel 8051/52 serial port Interface 2

Software

The combined software routines for Interface 1 and Interface 2 are listed below. The serial port mode 0 is used. If desired, the software can be incorporated into a subroutine so the designer can address the software with a simple subroutine call. Particular attention must be exercised when placing the address bits in the serial buffer since the serial port sends the least significant bit of the address first and the A/D converter accepts this bit as the most significant bit of the address. A similar process occurs when the serial port receives the conversion result.

Software Listing for Intel 8051/52−TLC545/546 Serial Interface 1 and 2

```
SR545L;         MOV A,#0AH          ;A/D IC address of A16 in this example
;
;The serial port will send 0A(00001010) with the least significant bit first. Therefore, the A/D
;IC will see (01010000), which will load an address of A16 into the control register.
;
                CLR P1.7            ;Disable 125
                CPL P1.7
                CLR P1.6            ;Lower chip select
                ANL SCON,#EDH       ;Reset REN & TI flags
                MOV SBUF,A          ;Send 545 address (LSB FIRST)
SNDTST:         JNB SCON.1,SNDTST   ;TI flag not set; branch
                                    ;until transmission is complete
;
;A delay must occur here to allow the A/D IC to complete
:conversion. The delay must allow 36 A/D IC system clock
;cycles to occur.
;
                MOV R3,#09H         ;This software loop allows a
                                    ;conservative 40 A/D system clocks,
                                    ;since only 36
DELAY:          DJNZ R3,DELAY       ;clocks are required to perform
                                    ;conversion, to be emitted from the
                                    ;microprocessor ALE pin
;
                CPL P1.7            ;Enable 125
                ORL SCON,#10H       ;Set REN
                ANL SCON,#FEH       ;Reset RI
RCVTST:         JNB SCON.0,RCVTST   ;RI FLAG not set; branch
                                    ;until reception is complete
                CPL P1.6            ;Raise chip select
                MOV A,SBUF          ;Set SBUF
;
;The serial port read reverses the data conversion bits coming
;to the microprocessor so that they are in the following order:
;b0(lsb),b1,b2,b3,b4,b5,b6,b7(msb). These bits (01234567) along
;with the carry bit (C) in the following instruction comments are
;presented so that the reader will understand the technique, which
;is used to place the bits in their proper order.
;
                RLC A               ; 6543210C 7; b7 is now in carry
                RLC A               ; 543210C7 6; b6 is now in carry
                MOV ACC.1,C         ; 54321067 6; put b6 into ACC.1
                MOV C,ACC.2         ; 54321067 0; put b0 into C
                RLC A               ; 43210670 5; b5 is now in carry
                MOV ACC.3,C         ; 43215670 5; put b5 into ACC.3
                MOV C,ACC.4         ; 43215670 1; put b1 into C
                RLC A               ; 32156701 4; b4 is now in carry
                MOV ACC.5,C         ; 32456701 4; put b4 into ACC.5
                MOV C,ACC.6         ; 32456701 2; put b2 into C
                RLC A               ; 24567012 3; b3 is now in carry
                MOV ACC.7,C         ; 34567012 3; put b3 into ACC.7
                RL A                ; 45670123  ; prepare for SWAP A
                SWAP A              ; 01234567  ; bits are ordered correctly
                                    ; conversion result is in accumulator
```

TLC545 A/D Converter Interface to Rockwell 6502 Microprocessors using the 6522 VIA

This application presents the design of an interface using the 6522 VIA. Another method using TTL gates is described in the following application. Cost and performance are the basic trade-offs between the two designs. The 6522 device interface is faster, but the TTL method costs less.

Principles of Operation

The interface circuit diagram is shown in Fig. 11.104. Timing for a data read cycle and an address write cycle is shown in Figs 11.105 and 11.106, respectively. Software listings for the initialization of the 6522 device and interface control follow the timing diagrams.

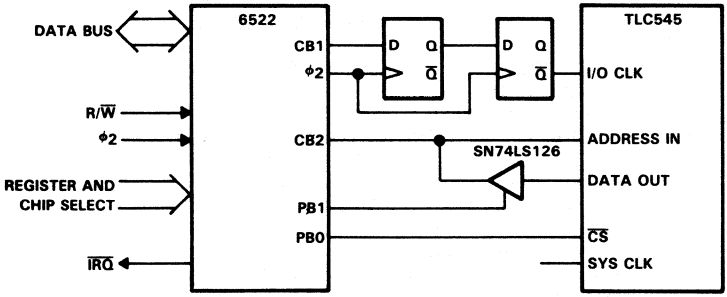


Fig. 11.104 6522 to TLC545 interface circuit diagram

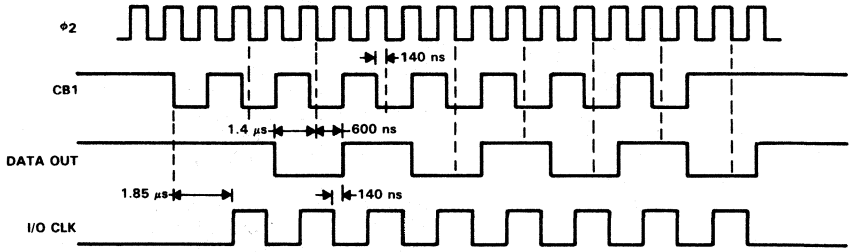


Fig. 11.105 6522 to TLC545 data read cycle timing diagram

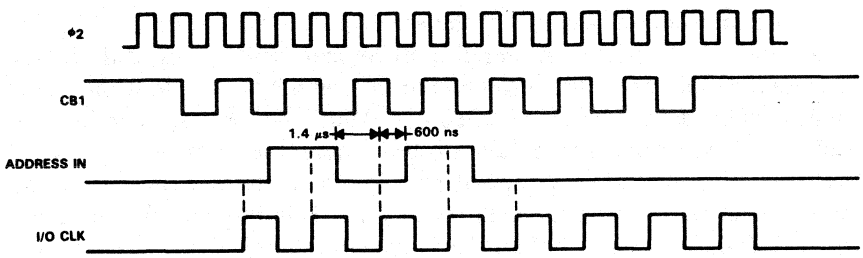


Fig. 11.106 6522 to TLC545 address write cycle timing diagram

```
;                REGISTER ASSIGNMENTS
;
ORB              .EQU 0000H
DDRB             .EQU 0002H
SR               .EQU 000AH
ACR              .EQU 000BH
PCR              .EQU 000CH
IFR              .EQU 000DH
IER              .EQU 000EH
;
;

                 LDA  #$03          ;

                 STA  DDRB          ; Initialize port B I/O pins
                 LDA  #$01          ;
                 STA  ORB           ; Bring chip select high
                 LDX  #$50          ; Initialize muxaddress = 01010


                 LDA  #$18          ; Shift out under phi 2 control
                 STA  ACR           ;
                 LDA  #$00          ;
                 STA  ORB           ; Disable data out, bring /CS low
                 STX  SR            ; Shift out muxaddress to 545
                 LDY  #$02          ; Load delay loop counter
DELAY1           DEY                ; Decrement delay counter
                 BNE  DELAY1        ; Branch if not zero
                 NOP                ;
                 LDX  #$02          ;
                 LDA  #$08          ;
                 STA  ACR           ; Shift in under phi 2 control
                 STX  ORB           ; Enable output of 74LS126
                 LDA  SR            ; Dummy load to shift results in
                 LDY  #$03          ; Load delay loop counter
DELAY2           DEY                ; Decrement delay loop counter
                 BNE  DELAY2        ; Branch if not zero
                 LDA  #$01          ;
                 STA  ORB           ; Disable data out, bring /CS high
                 LDA  SR            ; Read conversion results into 6502
```

The interface makes use of the serial port available on port B pins CB1 and CB2. Since the serial port is not capable of full duplex communication, previous conversion results cannot be read in while a new multiplexer address is being shifted out. Thus, the interface can be used only for individual conversion cycles where the port is configured dynamically; that is, as an output port for the address write cycle, and as an input port during the data read cycle. This requires the inclusion of an SN74LS126 3-state buffer. The D-type flip-flops are used to effectively delay the I/O clock to ensure that the set up and hold times for shifting data in and out are met. Port B pins PB0 and PB1 are used to generate $\overline{CS}$ and the output enable signal for the 3-state buffer.

A data conversion cycle begins by bringing $\overline{CS}$ low. This is accomplished by writing a low to PB0. The analog multiplexer address, which is stored in the five most significant bits of the X index register, is shifted out by writing to the SR register of the 6522 device. A delay loop is inserted to wait until the multiplexer address has been shifted out. The serial port is then configured as an input port to shift in the A/D conversion results. Conversion requires 36 system clock cycles, therefore, an appropriate delay loop dependent upon system clock frequency may be required. The output of the 3-state buffer is enabled by writing a high to PB1, and data is shifted into the SR register of the 6522 device. Again, a delay loop is included to wait until the data is shifted in. $\overline{CS}$ is then brought high, and the 3-state buffer is disabled to complete one data acquisition cycle. A data acquisition cycle can be completed in 55 μs.

TLC545 A/D Converter Interface to Rockwell 6502 Microprocessors using TTL Gates

This application presents the design of an interface using TTL gates. Another method using the 6522 VIA is described in the previous application. Cost and performance are the basic trade-offs between the two designs. The 6522 device interface is faster, but the TTL method costs less.

Principles of Operation

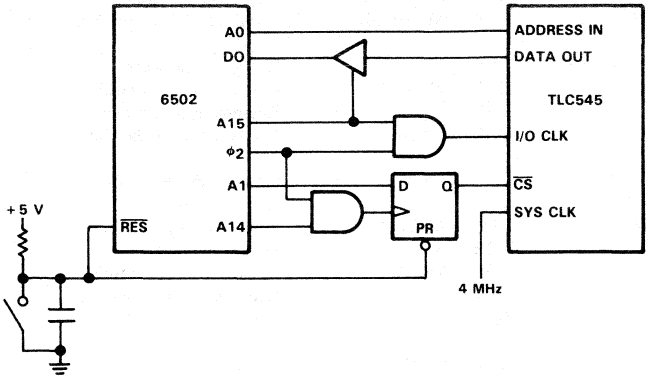


Fig. 11.107 6502 to TLC545 interface circuit diagram

The basic premise of the interface circuit shown in Fig. 11.107 is that all timing control signals are generated under software control. Circuit timing is shown in Fig. 11.108. This is followed by the control software listing.

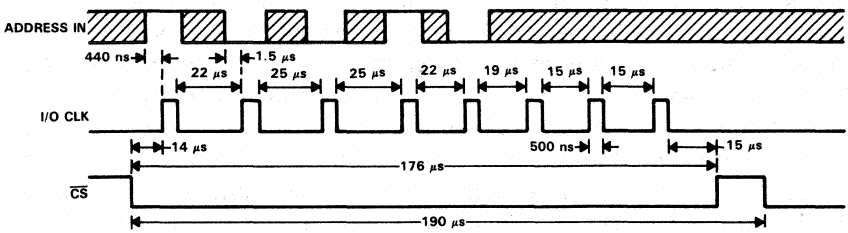


Fig. 11.108 6502 to TLC545 interface timing diagram

```
CSLOW       .EQU 4000H
CSHIGH      .EQU 4002H
ADDLOW      .EQU 8000H
ADDHIGH     .EQU 8001H
MUXADDRESS  .EQU 0000H
RESULT      .EQU 0001H
;
;
;
            LDA   #$90          ; Initialize muxaddress
            STA   MUXADDRESS    ; To 10010=channel 18
            STA   CSLOW         ; Bring /CS low
            LDX   #$05          ; Set bit counter for 5 MSB'S
```

```
LOOP1:      ROL   MUXADDRESS   ; Rotate muxaddress bit into carry
            BCS   SET          ; Branch if bit is set
            LDA   ADDLOW       ; Write out A low on A0, clock data in
            JMP   CONTINUE     ; Skip next instruction
SET:        LDA   ADDHIGH      ; Write out A high on A0, clock data in

CONTINUE:   ROR   A            ; Rotate do into carry
            ROL   RESULT       ; Rotate carry into result
            DEX                ; Decrement bit counter
            BNE   LOOP1        ; Go back for another bit

            LDX   #$03         ; Set counter for 3 LSB'S

LOOP2:      LDA   ADDLOW       ; Read in data bit
            ROR   A            ; Rotate into carry
            ROL   RESULT       ; Rotate carry into result
            DEX                ; Decrement bit counter
            BNE   LOOP2        ; Get another bit

            STA   CSHIGH       ; Bring /CS
```

A data conversion cycle is initiated by bringing $\overline{CS}$ low. This is accomplished by latching a low into the D-type flip-flop from address line A1 on the positive edge of system clock $\phi 2$. Address bit A14 is used as a gating signal to prevent the TLC545 device from being inadvertently selected during normal program execution. After $\overline{CS}$ is brought low, I/O clock pulses shift the multiplexer address that is stored in the five most significant bits of a byte in RAM while shifting out the previous conversion results. The I/O clock is enabled by gating the positive going pulse of $\phi 2$ to the TLC545 device I/O CLK input. The gating occurs by addressing a location so A15 is high. The high on A15 also enables the output of the 3-state buffer onto the data bus. Address bit A0 determines whether the multiplexer address bit is as high or a low. This multiplexer address bit is shifted into the TLC545 device on the positive edge of clock $\phi 2$ and a data bit is placed onto the data bus at this time by the TLC545 device. The data bit is latched into the 6502 device on the negative edge of clock $\phi 2$.

Once the data is loaded into the accumulator, it is rotated into the carry bit and then into memory. $\overline{CS}$ is brought high again by writing a high into the D-type flip-flop. This is done by placing a high on address bit A1. This cycle can be completed every 176 μs.

This interface circuit uses an address decoding scheme that requires a minimum of decoding hardware. Small modifications to the address decoding scheme may be necessary to fit the interface to a particular application.

TLC545 and TLC546 A/D Converter Interface to Motorola 6805 Microcomputers

This application describes techniques for operating the TLC545 and TLC546 A/D converters with the 6805 microcomputer.

Principles of Operation

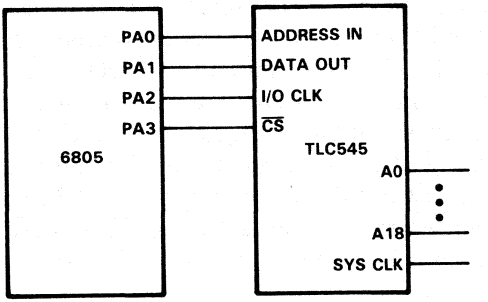


Fig. 11.109 6805 to TLC545 interface circuit diagram

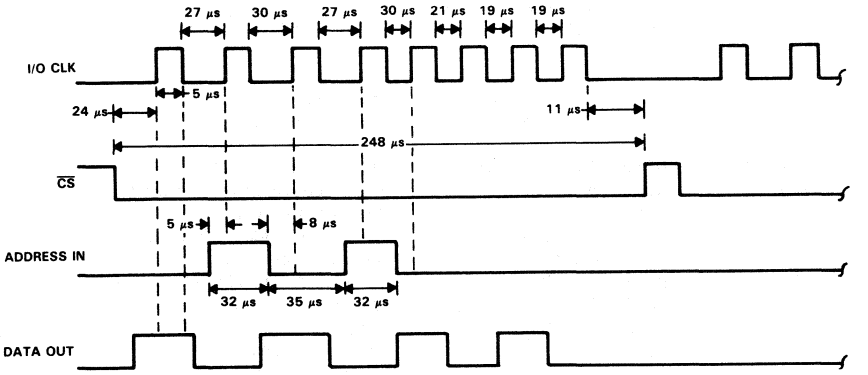


Fig. 11.110 6805 to TLC545 interface timing

Circuit diagram for the software controlled 6805 microcomputer to TLC545 and TLC546 device interface is shown in Fig. 11.109. Circuit timing is shown in Fig. 11.110. The software controlled interface makes use of four pins on port A of the 6805 microcomputer. Three of these pins are used as outputs to generate the $\overline{CS}$, I/O clock, and multiplexer address

input. The remaining pin is used as an input to receive the conversion results from the TLC545 and TLC546 devices. A software program segment illustrates a method that can be used to initialize the input/output pins.

A program listing which controls actual transfer of data follows the initialization method. This program block sends out the 5-bit analog multiplexer address that is shifted into the TLC545 and TLC546 devices on the first five I/O clock rising edges. Sampling of the addressed input begins at the falling edge of the fifth I/O clock and continues until the falling edge of the eighth I/O clock occurs. Conversion requires 36 clock cycles of the system clock, which can run up to 4 MHz to allow conversion in as little as 9 μs. Conversion results are shifted out of the TLC545 and TLC546 devices on the negative edge of the I/O clock and the program block leaves these results in the accumulator. With the 6805 device running at 5 MHz and the TLC545 device system clock at 4 MHz, one conversion cycle loads an analog multiplexer address and reads the results from the previous conversion. This can be completed in 248 μs.

```
PORTA     .EQU 0000H
DDRA      .EQU 0004H
          .ORG 0100H

START:    LDA   #0DH              ;
          STA   DDRA              ; Initialize port A I/O pins
          BSET  3,PORTA           ; Bring chip select high
          LDA   #50               ; Initialize multiplexer address


          LDX   #05               ; Load counter for first 5 bits
          BCLR  3,PORTA           ; Bring chip select low

LOOP1:    BRSET 1,PORTA,LABEL1    ; Read data bit into carry bit
LABEL1:   ROL   A                 ; Rotate muxadd bit and result bit
          BCS   SET               ; Go to set if muxadd bit is 1
          BCLR  0,PORTA           ; Write out A 0 to TLC545 address in
          JMP   CLOCK             ; Skip next instruction
SET:      BSET  0,PORTA           ; Write out A 1 to TLC545 address in
CLOCK:    BSET  2,PORTA           ; Bring I/O clock high
          BCLR  2,PORTA           ; Bring I/O clock low
          DECX                    ; Decrement counter
          BNE   LOOP1             ; Continue if counter is not zero
;
          LDX   #03               ; Load counter for last 3 data bits
;
LOOP2:    BRSET 1,PORTA,LABEL2    ; Read data bit into carry bit
LABEL2:   ROL   A                 ; Rotate new data bit into result
          BSET  2,PORTA           ; Bring I/O clock high
          BCLR  2,PORTA           ; Bring I/O clock low
          DECX                    ; Decrement counter
          BNE   LOOP2             ; Continue if counter is not zero
;
          BSET  3,PORTA           ; Bring chip select high
```

TLC548 and TLC549 8-Bit Analog-to-Digital Converters with Serial Output

The TLC548 and TLC549 are single input LinCMOS 8-bit A/D converters that share the same basic features as other members of the TLC540 family. This includes an on-chip sample and hold and serial control, using switch capacitor successive approximation technology. The TLC548/549 has a three line interface to a microprocessor and uses Chip Select ($\overline{CS}$) and Input/Output clock (I/O clock) inputs to control the three-state, Data Out, serial output. The addition of an on-chip system clock to these devices, which operates independently of the I/O clock, allows the use of a cost effective 8-pin package, Fig. 11.111, either dual-in-line or "small outline" surface mount.

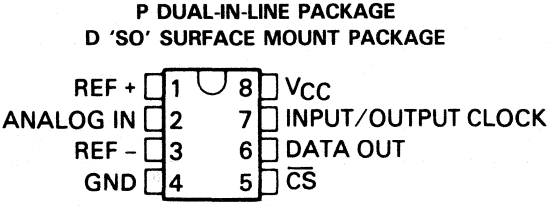


Fig. 11.111 TLC548 and TLC549 pinout (top view)

The guaranteed operation of TLC548 at a 2.048 MHz I/O clock gives it a faster access time than the TLC549 which is guranteed to operate at an I/O clock of 1.1 MHz. Both devices have a maximum total unadjusted error of ±0.5 LSB and operate over a 3 V to 6 V power supply range, see Table 11.15 for further performance details and Fig. 11.112 the Functional Block Diagram.

Table 11.15

Performance	**TLC548**	**TLC549**
Acquisition Time (max)	2 μs	3.6 μs
Conversion Time (max)	17 μs	17 μs
Sampling rate (max)	45.5 × 10^3sps	40 × 10^3sps
Power Dissipation (typ)	6 mW	6 mW

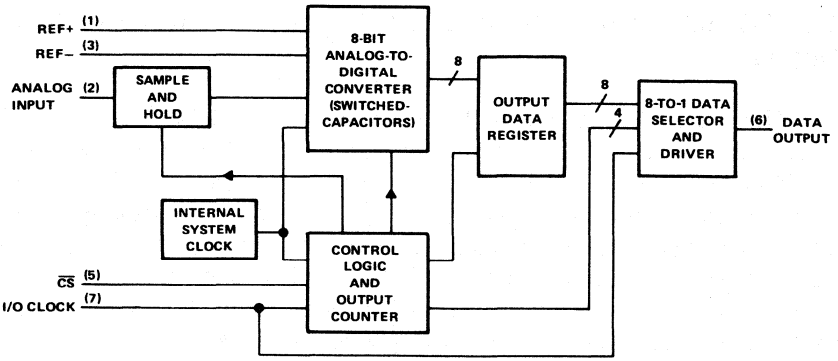
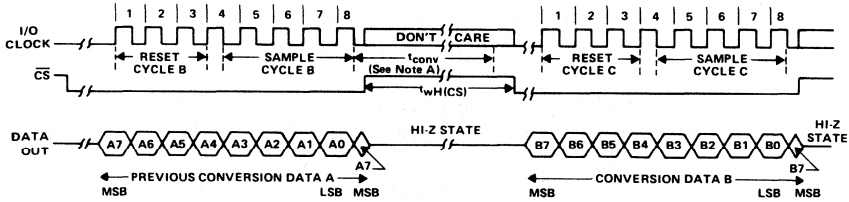


Fig. 11.112 Functional block diagram

Control Sequence



NOTE A: The conversion cycle, which requires 36 system clock periods, is initiated with the 8th I/O clock↓.

Fig. 11.113 Operating sequence

Operation is similar to that of the TLC540 (page 11.79) with the removal of the address input. The timing diagram of Fig. 11.113 shows the signals on the three control lines. When $\overline{CS}$ is high the Data Out pin is in a high impedance condition and the I/O clock input is disabled. An abbreviated description of the control sequence from that of the TLC540 (page 11.79) follows.

1. $\overline{CS}$ is brought low and waits for the two rising and one falling edge of the internal system clock before the high-low transition is recognized. This inhibits false enabling of Data Out due to noise on $\overline{CS}$. When $\overline{CS}$ is recognized the MSB of the previous conversion result will appear on Data Out.
2. The negative edges of the first four I/O clocks shift out the 2nd, 3rd, 4th and 5th MSB's of the previous conversion. The on-chip sample and hold begins sampling after the 4th falling edge.
3. The negative edges of the next three I/O clocks shift out the 6th, 7th and 8th MSB's of the previous conversion.

4. On the 8th and final negative I/O clock edge the sample and hold function starts, the conversion is completed in a time consisting of a maximum of 36 internal system clock cycles. After the 8th I/O clock cycle the I/O clock input must either remain low for the duration of the conversion time or $\overline{CS}$ must go high.

$\overline{CS}$ can be kept low for periods of multiple conversion, if held permanently low during operation, it must be taken from high to low at power-up to reset internal registers. $\overline{CS}$ may be taken high during a conversion but must remain so otherwise the conversion in progress will be aborted by the falling edge of $\overline{CS}$. A new conversion may be started, with the current conversion being aborted, by repeating steps 1 to 4 before the conversion time is over. The previous conversion result will be output and not the aborted conversion.

Delaying the fall of the negative edge of the 8th I/O clock until the required instant will allow direct control over the sample to hold transition, and hence sampling.

TLC549 A/D Converter Interface to Zilog Z80A Microprocessors

This application describes a technique for interfacing the TLC549 A/D converter to the Z80A microprocessor.

Principles of Operation

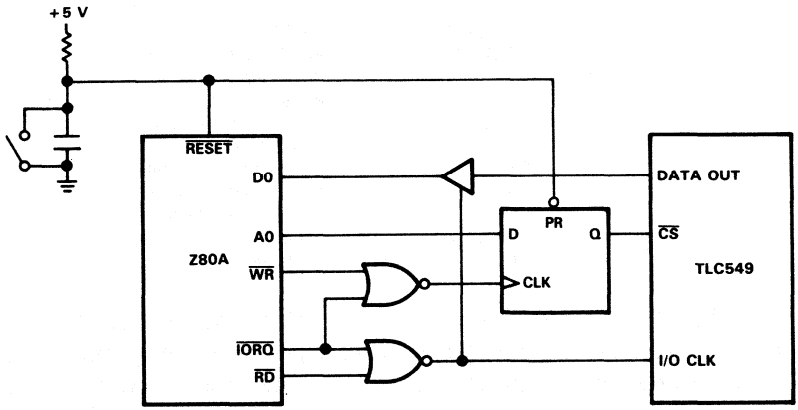


Fig. 11.114 Z80A to TLC549 interface circuit diagram

The circuit diagram for the TLC549 device to Z80A microprocessor interface is shown in Fig. 11.114. The circuit timing is shown in Fig. 11.115.

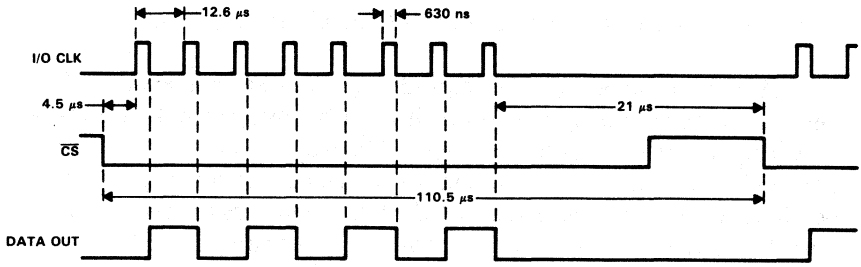


Fig. 11.115 Z80A to TLC549 interface timing diagram

$\overline{CS}$ is brought low by latching a low from address bit A0. Execution of an IN instruction causes $\overline{RD}$ and $\overline{IORQ}$ to become active and generate one I/O clock pulse. A data bit is read before the falling edge of the I/O clock and the falling edge shifts out the next data bit. Sampling of the analog input begins at the falling edge of the fourth I/O clock and continues until the falling edge of the eighth I/O clock. At that time, conversion begins and $\overline{CS}$ is brought high. Since $\overline{CS}$ high disables all inputs and outputs, conversion may proceed. Conversion requires 17 µs. A simple program segment that starts a conversion and reads in previous conversion results is shown in the software listing. If this program segment is placed in a loop, it is possible to initiate a conversion and read previous results in 111 µs.

```
        LD C,08H          ; Load bit counter
        LD B,00H          ; Initialize result register
        OUT (CSLOW),A     ; Bring chip select low
LOOP    RLC B             ; Rotate result reg. left
        IN A, (BIT)       ; Read in a bit & clock next
        AND 01H           ; Mask off bit 0
        OR B              ; Or new bit with result
        LD B,A            ; Store in results register
        DEC C             ; Decrement bit counter
        JP NZ,LOOP        ; Get another bit if not zero
        OUT (CSHIGH),A    ; Bring chip select high
```

Fig. 11.114 presents a software controlled interface for the Z80A and the TLC549 devices. It is possible to increase performance by introducing additional hardware into the interface. For more description of such an interface, refer to the **TLC540 A/D Converter Interface to Zilog Z80 and Z80A Microprocessors** application.

TLC549 A/D Converter Interface to Rockwell 6502 Microprocessors using the 6522 VIA

This application presents the design of an interface circuit for the TLC549 A/D to the 6502 microprocessor through the 6522 VIA. Interfacing techniques for the TLC549 are very similar to those for the TLC540. The interface presented in this application utilizes the 6522 VIA, and is similar to the **TLC540 A/D Converter Interface to Rockwell 6502 Microprocessors using the 6522 VIA** presented in a previous application. Since the TLC549 converts only one analog input, no multiplexer address is required, thus, less software is required to control the interface.

Principles of Operation

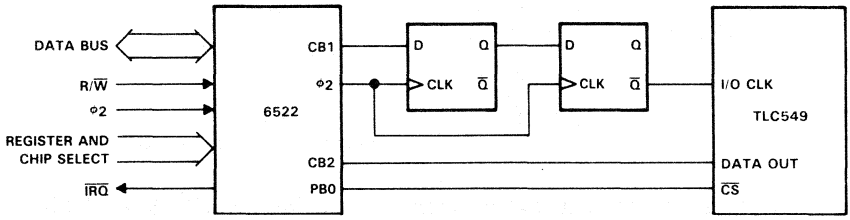


Fig. 11.116 6522 to TLC549 interface circuit diagram

The TLC549 to 6522 interface circuit is shown in Fig. 11.116. The timing diagram is shown in Fig. 11.117. An initialization software listing and an interface control software listing are included below.

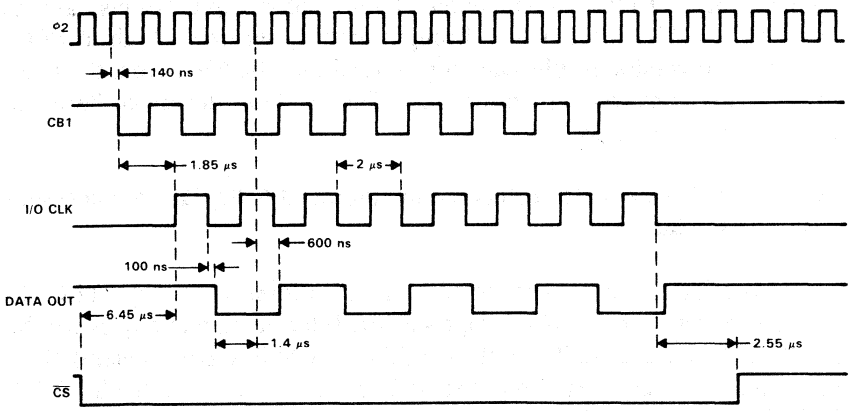


Fig. 11.117 6522 to TLC549 interface timing diagram

```
                    REGISTER ASSIGNMENTS
;
;
ORB         .EQU 0000H
DDRB        .EQU 0002H
SR          .EQU 000AH
ACR         .EQU 000BH
;
;
            LDA  #$01       ;
            STA  DDRB       ; Initialize port B I/O pins
            LDA  #$01       ;
            STA  ORB        ; Bring chip select high
            LDA  #$08       ; Shift in on phi 2
            STA  ACR        ;


            LDX  #$00       ;
            STX  ORB        ; Bring /CS low
            LDA  SR         ; Clock in previous results
DELAY:      LDY  #$03       ;
            DEY             ; Delay while results clocked in
            BNE  DELAY      ;
            LDX  #$01       ;
            STX  ORB        ; Bring /CS high
            LDA  SR         ; Load previous results into acc
```

A data conversion cycle begins by bringing $\overline{CS}$ low. This is accomplished by writing a low to port B output pin PB0. Previous conversion results are shifted in by reading the SR register of the 6522 (Fig. 11.116). The D-type flip-flops (Fig. 11.116) are included to effectively delay the I/O clock in order to meet all the set up and hold time requirements for shifting the data. A delay loop is included (Fig. 11.117) to allow for the data to be shifted in. On the eighth falling edge of the I/O clock, conversion of the analog input begins, and is completed in 17 μs. $\overline{CS}$ is then brought high by writing a high to PB0. Previous conversion results can then be read into the 6502 from the 6522 while a new conversion is in progress. One cycle as described here can be completed in 48 μs.

TLC549 A/D Converter Interface to Rockwell 6502 Microprocessors using TTL Gates

This application presents the design of an interface for the TLC549 A/D converter to the 6502 microprocessor. Interfacing techniques for the TLC549 device are similar to those of the TLC540 A/D peripheral chip. The interface uses a small number of TTL gates similar to the **TLC540 A/D Converter Interface to Rockwell 6502 Microprocessors using the TTL gates** presented in a previous application. Since the TLC549 device

converts only one analog input channel no multiplexer address is required. This reduces the amount of software code and execution time required.

Principles of Operation

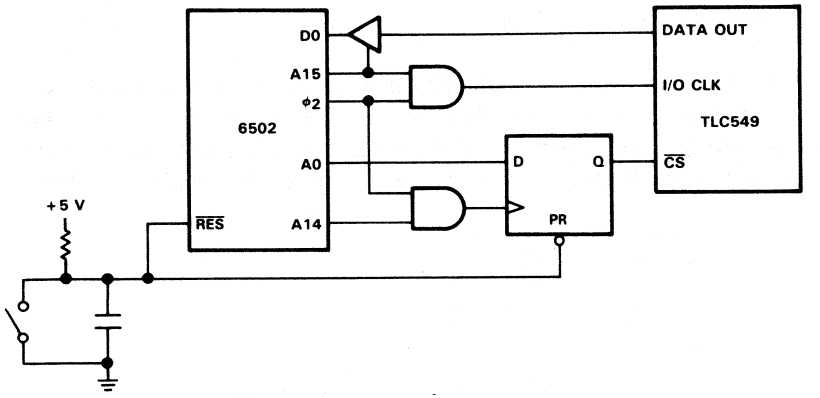


Fig. 11.118 6502 to TLC549 interface circuit diagram

The TLC549 device to 6502 microprocessor interface circuit is shown in Fig. 11.118. The timing diagram is shown in Fig. 11.119.

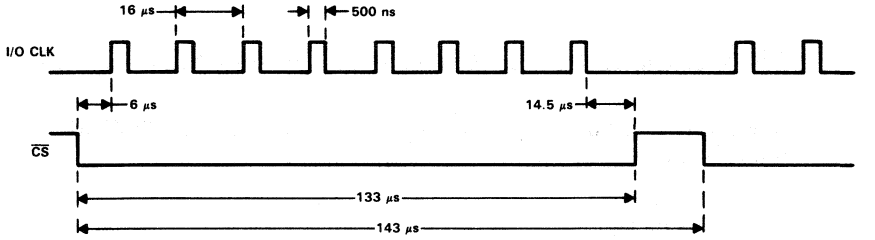


Fig. 11.119 6502 to TLC549 interface timing diagram

A data conversion cycle is initiated when $\overline{CS}$ is brought low. This occurs when a low on address line A0 is latched into the D-type flip-flop on the positive edge of clock $\phi 2$. Address bit A14 is used as a gating signal to prevent the TLC549 device from inadvertently being selected during normal program execution.

Data is shifted out of the TLC549 device on the positive edge of the I/O clock which is generated by gating the positive going edge of clock $\phi 2$ with address line A15. When A15 goes high, the output of the 3-state

buffer is enabled onto the data bus and the conversion result bit is shifted into the 6502 device on the trailing edge of clock $\phi 2$. Once the data is in the accumulator, it is rotated into the carry bit, and then rotated into memory. $\overline{CS}$ is then brought high by latching a high into the D-type flip-flop. This is done by placing a high on address bit A0. This cycle can be completed every 133 μs.

It should be noted that this circuit uses an address decoding scheme that requires a minimum of decoding hardware. Small modifications to the decoding scheme may be necessary to match the interface properly to a particular application. The software listing for this application follows.

```
CSHIGH      .EQU 4001H
CSLOW       .EQU 4000H
RESULT      .EQU 0000H
CLOCK       .EQU 8000H
;
;
;
            STA  SLOW       ; Bring /CS low
            LDX  #$08       ; Initialize bit counter
LOOP        LDA   LOCK      ; Clock in data bit
            ROR  A          ; Rotate do into carry bit
            ROL  RESULT     ; Rotate carry bit into result
            DEX             ; Decrement bit counter
            DNE  LOOP       ; Get another bit if not zero
            STA  CSHIGH     ; Bring /CS high
```

TLC549 A/D Converter Interface to Motorola 6805 Microcomputers

This application describes techniques for operating the TLC549 A/D peripheral chip with the 6805 microcomputer.

Principles of Operation

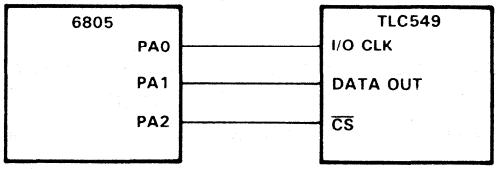


Fig. 11.120 6805 to TLC549 interface circuit diagram

The circuit diagram for the software controlled TLC549 device to 6805 microcomputer interface is shown in Fig. 11.120. Circuit timing is shown in Figure 11.121.

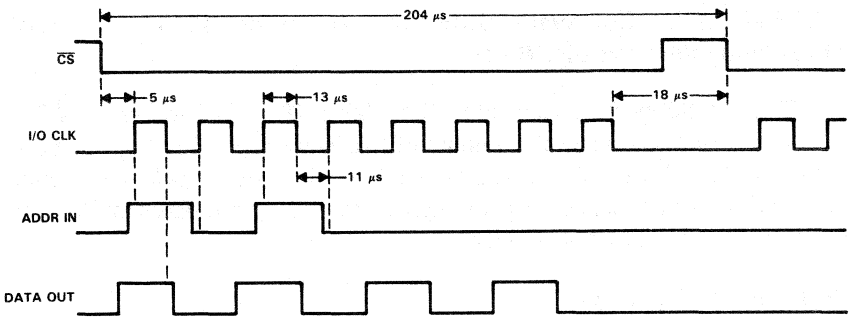


Fig. 11.121 6805 to TLC549 interface timing diagram

The software controlled interface makes use of three pins on port A of the 6805 microprocessor. Two of these lines are used as outputs to generate $\overline{CS}$ and I/O clock. The other pin is used as an input to receive the conversion results from the TLC549 device. A short program listing that can be used to initialize the input/output pins is given below.

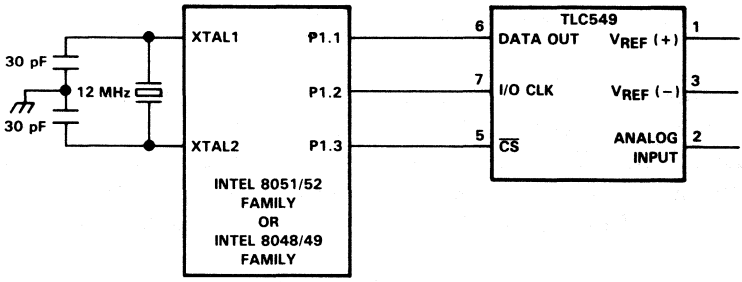
A program listing that controls the actual transfer of data is also shown below. This program block brings $\overline{CS}$ low and generates the I/O clock that shifts the previous conversion results out of the TLC549 device. Sampling of the analog input begins at the falling edge of the fourth I/O clock and continues until the falling edge of the eighth I/O clock occurs. When sampling ends, conversion begins and is completed in 17 μs. With the 6805 device running at 5 MHz, one conversion cycle that initiates a new conversion and reads the results of the previous conversion can be completed in 204 μs.

```
PORTA     .EQU  0000H
DDRA      .EQU  0004H
          .ORG  0100H
START:    LDA   #05H          ;
          STA   DDRA          ; Initialize port a I/O pins
          BSET  2,PORTA       ; Bring chip select high



          LDX   #08           : Initialize counter
          BCLR  2,PORTA       ; Bring chip select low
LOOP:     BSET  0,PORTA       ; Bring I/O clock high
          BRSET 1,PORTA,LABEL ; Read data bit into carry bit
LABEL:    ROLA                ; Rotate carry into accumulator
          BCLR  0,PORTA       ; Bring I/O clock low
          DECX                ; Decrement counter
          BNE   LOOP          ; Continue if counter is not zero
          BSET  2,PORTA       ; Bring chip select high
```

TLC549 A/D Converter Interface to Intel 8048 and 8049 Microcontroller Parallel Ports

The parallel port interface of the Intel 8048/49 to the low cost TLC549 A/D converter minimises the amount of hardware required to operate this serial interface device. The inclusion of an on-chip system clock and deletion of the address line makes further simplifications over the other multi-input members of the TLC540 family.

Principles of Operation



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751AH/80C51
INTEL 8048/49 FAMILY: 8048AH/8748H/8035AHL/8049AH/8749H/8039AHL/8050AH/8040AHL/80C49

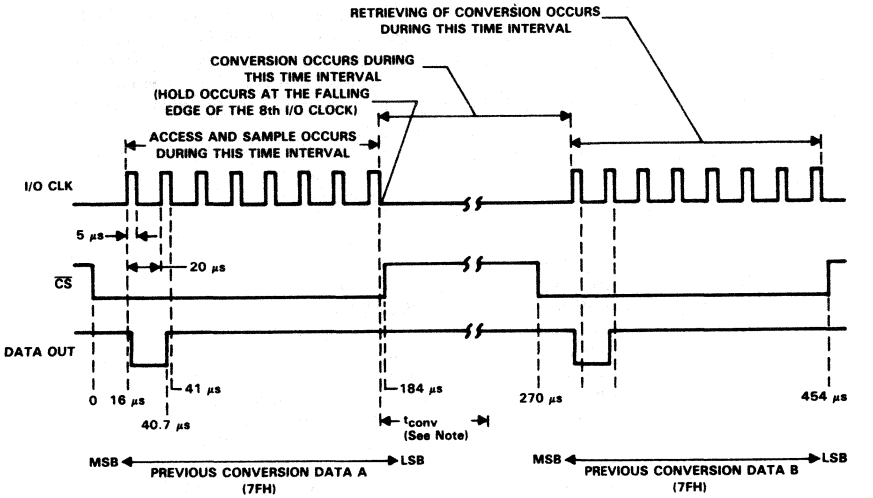
Fig. 11.122 The Intel 8051/52 or 8048/49 parallel port to TLC549 interface

Fig. 11.122 shows the circuit for the 8048 and 8049 microcontroller parallel port interface to the TLC549 device which operates under software control. This interface uses only three port pins of the microcontroller. While the TLC549 is outputting conversion results and concurrently the sample and hold is acquiring a new analog signal, Port 1.3 and $\overline{CS}$ must be low. I/O clock is transmitted through Port 1.2 to pin 7 of TLC549 and this causes conversion result data to be communicated from the Data Out pin to Port 1.1.

Timing Diagram

Fig. 11.123 shows the timing diagram for the interface. The timing diagram indicates that one complete conversion cycle requires 454 μs.

After $\overline{CS}$ goes low, eight I/O clocks access and sample the new analog signal from the analog input. At the same time, I/O clock falling edges shift out the previous conversion result. This conversion result is only valid for the previous analog signal sample.



NOTE; t_{conv}: The conversion cycles, which requires 36 internal system clock periods, is initiated with the 8th I/O clock↓ after $\overline{CS}$↓. Conversion time requires a minimum of 17μs.

Fig. 11.123 Timing diagram for the Intel 8048/49 parallel port – TLC549 interface

Conversion occurs in the time interval starting when the eighth I/O clock goes low. The conversion time interval requires 36 internal system clock cycles. The designer cannot see the system clock, but the designer can obtain the conversion time from the data sheets. Typically, a time interval of 17 μs is required for conversion.

See the **TLC540 and TLC541 A/D Converter Interface to Intel 8048 and 8049 Microcontrollers Parallel Ports** application for more details.

Software

The interface program listing follows and can be used as a subroutine using a simple call as shown.

```
                        ; Software Listing for Intel 8048/49
                        ; Interface — TLC549
                        ;
                        ;
                        ;
CALL S549D              ; Access, sample and hold the new
                        ; analog signal.
                        ;
```

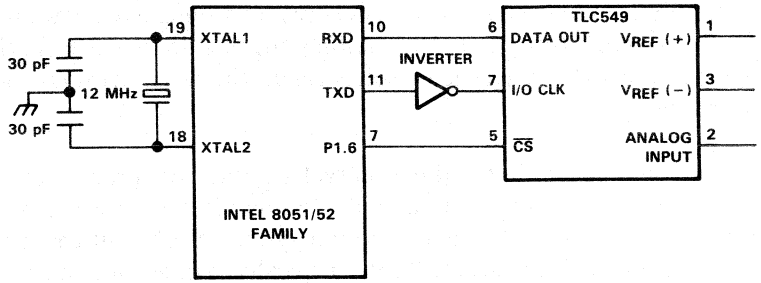
```
                                ; A delay must occur here to allow the A/D
                                ; chip to complete conversion. The delay
                                ; must allow 36 A/D chip internal system
                                ; clock cycles to occur. Conversion time
                                ; requires a minimum of 17 microseconds.
                                ;
                CALL S549D      ; Access, sample and hold the new
                                ; analog signal. Bring out the previous
                                ; conversion result.
                                ;
                                ; Subroutine CALL
                                ;
S549D           ANL P1, #FBH    ; Lower I/O clock
                MOV R0, #08H    ; Set bit counter to 8
                ANL P1, #F7H    ; Lower Chip Select
S549H           CLR C           ; Initialize C = 0
                IN A,P1         ; Get Port 1
                CPL A           ; Compliment Accumulator
                JB1 S549J       ; If 549 data out = 0; branch
                CPL C           ; 549 data out = 1; Set C = 1
S549J           MOV A,R1        ; Get serial buffer
                RLC A           ; Shift data out bit into serial buffer
                MOV R1,A        ; Store serial buffer
                ORL P1, #04H    ; Raise I/O clock
                NOP             ; Delay to slow I/O clock
                ANL P1, #FBH    ; Lower I/O clock
                DJNZ R0,S549H   ; Do 8 times
                ORL P1, #08H    ; Raise Chip Select
                MOV A,R1        ; Conversion data in A
                RET             ;
                END             ;
```

TLC549 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Serial Ports

Since the TLC549 device needs only the I/O clock input and $\overline{CS}$ input for data control, this serial port interface offers another simple interface example. Using the dedicated serial port provides faster communication than parallel port implementations. With minimum hardware, this allows excellent performance from the TLC549 with its low pin count and on-chip system clock and provides a cost effective solution.

Principles of Operation

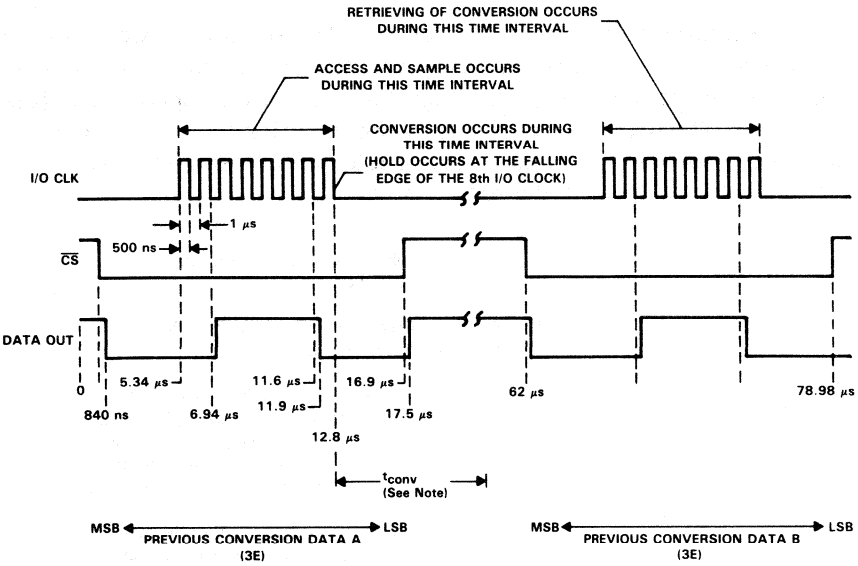
Fig. 11.124 shows the 8051 and 8052 device serial port to the TLC549 device interface circuit. By using the inverted TXD shift clock as an I/O clock for the A/D converter, previous conversion data can be transferred serially from the TLC549 device to the RXD pin of the microcontroller.



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751H/80C51
INVERTER: 74HC04, 74LS04

Fig. 11.124 The Intel 8051/52 serial port to TLC549 interface

Timing Diagram



NOTE; t_{conv}: The conversion cycles, which requires 36 internal system clock periods, is initiated with the 8th I/O clock↓ after $\overline{CS}$↓. Conversion time requires a minimum of 17µs.

Fig. 11.125 Timing diagram for the Intel 8051/52 serial port to TLC549 interface

Fig. 11.125 shows the timing diagram for a complete A/D conversion. The timing diagram shows that one complete conversion cycle requires 78.98 µs.

After $\overline{CS}$ goes low, eight I/O clocks access and sample the new analog input. At the same time, I/O clock falling edges shift out the previous conversion result. This conversion result is valid only for the previous analog signal sample.

Conversion occurs in the time interval starting when the eighth I/O clock goes low. The conversion time interval requires 36 internal system clock cycles. The designer cannot see the system clock, but can obtain the conversion time from the data sheets. Typically, a time interval of 17 μs is required for conversion.

See the **TLC540 and TLC541 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers Serial Ports** application for more details.

Software

The interface software listing follows. The serial port Mode 0 state is used to permit 8-bit transmission and reception. Note that the A/D converter device sends the most significant bit of the conversion result first and the serial buffer receives this bit as the least significant bit. The latter part of the software program is responsible for reversing the conversion bits and placing them in the proper order.

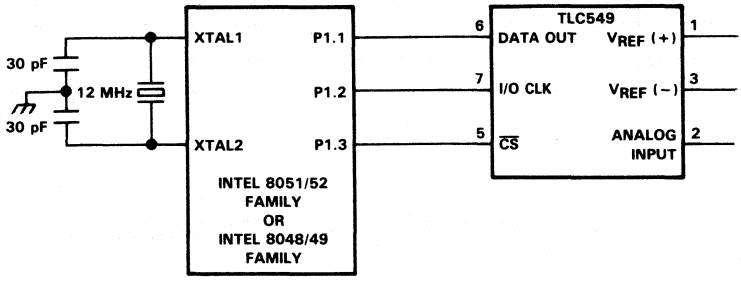
```
                    ; Software Listing for Intel 8051/52
                    ; Serial Port — TLC549
                    ;
ACALL SR549D        ; Access and sample and hold the new
                    ; analog signal.
                    ;
                    ; A delay must occur here to allow the A/D
                    ; chip to complete conversion. The delay
                    ; must allow 36 A/D chip internal system
                    ; clock cycles to occur. Conversion time
                    ; requires a minimum of 17 microseconds.
                    ;
ACALL SR549D        ; Access and sample and hold the new
                    ; analog signal. Bring out the previous
                    ; conversion result.
                    ;
                    ; The serial port read reverses the data con-
                    ; version bits coming to the microprocessor
                    ; so that they are in the following order:
                    ; b0(LSB),b1,b2,b3,b4.b5.b6.b7
                    ; (MSB). These bits (01234567) along with
                    ; the Carry bit (C) in the following
                    ; instruction comments are presented so
                    ; that the reader will understand the
                    ; technique, which is used to place the
                    ; bits in their proper order.
```

```
                        ;
                RLC A           ; 6543210C 7; b7 is now in Carry
                RLC A           ; 543210C7 6; b6 is now in Carry
                MOV ACC.1,C     ; 54321067 6; put b6 into ACC.1
                MOV C,ACC.2     ; 54321067 0; put b0 into C
                RLC A           ; 43210670 5; b5 is now in Carry
                MOV ACC.3,C     ; 43215670 5; put b5 into ACC.3
                MOV C,ACC.4     ; 43215670 1; put b1 into C
                RLC A           ; 32156701 4; b4 is now in Carry
                MOV ACC.5,C     ; 32456701 4; put b4 into ACC.5
                MOV C,ACC.6     ; 32456701 2; put b2 into C
                RLC A           ; 24567012 3; b3 is now in Carry
                MOV ACC.7,C     ; 34567012 3; put b3 into ACC.7
                RL A            ; 45670123   ; prepare for SWAP A
                SWAP A          ; 01234567   ; bits are ordered correctly
                                ; Conversion result is in Accumulator
                                ;
                                ; Subroutine ACALL
SR549D          CLR P1.6        ; Lower Chip Select
                ORL SCON, # 10H ; Set REN
                ANL SCON, # FEH ; Reset RI
                JNB SCON.0,RCV  ; RI flag not set; BRANCH
                                ; until reception is complete
                CPL P1.6        ; Raise Chip Select
                RET             ; Conversion is in SBUF
                END             ;
```

TLC549 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Parallel Ports

The parallel port interface of the Intel 8051/52 to the low cost TLC549 A/D converter minimises the amount of hardware required to operate this serial interface device. The inclusion of an on-chip system clock and deletion of the address line makes further simplifications over the other multi-input members of the TLC540 family.

Principles of Operation



INTEL 8051/52 FAMILY: 8031AH/8051AH/8032AH/8052AH/8751AH/80C51
INTEL 8048/49 FAMILY: 8048AH/8748H/8035AHL/8049AH/8749H/8039AHL/8050AH/8040AHL/80

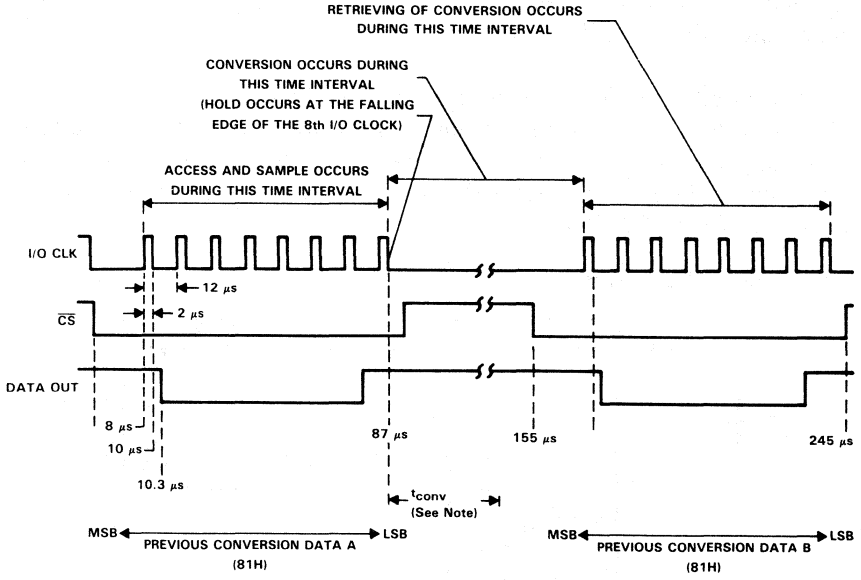
Fig. 11.126 The Intel 8051/52 or 8048/49 parallel port to TLC549 interface

Fig. 11.126 shows the circuit for the 8051 and 8052 microcontroller parallel port interface to the TLC549 device which operates under software control. This interface uses only three port pins of the microcontroller. While the TLC549 is outputing conversion results and concurrently the sample and hold is acquiring a new analog signal, Port 1.3 and $\overline{CS}$ must be low. I/O clock is transmitted through Port 1.2 to pin 7 TLC549 and this causes conversion result data to be communicated from the Data Out pin to Port 1.1.

Timing Diagram

Fig. 11.127 shows the timing diagram for the interface. The timing diagram shows that one complete conversion cycle requires 245 μs.

After $\overline{CS}$ goes low, eight I/O clocks access and sample the new analog input. At the same time, I/O clock falling edges shift out the previous conversion result. This conversion result is valid only for the previous analog signal sample.



NOTE: t_{conv}: The conversion cycles, which requires 36 internal system clock periods, is initiated with the 8th I/O clock↓ after $\overline{CS}$↓. Conversion time requires a minimum of 17μs.

Fig. 11.127 Timing diagram for the Intel 8051/52 parallel port–TLC549 interface

Conversion occurs in the time interval starting when the eighth I/O clock goes low. The conversion time interval requires 36 interval system clock cycles. The designer cannot see the system clock, but can obtain the conversion time from the data sheets. Typically, a time interval of 17 μs is required for conversion.

See the **TLC540 and TLC541 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers Parallel Ports** application for more details.

Software

The interface program listing follows. As shown in the Parallel Port Interface for TLC540 and TLC541 Devices to Intel 8051 and 8052 Microcontrollers application, the instruction RLC A (Rotate Accumulator Left the Carry flag) loads each conversion bit successively into the accumulator from the data output of the TLC549 device. The routine can be used from a simple subroutine call as shown.

```
                        ;
                        ;
        ACALL S549D     ; Access, sample and hold the new
                        ; analog signal.
                        ;
                        ; A delay must occur here to allow the A/D
                        ; chip to complete conversion. The delay
                        ; must allow 36 A/D chip internal system
                        ; clock cycles to occur. Conversion time
                        ; requires a minimum of 17 microseconds.
                        ;
        ACALL S549D     ; Access, sample and hold the new analog
                        ; signal. Bring out the previous
                        ; conversion result.
                        ;
                        ; Subroutine ACALL
S549D   CLR P1.2        ; Lower I/O Clock
        MOV R0, #08H    ; Set bit counter to 8
        CLR P1.3        ; Lower Chip Select
S549I   CLR C           ; Initialize C=0
        JNB P1.1,S549J  ; If 549 data out=0; BRANCH
        CPL C           ; 549 data out=1; set C=1
S549J   MOV A,R1        ; Get serial buffer
        RLC A           ; Shift data out bit into serial buffer
        MOV R1,A        ; Store serial buffer
        CPL P1.2        ; Raise I/O clock
        NOP             ; Delay to slow I/O clock
        CLR P1.2        ; Lower I/O clock
        DJNZ R0,S549I   ; Do 8 times
        CPL P1.3        ; Raise Chip Select
        MOV A,R1        ; Conversion data in A
        RET             ;
        END             ;
```

TLC549 A/D Converter Interface to Texas Instruments TMS32020 Digital Signal Processor

This application describes the interface of the TLC549 8-Bit serial control A/D to the Texas Instruments TMS32020 Digital Signal Processor.

Hardware

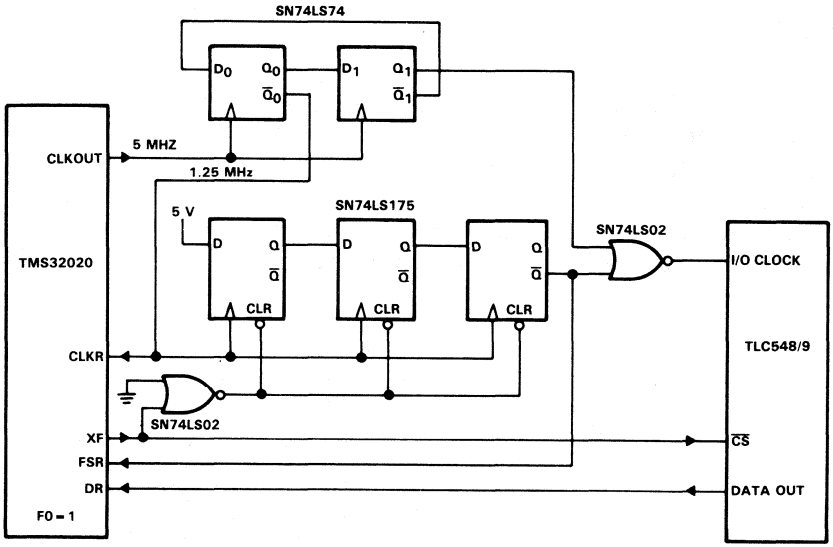


Fig. 11.128 TLC549 to TMS32020 interface circuit

The interface shown in Fig. 11.128 employs three packages:

1	SN74LS175	Quad Positive-edge Triggered D-type Flip-flop with Common Clock and Clear
1	SN74LS02	Quad Positive-NOR Gate
1	SN74LS74	Dual Positive-edge Triggered D-type Flip-flop with Independent Clocks, Clear and Preset

Note: One 74LS175 may be substituted for the 74LS74.

The 74LS74 (or, ½ of 74LS175) is used to implement a ring counter which divides the frequency of the CLKOUT signal, provided by the TMS32020, by four and yields four separate output phases. For example, if the frequency of CLKOUT is 5 MHz, then the output frequency of the counter is 1.25 MHz. If the 74LS74 is used, the clocks should be tied together and all PRESET and CLEAR inputs should be tied to the positive supply voltge. If the 74LS175 is used, only two of the four flip-flops are needed and the common CLEAR shoud be tied to the positive supply.

Principles of Operation

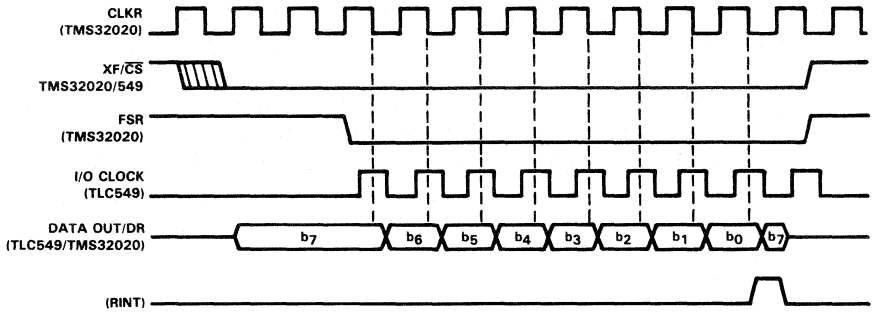


Fig. 11.129 TLC549 to TMS32020 interface timing

As shown in Fig. 11.129, the timing diagram, XF is normally high and is brought low under software control (an RXF instruction as shown in the software listing). Since XF is tied to $\overline{CS}$ of the TLC549, $\overline{CS}$ is also brought low. The TLC548/9 requires a maximum of 1.4 microseconds of delay from the falling edge of $\overline{CS}$ until valid data appears at the output. This is implemented with a 74LS175 which provides 2 microseconds of delay between the falling edge of $\overline{CS}$ and the falling edge of FSR, which begins data transfer.

After the falling edge of FSR, the first data bit is clocked into the DRR (Data Receive Register) of the TMS32020 on the falling edge of CLKR. Two-hundred nanoseconds later, the TLC548/9 sees a falling edge of I/O CLOCK and places the next bit on the DATA OUT pin. As shown in Fig. 11.247, the phase relationship between CLKR and I/O CLOCK allows 600 ns from the falling edge of I/O CLOCK (which clocks-out the next data bit) and the next falling edge of CLKR (which clocks the bit into the DRR register of the TMS32020).

Since the TLC548/9 is an 8-bit A/D converter, the serial port of the TMS32020 should be configured to read 8-bits at a time. This is accomplished by setting FO, the internal format bit, equal to 1. On the eighth falling edge of CLKR after FSR goes low, the last data bit (the LSB) is clocked into the DRR register, and the TMS32020 generates RINT, the serial port read interrupt. If the interrupt is enabled (i.e. the TMS32020 has been instructed to recognize the interrupt), the TMS32020 executes the instruction residing in memory location 26, which should be a branch to a routine which reads the data in the DRR register and raises XF (with an SXF instruction).

If the interrupt (RINT) has been disabled, XF should be raised (with an SXF instruction) after the eighth falling edge of I/O CLOCK (i.e. a minimum of 44 CLKOUT cycles after it was initially lowered).

Because the eighth falling edge of I/O CLOCK initiates the hold-mode of the TC548/9's internal sample-and-hold and starts the next conversion, it is important that the eighth falling edge of I/O CLOCK occur before XF is raised. XF should be kept high for a minimum of 17 microseconds (85 CLKOUT cycles) before the next RXF instruction, thus allowing the TLC548/9 enough time to complete the next conversion.

If XF is lowered before the conversion is complete, the TLC548/9 will halt conversion and the results of the previous conversion will be transferred to the DRR register of the TMS32020.

```
*
*
* Software for TLC548/9 to TMS32020 Interface
*
*
        AORG    >0
        B       INIT    Branch to initialization routine
*
* Interrupt Service Routine:
* After the conversion data has been transferred to the
* TMS32020 Data Receive Register, the Receive Interrupt
* causes the processor to branch to location 26.
*
        AORG    26
        SXF             Raise /CS (XF pin of TMS32020).
        LAC     >0      DRR is data-memory location 0.
        SACL    >60     Save result in data-memory address >60.
        RPTK    85      Delay for conversion.
        NOP
        B       GETDAT
*
* Initialization Routine:
*
```

```
INIT    DINT            Disable interrupts.
        SXF             Raise /CS.
        LDPK    >0      Set data-memory page pointer.
        LACK    >10     Load Interrupt Mask Register
        SACL    >4      with correct data to enable RINT.
        FORT    1       Set data format to 8 bits.
*
* Data Transfer Routine:
* This routine lowers /CS of the TLC548 and begins data transfer.
*
GETDAT  RXF             Lower /CS.
        EINT            Enable Interrupts.
WAIT    B       WAIT    Wait for Receive Interrupt.
        END
```

TLC1540 and TLC1541 10-Bit Analog-to-Digital Converters with 11-Inputs and Serial Input/Output

The TLC1540 and TLC1541 have the same basic features as the TLC540/541 with 11 analog inputs but provide 10-bit resolution. They have total unadjusted error specifications of ±0.5 LSB for the TLC1540 and ±1 LSB for the TLC1541. Table 11.16 gives more performance details and Fig. 11.130 shows the functional block diagram.

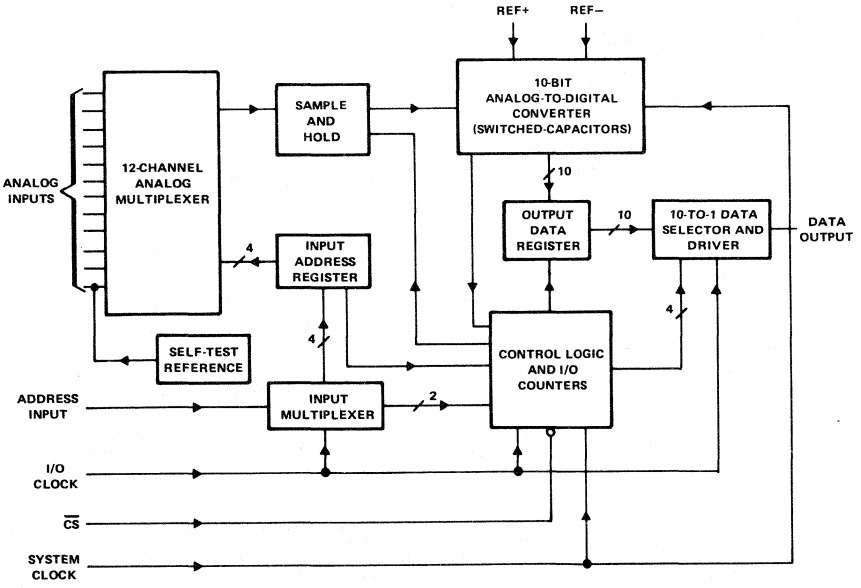


Fig. 11.130 TLC1540/1541 functional block diagram

The pin layout is shown in Fig. 11.131.

Table 11.16

Performance	**TLC1540/1541**
Acquisition Time (max)	5·5 μs
Conversion Time (max)	21 μs
Sampling rate (max)	32 × 10^3sps
Power Dissipation (typ)	6 mW

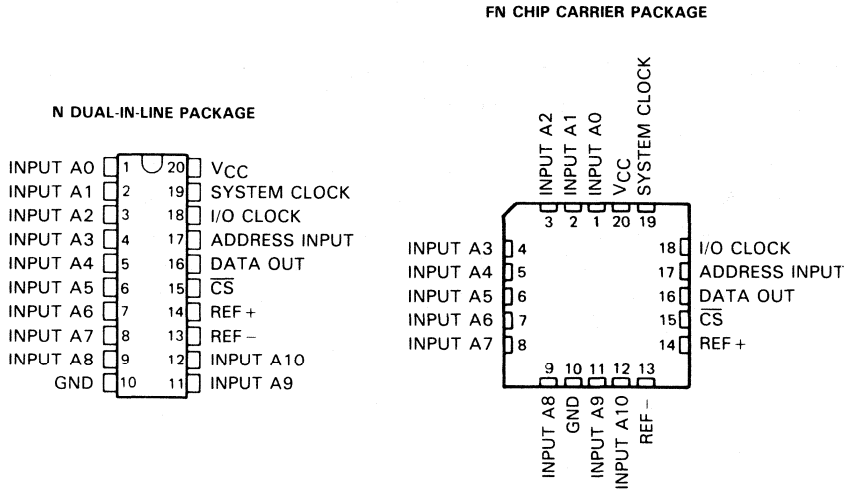
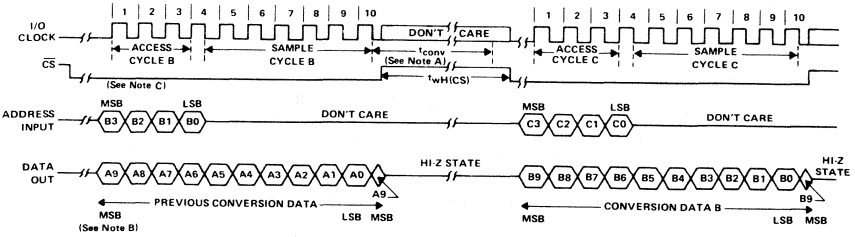


Fig. 11.131 TLC1540/1541 pinouts (top view)



NOTES: A. The conversion cycle, which requires 44 System Clock periods, is initiated on the 10th falling edge of the I/O Clock↓ after CS↓ goes low for the channel whose address exists in memory at that time. If $\overline{CS}$ is kept low during conversion, the I/O Clock must remain low for at least 44 System Clock cycles to allow conversion to be completed.

B. The most significant bit (MSB) will automatically be placed on the DATA OUT bus after $\overline{CS}$ is brought low. The remaining nine bits (A8-A0) will be clocked out on the first nine I/O Clock falling edges.

C. To minimize errors caused by noise at the CS input, the internal circuitry waits for three System Clock cycles (or less) after a chip-select falling edge is detected before responding to control input signals. Therefore, no attempt should be made to clock-in address data until the minimum chip-select setup time has elapsed.

Fig. 11.132 TLC1540/1541 interface timing diagram

Operation of the TLC1540 and TLC1541 is the same as TLC540 (see page 11−79) in all respects except for two additional I/O clock pulses to give ten in total which clock out the 10-bit conversion result in a serial format on Data Out. This is shown on the timing diagram of Fig. 11.132. This modifies the control sequence step 3 of page 11−79 in that five I/O clock pulses are used to clock out the 6th, 7th, 8th, 9th and 10th conversion bits. In step 4 it is the falling edge of the 10th I/O clock pulse which completes the sampling function and initiates the hold sequence. The conversion is then performed during the next 44 system clock cycles.

TLC1540 and TLC1541 A/D Converter Interface to Zilog Z80A Microprocessors

This application describes a technique for interfacing the Z80A microprocessor to the TLC1540 10-bit A/D converter using software control.

Principles of Operation

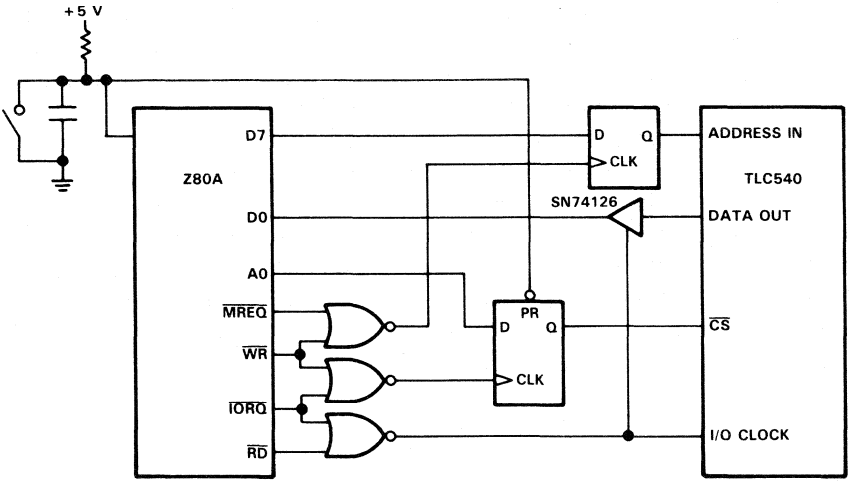


Fig. 11.133 Z80A to TLC1540 interface circuit diagram

A circuit diagram for the software controlled Z80A microprocessor to TLC1540 device interface is shown in Fig. 11.133. A timing diagram for the interface is shown in Fig. 11.134.

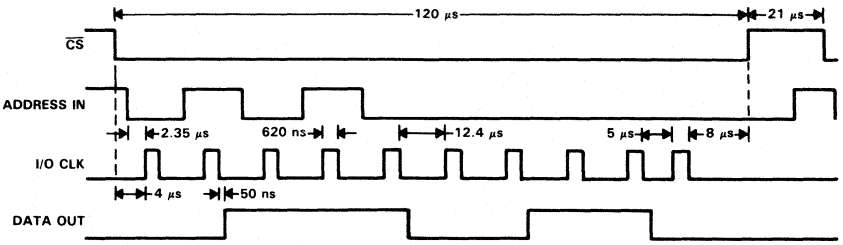


Fig. 11.134 Z80A to TLC1540 interface timing diagram

A data conversion cycle is initiated by bringing $\overline{CS}$ low. This is accomplished by latching a low on address line A0 into the D-type flip-flop. Execution of an IN instruction causes the RD line and the $\overline{IORQ}$ line to become active and generate one I/O clock pulse.

A multiplexer address bit is shifted in on the rising edge of the I/O clock pulse and a conversion result data bit is shifted out on the trailing edge of the pulse. The I/O clock pulse also connects the output of the SN74126 3-state buffer onto the data bus that allows a data bit to be read into the Z80A accumulator. Once in the accumulator, the data bit is rotated through the carry bit location and into the register B or C. Register B contains the eight most significant bits while register C contains the two least significant bits so the data is left justified. Register D contains the multiplexer address in the 4 most significant bit positions and is shifted out to the left. On the falling edge of the fourth I/O clock pulse, sampling of the addressed analog input begins and continues until the falling edge of the tenth I/O clock pulse occurs. At that time, $\overline{CS}$ is brought high and conversion begins. Conversion requires 44 system clock cycles, and $\overline{CS}$ should remain high until conversion is complete. If these conditions change, the ongoing conversion will be aborted and a new conversion cycle will begin. It is possible to read conversion results every 141 µs.
The control software routine listing follows

Software for Z80A to TLC1540
and TLC1541 Interface

```
        LD   E,08H          ; Load bit counter
        LD   D,50H          ; Initialize muxaddress to Ch 5
        OUT  (CSLOW),A      ; Bring /CS low
LOOP:   LD   (HL),D         ; Write out muxaddress bit
        IN   A,(BIT)        ; Read in a bit & clock next bit
        RRA                 ; Rotate bit into carry
        RL   B              ; Rotate into result register B
        RLC  D              ; Rotate muxaddress left
        DEC  E              ; Decrement bit counter
```

```
JP    NZ,LOOP       ; Get another bit if not zero
IN    A,(BIT)       ; Read in bit 1 of result
RRA                 ; Rotate into carry
RL    C             ; Rotate into result register C
IN    A,(BIT)       ; Read in bit 0 of result
RRA                 ; Rotate into carry
RR    C             ; Rotate result register C
RR    C             ; Rotate result register C
OUT   (CSHIGH),A    ; Bring /CS high
```

TLC1540 and TLC1541 A/D Converter Interface to Rockwell 6502 Microprocessors

This application describes a technique for operating the TLC1540 10-bit A/D converter with the 6502 microprocessor software generated control signals. These signals are $\overline{CS}$, I/O clock, address input, and system clock. The system clock signal is required to drive the successive-approximation conversion process.

Hardware

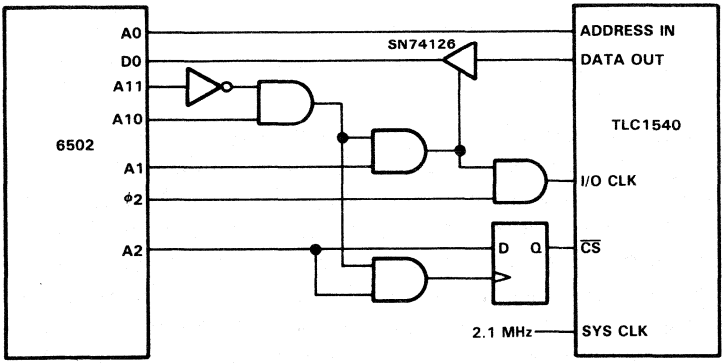


Fig. 11.135 6502 to TLC1540 interface circuit diagram

A circuit diagram for the 6502 microprocessor to TLC1540 device interface is shown in Fig. 11.135. I/O clock pulses are generated by enabling the positive going pulse of the $\phi 2$ clock to the TLC1540 device. Address input bits are latched into the TLC1540 device on the leading edge of the $\phi 2$ pulse and data bits are shifted out of the TLC1540 device on the trailing edge of the pulse. Address lines A1, A10, and A11 are decoded to enable the output of the SN74126 3-state buffer onto the data bus and the data bit is read by the 6502 microprocessor on the negative edge of $\phi 2$. $\overline{CS}$ is controlled by clocking the level of A2 into the D-type flip-

flop. Thus, by writing to the proper address, $\overline{CS}$ can be lowered or raised. A simple address decoding scheme is presented in this application. Small modifications may be necesssary to fit a particular application.

Timing Diagram

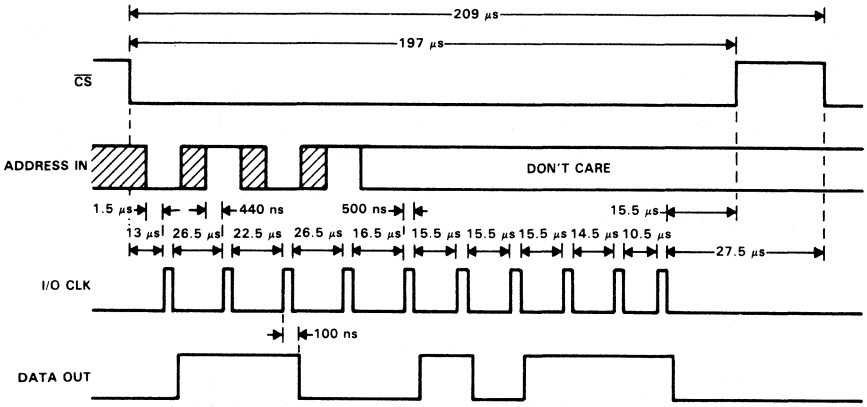


Fig. 11.136 6502 to TLC1540 interface timing diagram

A conversion cycle is initiated by bringing $\overline{CS}$ low. This is accomplished by writing to an address so that A2 is low. The multiplexer address should be placed in the most significant 4 bits of the MUXADDRESS memory location. A multiplexer address is rotated left into the carry, and then tested and written out. The LDA instruction causes one I/O clock pulse to be generated and one data bit to be read. Once the data bit is read in, it is rotated into the carry bit location, and then into RAM. The conversion result is stored in a left justified format in memory locations 0001H and 0002H. After the tenth I/O clock pulse, $\overline{CS}$ is brought high and conversion starts. Conversion requires 44 system clock cycles, and a delay loop may be needed to allow enough time for conversion when using low system clock frequencies. When conversion is complete, $\overline{CS}$ is brought low. The time for one conversion cycle, 209 μs, is indicated in Fig. 11.136.

Software

A control software listing follows.

Software Listings for TLC1540 and TLC1541
to Rockwell 6502 Interface

```
;                *** Register Assignments ***
;
CSLOW           .EQU 0400H
CSHIGH          .EQU 0404H
ADDLOW          .EQU 0402H
ADDHIGH         .EQU 8403H
MUXADDRESS      .EQU 0000H
MSB             .EQU 0001H
LSB             .EQU 0002H
;
;                *** Main Program ***
;
START:          STA  CSHIGH       ; Make sure chi select is high
                LDA  #$50         ;
                STA  MUXADDRESS   ; Initialize muxaddress to Ch 5
                STA  CSLOW        ; Bring chip select low
                LDX  #$04H        ; Load counter

LOOP1:          ROL  MUXADDRESS   ; Rotate muxaddress into carry
                BCS  HIGH         ; Branch if bit is set
                LDA  ADDLOW       ; Write out A low on A0, clock data in
                JMP  OVER         ; Skip next instruction
HIGH:           LDA  ADDHIGH      ; Write out A high on A0, clock data in
OVER:           ROR  A            ; Rotate do into carry
                ROL  MSB          ; Rotate carry into MSB result
                DEX               ; Decrement counter
                BNE  LOOP1        ; Go back for another bit

                LDX  #$04H        ; Load counter

LOOP2:          LDA  ADDLOW       ; Read in data bit
                ROR  A            ; Rotate into carry
                ROL  MSB          ; Rotate into MSB result
                DEX               ; Decrement counter
                BNE  LOOP2        ; Go back for another bit

                LDA  ADDLOW       ; Read in bit 1 of result
                ROR  A            ; Rotate into carry
                ROL  LSB          ; Rotate into LSB result
                LDA  ADDLOW       ; Read in bit 0 of result
                ROR  A            ; Rotate into carry
                ROR  LSB          ; Rotate LSB result
                ROR  LSB          ; Rotate LSB result
                STA  CSHIGH       ; Bring chip select high
```

TLC1540 and TLC1541 A/D Converter Interface to Motorola 6802 Microprocessors

This application describes a technique that uses software control in a 6802 microprocessor to TLC1540 10-bit A/D converter.

Principles of Operation

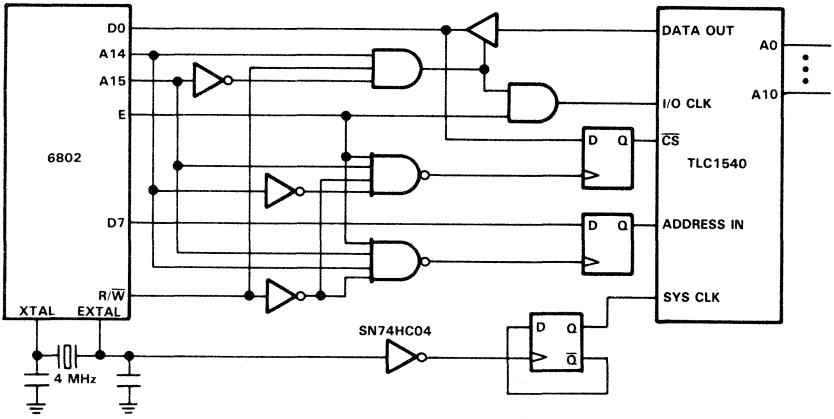


Fig. 11.137 6802 to TLC1540 interface circuit diagram

A circuit diagram for the software controlled TLC1540 device to 6802 microprocessor interface is shown in Fig. 11.137. The timing diagram is shown in Fig. 11.138.

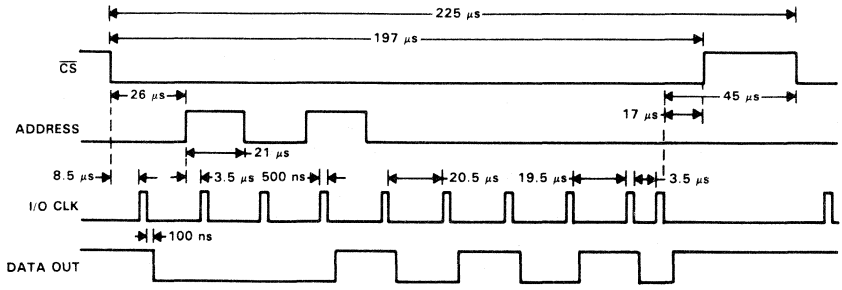


Fig. 11.138 6802 to TLC1540 interface timing diagram

A data conversion cycle is initiated by bringing $\overline{CS}$ low. This is accomplished by latching a low from data bus line DO into the D-type flip-flop. Executing a STAA instruction latches a multiplexer address bit into the D-type flip-flop to ensure that set-up time requirements are met. Execution of an LDAB instruction enables the E clock to the TLC1540

device to generate one I/O clock pulse. It also enables the output of the 3-state buffer onto the data bus to read in a data bit on the trailing edge of the E clock. The trailing edge of E clock also shifts out the next bit of the conversion result. Once a data bit is read into accumulator B, it is rotated into the carry bit and then rotated into accumulator A. This rotates the multiplexer address one bit to the left. Since the eight most significant bits are stored in the first byte of RAM and the two least significant bits are stored in the second byte of RAM, the data is left justified. Conversion of the addressed channel begins on the falling edge of the tenth I/O clock pulse and requires 44 system clock cycles. $\overline{CS}$ is brought high at the beginning of conversion and should remain high until conversion is complete. A 2 MHz system clock can be obtained by tapping off the EXTAL pin into an SN74HC04 inverter and frequency divide-by-two circuit. The high impedance of the HCMOS device prevents the oscillator from being loaded excessively. It should be noted that many of the gates in this interface are used for the address decoding scheme for the control software presented. One data conversion cycle may be completed in 225 μs as indicated in Fig. 11.138.

A control software routine listing follows:

```
;
;                    Register Assignments for a TLC1540 or TLC1541 to
                                Motorola 6802 Interface

;
ADDRESS     .EQU   $C000H      ; Address to select address FF
DATA        .EQU   $4000H      ; Address to select 3-state buff
CS          .EQU   $8000H      ; Address to select CS FF
;
;
;
START:      LDAA   #50H        ; Initialize muxaddress to channel 5
            LDX    #08H        ; Load counter
            LDAB   #00H        ;
            STAB   CS          ; Bring chip select low
LOOP:       STAA   ADDRESS     ; Write out muxaddress bit
            LDAB   DATA        ; Read data bit and clock
            RORB               ; Rotate data bit into carry
            ROLA               ; Rotate carry into result
            DEX                ; Decrement counter
            BNE    LOOP        ; Go back for another bit
            STAA   $0000H      ; Store 8 MSB'S in RAM
            LDAB   DATA        ; Read bit 1
            LDAA   DATA        ; Read bit 0
            RORA               ; Rotate bit 0 into carry
            RORB               ; Rotate bits 0 & 1
            RORB               ; Rotate bits 0 & 1
            STAB   $0001H      ; Store LSB'S in RAM
            LDAA   #01H        ;
            STAA   CS          ; Bring chip select high
```

TLC1540 and TLC1541 A/D Converter Interface to Motorola 6805 Microcomputers

This application describes an interface for the 6805 microcomputer to TLC1540 10-bit A/D converter using software generated control signals. These signals are $\overline{CS}$, I/O clock, address input, and system clock. The system clock signal is required to drive the successive-approximation conversion process.

Hardware

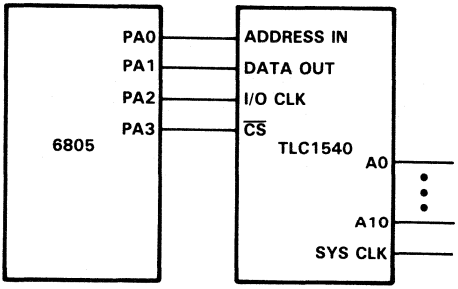


Fig. 11.139 6805 to TLC1540 interface circuit diagram

A circuit diagram for the 6805 microcomputer to TLC1540 device interface is shown in Fig. 11.139. Four port pins in PORTA are used to transfer control signals and data. A system clock signal may be obtained by tapping off the oscillator input into a high impedance buffer or inverter, such as a SN74HC04, to prevent loading of the oscillator. The signal can then be divided to the required frequency. Care should be used to ensure that system clock pulse width requirements of the TLC1541 device are met.

Software

All interface control signals are generated through software manipulation of port pins. The single loop control software listing follows.

```
;                    Software Listing for TLC1540 and TLC1541 to
                                Motorola 6805 Interface
                *** Register Assignments***
;
PORTA       .EQU  0000H          ; Port A I/O pins
DDRA        .EQU  0004H          ; Data direction register A
MSBS        .EQU  0010H          ; Conversion result MSB'S
LSBS        .EQU  0011H          ; Conversion result LSB'S
;
;         *** MAIN PROGRAM ***
;
START:      LDA   #0DH           ;
            STA   DDRA           ; Initialize port A I/O pins
            BSET  3,PORTA        ; Make sure /CS is high
            LDA   #50H           ; Initialize muxadd to channel 5
            LDX   #08H           ; Initialize counter
            BCLR  3,PORTA        ; Bring /CS low
;
LOOP:       BRSET 1,PORTA,LABEL1 ; Read data bit into carry
LABEL1      ROLA                 ; Rotate into accumulator
            BCS   HIGH           ; Go to high if muxadd bit is 1
            BCLR  0,PORTA        ; Write a 0 to TLC1540 address in
            JMP   CLOCK          ; Skip next instruction
HIGH:       BSET  0,PORTA        ; Write a 1 to TLC1540 address in
CLOCK:      BSET  2,PORTA        ; Bring I/O clock high
            BCLR  2,PORTA        ; Bring I/O clock low
            DECX                 ; Decrement counter
            BNE   LOOP           ; Go back for another bit
;
            STA   MSBS           ; Store MSB'S in RAM
            BRSET 1,PORTA,LABEL2 ; Read bit 1 into carry
LABEL2      ROLA                 ; Rotate into accumulator
            BSET  2,PORTA        ; Bring I/O clock high
            BCLR  2,PORTA        ; Bring I/O clock low
            BRSET 1,PORTA,LABEL3 ; Read bit 0 into carry
LABEL3      BSET  2,PORTA        ; Bring I/O clock high
            BCLR  2,PORTA        ; Bring I/O clock low
            RORA                 ; Rotate accumulator
            RORA                 ; Rotate accumulator
            STA   LSBS           ; Store LSB'S in RAM
            BSET  3,PORTA        ; Bring /CS high
```

The multiplexer address should be loaded in the most significant 4 bits of the accumulator during intitialization. A data bit is read into the carry bit location, rotated into the accumulator, which then rotates a multiplexer address bit out. One I/O clock pulse is then generated by toggling bit 2 of PORTA. The multiplexer address bit is latched in on the leading edge of the I/O clock, and a data bit is shifted out on the trailing edge of the I/O clock. This procedure is placed in a loop and repeated eight times. The most significant bits are then stored in the first byte of the on-board RAM. The last two result bits are read in and stored in the second byte of RAM causing the data to be stored in a left justified format.

After the results are read in, $\overline{CS}$ is brought high and should remain high until conversion is complete. Conversion requires 44 system clock cycles and a possible time delay depending on the system clock frequency. Using this program, a conversion cycle can be completed in 360 μs.

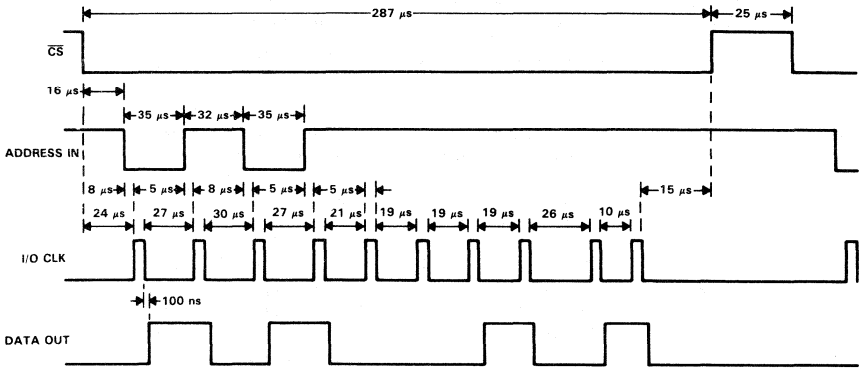


Fig. 11.140. 6805 to TLC1540 interface timing diagram

```
;       *** Register Assignments ***
;
PORTA       .EQU  0000H          ; Port A I/O pins
DDRA        .EQU  0004H          ; Data direction register A
MSBS        .EQU  0010H          ; Conversion result MSB'S
LSBS        .EQU  0011H          ; Conversion result LSB'S
;
;       *** MAIN PROGRAM ***
;
START:      LDA   #0DH           ;
            STA   DDRA           ; Initialize PortA I/O pins
            BSET  3,PORTA        ; Make sure CS is high
            LDA   #50H           ; Initialize Muxaddress to channel 5
            LDX   #04H           ; Initialize counter
            BCLR  3,PORTA        ; Bring CS low
;
LOOP1:      BRSET 1, PORTA,LABEL1 ; Read data bit into carry
LABEL1:     ROLA                 ; Rotate into accumulator
            BCS   HIGH           ; Go to high if Muxadd bit is 1
            BCLR  0,PORTA        ; Write out A0 to TLC1540 address in
            JMP   CLOCK          ; Skip next instruction
HIGH:       BSET  0,PORTA        ; Write out A1 to TLC1540 address in
CLOCK:      BSET  2,PORTA        ; Bring I/O clock high
            BCLR  2,PORTA        ; Bring I/O clocl low
            DECX                 ; Decrement counter
            BNE   LOOP1          ; Go back for another bit
;
            LDX   #04H           ; Load bit counter
;
LOOP2       BRSET 1,PORTA,LABEL2 ; Read data bit and clock
```

```
LABEL2:   ROLA                    ; Rotate into accumulator
          BSET  2,PORTA           ; Bring I/O clock high
          BCLR  2,PORTA           ; Bring I/O clock low
          DECX                    ; Decrement counter
          BNE   LOOP2             ; Go back for another bit

          STA   MSBS              ; Store MSB'S in RAM
          CLRA                    ; Clear accumulator
          BRSET 1,PORTA,LABEL3    ; Read bit 1 into carry
LABEL3:   ROLA                    ; Rotate into accumulator
          BSET  2,PORTA           ; Bring I/O clock high
          BCLR  2,PORTA           ; Bring I/O clock low
          BRSET 1,PORTA,LABEL4    ; Read bit 0 into carry
LABEL4:   BSET  2,PORTA           ; Bring I/O clock high
          BCLR  2,PORTA           ; Bring I/O clock low
          RORA                    ; Rotate accumulator
          RORA                    ; Rotate accumulator
          STA   LSBS              ; Store LSB'S in RAM
          BSET  3,PORTA           ; Bring /CS high
```

A timing diagram for a 2-loop control program is shown in Fig. 11.140. The 2-loop control program listing follows Fig. 11.140. This program has a loop to shift out the four multiplexer address bits and has another loop to shift in the next four bits. There are two individual clock cycles for the last two bits. Although this program uses a few more bytes of program memory, cycle time can be reduced to 312 μs. It is also possible to write a brute-force routine that uses no loops and generates 10 individual clock pulses. This routine reduces cycle time to 260 μs but requires more than twice as much program memory space.

TLC1540 and TLC1541 A/D Converter Interface to Intel 8051 and 8052 Microcontroller Parallel Ports

This application describes a technique for operating the TLC1540 10-bit A/D converter with the 8051 microprocessor using software generated control signals. These signals are $\overline{CS}$, I/O clock, address input, and system clock. The system clock signal is required to drive the successive approximation conversion process.

Hardware

The system clock is derived from the ALE signal of the 8051 device. Another method uses a signal that is tapped off the oscillator through a high impedance buffer or inverter and divided down to the appropriate frequency. Care should be taken when using the ALE signal for the

TLC1541 system clock to ensure that the high and low pulse widths are within specifications. The ALE signal is dependent upon the oscillator frequency and may not meet pulse width specifications at high oscillator frequencies. A circuit diagram is shown in Fig. 11.141.

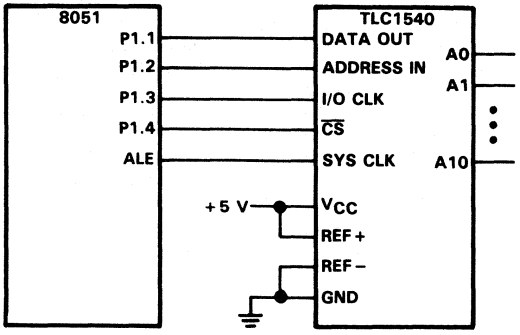


Fig. 11.141 8051 to TLC1540 interface circuit diagram

Timing Diagram

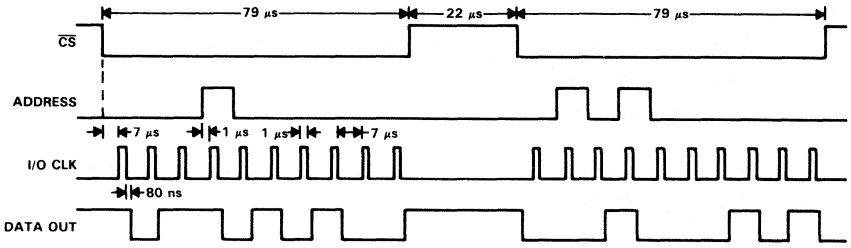


Fig. 11.142 8051 to TLC1540 interface timing diagram

A timing diagram for the interface is shown in Fig. 11.142. The subroutine can be executed in 79 µs. With a system clock of 2.1 MHz from the ALE pin, conversion results may be read every 101 µs.

Software

All interface control signals are generated through software manipulation of port pins. A subroutine is used to load a new multiplexer address and retrieve a previous conversion result. A listing of the subroutine follows the discussion.

A multiplexer address should be loaded into the most significant 4 bits of the accumulator before calling the subroutine. Previous conversion results are returned left justified, with R2 holding the eight most significant bits and R3 holding the two least significant bits. After returning from the subroutine, a delay loop is executed to allow time for the conversion. Conversion requires 44 system clock cycles, therefore, delay loops of appropriate length should be included according to the system clock frequency.

```
;                Software List for TLC1540 and TLC1541 to
                   Intel 8051 and 8052 Microprocessors
            *** Main Program ***
;
START:      MOV   P1,#02H      ; Initialize port 1 I/O pins
            CLR   P1.3         ; Make sure I/O CLK is low
            SETB  P1.4         ; Make sure /CS is high
CONTINUE:   MOV   A,#10H       ; Initialize muxaddress to channel 1
            ACALL TLC1540      ; Shift muxaddress/results
            MOV   R5,#07H      ; Load counter
DELAY1:     DJNZ  R5,DELAY1    ; Delay for conversion
            MOV0A,#0H          ; Initialize muxaddress to channel 5
            ACALL TLC1540      ; Shift muxaddress/results
            MOV   R5,#07H      ; Load counter
DELAY2:     DJNZ  R5,DELAY2    ; Delay for conversion

;           *** Subroutine ***
;
TLC1540     MOV   R4,#08       ; Load counter
            CLR   P1.4         ; Bring /CS low
            NOP                ; Delay for /CS to go low
            NOP                ;
LOOP:       MOV   C,P1.1       ; Read data bit into carry
            RLC   A            ; Rotate into accumulator
            MOV   P1.2,C       ; Write muxadd bit out
            SETB  P1.3         ; I/O clock high
            CLR   P1.3         ; I/O clock low
            DJNZ  R4,LOOP      ; Go back and get another bit
            MOV   R2,A         ; Store MSB'S in R2
            MOV   C,P1.1       ; Read data bit into carry
            CLR   A            ; Clear accumulator
            RLC   A            ; Rotate data bit into accumulator
            SETB  P1.3         ; I/O clock high
            CLR   P1.3         ; I/O clock low
            MOV   C,P1.1       ; Read data bit into carry
            RRC   A            ; Rotate right into accum MSB
            RRC   A            ; Rotate right into accum MSB
            MOV   R3,A         ; Store LSB'S in R3
            SETB  P1.3         ; I/O clock high
            CLR   P1.3         ; I/O clock low
            SETB  P1.4         ; Bring /CS high
            RET                ; Return to main program
```

TLC1540/1 and TLC540/1 A/D Converters Interface to Texas Instruments TMS32020 and TMS32025 Digital Signal Processors

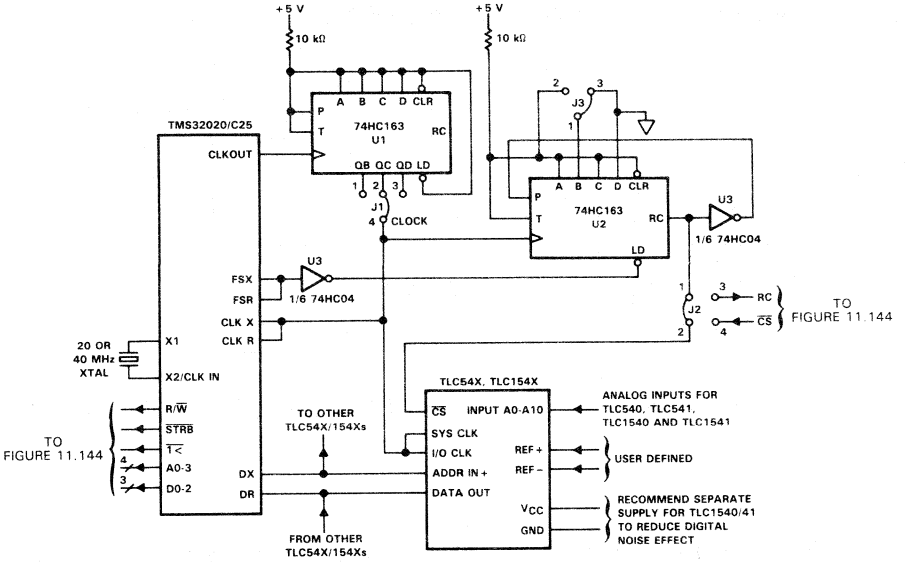
The TMS32020/C25 can be easily interfaced to the TLC154X and TLC54X family of data acquisition devices via the serial port. The TLC1540/1 and TLC540/1 are multiple-analog-input devices that are useful in cases where more than one analog signal needs to be monitored. The TLC540/1 contains an 8-bit ADC with 11 external analog inputs. If more precision is required, the TLC1540/1 contains a 10-bit ADC with 11 external analog inputs. These devices contain separate input and output serial ports. The input port is used for selecting which analog input is to be monitored. The host processor simply sends the address of the desired analog input. The output port is used for transmitting the digital representation of the processor.

Hardware

As shown in Fig. 11.143, glue logic is required to generate the clock for the serial interface and the chip select for the TLC54X/154X. A total of three packages is required: two 74HC163 synchronous counters and two inverters from a 74HC04. Timing is not critical, therefore other technologies such as S, LS, AS, ALS, F, and ACL can be used. In this application, the glue logic can be replaced by one PAL. A '16R8 would be appropriate.

Please refer to Fig. 11.143 for the following explanation. U1 divides the CLKOUT signal from the TMS32020/C25 to provide the clock for the serial interface. The TMS32020/C25 internally divides the input crystal clock by four. In the case of a TMS320/C25 using a 40 MHz crystal, U1 divides by eight and by sixteen to provide a 1.25 MHz and 625 kHz clock respectively. The 1.25 MHz clock can be used for the TLC540, and the 625 kHz clock can be used for the TLC541 and TLC1540/1. The TMS32020/C25 generates a framing signal on pin FSX that is used to trigger a chip-select from U2. When triggered, U2 provides a chip-select 8 and 10 clocks wide for the TLC540/1 and TLC1540/1, respectively. The ripple carry signal from U2 is used as the chip-select signal. Triggering occurs when the load input is driven low. This occurs when the FSX signal from the TMS32020/C25 goes high. On the next rising clock edge, U2 is loaded with the values on its preset inputs. When U2 is loaded, ripple carry goes low until a count of 15 is reached. Then, ripple carry goes high, disabling the count. The count disable is accomplished by connecting the inverted ripple carry to ENABLE P input. The preset input values are 7 and 5 to generate a chip-select 8 and 10 clocks wide, respectively.

If several devices need to be networked together, multiple TLC54X and TLC154Xs can be connected together on the same serial bus. This is possible because the serial output ports are in a high-impedance state when not selected.



TMS32020/C25-TLC54X, TLC154X INTERFACE:

- Data transfer software initiated on TMS32020/C25
- TMS32020/C25 using burst mode for transfers on serial port
- 3 packages used for glue logic: 74HC04 and 2X 74HC163
- Glue logic can be replaced with one PAL, 16R8

J1 – TMS32020 operation:
Connect 1 and 4 for 1.25 MHz clock
Connect 2 and 4 for 625 kHz clock
TMS320C25 operation:
Connect 2 and 4 for 1.25 MHz clock
Connect 3 and 4 for 625 kHz clock
J2 – Connect 1 and 2 for single TLC54X/154 X operation
Connect 1 and 3, 2 and 4 for multiple TLC54X/154X operation
J3 – Connect 1 and 3 for TLC154X
Connect 1 and 2 for TLC540/1.

Fig. 11.143 TLC54X, TLC154X to TMS32020/C25 interface

Additional glue logic is required for this application. A latched decoder such as a 74HCT137 (U4) as shown in Fig. 11.144 is a possible solution. The appropriate device is selected by writing a device value to U4. The device value is the address of the output pin on U4 that routes the chip-select to the desired device. U4 is I/O mapped in this application. Memory mapping is possible by using the data-space select (-DS) rather than the I/O-space select (-IS) on the TMS32020/C25.

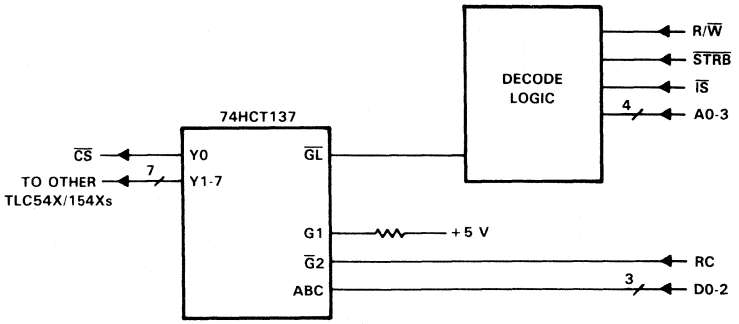


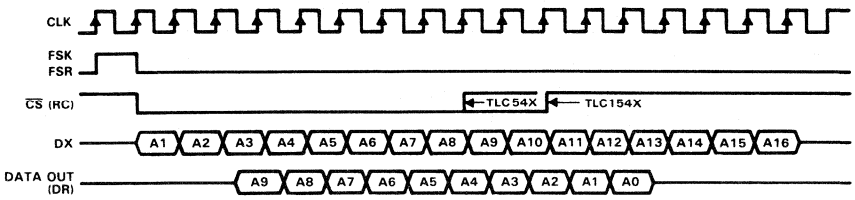
Fig. 11.144 Expansion circuitry for TMS32020/C25 to TLC154X/54X interface

Software

For proper operation of the TMS32020/C25 with the hardware, two of its status bits, TXM and FO, must be correctly set as shown in the following software listing. TXM is the transmit mode bit that configures the FSX pin as an input or an output. This application requires FSX to be an output. Consequently, the TXM bit must be set high. The value of the TXM bit can be modified by using the STXM or RTXM instructions. Note that TXM is set low upon reset. The TXM bit should be set high using the STXM instruction during the reset initialization routine. FO is the serial I/O format bit that defines the data transfer to be either 8 or 16-bit formats. This application requires data transfers to be in a 1-bit format. Therefore, the FO bit must be set low. The FO bit can be modified using the FORT instruction. Note that the FO bit is correctly set low at reset. TLM and FO can also be modified using the LST1 instruction.

```
* Software for TLC54X/154X to TMS32020/C25
* Interface
*
*
        AORG    >0
        B       INIT
```

```
*
* Interrupt Service Routine:
* After the conversion data has been transferred to the
* TMS32020/C25 Data Receive Register, the Receive Interrupt
* causes the processor to branch to location 26.
*
        AORG   26
        ZALS   >0       Save unshifted result.
        ANDK   >1FFF    Mask receive bits.
        RPTK   2        Shift receive bits.
        SFR
        SACL   >60      Save conversion result in >60.
        RPTK   127      Delay for conversion.
        NOP
        RPTK   127
        NOP
        RPTK   127
        NOP
        B      TXRX     Get new conversion result.
*
* Initialization Routine:
*
INIT    DINT            Disable interrupts.
        LDPK   >0       Set Data-Memory Page Pointer to >0.
        LACK   >10      Enable (RINT) interrupt.
        SACL   >4
        STXM            Set transmit mode (FSX is an output).
        FORT   0        Set data format to 16 bits.
*
* Data Transfer Routine:
*
TXRX    LACK   >5       Load Accumulator with address (channel 5).
        RPTK   9        Shift address 10 places to the left.
        SFL
        SACL   >1       Place channel address in Transmit Register.
        EINT            Enable Interrupts.
WAIT    B      WAIT     Wait for receive interrupt.
```



TMS32020/C25

NOTES:
- Burst mode for serial data transfer
- FSX is an output, TXM status bit set to "1"
- 16-bit word transfers required, FO status bit set to "0"

Fig. 11.145 Timing diagram for TLC154X

Attention must be paid to the bit positions of the data in the 16-bit word on the TMS32020/C25. Bit positions are skewed as a result of the 2.5 system clock delay in the TLC54X/154X to recognize a chip-select. The TMS32020/C25 must embed the analog input address at the proper bit locations to be correctly recognized by the TLC540/1 and TLC1540/1. Similarly, the TMS32020/C25 must correctly interpret the proper bit locations of the data sent by the TLC54X/154X. The timing diagrams (see Fig. 11.145) illustrate the skew in the bit locations. Bit manipulations are easily performed using logical operators and bit shifting such as the AND, ANDK, OR, ORK, XOR, XORK, LAC and LACT instructions. Table 11.17 shows the proper bit locations.

Table 11.17 TMS320/C25 Data Bit Positions

BIT POSITIONS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TLC540/1 AND TLC1540/1 ADDRESS IN			B3	B2	B1	B0										
TLC154XDATA OUT				A9	A8	A7	A6	A5	A4	A3	A2	A1	A0			
TLC54X DATA OUT				A7	A6	A5	A4	A3	A2	A1	A0					

Other Considerations

Data transfers between the TMS32020/C25 and the TLC54X/154X are initiated under software control. External stimulus for these transfers is possible by using the TMX32020/C25 external interrupt inputs ($\overline{\text{INT}}$) and the branch control input ($\overline{\text{BIO}}$).

The TLC54X/154X require a minimum number of system clock cycles to occur between valid transfers to allow for ADC conversion. Conversion time periods are specified on the TLC54X/154X data sheets. Be sure to wait the conversion time period between transfers to obtain valid data. Wait periods can be timed by using the timer function in the TMS32020/C25.

TLC0820 Advanced LinCMOS™ HIGH SPEED 8-BIT ANALOG-TO-DIGITAL CONVERTER

The TLC0820 is a high speed 8-Bit analog to digital converter built using Advanced LinCMOS technology. It employs a modified flash analog-to-digital conversion technique, sometimes called semi-flash, which gives an access and conversion time of 1.18 μs in the fastest Write-Read mode. Each converter consists of two 4-bit flash analog-to-digital converters, a 4-bit digital to analog converter, a subtracting circuit, control logic and a conversion result latch. The device uses sample-data comparator techniques which provide an on-chip sample and hold circuit which has a 100 ns sample window and allows the TLC0820 to convert continuous analog input signals with slew rates typically of 100 mV/μs. Parallel output through TTL-compatible three-state output drivers combined with two modes of operation allow interfacing to a variety of microprocessors. Performance information is shown in Table 11.18. Fig. 11.146 shows the pin layout.

Table 11.18

PERFORMANCE	TLC0820A	TLC0820B
Conversion and Access Time (max):		
Write-Read Mode $\overline{RD}$ low before $\overline{INT}$	1.18 μs	1.18 μs
Write-Read Mode $\overline{RD}$ low after $\overline{INT}$	1.92 μs	1.92 μs
Read Mode	2.6 μs	2.6 μs
Total Unadjusted Error (max)	1 LSB	0.5 LSB
Power Dissipation (typ)	50 mW	50 mW

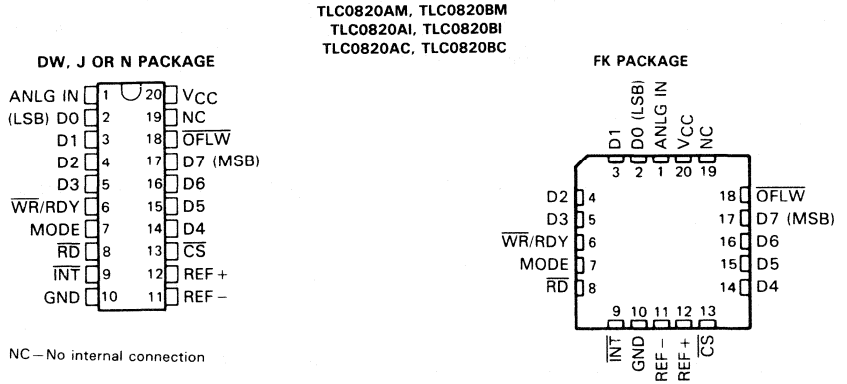


Fig. 11.146 TLC0820 pinouts (top view)

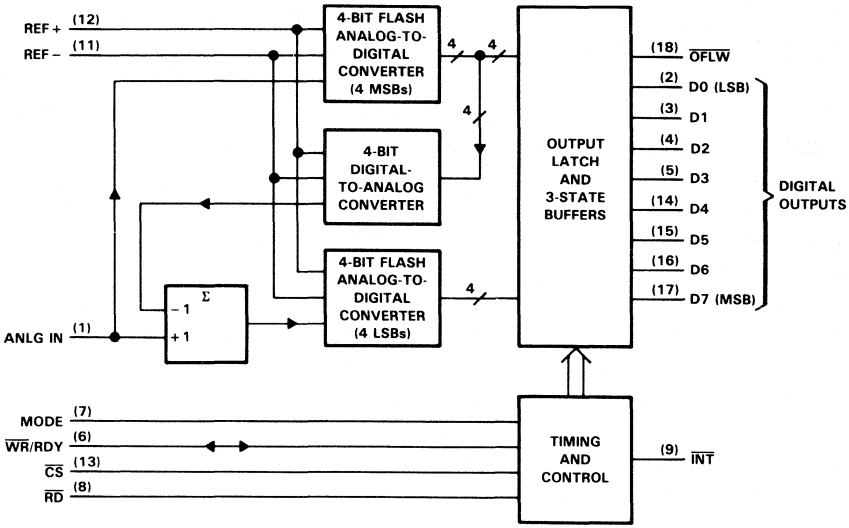


Fig. 11.147 TLC0820 functional block diagram

The functional block diagram is shown in Fig. 11.147. The high-order 4-bit flash analog to digital converter measures the input by means of 16 internal comparators operating simultaneously. A precision 4-bit digital to analog converter then generates a discrete analog voltage from the result of that conversion. After a time delay, a second bank of 16 comparators does a low-order conversion on the difference between the input level and the high-order digital to analog converter output. The use of sampled data comparators allows this time delay between the conversion of the high and lower order bits while actually sampling the input signal at only one instant. The results from each conversion enter an 8-bit latch and are output to the three-state buffers on the falling edge of $\overline{RD}$.

Principles of Operation

The device operates in two modes, Read and Write-Read, selected by the Mode pin 7.

Read Mode

The converter is set to the Read Mode with the Mode pin 7 low, see Fig. 11.148. The Mode pin is internally connected to GND by a 50 μA current source which acts like a pull-down resistor. In the Read Mode,

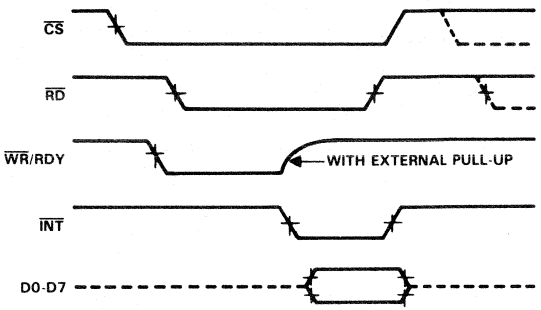


Fig. 11.148 TLC0820 read mode waveforms (mode pin low)

the WR/RDY pin is used as an output and is referred to as the RDY pin. In this Mode the RDY pin, which has an open drain output, will go low after the falling edge of CS and remains low to indicate that the device is busy. Conversion starts on the falling edge of RD and is completed in not more than 2.5 μs. Then INT falls and the RDY pin returns to a high impedance state when the conversion result is strobed into the output latch. Data outputs also change from high-impedance to active states at this time. After data is read, RD is taken high, INT returns high, and the data outputs return to their high impedance states.

Write-Read Mode

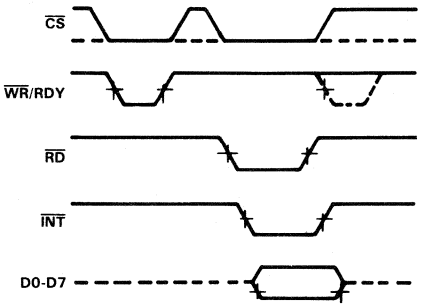


Fig. 11.149 TLC0820 write-read (fastest) mode with RD low before INT

Write-Read mode is selected with the Mode pin 7 high. The WR/RDY pin then becomes an input and is referred to as WR pin. Taking CS and WR pins low selects the converter, while WR is low the analog input signal is being sampled (acquired). In the Write-Read Mode at the rising edge

of $\overline{WR}$ the sample to hold transition is made and conversion started. Conversion is available approximately 600 ns after the $\overline{WR}$ pin returns high.

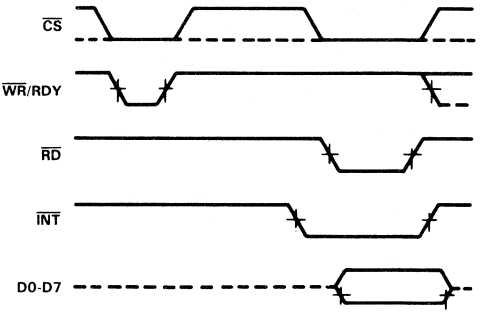


Fig. 11.150 TLC0820 write-read mode with $\overline{INT}$ low before $\overline{RD}$

There are two ways to read the conversion result output data, the choice depends on interface timing. For an interrupt driven system a microprocessor can respond to $\overline{INT}$ output going low by bringing $\overline{RD}$ input low to read the data output, Fig. 11.150. Alternatively, for fastest conversion, $\overline{RD}$ may be brought low in a minimum of 400 ns after $\overline{WR}$ goes high, Fig. 11.149, this will cause $\overline{INT}$ to go low and put the conversion result on the data output. $\overline{INT}$ is reset by the rising edge of either $\overline{CS}$ or $\overline{RD}$;.

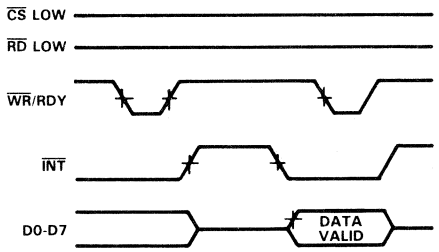


Fig. 11.151 TLC0820 write-read mode stand alone operation CS and RD low

Stand alone operation is possible, Fig. 11.151, by tying both $\overline{CS}$ and $\overline{RD}$ low with $\overline{WR}$ used as a clock input whose waveform has a minimum low period of 500 ns and minimum high period of 1.9 μs.

The recommended analog input voltage range for conversion is -0.1 V to V_{CC} + 0.1 V. Analog signals that are less than V_{REF-} + ½ LSB or greater that $V_{REF+} - ½$ LSB convert to 00000000 or 11111111 respectively. The reference inputs are fully differential with common-mode limits defined by the supply rails. The reference input values define the full

scale range of the analog input. This allows the gain of the analog to digital converter to be varied for ratiometric conversion by changing the V_{REF-} and V_{REF+} voltages.

Normally the overflow pin $\overline{OFLW}$ is at logical high, if the analog input voltage is higher than V_{REF+}, $\overline{OFLW}$ will be low at the end of conversion. It can be used to cascade two or more devices to improve resolution, 9 or 10 bits.

TLC0820 A/D Converter Interface to Texas Instruments TMS32010 Digital Signal Processor

Hardware

Because the control circuitry of the TLC0820 8-bit A/D converter operates much more slowly than the TMS32010, it cannot be directly interfaced. The following describes in detail the circuit shown in Fig. 11.152. All of the logic functions are implemented with 74ALS and 74LS Low-power Schottky Logic. The devices used are:

1	SN74ALS679	12-bit Address Comparator
1	SN74LS74	Dual Positive-edge Triggered D-type Flip-flops
1	SN74ALS465	Octal Buffer with 3-state Output
1	SN74LS32	Quad 2-Input OR-Gate

Principles of Operation

The TMS32010 Digital Signal Processor operates the TLC0820 by writing to the D flip-flops the necessary states of the $\overline{WR}$ and $\overline{RD}$ pins. The $\overline{WR}$ and $\overline{RD}$ signals must initially be at a logic high level, corresponding to a high level at the output of both flip-flops. To begin conversion, $\overline{WR}$ must be lowered for a minimum of 600 nanoseconds and then raised. The conversion begins on the rising edge of the $\overline{WR}$ pulse, and approximately 600 nanoseconds later, the conversion is complete. Since the OUT instruction requires two CLKOUT cycles (each of which lasts 200 ns when the TMS32010 is running at 20 MHz), the write pulse may be implemented with the following code (also shown in the software listing).

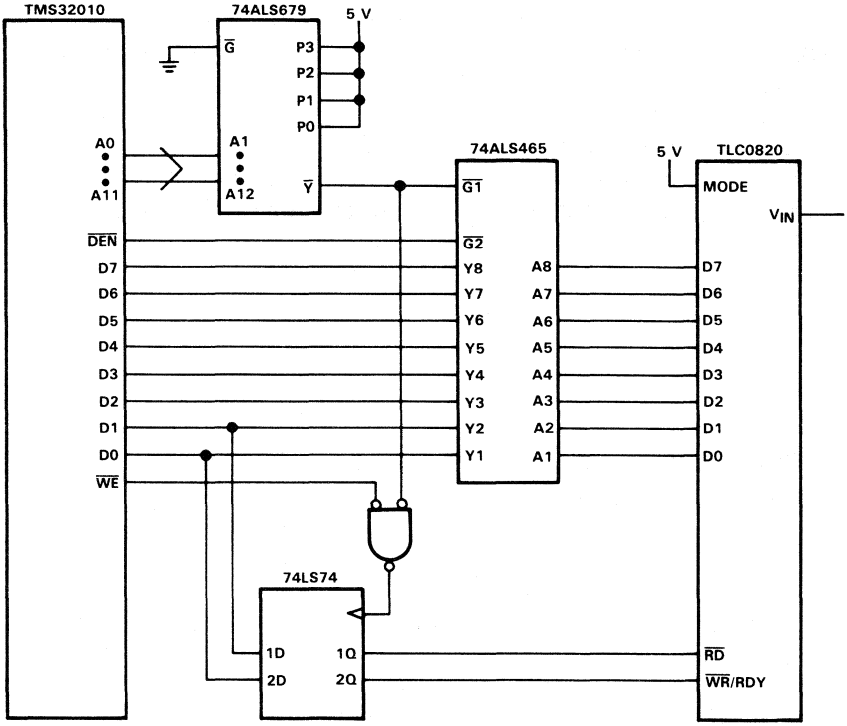


Fig. 11.152 TLC0820 to TMS32010 interface

```
OUT     <dma1>,<PA>
NOP                     or any 1-cycle instruction
OUT     <dma2>,<PA>     2-cycle instruction
```

where <dma1> is a data-memory address of a 16-bit word in which the bit assigned to $\overline{WR}$ is a logic-0 and the bit assigned to $\overline{RD}$ is a logic-1.

<PA> is the port address assigned to the TLC0820 and programmed into the 12-bit address comparator. The port address appears on pins A2 through A0 of the address bus during an OUT or an IN operation, pins A3 through A11 are driven low.

<dma2> is a data-memory address of a 16-bit word in which the bits assigned to both $\overline{WR}$ and $\overline{RD}$ are logic 1s.

After conversion has begun, the TLC0820 must have time to complete the A/D conversion before the result can be read by the DSP. There are two possible ways in which the TMS32010 may operate the TLC0820 (See Figs. 11.149 and 11.150). In the first method, the $\overline{RD}$ pin is lowered at the end of conversion (400 ns after the rising edge of $\overline{WR}$), before the TLC0820 signals that the conversion result is ready. The $\overline{RD}$ pin may be lowered in the following manner:

```
NOP                                  any 1-cycle instruction for 200 ns delay
OUT      <dma3>,<PA>                 2-cycle instruction
```

where <dma3> is the data-memory address of a 16-bit word in which the bit assigned to $\overline{RD}$ is a logic 0 and the bit assigned to $\overline{WR}$ is a logic 1.

<PA> is the port address assigned to the TLC0820 and programmed into the 12-bit address comparator. The port address appears on pins A2 through A0 of the address bus during an OUT or an IN operation, pins A3 through A11 are driven low.

No more than 300 ns after the $\overline{RD}$ pin is lowered, the conversion result will appear on the data bus of the TLC0820, where it can then be read by the TMS32010. Since the IN instruction requires 400 ns to execute, no additional delay is necessary. The data may be read and the $\overline{RD}$ pin subsequently raised in the following manner:

```
IN       <dma4>,<PA>     2 cycles
OUT      <dma2>,<PA>     2 cycles
```

where <dma4> is the data-memory address where the conversion result will be stored.

<PA> is the port address of the TLC0820.

<dma2> is the data-memory address of a 16-bit word in which both the bits assigned to $\overline{RD}$ and $\overline{WR}$ are logic 1s.

After the conversion result has been read it will be necessary to mask the data to obtain the 8-bits of interest. The interface timing is shown in Fig. 11.153.

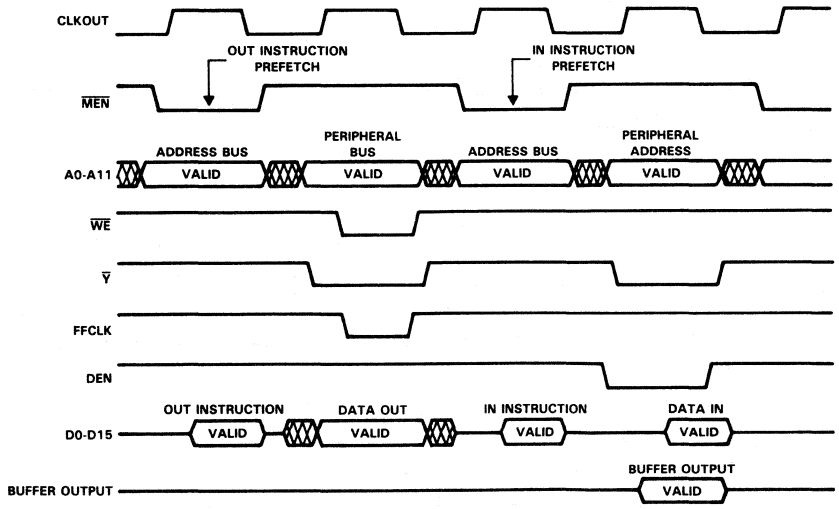


Fig. 11.153 TLC0820 to TMS32010 interface timing

```
*
*
* Software for TLC0820 Interface to TMS32010
*
*
        AORG   0
        LDPK   0        Set Data-Memory Page-Pointer to 0.
*
* Data memory locations 1, 2, and 3 should be initialized as follows:
*
*    ADDRESS   DATA
*       1      0001H
*       2      0002H
*       3      0003H
*
* This is necessary because the LSB of the data will be written to the /WR line of the TLC0820
* and the next-to-LSB will be written to the /RD line of the TLC0820.
*
* The following program segment manipulates the /WR and /RD control inputs of the TLC0820.
*
*
START   OUT    3,0      Raise /WR and /RD.
        NOP
        OUT    2,0      Lower /WR.
        NOP
        OUT    3,0      Raise /WR.
        NOP
        OUT    1,0      Lower /RD.
        IN     0,0      Read conversion result and store in data-memory location 0.
        OUT    3,0      Raise /RD.
        B      START
        END
```

TLC0820 A/D Converter Interface to Texas Instruments TMS32020 Digital Signal Processor

Hardware

Because the control circuitry of the TLC0820 8-bit A/D converter operates much more slowly than the TMS32020, it cannot be directly interfaced. The following describes in detail the interface circuit shown in Fig. 11.154. As drawn, the interface circuitry employs the following logic devices.

1	SN74LS138	3-Line to 8-Line Decoder
1	SN74LS00	Quad 2-Input NAND Gate
1	SN74LS04	Hex Inverters
1	SN74LS32	Quad 2-Input OR Gate
1	SN74LS175	Quad D-type Flip-flop

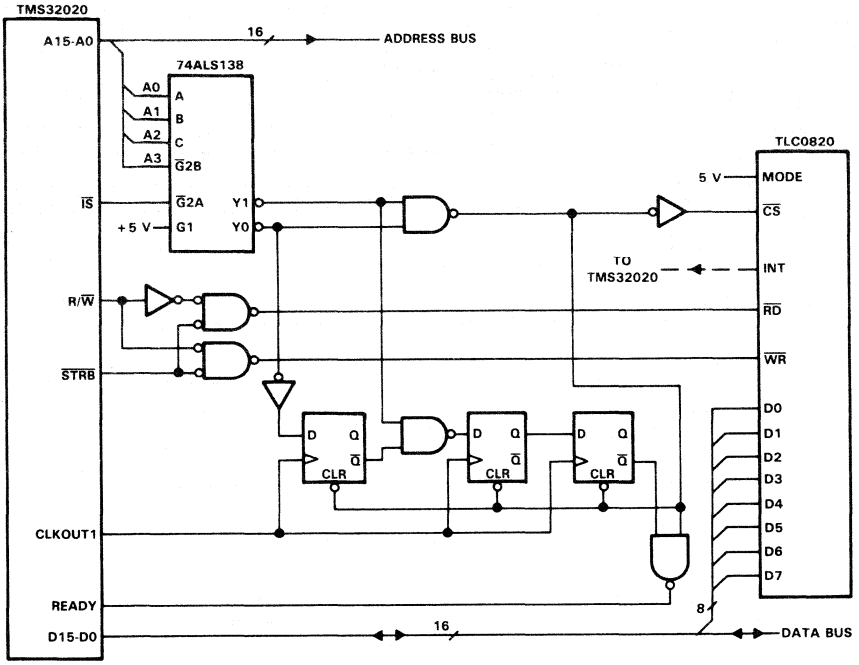


Fig. 11.154 TLC0820 to TMS32020 interface

Principles of Operation

The 74LS138 decodes the addresses assigned to the TLC0820. One of the addresses is used when performing a write operation; the other, for a read. The two different addresses are necessary to ensure that the correct number of wait states is provided for the write and read operations.

The controlling software is shown in the following listing. To begin conversion, the TMS32020 must supply the TLC0820 with a write pulse ($\overline{WR}$) of at least 600 ns. With the TMS32020 running at 20 MHz and the TLC0820 configured as slow memory, three wait states are necessary to provide a write pulse of sufficient length.

```
OUT     *,0     output the data word specified by the current auxiliary register
                to peripheral on port address 0. A low at Y0 of the
                74ALS138 will allow 3 wait states.
```

After conversion has begun (with the rising edge of the $\overline{WR}$ signal), the TMS32020 must wait at least 600 ns before the conversion result can be read. Sufficient delay should be provided in software.

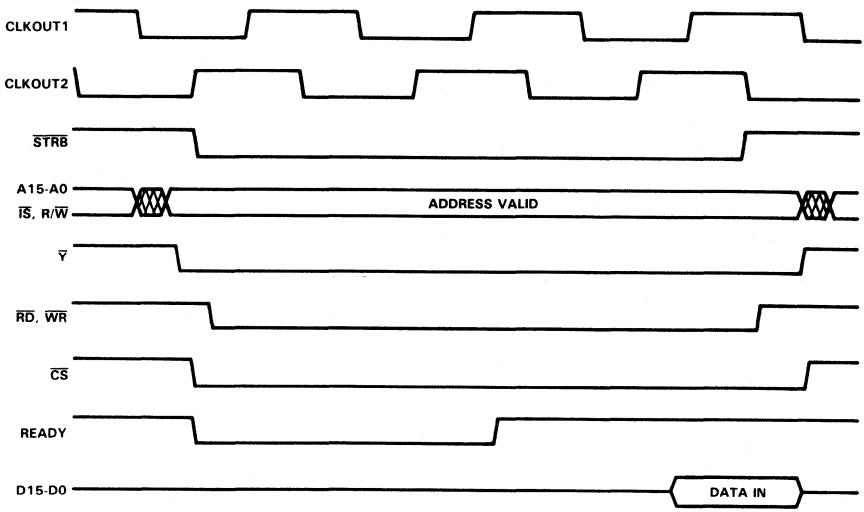


Fig. 11.155 TLC0820 IN instruction timing – two wait states

IN	<dma>,1	read a data word from peripheral on port address 1. Store in data memory location <dma>. A low at Y1 of the 74ALS138 will provide 2 wait states.

To read the conversion result, sufficient wait states should be provided to allow for the data access time (320 ns minimum) of the TLC0820. As shown in the timing diagram of Fig. 11.155, two wait states are provided when accessing port 1.

```
*
*
* Software for the TLC0820 to TMS32020 Interface
*
*
        AORG   >0
        LDPK   >0       Set Data-Memory Page Pointer to >0.
*
* The following statement writes the data contained in the data-memory location specified by the
* current address-register pointer to the TLC0820. The data is not important* the write pulse is.
* The interface provides for 3 wait states.
*
START   OUT    *,0
*
* The following instructions provide a 600 ns delay between the rising edge of the write pulse to
* the falling edge of the subsequent read.
*
        NOP
        NOP
```

```
*
* The following IN instruction reads the conversion result and stores the result in data-memory
* location >60. The interface provides for 2 wait states.
*
        IN      >60,1
        B       START
        END
```

TLC7524 Advanced LinCMOS™ 8-BIT MULTIPLYING DIGITAL-TO-ANALOG CONVERTER

The TLC7524 is an 8-bit multiplying digital-to-analog converter (DAC) built using Advanced LinCMOS technology. The input latch, with a load cycle similar to the "write" cycle of a random access memory, allows easy interface to most popular microprocessors. Operation from 5 to 15 V single supply with two or four quadrant multiplying capability makes the TLC7524 an ideal choice for many microprocessor-controlled gain-setting and signal control applications. The reduced set-up time requirements of the digital input can allow direct interface to microprocessors and digital signal processors without extra latches. The TLC7524 is manufactured without the need for thin-film resistors and laser trimming to achieve 0.5 LSB accuracy. Performance information is shown in Table 11.19 and the pin layout in Fig. 11.156 showing both dual-in-line and 'SO" surface mount packages.

Table 11.19

PERFORMANCE	**TLC7524**
Resolution	8 Bits
Linearity Error (max)	0.5 LSB
Chip-Select Set-up Time (min)	40 ns
Data Bus Set-up Time (min)	25 ns
Write Pulse Duration (min)	40 ns
Settling Time (max)	100 ns
Propagation Delay (max)	80 ns
Power Dissipation at VDD=5 V (max)	5 mW

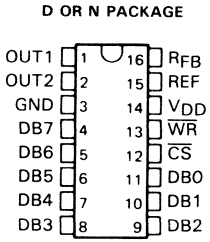


Fig. 11.156 TLC7524 pinout (top view)

Principles of operation

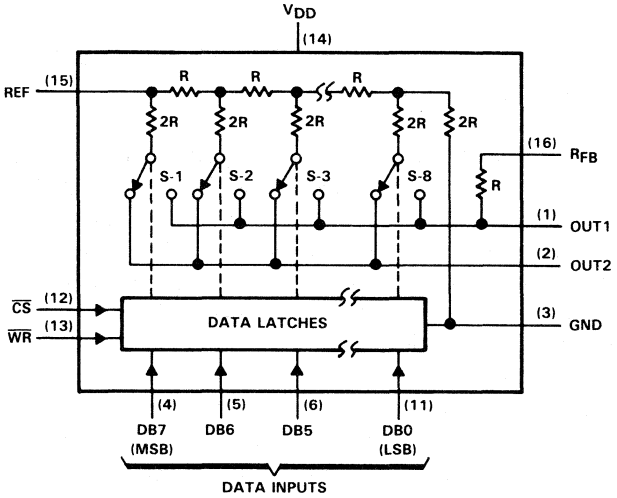
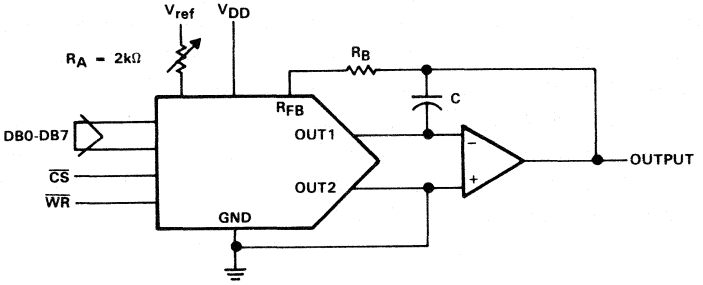


Fig. 11.157 TLC7524 functional block diagram

Functional block diagram, Fig. 11.157, shows the inverted R-2R ladder, analog switches and data input latches used in the implementation of the TLC7524 8-bit multiplying DAC. Binary weighted currents are switched between the OUT1 and OUT2 output lines which, if held at the same voltage, maintain a constant current in each resistive ladder leg independent of the switch state. In the current mode of operation shown in Fig. 11.158 OUT2 is connected to analog ground while OUT1 is held at ground potential by the op-amp's virtual ground.

Glitch impulse on the output is usually highest and poses the greatest problem during the most-significant bit (MSB) change. This is minimised

in the TLC7524 design by using segmentation which, through a modification of the R-2R ladder, decodes higher order bits to control three equally weighted current sources (resistor ladder legs). This is in place of two current sources with a ratio of 2:1 between them. For the MSB-1 to MSB change instead of selecting between 2:1 ratio current sources an second MSB-1 weighted current source is added to that existing to give the MSB current. A third MSB-1 current source is added when MSB + (MSB-1) current output is required. Lower order bits use the usual R-2R ladder network.



UNIPOLAR BINARY CODE

DIGITAL INPUT MSB LSB	ANALOG OUTPUT
11111111	$-V_{ref}$ (255/256)
10000001	$-V_{ref}$ (129/256)
10000000	$-V_{ref}$ (128/256) = $-V_{ref}/2$
01111111	$-V_{ref}$ (127/256)
00000001	$-V_{ref}$ (1/256)
00000000	0

LSB = 1/256 (V_{ref}).

Fig. 11.158 TLC7524 unipolar (2-quadrant) operation

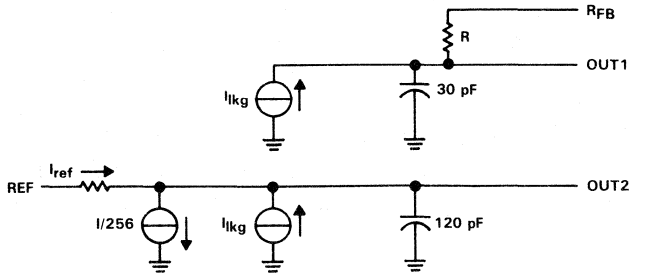


Fig. 11.159 TLC7524 equivalent circuits with all digital inputs low

The equivalent circuit for all digital inputs low is shown in Fig. 11.159, in this condition the entire reference current is switched to OUT2. The current source I/256 represents the constant current flowing through the 2R terminating resistor of the R-2R ladder, while I_{lkg} represents leakage currents to the substrate. The equivalent circuit for all digital inputs high is similar to that of Fig. 11.159 except that I_{ref} is switched to OUT1.

The capacitances appearing at OUT1 and OUT2 are dependent upon the digital input code. With all digital inputs high, the off-state switch capacitance of 30 pF (max) appears at OUT2 and the on-state switch capacitance of 120 pF (max) appears at OUT1. With the digital inputs low the situation is reversed and is shown in this condition in Fig. 11.159. For a mixture of digital inputs the output capacitance will vary between these values. This DAC output capacitance will appear across the input of the op-amp, shown in Fig. 11.158, and cause a feedback pole. Depending on the op-amp used, to prevent overshoot, ringing or even oscillation, a compensation capacitor C, see Fig. 11.158, of typically 15 pF may be needed.

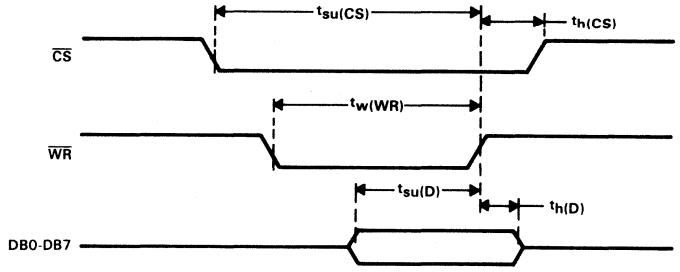
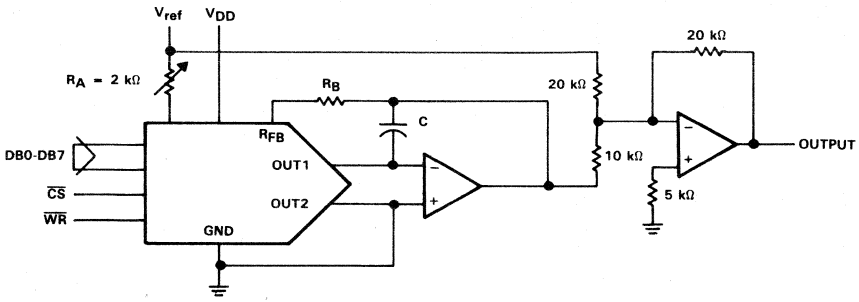


Fig. 11.160 TLC7524 timing diagram

Interfacing the TLC7524 DAC to a microprocessor is accomplished via the data bus and the $\overline{CS}$ and $\overline{WR}$ control signals, see Fig. 11.160. The input latches are transparent when $\overline{CS}$ and $\overline{WR}$ are both low, with the TLC7524 analog output directly corresponding to the data activity on the DB0-DB7 data bus inputs. When either the $\overline{CS}$ or $\overline{WR}$ signal goes high, the data on the DB0-DB7 inputs are latched until the $\overline{CS}$ and $\overline{WR}$ go low again. When $\overline{CS}$ is high, the data inputs are disabled regardless of the state of the $\overline{WR}$ signal.

Circuit configurations and input coding for 2 and 4-quadrant multiplication are given in Figs. 11.158 and 11.161 respectively. R_A and R_B are

used only if gain adjustment is required, C is for phase compensation as discussed above.



BIPOLAR (OFFSET BINARY) CODE

DIGITAL INPUT MSB LSB	ANALOG OUTPUT
11111111	V_{ref} (127/128)
10000001	V_{ref} (1/128)
10000000	0
01111111	$-V_{ref}$ (1/128)
00000001	$-V_{ref}$ (127/128)
00000000	$-V_{ref}$

LSB = 1/128 (V_{ref}).

Fig. 11.161 Bipolar (4-quadrant) operation

TLC7524 Digital-to-Analog Converter Interface to Texas Instruments TMS32010 Digital Signal Processor

Hardware

Due to the high-speed operation of the internal logic circuitry of the TLC7524 8-bit digital-to-analog converter, the interface to the TMS32010 Digital Signal Processor requires a minimum of external circuitry. As shown in Fig. 11.162, the interface circuitry consists of logic to decode the address of the peripheral. Here we have used one SN74ALS679 12-bit Address Comparator.

Principles of Operation

As shown in Fig. 11.163, when the TMS32010 executes an OUT instruction, the $\overline{MEN}$ output remains high and the address of the peripheral specified by the instruction is placed on the address bus. The three-bit port

address will appear on pins A0 through A2, and pins A3 through A11 will be driven low.

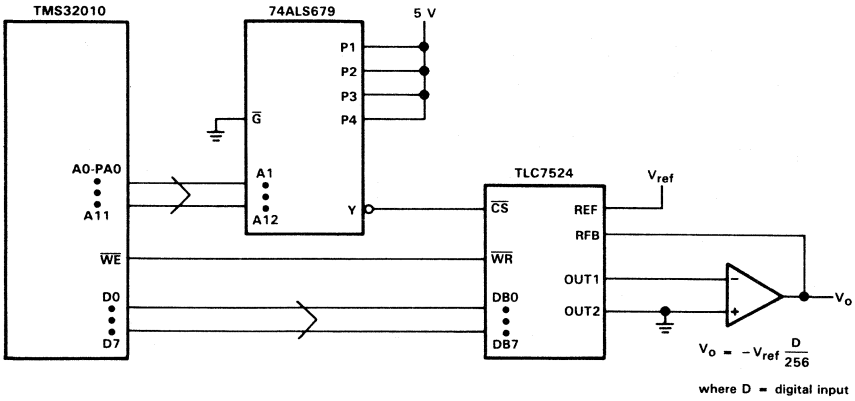


Fig. 11.162 TLC7524 to TMS32010 interface

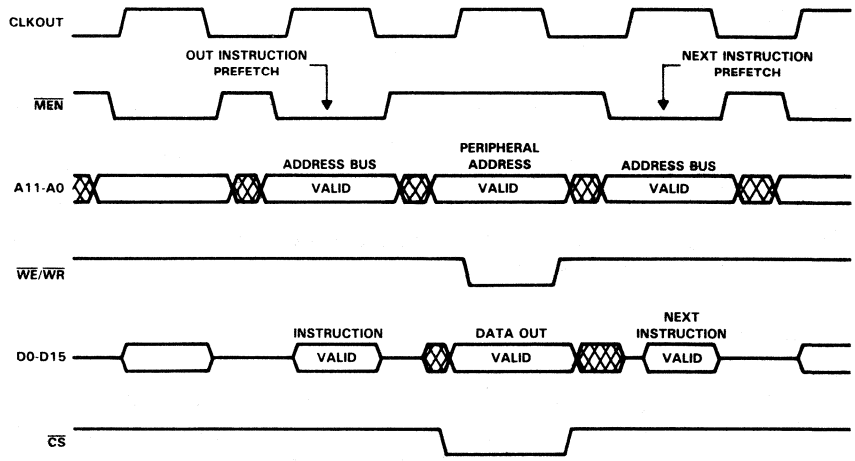


Fig. 11.163 OUT instruction timing for TMS32010

The P-inputs of the SN74ALS679 Address Comparator should be hard wired with the preprogrammed address of the I/O port which corresponds to the TLC7524. For example, if the TLC7524 is at I/O port 0, all of the P-inputs of the 74ALS679 should be tied to logic high. If however the TLC7524 is at I/O port address 1, input P2 of the 74ALS679 should be tied to logic low and P1, P3 and P4 should be tied to logic high (detailed

information on the 74ALS679 may be found in the ALS/AS Logic Data Book).

When the address appearing at the address bus of the TMS32010 corresponds to the address programmed into the 74ALS679, the output of the 74ALS679 will be driven low, enabling the TLC7524 (see Fig. 11.163, the timing diagram). The data to be written to the TLC7524 should then be on the data bus of the TMS32010 so that when $\overline{WE}$ goes low, the data can be latched into the TLC7524. The controlling software follows.

```
*
* Software for TLC7524 to TMS32010 Interface
*
*
*
*
DAT     EQU     0       DAT is defined to be data-memory address zero.
*
        AORG    0
        LDPK    0       Set Data-Memory Page Pointer to 0.
*
START   OUT     DAT,0   Place the contents of data-memory address 0 in
*                       the data latch of the DAC. The TLC7524 is at port address 0.
*       B       START
        END
```

TLC7524 Digital-to-Analog Converter Interface to Texas Instruments TMS32020 Digital Signal Processor

Hardware

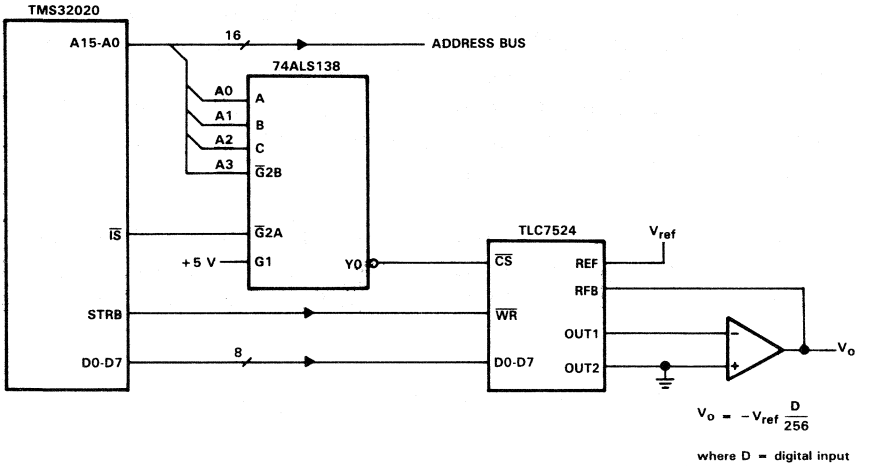


Fig. 11.164 TLC7524 to TMS32020 interface

Due to the high-speed operation of the internal logic ciruitry of the TLC7524 8-bit digital-to-analog converter, the interface to the TMS32020 Digital Signal Processor requires a minimum of external circuitry.

As shown in Fig. 11.164, the interface circuitry consists of logic to decode the address of the peripheral. Here we have used one SN74ALS138 3-to-8-Line Decoder.

Principles of Operation

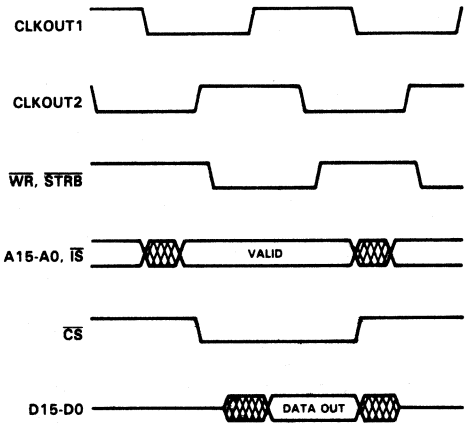


Fig. 11.165 TMS32020 OUT instruction cycle

As shown in Fig. 11.165, when the TMS32020 executes an OUT instruction, the peripheral address is placed on the address bus and the $\overline{IS}$ line goes low, indicating that the address on the bus corresponds to an I/O port and not external program or data memory. A low level at $\overline{IS}$ will enable the 74ALS138 decoder, and the Y-output, corresponding to the address on the bus, is brought low.

When the Y-output is brought low, the TLC7524 is enabled and the data appearing on the databus will be latched into the D/A converter by $\overline{STRB}$. The controlling software is shown in the following listing.

```
*
*
* Software for TLC7524 to TMS32020 Interface
*
*
        AORG   >0
        LDPK   >0       Set data memory page pointer to 0.
```

```
*
* The following program segment transfers the data in the lower 16 bits of the accumulator to the
* data latches of the TLC7524 digital-to-analog converter.

*
START   SACL    >60     Save low accumulator in >60.
        OUT     >60,0   Move contents of >60 to TLC7524.
        B       START
        END
```

TLC7528 Advanced LinCMOS™ DUAL 8-BIT MULTIPLYING DIGITAL-TO-ANALOG CONVERTER

The TLC7528 is a dual 8-bit multiplying digital-to-analog converter (DAC) built using Advanced LinCMOS technology and featuring excellent DAC-to DAC matching. Separate on-chip data latches for each DAC use data transferred from a common 8-bit input port. A single input controls which data latch is loaded, in a load cycle similar to the "write" cycle of a random access memory, this allows easy interface to most popular microprocessors. The reduced set-up times of the digital input can allow direct interface to microprocessors and digital signal processors without extra latches. Key performance information is shown in Table 11.20 and the pin layout in Fig. 11.166 showing both dual-in-line and surface mount "SO" packages.

Table 11.20

PERFORMANCE (V_{DD} = 5 V)	**TLC7524**
Resolution	8 Bits
Linearity Error (max)	0.5 LSB
Chip-Select Set-up Time (min)	50 ns
DAC Select Set-up Time	50 ns
Data Bus Set-up Time (min)	25 ns
Write Pulse Duration (min)	50 ns
Settling Time (max)	100 ns
Propagation Delay (max)	80 ns
Power Dissipation at V_{DD} = 5 V (max)	5 mW

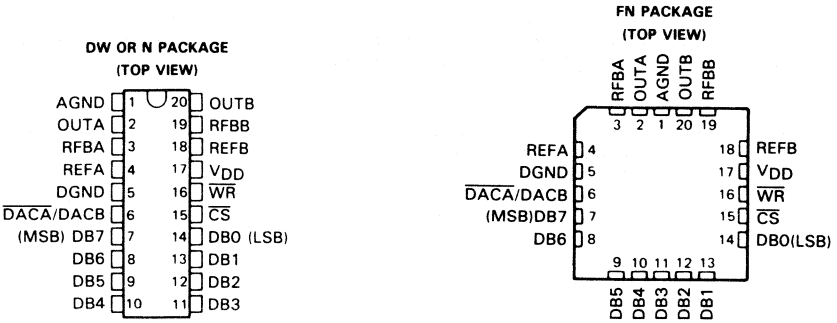
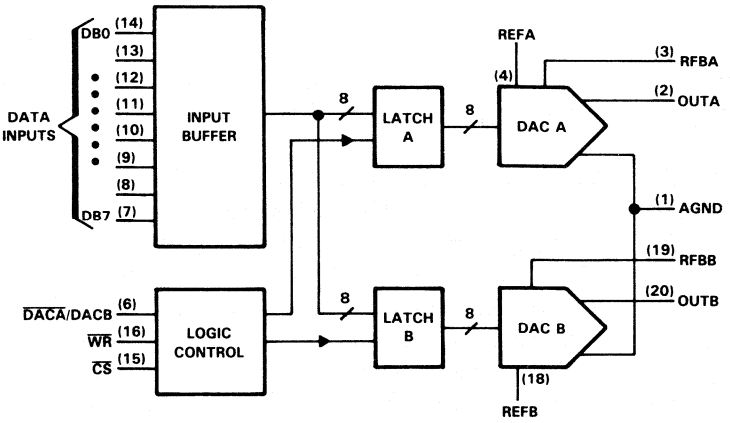


Fig. 11.166 TLC7528 pinouts (top view)



MODE SELECTION TABLE

DACA/ DACB	CS	WR	DACA	DACB
L	L	L	WRITE	HOLD
H	L	L	HOLD	WRITE
X	H	X	HOLD	HOLD
X	X	H	HOLD	HOLD

L = low level, H = high level, X = don't care

Fig. 11.167 TLC7528 functional block diagram

Functional block diagram Fig. 11.67, shows the two identical digital to analog converters DACA and DACB. Each consists of an inverted R-2R ladder, analog switches, and input data latches. Segmentation of the high-order bits is used to minimise glitches during most significant bit changes, see TLC7524 for more details.

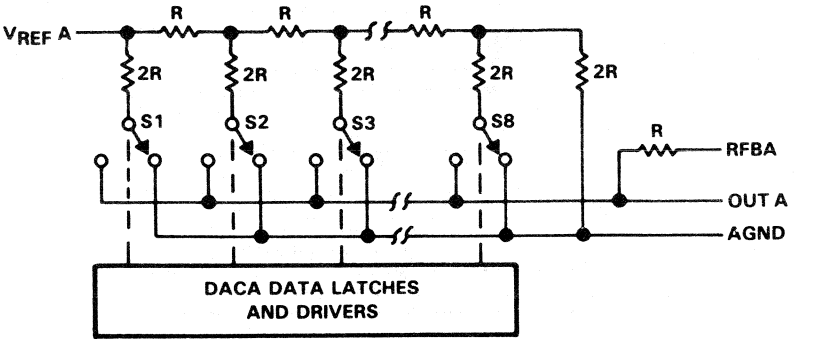


Fig. 11.168 TLC7528 simplified functional circuit

Fig. 11.168 shows how the binary-weighted currents of DACA (which is identical to DACB) are switched between output OUTA and analog ground AGND. With equal output voltages on these pins, a constant current is maintained in each switch leg independant of switch position. Output OUTA is usually held by the virtual ground of an op-amp to the same analog ground as AGND (see Fig. 11.171).

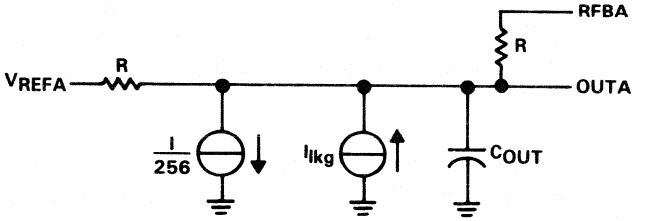


Fig. 11.169 TLC7528 DACA equivalent circuit with all digital inputs high

Equivalent Circuits for DACA and DACB are identical, DACA is shown in Fig. 11.169. The output current depends on the number of switches connected to OUTA, with all digital inputs high this is the reference current less the ladder terminating resistor current I/256. Due to internal junctions a small leakage current I_{lkg} flows which doubles every 10°C. Output capacitance C_o has a value which varies from 50 pF to 120 pF and depends on the parallel combination of NMOS switches connected to the output. Output resistance also depends on the input code and varies from the nominal ladder resistance, R, over the range 0.8R to 3R. This variation will cause a gain change of the follow op-amp circuit (Fig. 11.171) with code change and dictates the allowable value of op-amp input offset voltage without causing large differential non-linearity and a non-monotonic output.

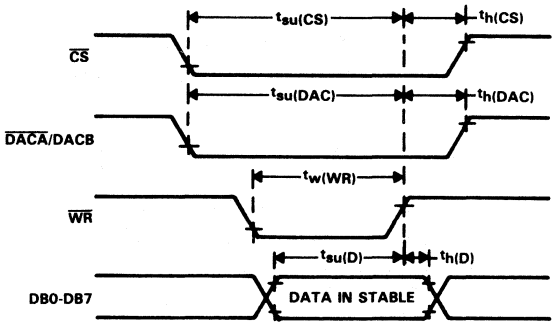
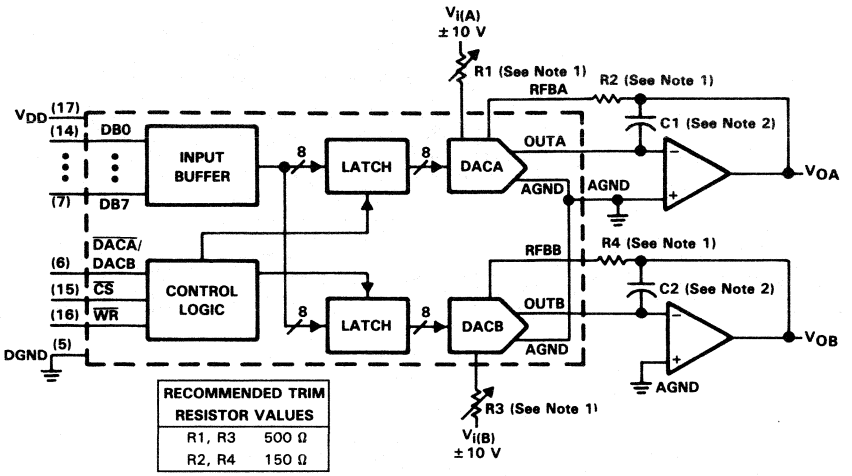


Fig. 11.170 TLC7528 timing diagram

Control signals, $\overline{CS}$, $\overline{WR}$ and $\overline{DACA}$/DACB, and the data bus are used to interface the TLC7528 to a microprocessor. When both $\overline{CS}$ and $\overline{WR}$ are low the analog output, specified by the DACA/DACB control line, responds to the activity on the DB0-DB7 data bus inputs (see Fig. 11.170). With $\overline{CS}$ and $\overline{WR}$ low the input latches are transparent and the input data directly affects the selected analog output. When either $\overline{CS}$ or $\overline{WR}$ goes high the data on DB0-DB7 is latched until both $\overline{CS}$ and $\overline{WR}$ go low again. When $\overline{CS}$ is high the data inputs are disabled regardless of the state of the $\overline{WR}$ signal.

The TLC7528 can be operated over a supply voltage range of 5 to 15 V, however the inputs are TTL compatible only when the supply voltage is 5 V.

Circuit configuration for unipolar (2-quadrant multiplication) and bipolar (4-quadrant) multiplication are shown in Fig. 11.171 and 11.172 respectively.



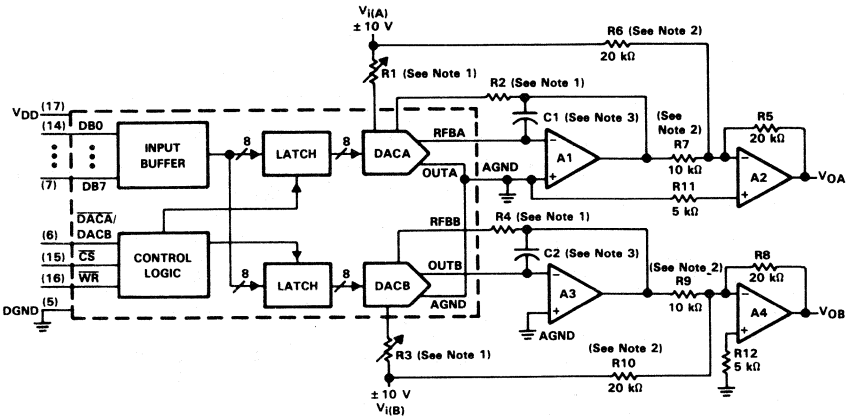
NOTES: 1. R1, R2, R3, and R4 are used only if gain adjustment is required. See table for recommended values. Make gain adjustment with digital input of 255.
2. C1 and C2 phase compensation capacitors (10 pF to 15 pF) are required when using high-speed to prevent ringing or oscillation.

UNIPOLAR BINARY CODE

DAC LATCH CONTENTS MSB LSB†	ANALOG OUTPUT
11111111	$-V_i$ (255/256)
10000001	$-V_i$ (129/256)
10000000	$-V_i$ (128/256) = $-V_i/2$
01111111	$-V_i$ (127/256)
00000001	$-V_i$ (1/256)
00000000	$-V_i$ (0/256) = 0

† 1 LSB = $(2^{-8})V_i$

Fig. 11.171 TLC7528 unipolar (2-quadrant) operation



NOTES: 1. R1, R2, R3, and R4 are used only if gain adjustment is required. See table in Fig. 11.171 for recommended values. Adjust R1 for V_{OA} = 0 V with code 10000000 in DACA latch. Adjust R3 for V_{OB} = 0 V with 10000000 in DACB latch.
2. Matching and tracking are essential for resistor pairs R6, R7, R9, and R10.
3. C1 and C2 phase compensation capacitors (10 pF to 15 pF) may be required if A1 and A3 are high-speed amplifiers.

BIPOLAR (OFFSET BINARY) CODE

DAC LATCH CONTENTS MSB LSB‡	ANALOG OUTPUT
11111111	V_i (127/128)
10000001	V_i (1/128)
10000000	0 V
01111111	$-V_i$ (1/128)
00000001	$-V_i$ (127/128)
00000000	$-V_i$ (128/128)

‡ 1 LSB = $(2^{-7})V_i$

Fig. 11.172 TLC7528 bipolar (4-quadrant) operation

TLC32040 FAMILY OF ANALOG INTERFACE CIRCUITS (AIC)

Introduction to Data Acquisition Devices for Digital Signal Processing

Digital Signal Processing (DSP) involves the representation, transmission, and manipulation of signals using numerical techniques and digital processors. It has been a fast growing technology during the past few years, with the impact of programmability and VLSI broadening the range of problems that can be served with DSP. Its applications have been expanded to encompass not only traditional radar signal processing but also today's image processing, speech processing, and telecommunications.

Both the theoretical and practical aspects of DSP have made tremendous progress. While more DSP algorithms are being discovered, better tools are also being developed to implement these algorithms. One of the most important recent breakthroughs in electronic technology is the high-speed digital signal processor. These single-chip processors are now commercially available in very large-scale integration (VLSI) circuits from semiconductor suppliers. Digital signal processors are essentially high-speed microprocessors/microcomputers, designed specifically to perform computation-intensive algorithms. By taking advantage of the advanced architecture, parallel processing, and dedicated DSP instruction sets, these devices can execute millions of DSP operations per second.

With the advantages offered by VLSI, innovative engineers are discovering more and more applications where digital signal processors efficiently provide better solutions than their analog counterparts for reasons of reliability, flexibility, repeatibility, compactness, and long-term stability. Digital signal processors do, however, require accurate analog-to-digital (A/D) and digital-to-analog (D/A) converters. These converters interface the inputs and outputs of the DSP to the analog world.

Texas Instruments has developed a state-of-the-art analog interface IC, the TLC32040, to interface Texas Instruments TMS320 family of digital signal processors to the analog world. In addition to the A/D and D/A converters, the TLC32040 also contains filters for reducing noise and alaising, which is an undesired sampled-data phenomenon. The TLC32040 also contains a serial port, which allows direct interface to most TMS320 digital signal processors. More information on the TLC32040 is presented in the following pages.

TLC32040 AIC Interface to TMS320 Family Digital Signal Processors

Description

The TLC32040 is a complete analog-to-digital and digital-to-analog input/output system on a single monolithic CMOS chip. This device integrates a band-pass, switched-capacitor, antialiasing input filter, a 14-bit resolution A/D converter, four microprocessor-compatible serial port modes, a 14-bit resolution D/A converter, and a low-pass, switched-capacitor, output-reconstruction filter. The device offers numerous combinations of Master Clock input frequencies and conversion/sampling rates, which can be changed via digital processor control.

Typical applications for this IC include modems (7.2-, 8-, 9.6-, 14.4, and 19.2-kHz sampling rates), analog interface for digital signal processors, speech recognition/storage systems, industrial process control, biomedical instrumentation, acoustical signal processing, spectral analysis, data acquisition, and instrumentation recorders. Four serial modes, which allow direct interface to the TMS32011, TMS32020, and TMS320C25 digital signal processors, are provided. Also, when the transmit and receive sections of the Analog Interface Circuit (AIC) are operating synchronously, it will interface to two SN54299 or SN74299 serial-to-parallel shift registers. These shift registers can then interface in parallel to the TMS32010, other digital signal processors, or to external FIFO circuitry. Output data pulses are emitted to inform the processor that data transmission is complete, or to allow the DSP to differentiate between two transmitted bytes. A flexible control scheme is provided so that the functions of the IC can be selected and adjusted coincidentally with signal processing via software control.

The antialiasing input filter comprises seventh-order and fourth-order CC-type (Chebyshev/elliptic transitional) low-pass and high-pass filters, respectively, and a fourth order equalizer. The input filter is implemented in switched-capacitor technology and is preceded by a continuous time filter to eliminate any possibility of aliasing caused by sampled data filtering. When no filtering is desired, the entire composite filter can be switched out of the signal path. A selectable, auxiliary, differential analog input is provided for applications where more than one analog input is required.

The A/D and D/A converters each have 14 bits of resolution with 10 bits of integral linearity guaranteed over any 10-bit range. Currently the AIC is being evaluated from the perspective of offering several different versions of the AIC. These AICs would be screened at the factory for

enhanced performance in areas such as total harmonic distortion or A/D and D/A accuracy. One of these devices may provide integral linearity greater than 10 bits.

The A/D and D/A architectures guarantee no missing codes and monotonic operation. An internal voltage reference is provided to ease the design task and to provide complete control over the performance of the IC. The internal voltage is brought out to a pin and is available to the designer. Separate analog and digital voltage supplies and grounds are provided to minimize noise and ensure a wide dynamic range. Also, the analog circuit path contains only differential circuitry to keep noise to an absolute minimum. The only exception is the DAC sample-and-hold, which utilizes pseudo-differential circuitry.

The output-reconstruction filter is a seventh-order CC-type (Chebyshev/elliptic transitional low-pass filter with a fourth-order equalizer) and is implemented in switched-capacitor technology. This filter is followed by a continuous-time filter to eliminate images of the digitally encoded signal.

Principles of Operation

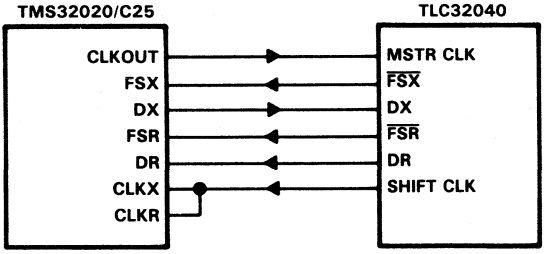


Fig. 11.173 AID interface to TMS32020/C25

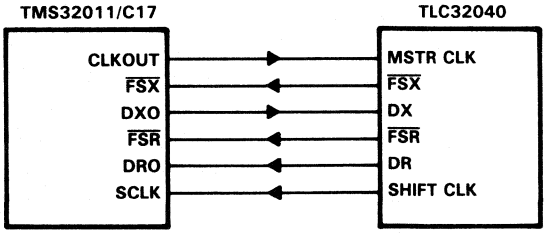


Fig. 11.174 AIC interface to TMS32011/C17

As shown in Figs. 11.173 and 11.176 the AIC is easily interfaced to the TMS32020, TMS320C25, TMS320C17 and TMS32011 serial ports. The TMS32020/C25 can communicate with the AIC either synchronously or asynchronously depending on the information in the control register. If d5 in the AIC control register is a 0, the transmit and receive sections of the AIC will operate asynchronously; if d5 is a 1, these sections will operate synchronously. The operating sequence for synchronous communication with the TMS32020 and TMS320C25 is shown in Fig. 11.175.

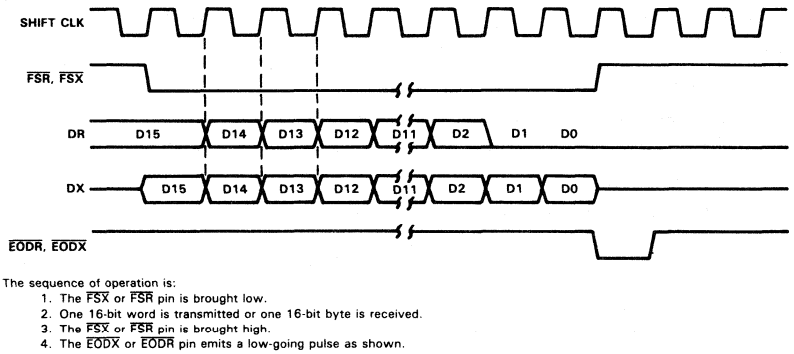


Fig. 11.175 Operating sequence for AIC-TMS32020/C25 interface – synchronous

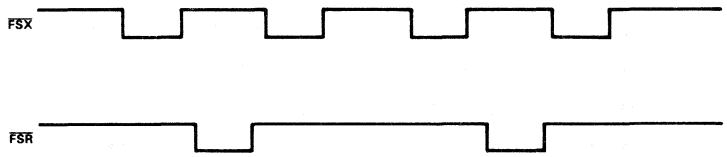
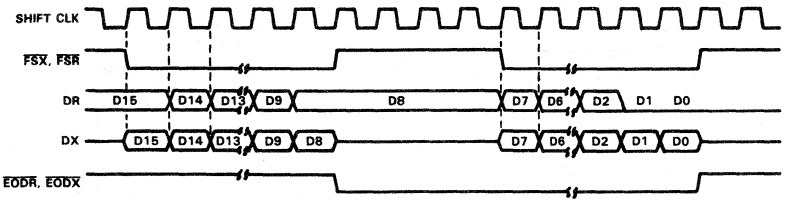


Fig. 11.176 Asynchronous communication: AIC-TMS32020/C25 interface

For asynchrounous communication, the operating sequence is similar but $\overline{\text{FSX}}$ and $\overline{\text{FSR}}$, in general, do not occur at the same time (see Fig. 11.176). For proper operation, the TXM bit in the TMS32020/C25 control register should be set to 0 so that the $\overline{\text{FSX}}$ pin of the TMS32020/C25 is configured as an input, the format status bit (FO) of the TMS32020/C25 should be set to 0, and the AIC WORD/BYTE pin should be at a logic high. After each receive and transmit operation, the TMS32020/C25 asserts an internal receive (RINT) and transmit (XINT) interrupt which may be used to control program execution.

The interface to the TMS32011 and TMS320C17 also requires no external circuity and, like the TMS32020 interface, can operate either synchronously or asynchronously. The operating sequence for synchronous communication is shown in Fig. 11.177. In this case, the AIC must transmit data in two 8-bit bytes and therefore the WORD/$\overline{\text{BYTE}}$ pin must be held low. As with the TMS32020/C25, the TMS32011/C17 control register should be configured to respond to external framing pulses ($\overline{\text{FSX}}$ and $\overline{\text{FSR}}$) and an external Shift Clock. The $\overline{\text{FSX}}$ and $\overline{\text{FSR}}$ interrupts may be used to control program execution.



The sequence of operation is:

1. The $\overline{\text{FSX}}$ or $\overline{\text{FSR}}$ pin is brought low.
2. One 8-bit is transmitted and one 8-bit byte is received.
3. The $\overline{\text{EODX}}$ and $\overline{\text{EODR}}$ pins are brought low.
4. The $\overline{\text{FSX}}$ and $\overline{\text{FSR}}$ emit a positive frame-sync pulse that is four shift clock cycles wide.
5. One 8-bit byte is transmitted and one 8-bit byte is received.
6. The $\overline{\text{EODX}}$ and $\overline{\text{EODR}}$ pins are brought high.
7. The $\overline{\text{FSX}}$ and $\overline{\text{FSR}}$ pins are brought high.

Fig. 11.177 Operating sequence for AIC-TMS32011/C17 interface – synchronous

Since the TMS32010 and the TMS320C15 have no serial port, a direct connection to the AIC is not possible. However, as shown in Fig. 11.178, the AIC can interface to two 74LS299 Serial-to-Parallel shift registers which can then be connected to the data bus of the TMS32010 or the TMS320C15. The AIC must be operated synchronously and in 16-bit (word) mode. The operating sequence is shown in Fig. 11.179. In this configuration, the device labeled A1 represents a data-delay. The data-delay may be realized with three cascaded AND gates connected between "I" and "O" ("C" is not connected), or a single D-type flip-flop with the D input connected to "I", the Q-output connected to "O" and the noninverting clock input connected to "C". All of the logic circuitry excluding the two 74LS299s and the 74LS138 3-to-8-line decoder may be replaced with a single PAL. Data is written to the AIC with an OUT instruction, specifying port address 1; a read is accomplished with an IN instruction specifying port address 0.

For further information concerning TLC32040 family devices contact Texas Instruments Customer Response Centre and ask for the application report entitled "Interface the TLC32040 with the TMS320 Family of Digital Signal Processors" and TLC32040 family data sheets.

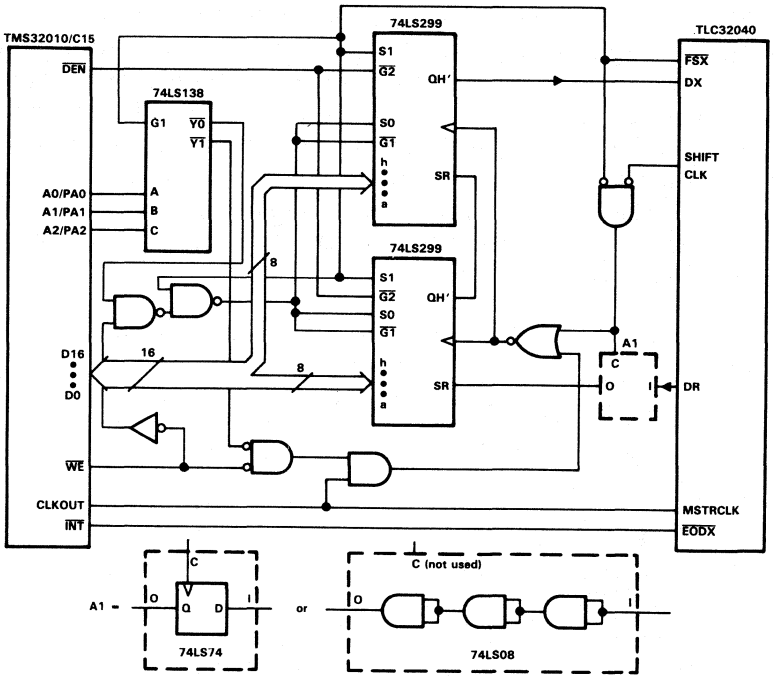


Fig. 11.178 AIC interface to TMS32010/C15

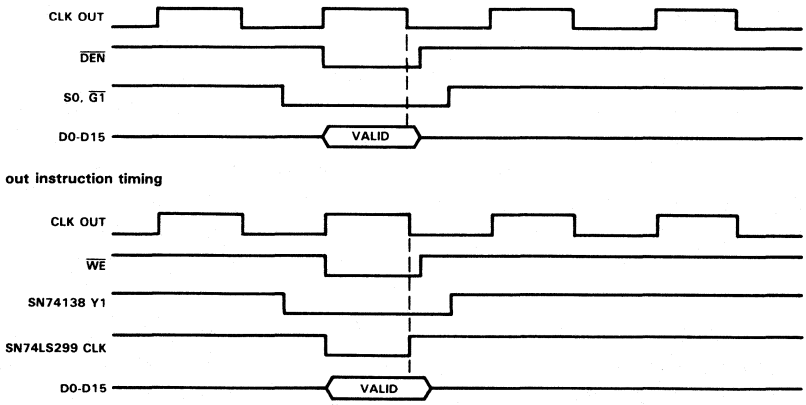
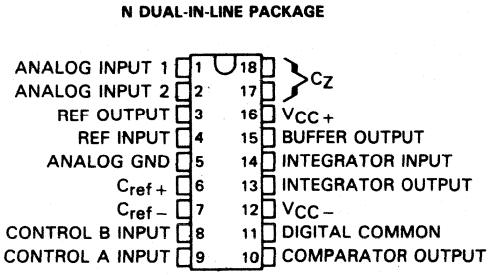


Fig. 11.179 TMS32010/C15-TLC32040 interface timing

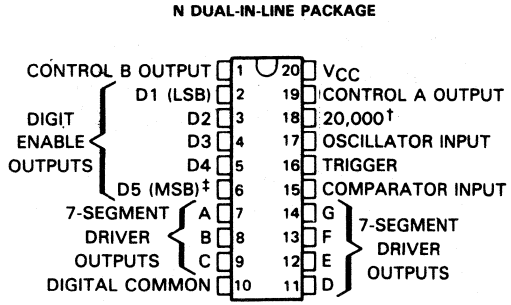
TL500C THRO' TL503C DUAL-SLOPE-INTEGRATING ANALOG-TO-DIGITAL CONVERTERS

The TL500C and TL501C analog processors and TL502C and TL503C digital processors provide the basic functions for a dual-slope-integrating analog-to-digital converter.

The TL500C and TL501C analog processors have true differential inputs and automatic zero. They feature automatic polarity status and have a high input impedance in the range of $10^9\ \Omega$ typically. Fig. 11.180 shows the pinout of the TL500C and TL501C. The TL500C has a resolution of 14 bits when used with the TL502C digital processor. It features a linearity error of 0.001% and has a 4-1/2 digit readout accuracy when used with an external reference. The TL501C is capable of 10 to 13 bits of resolution when used with the TL502C processor. It has a linearity error of 0.01% and a 3-1/2 digit readout accuracy.



(a) TL500C and TL501C



(b) TL502

Fig. 11.180 Pinouts (top view)

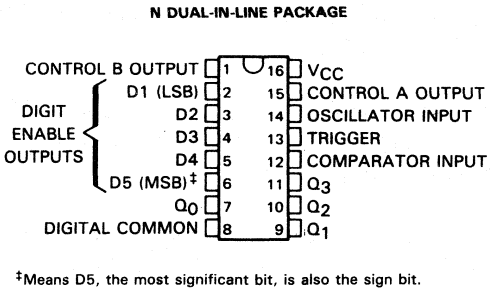


Fig. 11.181 TL503 pinout (top view)

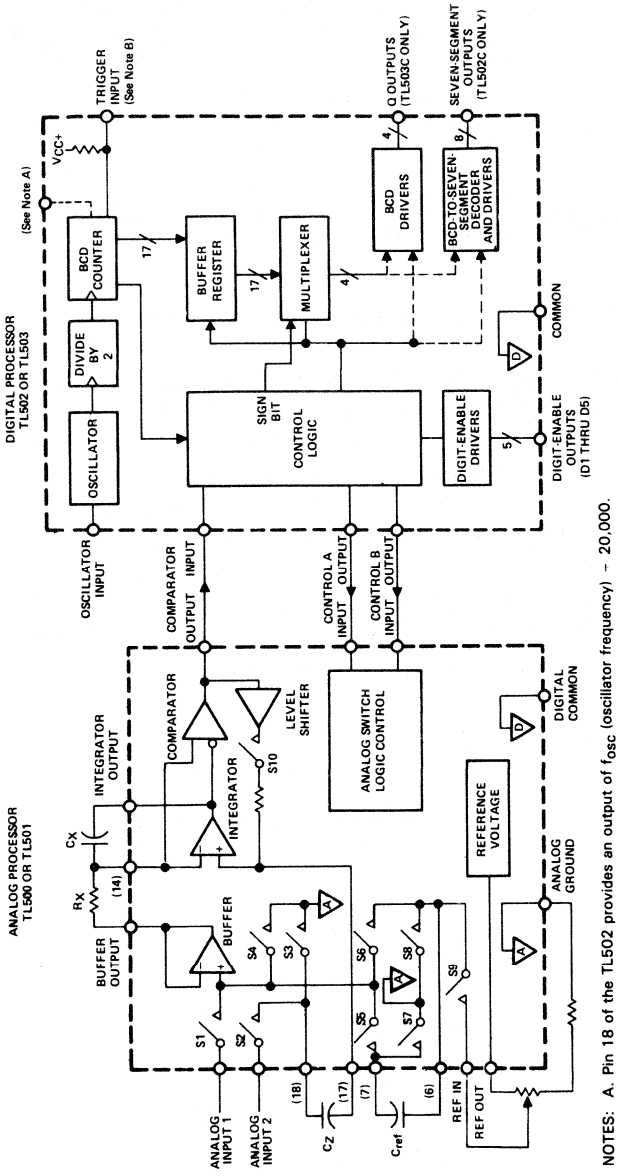
The TL502C and TL503C are digital processors which feature interdigit blanking as well as over-range blanking. They have fast display scan rates and the internal oscillator may be driven or free-running. These devices have 4-1/2 digit display circuitry. Figs 11.180 and 11.181 show the pinout of the TL502C and TL503C, respectively. The TL502C is compatible with popular 7-segment common-anode displays and features a high-sink-current segment driver for large displays. The TL503C features multiplexed BCD outputs with high sink current capability.

General Overall Description

The TL500C and TL501C contain the necessary analog switches and decoding circuits, reference voltage generator, buffer, integrator, and comparator. These devices may be controlled by the TL502C, TL503C, by discrete logic, or by a software routine in a microprocessor.

The TL502C and TL503C both include oscillator, counter, control logic, and digit enable circuits. The TL502C provides multiplexed outputs for 7-segment displays, while the TL503C has multiplexed BCD outputs.

When used in complementary fashion, these devices form a system that features automatic zero-offset compensation, true differential inputs, high input impedance, and capability for 4-1/2 digit accuracy. Applications include the conversion of analog data from high-impedance sensors of pressure, temperature, light, moisture, and position. Analog-to-digital-logic conversion provides display and control signals for weight scales, industrial controllers, thermometers, light-level indicators, and many other applications. See Fig. 11.182 and Table 11.21.



NOTES: A. Pin 18 of the TL502 provides an output of f_{OSC} (oscillator frequency) ÷ 20,000.
B. The trigger input assumes a high level if not externally connected.

Fig. 11.182 Block diagram of basic analog-to-digital converter using TL500C or TL501C and TL502C or TL503C

Table 11.21 TL500C and TL501C functional operation

MODE	ANALOG INPUT	COMPARATOR	CONTROLS A AND B	ANALOG SWITCHES CLOSED
Auto Zero	X	Oscillation	L L	S3, S4, S7, S9, S10
Hold†				
Integrate	Positive	H	H H	S1, S2
Input	Negative	L		
Integrate	X	L‡	L H	S3, S6, S7
Reference		H‡	H L	S3, S5, S8

H ≡ High, L ≡ low, X ≡ Irrelevant

†If the trigger input is low at the beginning of the auto-zero cycle, the system will enter the hold mode. A high level (or open circuit) will signal the digital processor to continue or resume normal operation.
‡This is the state of the comparator output as determined by the polarity of the analog input during the integrate input phase.

Description of TL500C and TL501C Analog Processors

The TL500C and TL501C analog processors are designed to compensate automatically for internal zero offsets, integrate a differential voltage at the analog inputs, integrate a voltage at the reference input in the opposite direction, and provide an indication of zero-voltage crossing. The external control mechanism may be a microcomputer and software routine, discrete logic, or a TL502C or TL503C digital processor. The TL500C and TL501C are designed primarily for simple, cost-effective, dual-slope analog-to-digital converters. Both devices feature true differential analog inputs, high input impedance, and an internal reference-voltage source. See Fig. 11.183 for the input schematic and Fig. 11.184 for the output schematic.

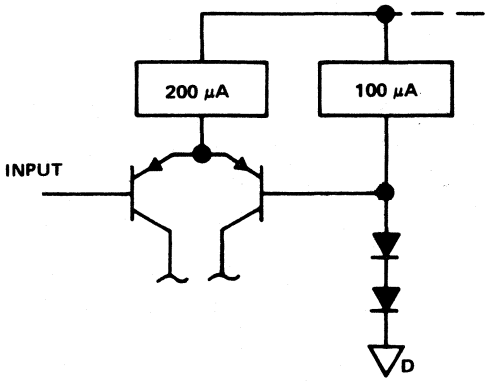


Fig. 11.183 TL500C and TL501C input schematic

The TL500C provides 4-½ digit readout accuracy when used with a precision external reference voltage. The TL501C provides 100-ppm linearity error and 3-½ digit accuracy capability. These devices are manufactured using TI's advanced technology to produce JFET, MOSFET, and bipolar devices on the same chip. The TL500C and TL501C are intended for operation over the temperature range of 0°C to 70°C.

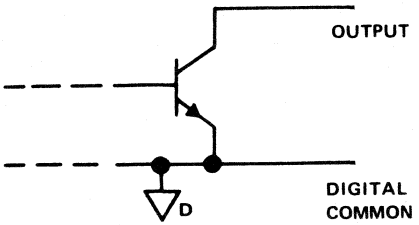


Fig. 11.184 TL500C and TL501C output schematic

Description of TL502C and TL503C Digital Processors

The TL502C and TL503C are control logic devices designed to complement the TL500C and TL501C analog processors. They feature an internal oscillator, interdigit blanking, and fast display scan rate. The internal-oscillator input is a Schmitt trigger circuit that can be driven by an external clock pulse or provide its own time base with the addition of a capacitor. The typical oscillator frequency is 120 kHz with a 470 pF capacitor connected between the oscillator input and ground.

The TL502C provides 7-segment-display output drivers capable of sinking 100 mA and compatible with popular common-anode displays. The TL503C has four BCD output drivers capable of 100 mA sink current. The code for each digit is multiplexed to output drivers in phase with a pulse on the appropriate digit-enable line at a digit rate equal to f_{osc} divided by 200 (Table 11.22 and Fig. 11.185). Each digit-enable output is capable of sinking 20 mA. Fig. 11.186 shows input/output schematics of the TL502C and TL503C. Table 11.23 is the Table of Special Functions.

Table 11.22

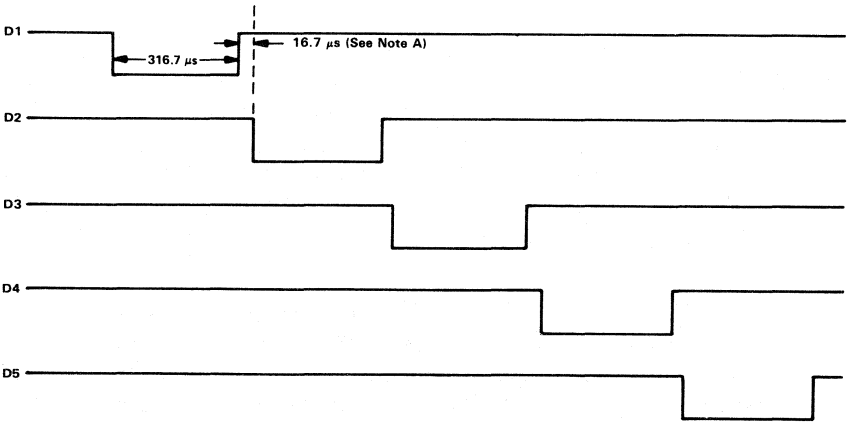
(a) DIGITS 1 THRU 4 NUMERIC CODE

NUMBER	TL502C SEVEN-SEGMENT LINES							TL503C BCD OUTPUT LINES			
	A	B	C	D	E	F	G	Q3	Q2	Q1	Q0
								8	4	2	1
0	L	L	L	L	L	L	H	L	L	L	L
1	H	L	L	H	H	H	H	L	L	L	H
2	L	L	H	L	L	H	L	L	L	H	L
3	L	L	L	L	H	H	L	L	L	H	H
4	H	L	L	H	H	L	L	L	H	L	L
5	L	H	L	L	H	L	L	L	H	L	H
6	L	H	L	L	L	L	L	L	H	H	L
7	L	L	L	H	H	H	H	L	H	H	H
8	L	L	L	L	L	L	L	H	L	L	L
9	L	L	L	L	H	L	L	H	L	L	H

H = high level, L = low level

(b) DIGIT 5 (MOST SIGNIFICANT DIGIT) CHARACTER CODES

CHARACTER	TL502C SEVEN-SEGMENT LINES							TL503C BCD OUTPUT LINES			
	A	B	C	D	E	F	G	Q3	Q2	Q1	Q0
								8	4	2	1
+	H	H	H	H	L	L	L	H	L	H	L
+1	H	L	L	H	L	L	L	H	H	H	L
–	L	H	H	L	H	H	L	H	L	H	H
–1	L	L	L	L	H	H	L	H	H	H	H



NOTE A: The BCD or seven-segment driver outputs are present for a particular digit slightly before the falling edge of that digit enable.

Fig. 11.185 TL502 anmd TL503 digit timing with 120 kHz clock signal at oscillator input

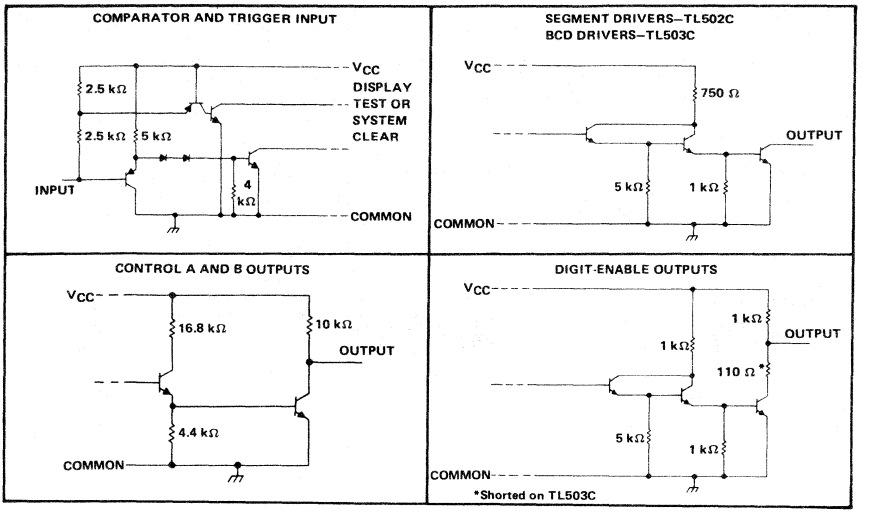


Fig. 11.186 TL502C and TL503C inputs and outputs schematics

*Table 11.23 Table of special functions**

TRIGGER INPUT	COMPARATOR INPUT	FUNCTION
$V_I \leq 0.8$ V	$V_I \leq 6.5$ V	Hold at auto-zero cycle after completion of conversion
2 V $\leq V_I \leq$ 6.5 V	$V_I \leq 6.5$ V	Normal operation (continuous conversion)
$V_I \leq 6.5$ V	$V_I \geq 7.9$ V	Display Test: All BCD outputs high
$V_I \geq 7.9$ V	$V_I \leq 6.5$ V	Internal Test
Both inputs go to $V_I \geq 7.9$ V simultaneously		System Clear: Sets CBC counter to 20,000. When normal operation is resumed, cycle begins with Auto Zero.

*V_{CC} = 5 V ± 10%

The comparator input of each device, in addition to monitoring the output of the zero-crossing detector in the analog processor, may be used in the display test mode to check for wiring and display faults. A high logic level (2 to 6.5 V) at the trigger input with the comparator input at or below 6.5 V starts the integrate-input phase. Voltage levels equal to or greater than 7.9 V on both the trigger and comparator inputs clear the system and set the BCD counter to 20,000. When normal operation resumes, the conversion cycle is restarted at the zero phase. These devices are manufactured using I^2L and bipolar technologies.

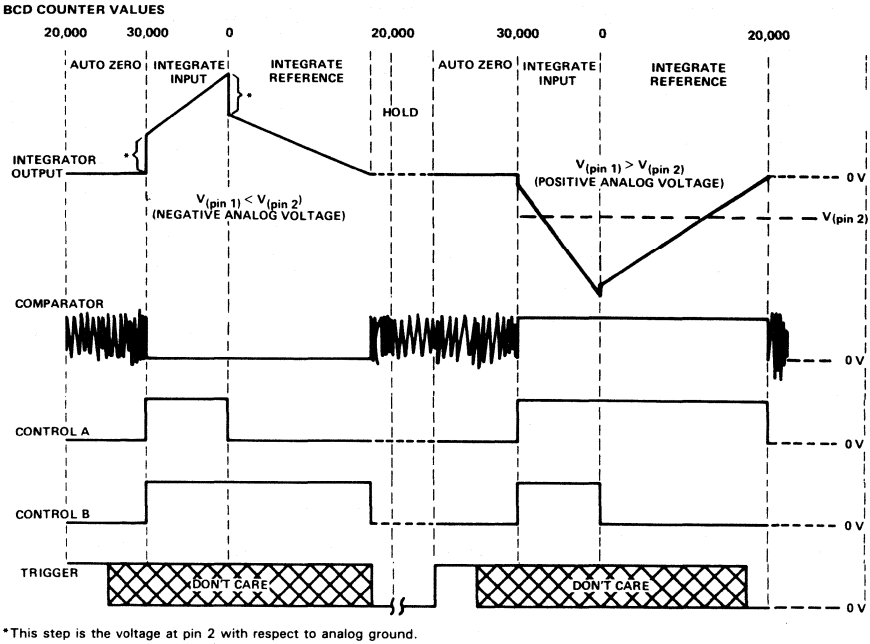


Fig. 11.187 Voltage waveforms and timing diagram

Principles of Operation

The basic principle of dual-slope-integrating converters is relatively simple (refer to Fig. 11.31) page 11−32. The relationship of the charge and discharge values is shown in Fig. 11.182. Capacitor, C_X, is charged through the integrator from comparator threshold voltage, V_{CT}, for a fixed period of time at a rate detemined by the value of the unknown voltage input. The capacitor is then discharged at a fixed rate determined by the reference voltage back to V_{CT} where the discharge time is measured precisely.

$$V_{CX} = V_{CT} - \frac{V_I \quad t_1}{R_X C_X} \text{ charge} \tag{1}$$

$$V_{CT} = V_{CX} - \frac{V_I}{V_{ref}} = -\frac{t_2}{t_1} \text{ discharge} \tag{2}$$

Combining equations 1 and 2 results in:

$$\frac{V_I}{V_{ref}} = -\frac{t_2}{t_1} \tag{3}$$

where:

V_{CT} = Comparator (offset) threshold voltage
V_{CX} = Voltage change across C_X during t_1 and t_2 (equal in magnitude)
V_I = Average value of input voltage during t_1
t_1 = Time period over which unknown voltage is integrated
t_2 = Unknown time period over which a known reference voltage is integrated

Equation (3) illustrates the major advantages of a dual-slope converter:

1. Accuracy is not dependent on absolute values of t_1 and t_2, but is dependent on their ratios. Long-term clock frequency variations will not affect the accuracy.
2. Offset values, V_{CT}, are not important.

The BCD counter in the digital processor (Fig. 11.182) changes at a rate equal to one-half the oscillator frequency. The BCD counter and the control logic divide each measurement cycle into three phases.

Auto-Zero Phase

The cycle begins at the end of the integrate-reference phase when the digital processor applies low levels to inputs A and B of the analog processor. If the trigger input is at a high level, a free-running condition exists and continuous conversions are made. However, if the trigger input is low, the digital processor stops the counter at 20,000, entering a hold mode. In this mode, the processor samples the trigger input every 400 oscillator pulses until a high level is detected. When this occurs, the counter is started again and is carried to completion at 30,000. The reference voltage is stored in the reference capcitor C_{ref}, comparator offset voltage is stored in the integration capacitor C_X, and the sum of the buffer and integrator offset voltages is stored on auto-zero capacitor C_Z. During the auto-zero phase, the comparator output is characterized by an oscillation (limit cycle) of indeterminate waveform and frequency that is filtered and dc shifted by the level shifter.

Integrate-Input Phase

The auto-zero phase is completed at a BCD count of 30,000, and high levels are applied to both control inputs to initiate the integrate-input phase. The integrator charges C_X for a fixed time of 10,000 BCD counts at a rate determined by the input voltage. Note that during this phase, the analog inputs see only the high impedance of the noninverting operational amplifier input. Therefore, the integrator responds only to the difference between the analog input terminals, providing true differential inputs.

Integrate-Reference Phase

At a BCD count of 39,999 + 1 = 40,000 or 0, the integrate-input phase is terminated and the integrate-reference phase is begun by sampling the comparator output. If the comparator output is low, corresponding to a negative average analog input voltage, the digital processor applies a low and a high to control inputs A and B, respectively, the positive side of the reference voltage stored on C_{ref} is applied to the buffer. If the comparator output is high, corresponding to a positive input, control inputs A and B are made high and low, respectively, and the negative side of the stored reference voltage is applied to the buffer. In either case, the processor automatically selects the proper logic state to cause the integrator to ramp back toward zero at a rate proportional to the reference voltage. The time required to return to zero is measured by the counter in the digital processor. The phase is terminated when either the integrator output crosses zero and the counter contents are transferred to the register, or the BCD counter reaches 20,000 and the overrange indication is activated. When activated, the overrange indication blanks all but the most significant digit and sign.

Seventeen parallel bits (4-½ digits) of information are strobed into the buffer register at the end of the integration phase. Information for each digit is multiplexed out to the BCD outputs (TL503C) or the 7-segment drivers (TL502C) at a rate equal to the oscillator frequency divided by 400.

TL500C Thru' TL503C Dual-Slope-Integrating Analog-to-Digital Converter Application Examples

The TL500 and TL501 devices are dual-slope A/D converters implemented with bipolar and MOSFET elements. These converters feature:

1. True differential inputs
2. Automatic zero

3. Automatic polarity
4. High input impedance ($10^9\ \Omega$ typically).

The major differences between the two devices are given in Table 11.24.

Table 11.24 Major differences between the TL500 and the TL501

	TL500	TL501
Resolution	14 bits (with TL502)	10-13 bits (with TL502)
Linearity Error	0.001%	0.01%
Readout Accuracy	4-1/2 Digits (with external ref.)	3-1/2 Digits

The TL502 and TL503 devices are digital processors, or control circuits, that contain an oscillator circuit and output devices. These are monolithic circuits using I^2L and bipolar techniques. These digital processing devices feature:

1. Fast display scan rates
2. Internal oscillator (free-running or driven)
3. Interdigit blanking
4. Overrange blanking
5. 4-½ digit display circuitry
6. High sink-current digit drivers.

The major difference between the two digital processors are given in Table 11.25.

Table 11.25 Major differences between the TL502 and the TL503

TL502	TL503
Compatible with Seven-Segment Common-Anode Displays	Multiplexed BCD Outputs
High Sink-Current Segment Driver for Large Displays	High Sink-Current BCD Outputs

The dual-slope technique used on the TL500 and TL501 devices results in a high degree of noise rejection due to input voltage integration.

External Components Selection Guide

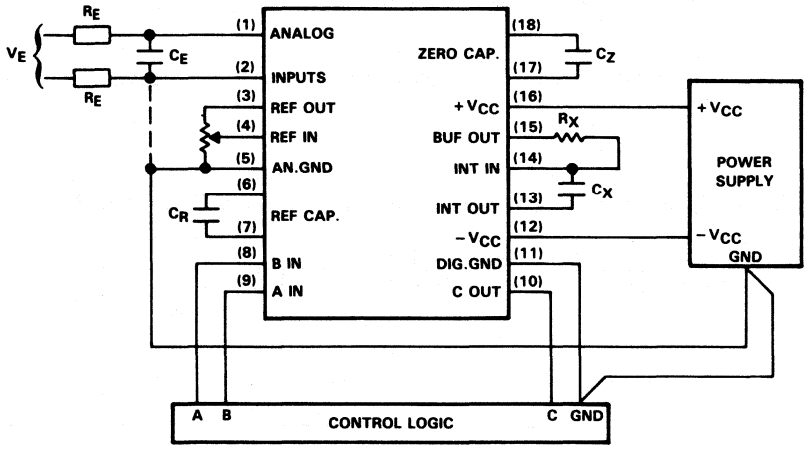


Fig. 11.188 TL500/501 wiring diagram

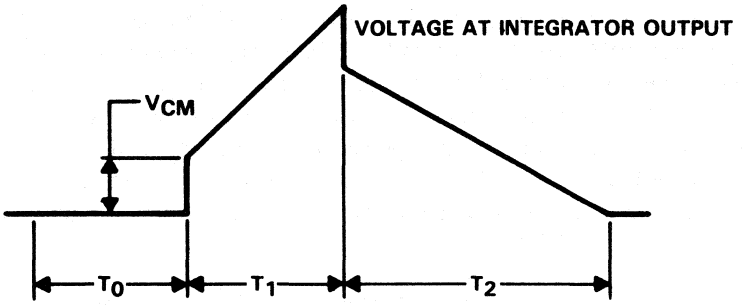
The proper selection of external components is required for the accurate and reliable operation of an A/D converter. These components are shown in Fig. 11.188. Capacitors C_Z and C_{ref} should be types with very low leakage; plastic foil capacitors are excellent. The capacitors should be at least 200 nF. A higher value should be used in 4-½ digit applications and circuits with low conversion speeds (one conversion per second). Capacitance of 1 μF is usually a suitable choice.

Ceramic or aluminium electrolytic capacitors should not be used because of their high leakage. The integrator capacitor C_X should also be within the recommended range and must have good voltage linearity and low dielectric absorption. A polypropylene-dielectric capacitor similar to TRW's X363UW is recommended for 4-½ digit accuracy. For 3-½ digit applications, polyester, polycarbonate, and other film dielectrics are usually suitable.

The value of resistor R_X should be in the range of 15 kΩ to 100 kΩ; a carbon film resistor would be suitable. The minimum RC time constant for the integrator can be calculated using the following equation:

$$R_X \times C_X = \frac{V_{EMAX} \times t_1}{(V_{OMAX}) - V_{CM}}$$

In the above equation, V_{EMAX} represents the maximum input voltage between pins 1 and 2. When using the TL502 or TL503 device, this value will be 200 mV. V_{OMAX} is the maximum output voltage of the integrator at pin 13. With a supply voltage of ±12 V, this should be +8 V or −5 V. V_{CM} is the common-mode voltage between pin 2 and the analog ground as indicated in Fig. 11.189.



V_{CM} = VOLTAGE BETWEEN ANALOG GROUND (PIN 5) AND ANALOG INPUT (PIN 2).

Fig. 11.189 Example with constant common-mode voltage

If a differential input is not required, connecting pin 2 to analog ground will set V_{CM} to zero. The duration of the V_E integration phase is equal to t_1. When using the TL502 or TL503 device, t_1 can be calculated from the oscillation frequency:

$$t_1 = \frac{20{,}000}{f_{osc}}$$

Example: A value of 470 pF for C_{osc} results in a frequency of approximately 160 kHz. Thus, the period of t_1 is 125 ms. (The oscillator circuit is part of the logic control circuit in TL502 or TL503, and is discussed later.)

Resistor R_E and capacitor C_E are not strictly necessary to the operation of the converter as resistor R_E constitutes a protection circuit for the analog inputs. If the input voltages cannot be guaranteed to remain below the value of the supply voltage V_{CC}, then R_E should be included to limit the current which will flow through the TL500 and TL501 internal clamp diodes. The value of R_E should not exceed 100 kΩ.

Each analog input is connected through an analog switch to the high-impedance buffer or integrator inputs. These switches are MOSFET devices having fixed drain-gate capacitance. If the inputs to the switches are also high impedance, then significant feed-through or interaction may occur between the switch control signals and the input voltages. Eventual converter errors can be avoided if the input impedance is decreased using capacitor C_E with a capacitance of 10 to 100 nF.

The reference voltage at pin 4, which is 100 mV for an input voltage range of 20 mV, may be derived from the internal reference voltage (REF OUT) through a voltage divider. The total resistance of the divider should be within the range of 1 kΩ to 10 kΩ. For applications that require high precision, a temperature-compensated reference voltage source is recommended. If the value of R_X is within the recommended range of 15 kΩ to 100 kΩ, the output impedance of the reference voltage source should not exceed 2 kΩ.

Supply voltages for the TL500 and TL501 devices may be selected within the range of 5 V to 18 V and −8 V to −18 V. Operation at voltages approaching the lower limits is not recommeded and should only be considered for applications which do not require high resolution.

Both the common-mode range and the maximum output voltage of the integrator are approximately 3 V. In the case of 4-½ digit applications, a power supply of ±12 V minimum is recommended because of the eventual increase in the dynamic range of the integrator.

The analog and digital ground connections of the TL500 and TL501 devices are internally isolated from each other. The digital ground connection serves as a reference point for the A and B control inputs and the comparator output. The operation of the TL500 and TL501 devices is guaranteed if the potential of the digital ground is within the range of $-V_{CC}$ to $+V_{CC}$ minus 4 V.

Printed Circuit Board Layout Notes

When constructing an A/D circuit on a printed circuit board, the layout has a significant effect on accuracy. Coupling of the digital control signals to the analog components can often be the cause of unexplainable errors. Stray coupling from the comparator output to any analog pin (in order of importance 17, 18, 14, 6, 13, 1, 2, 15) must be minimized to avoid oscillations.

It is essential that the TL500 and TL501 supply voltages be adequately decoupled. Tantalum capacitors are recommended and should be located as close as possible to the device. The analog and digital grounds should be connected together at the power supply as illustrated in Fig. 11.188. This ensures that no current from the digital circuitry can flow along the analog ground connections. The ground connection of the internal reference voltage-divider network or an external reference voltage should be connected directly to the analog ground pin of the TL500 and TL501 devices. The connections to C_R, C_Z, C_X, and R_X should be kept as short as possible and isolated from any digital control wiring.

TL502 and TL503 Control Circuits

The TL502 and TL503 contain not only the necessary logic to control the TL500 and TL501 A/D converters, but also have outputs for driving a 4-½ digit display in either the 7-segment or BCD code respectively. The TL502 and TL503 devices are also suitable for controlling the TL505 device.

Fig. 11.182, page 11−195, shows the block diagram of the TL502 or TL503 device connected to a TL500 or TL501 A/D converter. The oscillator in the TL502 and TL503 can operate either as a free-running oscillator with an external capacitor connected between the oscillator input and digital ground or may be driven by an external TTL signal. The oscillation frequency when operating in the free-running mode is approximately 160 kHz with a 470 pF external capacitor. The TL502 and TL503 devices require 80,000 cycles from the oscillator for a complete conversion. If a 470 pF capacitor is used, the result is a conversion speed of about 2 measurements per second.

For 4-½ digit applications, an external oscillator is recommended. This oscillator should not have any short-term frequency variations, although long-term variations do not have a direct effect on the conversion result. To obtain a stable display, it is necessary to reject the power line frequency. It is desirable that the oscillation frequency be an even multiple of the power line frequency. The external oscillator should be synchronized.

The oscillation frequency range is approximately 10 kHz to 1 MHz. Below 10 kHz the display begins to flicker, and because of the long conversion time, this low frequency is not recommended. Above 1 MHz, reliable operation of the TL502 and TL503 devices is no longer guaranteed for all operating conditions. Higher conversion speeds can be achieved with the A/D converters if a fast TTL or microprocessor control is used.

Logic Inputs and Outputs

The TL502 and TL503 devices have two inputs, trigger and comparator, that can control a total of five different functions.

The conversion may be interrupted using the trigger input; a logic low (0) at the trigger input will cause the converter to stop at the beginning of the next Auto Zero phase. Under these conditions, logic outputs A and B are reset low and the previous conversion result is displayed. However, the internal operation of the device continues, and if the trigger input receives another high level, a new conversion begins with the next Auto Zero phase.

If the trigger input rises more than approximately 2.5 V above V_{CC}, the device enters a test mode in which the operation of the multiplexer is inhibited and all the segments or BCD outputs are activated. Certain decades of the main counter are bypassed relative to the digit output under control to accelerate the operation of the controller.

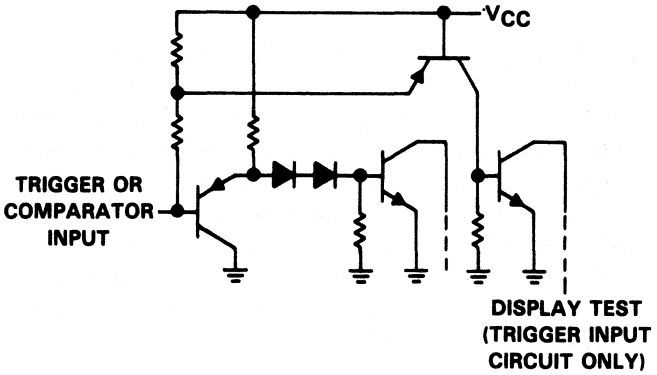


Fig. 11.190 Typical of both comparator and trigger input circuits

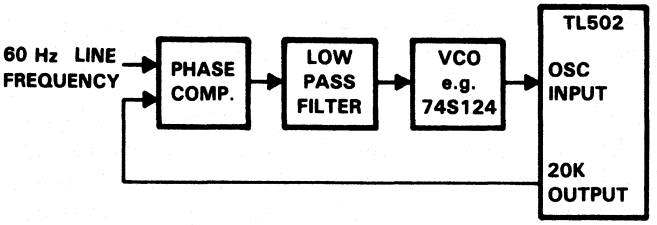


Fig. 11.191 Synchronization of the oscillator frequency

The comparator input receives information from the A/D converter regarding the polarity of the unknown voltage. This information determines whether the A or B output is set low in the following phase and whether a positive or negative reference voltage is integrated. The control logic recognizes the end of the integrating phase when the switching threshold of the comparator is crossed. The value from the main counter is then transferred to a register, and the A and B logic outputs are set low. Fig. 11.190 shows a circuit of both comparator and trigger inputs.

If the comparator input rises more than approximately 2.5 V above V_{CC}, a display test begins but the control logic continues to operate. However, normal conversion is not guaranteed in this case because comparator information cannot be communicated.

If signals are applied simultaneously to both the trigger and comparator inputs, the logic is reset. When this mode ends, control begins again with the start of the next Auto Zero phase; until then, the outputs display 19999.

The TL502 device has an oscillator output that is labelled 20,000. The frequency of this output is the oscillator frequency divided by a factor of 20,000 and has a mark-space ratio of 1:4. This output is suitable for controlling a phase locked loop (PLL) circuit so the oscillator frequency is locked to an even multiple of the line frequency. Fig. 11.191 shows the block diagram of a typical PLL circuit. This type of circuit is recommended when a stable display is required at relatively high conversion rates that will remain stable, even when the input voltage has superimposed line frequency hum.

The A and B outputs, as well as the 20,000 oscillator output, have internal pull-up resistors; therefore, they are TTL compatible and the A and B outputs can be connected directly to the relevant inputs of the TL500 and TL501 devices.

Driving a Seven Segment Display Using the TL502 Device

The TL502 device can drive up to a 4-½ digit display in a multiplexed mode. This has the advantage of minimizing the device outputs required and also simplifies the PC board layout.

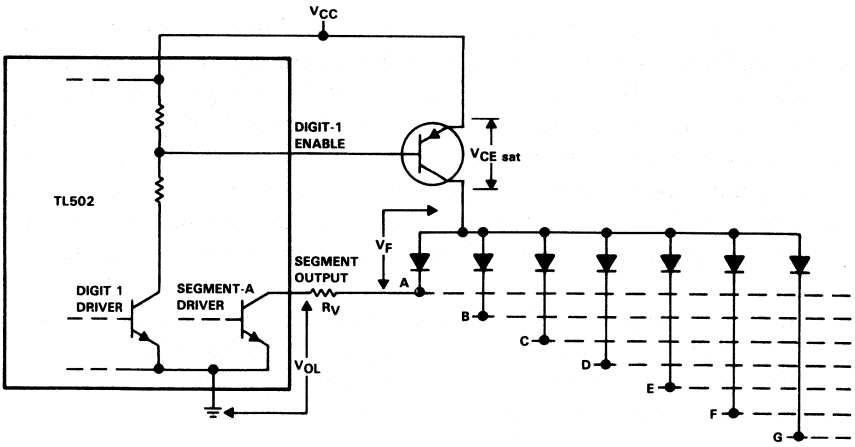


Fig. 11.192 Display driving

In most simple applications, the display will consist of 7-segment, common-anode LEDs. Because the I^2L technology does not permit the manufacture of suitable PNP drive transistors, it is necessary to use external drive transistors to drive each digit anode. The display cathodes are connected by a current-limiting resistance R_V to the relevant segment output shown in Fig. 11.192.

Calculation of the Current-Limiting Resistor

The segment current and the brightness of the display are set by resistor R_V. From Fig. 11.192 the voltage drop across R_V is:

$$\begin{aligned} V_{RV} &= V_{CC} - V_F - V_{CEsat} - V_{OL} \\ &= 5\ V - 1.7\ V - 2 \times 0.2\ V \\ &= 2.9\ V \end{aligned}$$

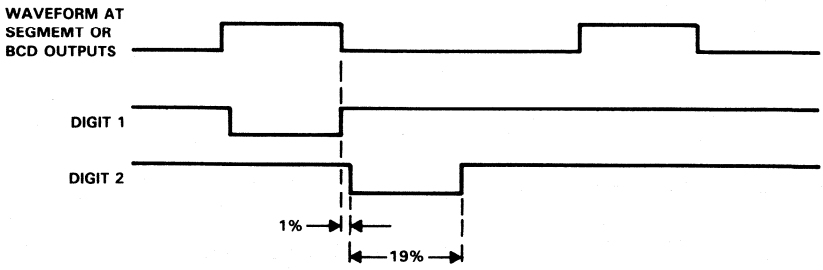


Fig. 11.193 Multiplexer timing diagram

As indicated in the segment current waveform in Fig. 11.193, each digit is active for 19% of the time. If the display requires an average segment current of 15 mA to achieve a specific brightness, the peak current (I_P) can be calculated by the equation:

$$I_P = \frac{15\ mA}{19\%}$$
$$= \text{approx. } 80\ mA$$

and R_V can be calculated by:

$$R_V = \frac{V_{RV}}{I_P}$$
$$= \frac{2.9\ V}{80\ mA}$$
$$= \text{approx. } 36\ \Omega$$

In practice, a value of 39 Ω would be chosen because it is the next higher standard value. A power rating of ¼ W is adequate.

Application of the TL503 BCD Outputs

The outputs of the TL503 device correspond to the multiplexer timing diagram in Fig. 11.193. The clock time of the multiplexer is 200 x ($1/f_{osc}$). The digital outputs are active low. The BCD outputs are the open-collector type and are active high.

If the conversion results are recorded by a microprocessor, the following procedure may be used. The trigger input must be low prior to conversion. When the controlling program requires a new conversion, a logic high signal is directed to the trigger input for a period of not less than 2000 x ($1/f_{osc}$) and the A/D converter immediately starts a conversion. When period t satisfies the condition shown below

$$2000 \times (1/f_{osc}) < t < 40{,}000 \times (1/f_{osc}),$$

the microcomputer awaits an end of conversion signal. This signal is processed from the A and B control signals by a NOR gate. After this signal has been received, the new result may be used.

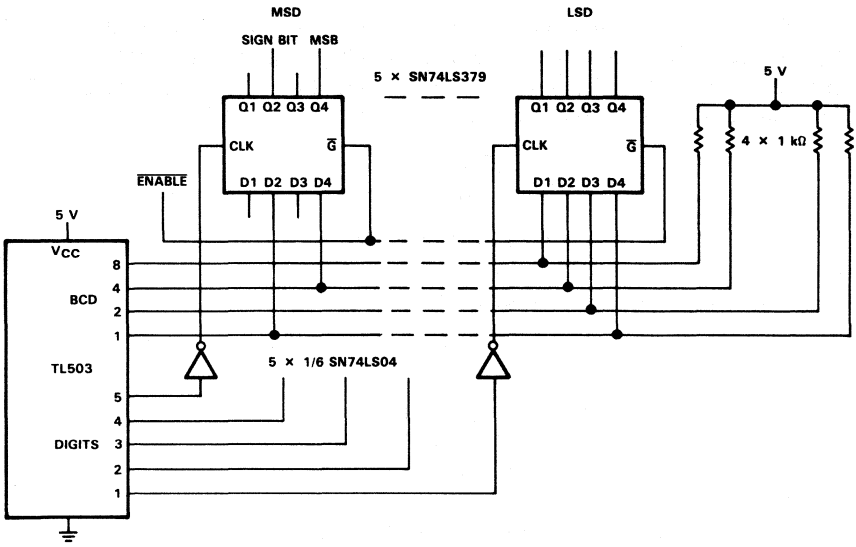


Fig. 11.194 Demultiplexer for TL503

The actual data transmission is synchronized using the digital outputs with each digit triggering a different interrupt. This guarantees the integrity of the BCD information. The minimum time required for a complete transfer, therefore, corresponds to the duration of the multiplexer period. This time can be decreased if two or more bits of digital information are combined.

It is possible to combine all 18 bits using the demultiplexer shown in Fig. 11.194. The BCD information from each digit is transmitted to the quadruple D-type flip-flop using the negative edge of the digital outputs. Fig. 11.193 indicates that this negative edge is available after a time delay of two clock cycles. If required, the inputs to the flip-flops may be inhibited during the read process by using the enable inputs.

Digital Panel Meter Using the TL501 and TL502 Devices

The input voltage range of the digital panel meter shown in Fig. 11.195 is ±2.0 V and when using the component values shown, a conversion rate of approximately 2 conversions per second is produced. Other conversion rates can be achieved by using different size capacitors for C_O and C_X. Pins 2 and 5 on the TL501 device may be connected together when no common-mode input voltage exists. The equation for selecting values for

R_X and C_X is given in the External Components Selection Guide section. When necessary, an RC time constant protection network may be connected to the input.

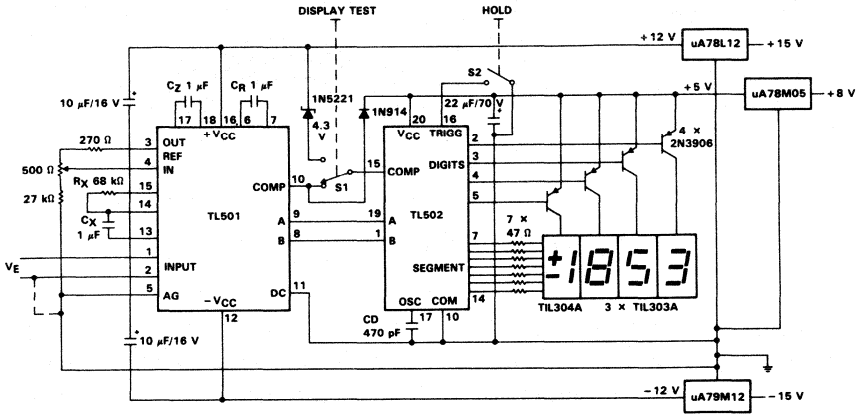


Fig. 11.195 Digital panel meter using TL501

Note that the recommended use of separate grounds is shown in Fig. 11.195. It is essential that the analog and digital grounds be connected only at the power supply.

Switch S1 provides a means of activating the display test mode. The required 7.5 V supply is provided from the 12 V line using a 1N5221 zener diode to provide a 4.3 V drop. A changeover switch is required because the comparator output of the TL501 device is not short circuit protected. Remember that after a display test, the display contains an invalid result.

By operating the HOLD switch, S2, conversion results can be stored or readings can be manually observed. The different pin-out of the TIL304A device (sign and MSB digits) means that pins 1 and 14 must be connected to the digit driver. Pin 7 connects to the limiting resistor of segment E or F and pin 8 connects to the limiting resistor of segment G.

Digital Thermometer

The digital thermometer diagram shown in Fig. 11.196 is an application that uses the differential inputs of the TL500 or TL501 device to measure

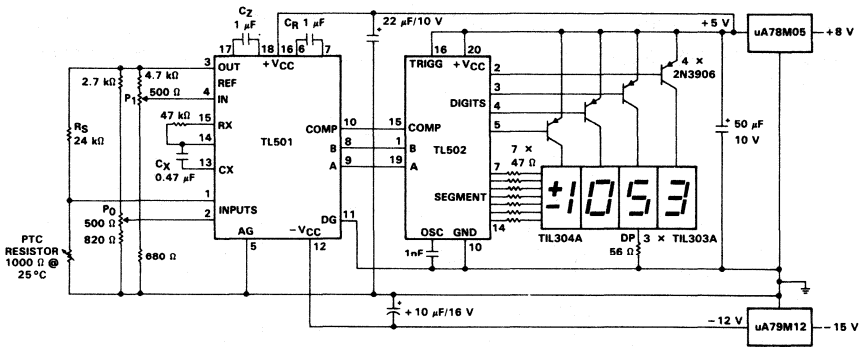


Fig. 11.196 Digital thermometer

a voltage from a bridge circuit. When the circuit is calibrated, the voltage that represents an actual temperature is indicated on the LED readout.

The temperature sensor is a silicon thermistor with a positive temperature coefficient and a low time constant (less than 1.5 seconds). Excluding linearity of the sensor, the linearity error of the circuit is less than 0.5% over the temperature range of −55°C to +125°C.

The calibration of the circuit is straightforward. First, potentiometer P0 is adjusted for the zero point. This is accomplished by adjusting P0 to a preset voltage of $0.258 \times V_{REF}$ V (approximately 0.31 V) at the wiper of P0. Second, the thermometer is calibrated by applying a known temperature to the thermistor and adjusting potentiometer P1 for about $0.153 \times V_{REF}$ V (approximately 0.19 V) at the wiper of P1.

The decimal point is fixed between digits 2 and 3 by connecting the relevant cathode of the display to digital ground through a 56-Ω resistor.

Precision Panel Meter Application

Fig. 11.197 shows a 4-½ digit precision panel meter using the TL500 and TL502 devices. The input voltage range is ±200 mV. This means that current measurements are possible with only an additional shunt resistor.

The use of a precision reference voltage source is recommended so the circuit may be used over a wide temperature range. The 100 mV reference voltage is obtained from a TL431 shunt regulator and a voltage divider comprised of a 250-Ω cermet trimmer potentiometer and a 2500-Ω

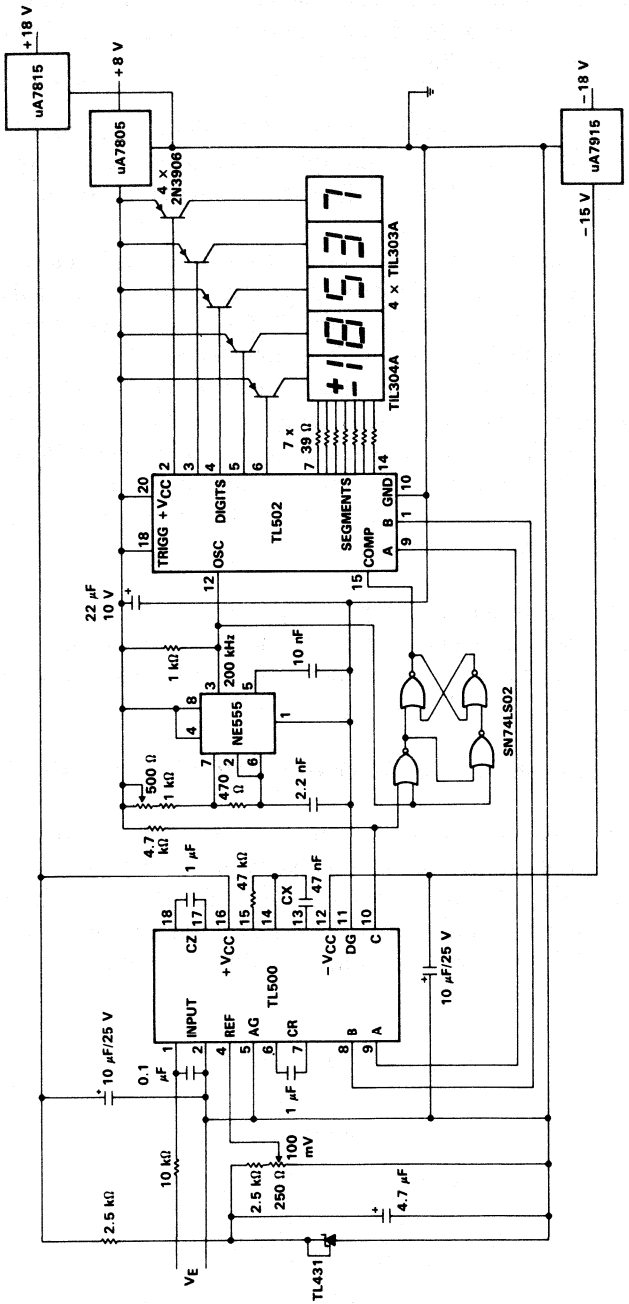


Fig. 11.197 Precision panel meter

metal film resistor. No allowance has been made for a common-mode input voltage. If this is required, the connection from pin 2 to pin 5 should be opened and a larger capacitor inserted.

The TL502 device is controlled by a NE555 external oscillator. Its frequency is adjusted to an even multiple of the power line frequency to guarantee a high degree of power line frequency suppression.

The latch, implemented with an SN74LS02 device, increases the performance of the circuit at input voltages of almost zero. If the input voltage is very low, the integrator of the TL500 device can only produce a small ramp signal. In this case, low-level interfering signals are sufficient to cause an incorrect comparator switching sequence. This result is especially noticeable during the switch-over from the up-integration phase to the down-integration phase. The comparator signal, delayed by one-half of a clock cycle, is directed to the control logic through the latch. Thus, interference impulses caused by crosstalk from the A and B control signals no longer have any influence on the comparator input.

The suggested display is composed of four TIL303A devices and a single TIL304A device. When connecting the TIL304A device, the minus sign should be connected to segment outputs A and D. All other connections are identical to the TIL303A device.

TL507 SINGLE–SLOPE ANALOG-TO-DIGITAL CONVERTER

The TL507 is an economical single-slope, 7-bit resolution A/D converter. It features guaranteed monotonicity and ratiometric conversion. Operating from a single supply, it consumes only 25 mW of power at a supply voltage of 5 V and has a conversion speed of approximately 1 ms. See Fig. 11.198 for the pin assignment of this converter.

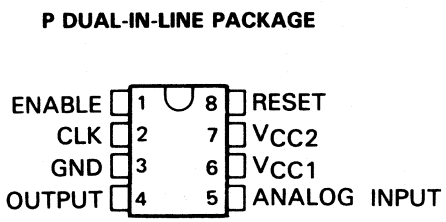


Fig. 11.198 TL507 pinout (top view)

Description

The TL507 is a single-slope A/D converter designed to convert analog input voltages between 0.25 V_{CC} and 0.75 V_{CC} into a pulse width modulated output code. It contains a 7-bit synchronous counter, a binary weighted resistor ladder network, and operational amplifier, two comparators, a buffer amplifier, an internal regulator, and necessary logic circuitry. Integrated injection logic (I^2L) technology makes it possible to offer this complex circuit at low cost in a small dual-in-line 8-pin package.

In continuous operation, it is possible to obtain conversion speeds up to 1000 per second. The TL507 requires external signals for clock, reset, and enable. Versatility and simplicity of operation coupled with low cost, make this converter especially useful for a wide variety of applications.

Definition of Terms

Zero Error—The intercept (b) of the A/D converter system function $y = mx + b$, where y is the output of the ladder network, x is the analog input, and m is the slope of the transfer function.

Overall Error—The magnitude of the deviation from a straight line between the endpoints of the transfer function.

Differential Nonlinearity—Maximum deviation of an analog value change that is associated with a 1 bit code change (1 clock pulse) from its theoretical value of 1 LSB.

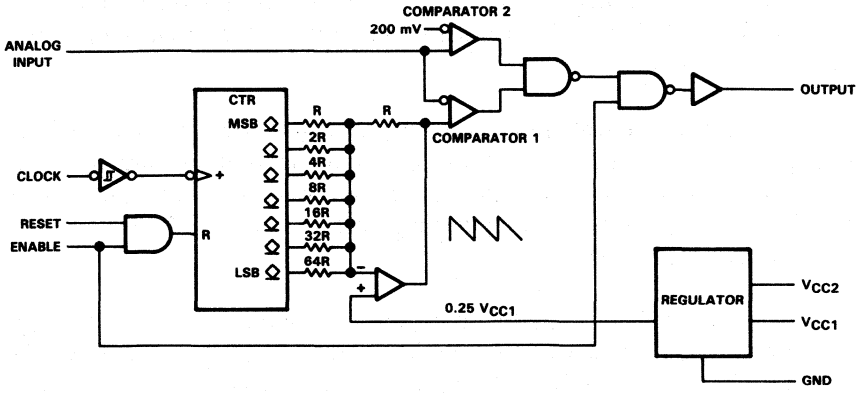


Fig. 11.199 TL507 block diagram

Principles of Operation

The operation of the TL507 device is explained with the block diagram in Fig. 11.199 and the timing diagram in Fig. 11.200.

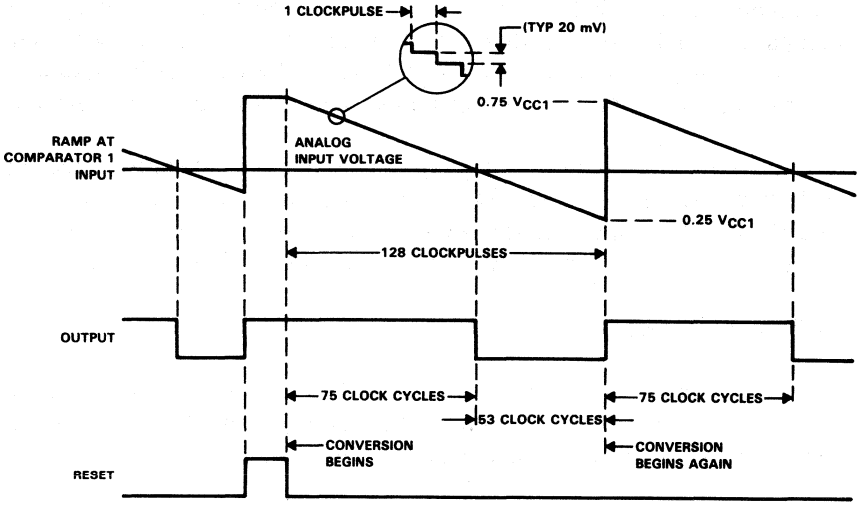


Fig. 11.200 TL507 timing diagram

The internal 7-bit counter drives a binary-weighted resistor network to produce a monotonically negative-going ramp as the circuit is clocked. Comparator 1 compares the analog input voltage with the ramp and produces a low state at the output when the ramp voltage is less than the analog input voltage. This function happens when the enable input is high and the reset input is low.

The reset and enable inputs are inputs to an AND gate and are TTL voltage-level compatible. When the reset and enable inputs are at a high level, the counter is reset and the ramp is at its maximum value ($0.75\ V_{CC1}$). When the reset input pin goes low, the TL507 device starts the A/D conversion cycle.

A conversion can begin by taking the reset input low or by clocking the circuit after the end of the previous conversion cycle as shown in Fig. 11.200. The ramp input to the comparator is always at its highest voltage at the beginning of a conversion and the TL507 device output is

high. The ramp voltage decreases with each succeeding clock cycle and the TL507 device output remains high until the ramp voltage becomes less than the analog input voltage; then, the output goes low. Thus, the number of clock cycles that occur during the period when the TL507 device output is low is directly proportional to the analog input voltage. A convenient method for obtaining a clock count, which is directly proportional to the analog input voltage, can be obtained with the simple NOR gate circuit in Fig. 11.201.

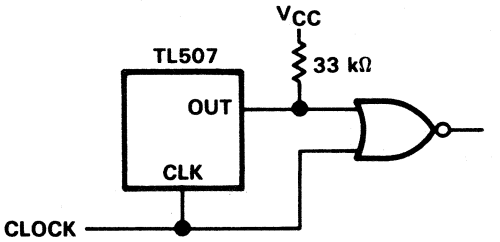


Fig. 11.201 Circuit for obtaining clock count proportional to the analog input voltage

The lower limit of the ramp is set by an internal voltage divider at 0.25 V_{CC1}. This lower limit corresponds to the highest possible count which is $2^7 - 1 = 127$. Since the counter outputs are open-collector transistors that are off at the highest count, no current flows through the binary-weighted resistor network. Thus, the output of the operational amplifier is equal to the voltage (0.25 V_{CC1}) at its positive input. The upper limit of the ramp corresponds to a count of zero when the binary-weighted resistor network is grounded. The output of the operational amplifier is calculated as follows:

$$V_{RAMPmax} = 0.25\ V_{CC1}\ (1 + R[1/R + - - - 1/64R])$$

$$\approx 0.75\ V_{CC1}$$

The counter is decreased by one least significant bit for each clock pulse; thus, the ramp voltage decreases by $V_{CC1}/256$ as illustrated in Fig. 11.200 and as calculated below:

$$\Delta V_{RAMP} = 0.25\ V_{CC1}\ (R)\ (1/64R)$$

$$= \frac{V_{CC1}}{256} \approx 20\ \text{mV, typically}$$

The input voltage , as illustrated in Fig. 11.200, resulted in the device output being low for 53 clock cycles. The value of this input voltage is therefore

$$V_{IN} = \frac{\text{low clock count}}{\text{total clock count}} \times \frac{V_{CC1}}{2}$$

(where $V_{CC1} = 5$ V)

$$= \frac{53}{128} \times 2.5 \text{ V} \approx 1 \text{ V}.$$

TL507 Device Inputs and Outputs

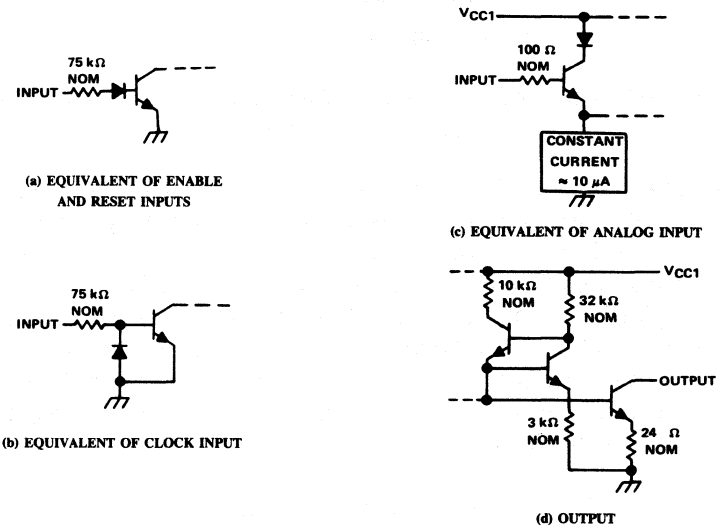


Fig. 11.202 Schematics of inputs and outputs

When the enable input, pin 1, is held low, the TL507 device output is forced high regardless of all other input states. Since the TL507 device output is an open-collector transistor, multiple TL507 outputs can be wire ORed together, and the enable input functions as a multiplexer. When the enable input of one of the wire ORed TL507 goes high, the circuit performs the conversion operation and only its output controls the bus. The enable input is TTL compatible and is active high.

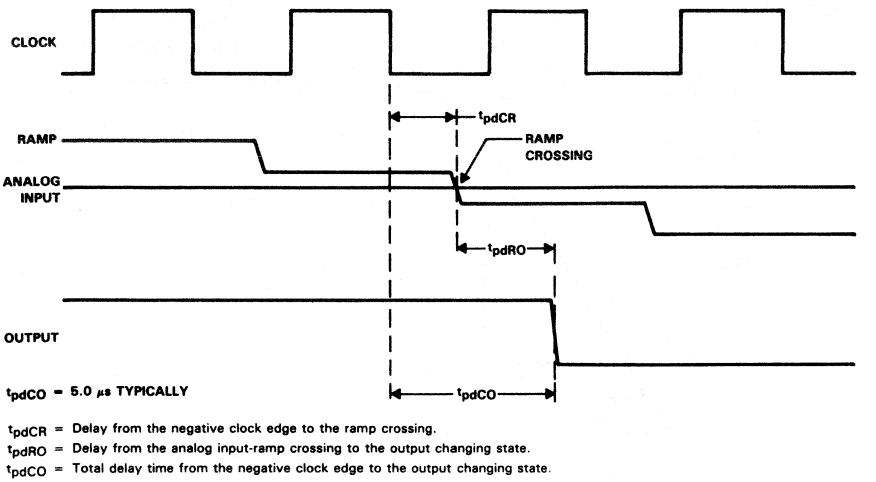


Fig. 11.203 Typical TL507 propagation delay

The clock input, pin 2, is connected to the counter input through a Schmitt-trigger gate that provides a large amount of hysteresis for noise immunity; therefore, the clock input can accept inputs with slow rise and fall times. Because of the hysteresis, the clock input is not compatible with TTL voltage levels and a pull-up resistor of approximately 4.7 kΩ must be used when this input is driven from a TTL output. The counter is clocked on the negative edge of the clock waveform as shown in Fig. 11.203. The propagation delay time t_{pdCO}, as illustrated in Fig. 11.203, is from the clock's negative edge to an output level change and is made up of two parts. One is the shift register delay (t_{pdCR}) and is from the clock input's negative edge to the analog input-ramp voltage crossing. The other (t_{pdRO}) is from the analog input-ramp voltage crossing to the output changing state. Although t_{pdRO} is a fixed value of about 3 μs, the value of t_{pdCR} may vary, depending on the value of the analog input voltage, from 1 μs to about 3 μs. Total delay, $t_{pdCO} = t_{pdCR} + t_{pdRO}$, must be less than one clock period. The recommended operating clock frequency is 125 kHz for an 8 μs time period. The maximum recommended clock frequency of 150 kHz yields a 6.67 μs time period.

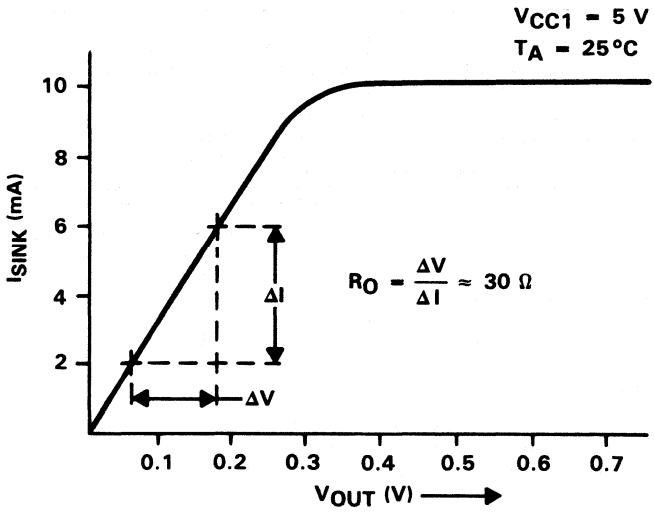


Fig. 11.204 Typical TL507 output characteristic

The TL507 output at pin 4, Fig. 11.202, is an open-collector npn transistor, capable of withstanding up to 18 V in the off state, that must be pulled up with a suitable resistor determined by the load current and voltage requirements. The typical output characteristic of the TL507 device is shown in Fig. 11.204 to aid in selecting the pull-up resistor value. Although the output acts like a current source at loads of about 4 mA, this characteristic should not be used to limit the output current when driving a low impedance load such as an optical coupler input diode. Also, the low state voltage should not exceed 5.5 V to keep from exceeding the power dissipation limits.

The analog input, pin 5, Fig. 11.202, is buffered by an emitter follower prior to being connected to the inputs of the comparators, and the buffer reduces the worst-case analog input current to 300 nA. The circuit or transducer which drives the analog input, must have a low enough input impedance to minimize the effects of the input current or it will create an input offset. Comparator 2 has a nominal offset voltage of 200 mV as indicated in Fig. 11.199. When the analog input voltage is less than 200 mV, the output is forced to a low state to indicate that the analog input voltage is out of range. Since the offset voltage has a ±50% tolerance, it is not suitable for measurement purposes. When the analog input exceeds the upper ramp voltage of 0.75 V_{CC1}, the output goes low again as an indication of overrange.

Since the analog input current of the TL507 device is typically 10 nA, it can be used in a sample-hold configuration as shown in Fig. 11.205.

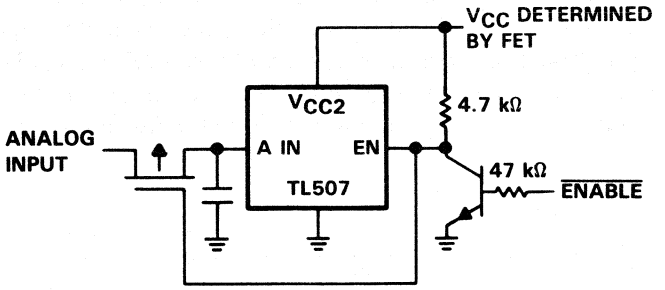


Fig. 11.205 Sample hold circuit

The TL507 device has an internal regulator that converts an unregulated 8 V to 18 V input into a regulated 5 V. The unregulated input, pin 7, is called V_{CC2}. The regulated voltage is available on pin 6 as V_{CC1}. If more current is needed, pin 7 can be left open and a regulated voltage between 3.5 V and 6 V can be applied to pin 6. The typical output characteristic for the regulator is shown in Fig. 11.206. Additional loading on the regulated V_{CC1} should not exceed 5 mA. When the internl regulator is used, both the V_{CC1} and V_{CC2} pins should be decoupled with a 10 μF capacitor to ground.

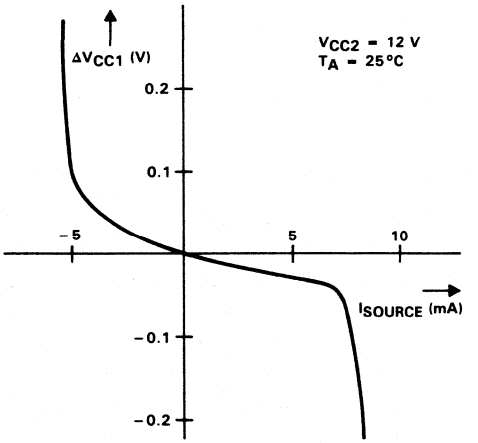


Fig. 11.206 Typical V_{CC1} characteristic

TL507 Single Slope Analog to Digital Converter Applications

Single-Wire, Power, Data, and Clock Cycle Transmitter

In this application, a remote sensor and the TL507 are used to provide a pulse-width-modulating signal representing a sensed analog input level. Connection between the remote location and a central control logic unit is over a single wire and common ground. This wire carries the clock pulses and unregulated power to the remote location and also returns the PWM signal.

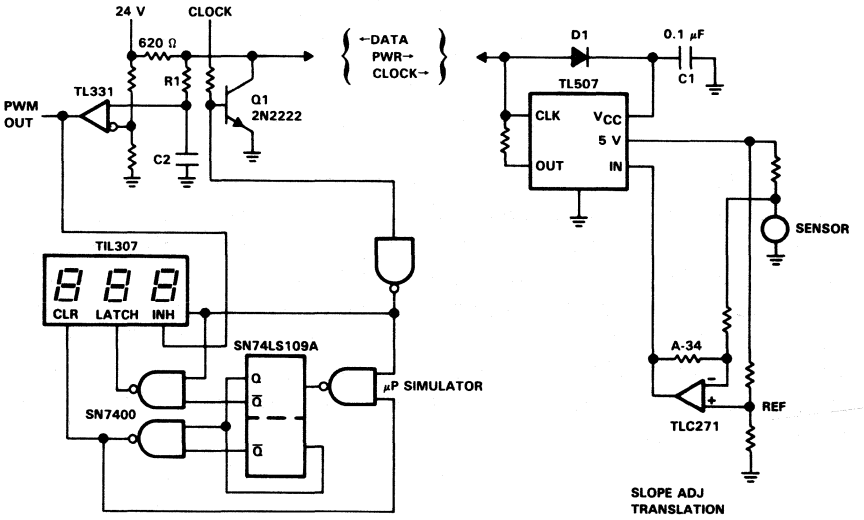


Fig. 11.207 Remote-sensor, single-wire A/D converter

Figs 11.207 and 11.208 show the basic circuitry and resulting waveforms. Q1 is used to chop the supply voltage and provide clock pulses to the remote location. The clock signal is fed directly to the TL507 clock input. It is also connected to diode D1 and capacitor C1 for filtering to provide the dc power for the TL507 A/D converter and the TLC271 signal conditioner. Adjusting the TLC271 gain provides the proper input levels for conversion by the TL507, allowing calibration of the output in degrees Celsius or Fahrenheit. Since the TL507 is continuously clocked, without any reset or enable, it continuously converts the analog input signal as illustrated in Fig. 11.208. The pulse-width-modulated output signal from the TL507 is fed back over the wire and through a filter (R1 and C2) to

remove the clock pulses. A TL331 comparator detects the PWM signal produced by the TL507. The comparator output, along with SN7400 and SN74ALS109A control logic, controls the TIL307 intelligent display.

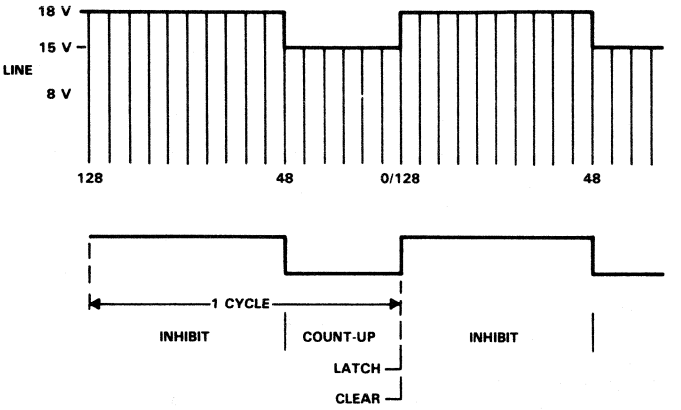


Fig. 11.208 TL507 waveforms

TL507 High Voltage Isolation Application and Signal Coupling

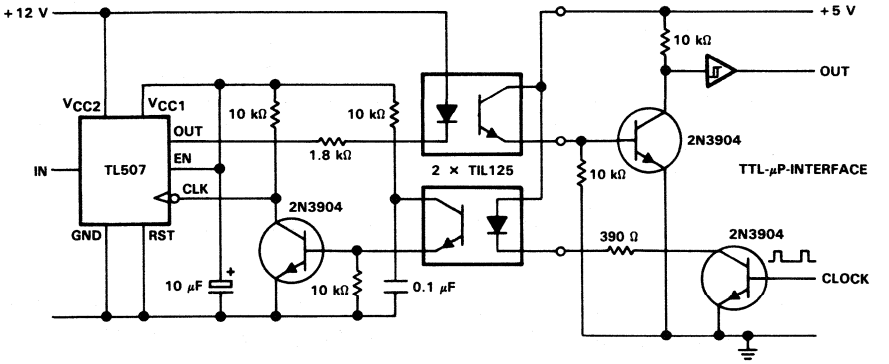


Fig. 11.209 Example of high voltage isolation and signal coupling

The circuit in Fig. 11.209 illustrates the use of an opto-coupler interface between a microprocessor and the TL507 device. In this circuit, the TL507 may be located at a high voltage point while the TL125 devices provide

isolation for the microprocessor. This design is intended to mimimize the opto-coupler switching times as much as possible without sacrificing performance.

TL507 to TMS1000 Interface

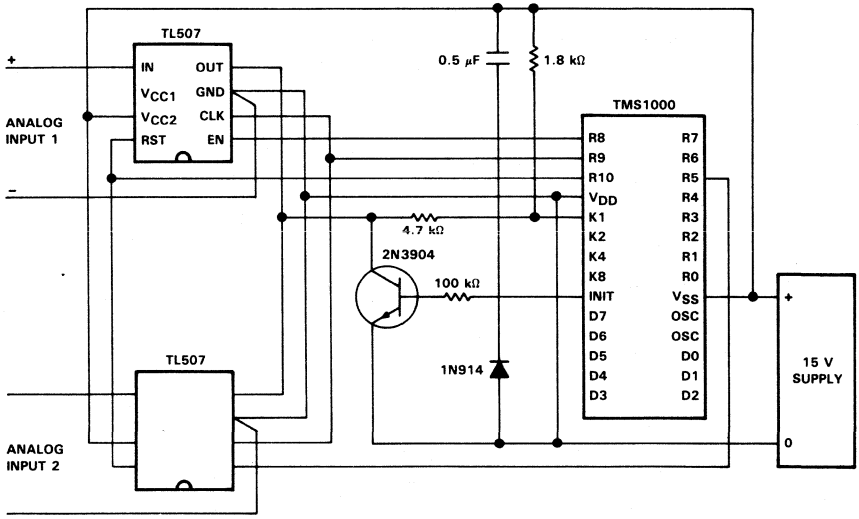


Fig. 11.210 TL507–TMS1000 interface

In some applications, the TL507 device is used with a single-chip microprocessor such as the TMS1000. Fig. 11.210 shows a TL507 device used with the PMOS TMS1000NLP microprocessor. Control of the single TL507 device is accomplished using serial control. Using parallel control would allow direct control of 4 TL507 devices. If necessary, up to 15 TL507s may be controlled if the 4 kΩ inputs of the TMS1000 are expanded using an external TMS1025 integrated circuit. The organization of the inputs and outputs of the TMS1000 microprocessor shown in Fig. 11.210 is given only as an example and may be changed as required.

TL507 to TMS1000C Interface

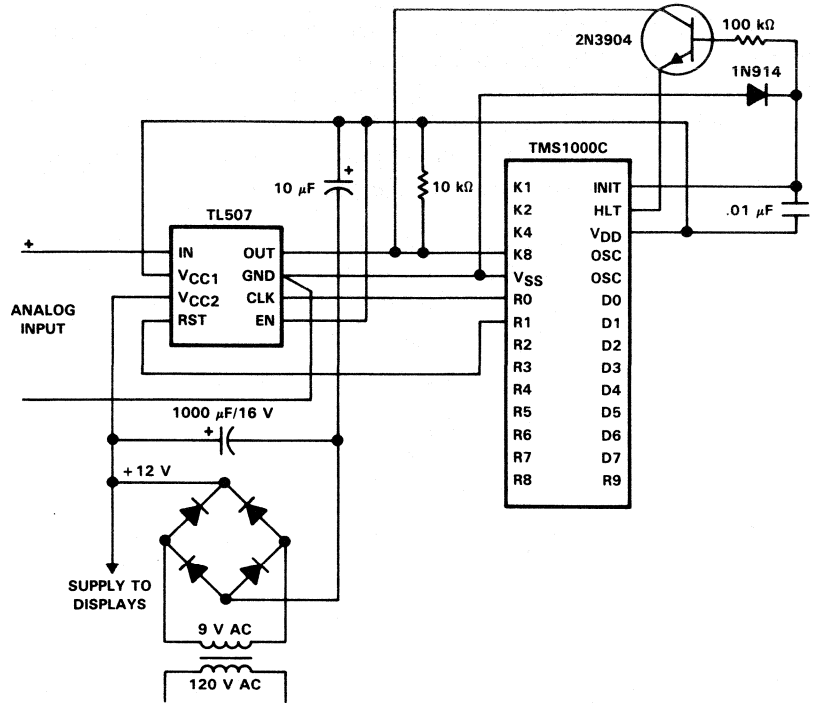


Fig. 11.211 TL507–TMS1000C interface

Fig. 11.211 shows the interface between a TL507 device and a TMS1000C microprocessor. This is the CMOS version of the single-chip TMS1000 microprocessor. This circuit is particularly relevant for applications requiring low power consumption. Since the current requirement of the TMS1000C microprocessor is less than 3 mA, it may be supplied from the V_{CC1} regulator of the TL507 device. Since the total current requirement of the circuit is less than 10 mA, it can be powered from a single 9 V battery. The transistor in parallel with the TL507 output may be omitted if the undefined state of the TL507 output is taken into account during initialization.

ADC0803 AND ADC0805 8-BIT SUCCESSIVE APPROXIMATION A/D CONVERTERS WITH DIFFERENTIAL INPUTS

The ADC0803 family of A/D converters are 8-bit devices which feature a conversion time of 100 μs and an access time of 135 ns. This converter requires no zero adjust, has an on-chip clock generator, and will operate from a single 5-V supply. As shown in Fig. 11.212 the pinout allows for easy PC board layout. The data output pins are grouped together and the ground and V_{CC} pins are located at the package corners.

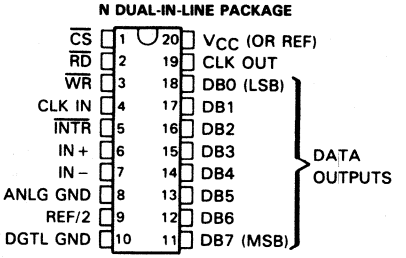


Fig. 11.212 ADC0803 and ADC0805 pinout (top view)

Description

The ADC0803 and ADC0805 are CMOS 8-bit successive-approximation analog-to-digital converters that use a modified potentiometric 256-resistor ladder network. These devices are designed to operate from common microprocessor control buses, with the 3-state output latches driving the data bus. The devices can be made to appear to the microprocessor as a memory location or an I/O port. Fig. 11.213 shows a functional block diagram and the timing diagrams are illustrated in Fig. 11.214.

A differential analog voltage input allows increased common-mode rejection and eliminates offset due to the zero-input analog voltage value. Although a reference input (REF/2) is available to allow 8-bit conversion over smaller analog voltage spans or to make use of an external reference, ratiometric conversion is possible with the REF/2 input open. Without an external reference, the conversion takes place over a span from V_{CC} to analog ground (ANLG GND). The devices can operate with an external clock signal or, the on-chip clock generator can be used independently by adding an external resistor and capacitor to set the time period.

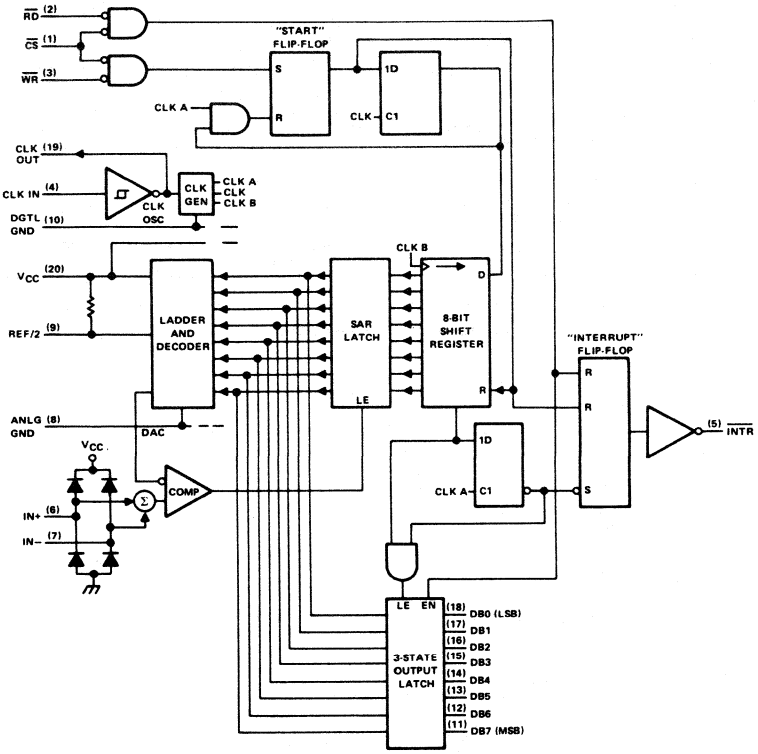


Fig. 11.213 Functional block diagram (positive logic)

Principles of Operation

The ADC0803 and ADC0805 each contain a circuit equivalent to a 256-resistor network. Analog switches are sequenced by successive-approximation logic to match an analog differential input voltage ($V_{I+} - V_{I-}$) to a corresponding tap on the resistor network. The most significant bit (MSB) is tested first. After eight comparisons (64 clock periods), an 8-bit binary code (1111 1111 = full scale) is transferred to an output latch and the interrupt ($\overline{INTR}$) output goes low. The device can be operated in a free-running mode by connecting the $\overline{INTR}$ output to the write ($\overline{WR}$) input and holding the conversion start ($\overline{CS}$) input at a low level. To ensure start-up under all conditions, a low-level $\overline{WR}$ input is required during the power-up cycle. Taking $\overline{CS}$ low any time after that will interrupt a conversion in process.

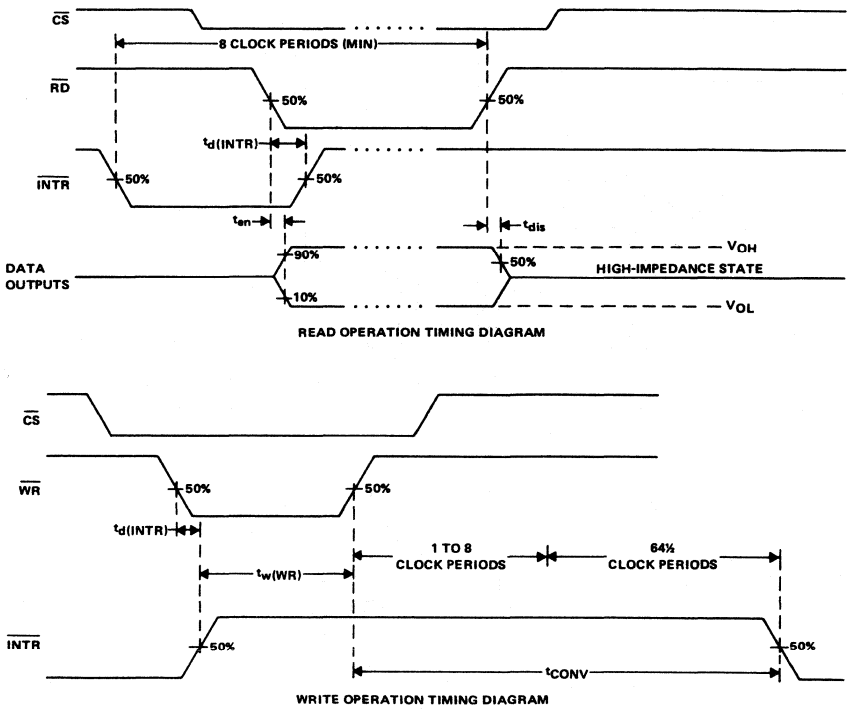


Fig. 11.214 Timing diagrams

When the $\overline{WR}$ input goes low, the internal successive-approximation register (SAR) and 8-bit shift register are reset. As long as both $\overline{CS}$ and $\overline{WR}$ remain low, the analog-to-digital converter will remain in its reset state. One to eight clock periods after $\overline{CS}$ or $\overline{WR}$ makes a low-to high transition, conversion starts. When the $\overline{CS}$ and $\overline{WR}$ inputs are low, the start flip-flop is set and the interrupt flip-flop and 8-bit register are reset. The next clock pulse transfers a logic high to the output of the start flip-flop. The logic high is ANDed with the next clock pulse, placing a logic high on the reset input of the start flip-flop. If either $\overline{CS}$ or $\overline{WR}$ have gone high, the set signal to the start flip-flop is removed, causing it to be reset. A logic high is placed on the D input of the 8-bit shift register and the conversion process is started. If the $\overline{CS}$ and $\overline{WR}$ inputs are still low, the start flip-flop, the 8-bit shift register, and the SAR remain reset. This action allows for wide $\overline{CS}$ and $\overline{WR}$ inputs with conversion starting from one to eight clock periods after one of the inputs goes high.

When the logic high input has been clocked through the 8-bit shift register, completing the SAR search, it is applied to an AND gate controlling the output latches and to the D input of a flip-flop. On the next clock pulse, the digital word is transferred to the 3-state output latches and the interrupt flip-flop is set. The output of the interrupt flip-flop is inverted to provide an $\overline{INTR}$ output that is high during conversion and low when the conversion is completed.

When a low is at both the $\overline{CS}$ and $\overline{RD}$ inputs, an output is applied to the DB0 through DB7 outputs and the interrupt flip-flop is reset. When either the $\overline{CS}$ or $\overline{RD}$ input returns to a high state, the DB0 through DB7 outputs are disabled (returned to the high-impedance state). The interrupt flip-flop remains reset.

ADC0803/4/5 A/D Converter Interface to Zilog Z80 and Z80A Microprocessors

This application illustrates the circuit configuration and the associated software that can be used to operate the ADC080X family of A/D converters with the ZILOG Z80A and Z80. The A/D circuits are 8-bit successive-approximation A/D converters that feature microprocessor-compatible control logic and parallel communication with the microprocessor via the data bus. The configuration features are as follows:

1. Minimum circuitry
2. Low cost
3. Very fast communication between the microprocessor and A/D converter
4. Optional microprocessor-interrupt acknowledgement of the end of conversion
5. Differential analog voltage inputs, which reject both common-mode voltages and the offset of the zero-input analog voltage value
6. Optional on-board generation of the A/D clock signal with an external resistor and capacitor.

The basic differences between the A/D converters in this family are given in Table 11.26.

Table 11.26 Differences between devices in the ADC080X A/D converter family (note 1)

	0803	0804	0805	UNIT
Total maximum adjusted error (with full-scale adjust)	±1/2	–	–	LSB
Total maximum unadjusted error ($V_{ref}/2$ = 2.5 V)	–	–	–	LSB
Total maximum unadjusted error ($V_{ref}/2$ = open)	–	±1	±1	LSB
Operating free-air temperature range	−40 to 85	0 to 70	−40 to 85	°C

NOTE 1: Conversion accuracies listed are with V_{CC} at 5 V and the clock frequency at 640 kHz. If a faster clock is used, conversion time will decrease proportionately and the accuracy will tend to decrease slightly.

Circuitry

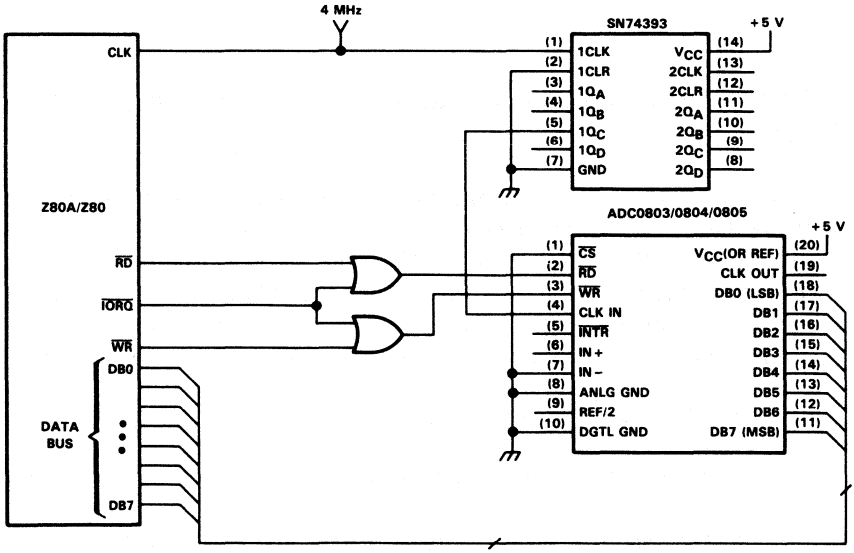


Fig. 11.215 Circuit diagram for Z80A/Z80 to ADC 0804/0805 interface

Fig. 11.215 shows the interconnection between the microprocessor and the A/D converter. The A/D converter write $\overline{WR}$ and read $\overline{RD}$ signals, which are generated by the microprocessor, are not masked by an addressing scheme. However, if additional I/O devices were placed on the data bus, masking could be easily designed.

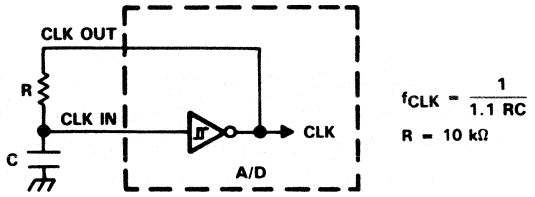


Fig. 11.216 Configuration and equations for on-board clock generator

The SN74393 dual binary counter generates a 500 kHz clock signal for the A/D converter. Any binary counter may be used instead of the SN74393. In addition, any clock signal within the clock frequency specification may be used. Another way to generate the clock signal is to use an external resistor and capacitor in conjunction with the CLK IN and CLK OUT pins of the A/D converter. The configuration and frequency equation for this method are shown in Fig. 11.216.

Timing Diagram

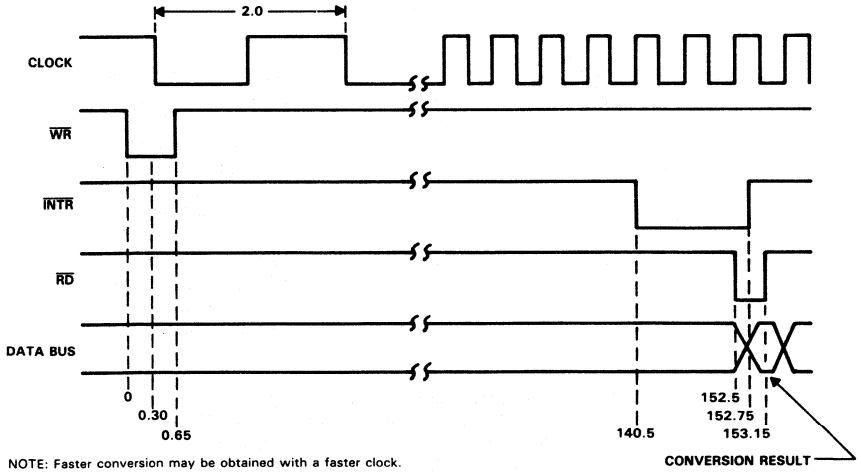


Fig. 11.217 Timing diagram for Z80A/Z80 to ADC0803/0804/0805 interface

Fig. 11.217 shows the timing diagram for the interface. With a 500 kHz clock, the conversion time is 140.5 μs and the A/D $\overline{\text{INTR}}$ pin is reset when the $\overline{\text{RD}}$ signal goes low.

Software

The following software listing presents the software for this interface. The software is minimal due to the microprocessor-compatibility of the A/D converter. In allowing time for conversion, the designer can use any delay that is convenient, including a timer function. Another method is to use the $\overline{\text{INTR}}$ signal on the A/D converter to start a microprocessor interrupt routine that reads the conversion result. If an on-board clock generator like the one shown in Fig. 11.216 is used, the conversion time will vary with the tolerances of the resistor and capacitor. In this situation, the designer can allow a conservative delay to assure that the conversion is complete before reading the result. Another method might be to wait for an acknowledgement of the $\overline{\text{INTR}}$ signal and use this to inform the microprocessor of a completed conversion.

```
                        ;               Software for Z80/Z80A, ADC0803,
                        ;               ADC0804, ADC0805, Interface
                        ;
                        ;
        FF 00           WRITE:          EQU FFH             ;Interface write address
        FF 00           READ:           EQU FFH             ;Interface read address
                                                            ;The above addresses could be
                                                            ;anything unless they are used
                                                            ;to drive an address decode
                                                            ;circuit, which addresses
                                                            ;the A/D IC
                        ;
0000    D3 FF           START:          OUT (WRITE),A       ;Start conversion of analog
                                                            ;channel; the contents of the
                                                            ;accumulator are not important;
                                                            ;the write pulse however, is.

                        ;
                        ;Before reading the conversion result, a slightly
                        ;greater than 140 microsecond delay (with a 500 kHz
                        ;A/D clock), to allow for conversion time, can be
                        ;selected in any way which is convenient for the designer.
                        ;                       OR
                        ;The microprocessor can continue to perform main
                        ;program software and the conversion result can be
                        ;retrieved by an interrupt routine, which is initiated
                        ;by the INTR(bar) signal, which signifies the end of
                        ;conversion, on pin 5.
                        ;
0002    DB FF                           IN A, (READ)        ;Read conversion result
                                                            ;into the A register
0004                                    END
```

ADC0803/4/5 A/D Converter Interface to Rockwell 6502 Microprocessors

This application shows the circuit configuration and the associated software that can be used to operate this family of A/D converters with the Rockwell 6502 microprocessor. The interface circuit diagram is shown in Fig. 11.218.

A data conversion cycle begins by performing a write to the ADC080X with a Start (STA) accumulator instruction. The conversion requires between 66 and 73 clock cycles. The ADC080X provides an interrupt request signal when the conversion is complete. The circuit timing is shown in Fig. 11.219.

An alternate method for retrieving the conversion result would be to use a wait state in a software delay loop until the conversion results are read into the 6502 with an LDA instruction. A software listing for a typical interrupt service routine is shown. Note that the circuit employs a minimum of address decoding hardware; some applications may require additional decoding hardware.

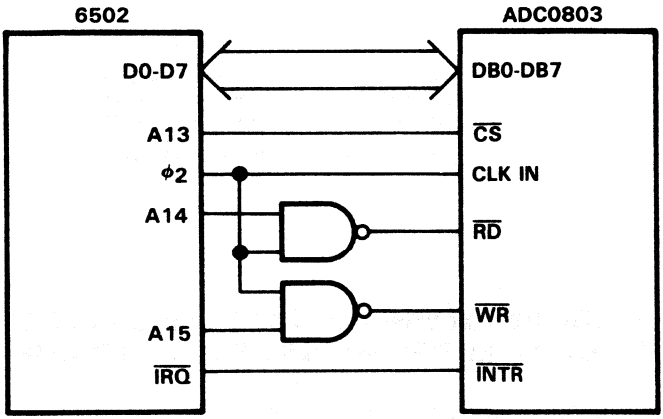


Fig. 11.218 6502 to ADC08030804/0805 interface circuit diagram

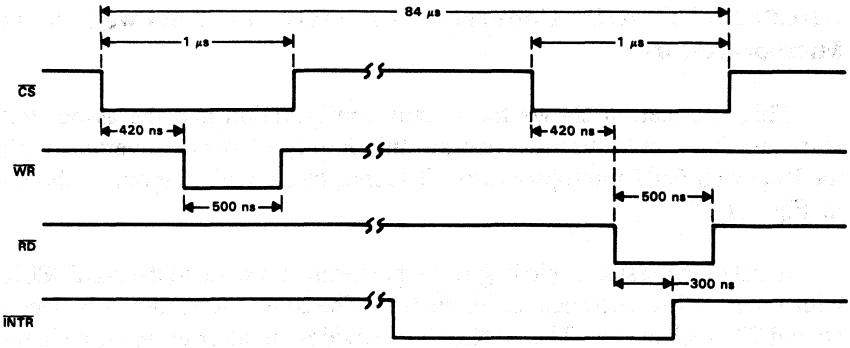


Fig. 11.219 6502 to ADC0803/0804/0805 interface timing diagram

```
;               Register Assignments for ADC0803, ADC0804,
;               and ADC0805 to 6502 Interface
;
WRITE:      .EQU 8800H
READ:       .EQU 4800H
;
;
MAIN:           .
                .
                .

                STA WRITE       ;Start conversion

                .
                .
                .

ISR:        PHA                 ;Save contents of accumulator
            LDA READ            ;Read conversion results
            STA DATA            ;Store results in memory
            PLA                 ;Restore accumulator
            RTI                 ;Return to main program
```

ADC0808, ADC0809, TL0808 and TL0809 CMOS 8-BIT ANALOG-TO-DIGITAL CONVERTERS WITH 8-CHANNEL MULTIPLEXERS

The ADC0808 and ADC0809 are 8-bit A/D converters that offer latched address inputs as well as latched 3-state outputs. The total unadjusted error in the ADC0808 is ±½ LSB max and for the ADC0809 is ±1.0 LSB max. These devices feature ratiometric conversion with a conversion time

of 100 μs. They may be powered from a single 5-V supply and are easily connected to a microprocessor. TL0808 and TL0809 are versions specified to operate over a 3 to 5.25 V supply range and have a low power consumption of typically 0.3 mW at 3 V supply voltage.

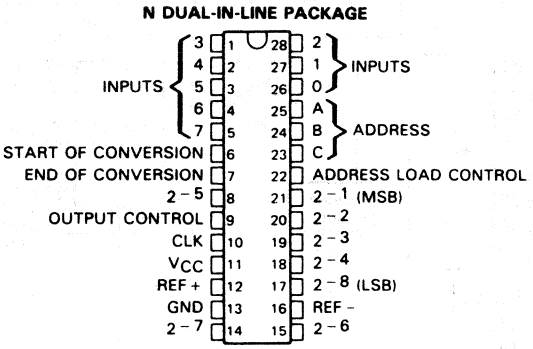


Fig. 11.220 ADC0808, ADC0809, TL00808 and TL0809 pinouts (top view)

Fig. 11.220 shows how the inputs and address lines have been grouped to facilitate PC board layout.

Description

The ADC0808 and ADC0809 are monolithic CMOS devices with an 8-channel multiplexer, an 8-bit analog-to-digital A/D converter, and microprocessor-compatible control logic. The 8-channel multiplexer can be controlled by a microprocessor through the 3-bit address decoder with address load control to select any one of eight single-ended analog switches connected directly to the converter. The 8-bit A/D converter uses the successive-approximation conversion technique featuring a high-impedance threshold detector, a switched-capacitor array, a sample-and-hold, and a successive-approximation register (SAR). The functional block diagram is shown in Fig. 11.221 and the operating sequence in Fig. 11.222.

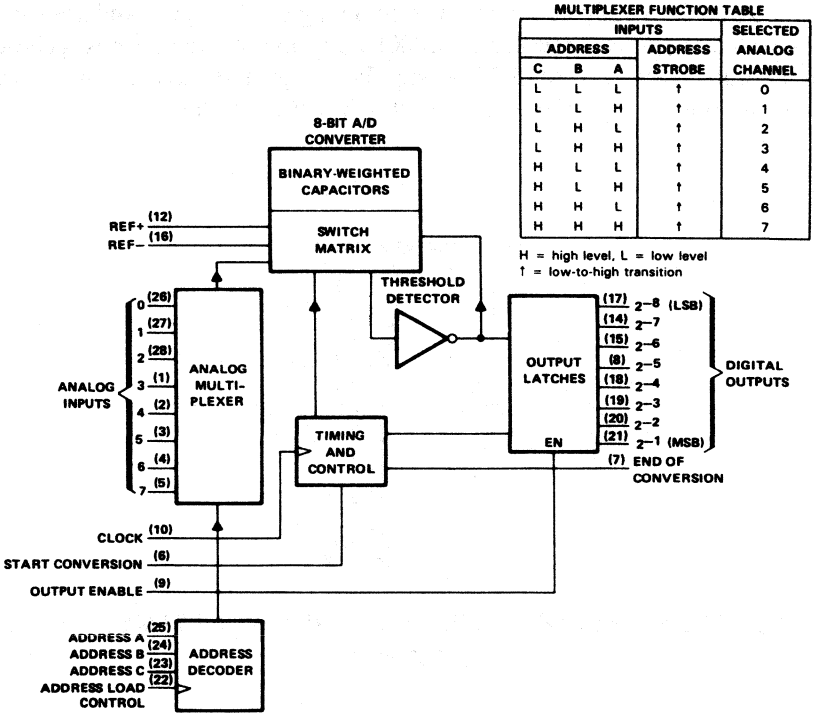


Fig. 11.221 Functional block diagram and function table

The comparison and conversion methods used eliminate the possibility of missing codes, nonmonotonicity, and the need for zero or full-scale adjustment. Also featured are latched 3-state outputs from the converter and latched inputs to the multiplexer address decoder. The single 5-V supply and low power requiremnts make the ADC0808 and ADC0809 especially useful for a wide variety of applications. Ratiometric conversion is made possible by access to the reference voltage input terminals.

Multiplexer

The analog multiplexer selects 1 of 8 single-ended input channels as determined by the address decoder. Address load control loads the address code into the decoder on a low-to-high transition.

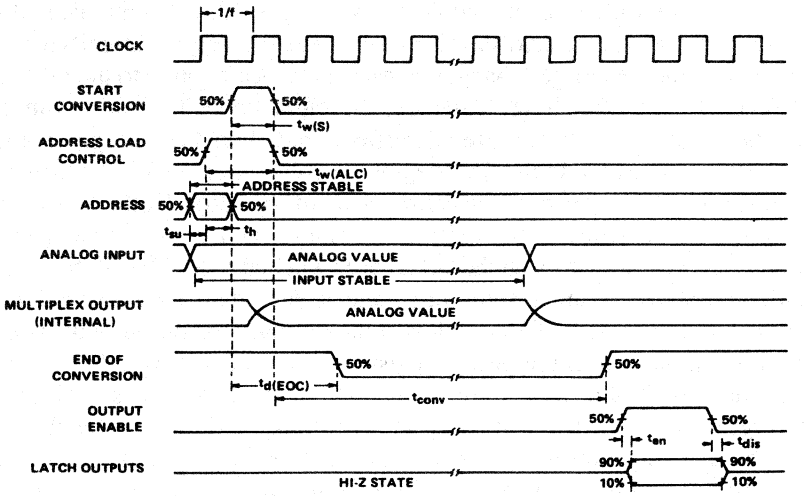


Fig. 11.222 Operating sequence

Converter

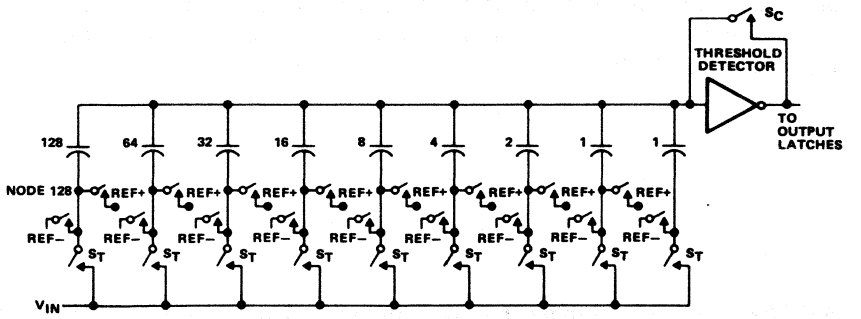


Fig. 11.223 Simplified model of the successive-approximation system

The CMOS threshold detector in the successive-approximation conversion system determines each bit by examining the charge on a series of binary-weighted capacitors (Fig. 11.223).

In the first phase of the conversion process, the analog input is sampled by closing switch S_C and all S_T switches, and by simultaneously charging all the capacitors to the input voltage.

In the next phase of the conversion process, all S_T switches and the S_C switch are opened and the threshold detector begins identifying bits by identifying the charge (voltage) on each capacitor relative to the reference voltage. In the switching sequence, all eight capacitors are examined separately until all 8 bits are identified, and then the charge -convert sequence is repeated. In the first step of the conversion phase, the threshold detector looks at the first capacitor (weight = 128). Node 128 of this capacitor is switched to the reference voltage, and the equivalent nodes of all the other capacitors on the ladder are switched to REF−. If the voltage at the comparator input summing node is less than its threshold level (approximately one-half V_{CC} voltage) the comparator output will be high and a bit is placed in the output register. Then the 128-weight capacitor remains switched to REF+ the remainder of the capacitor sampling (bit testing) process. If the voltage at the summing node is greater than the threshold level, a bit is not set in the output register and the 128-weight capacitor is switched back to REF−. The process is repeated for the 64-weight capacitor, then the 32 weight capacitor and so on until all bits are tested.

With each step of the capacitor-sampling process, the initial charge is redistributed among the capacitors. This conversion process is successive approximation, but relies on charge redistribution rather than a successive-approximation register (and reference D/A) to count and weigh the bits from MSB to LSB.

ADC0808/9 and TL0808/9 A/D Converter Interface to Zilog Z80A Microprocessors

This application presents the circuit configuration and the associated software that can be used to operate this family of A/D converters with the ZILOG Z80A and Z80. These A/D circuits are 8-bit successive-approximation A/D converters that feature microprocessor-compatible control logic and parallel communication with the microprocessor via the data bus. This configuration features:

1. Minimum circuitry
2. Low cost

3. Very fast communication between the microprocessor and A/D converter
4. Optional microprocessor-interrupt acknowledgement of the end of conversion
5. Convenient control of the analog multiplexer.

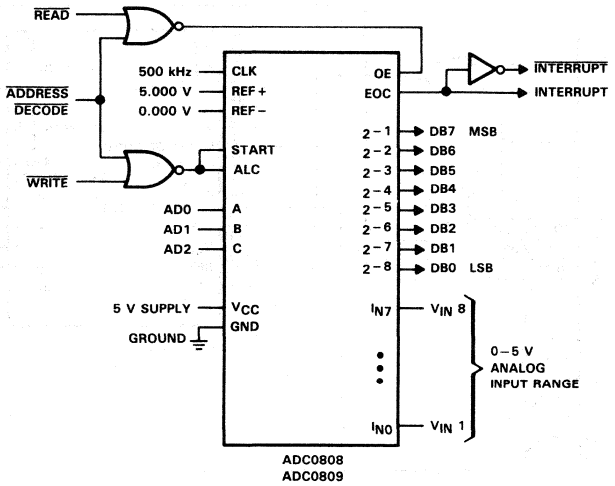


Fig. 11.224 Typical interface application

Fig. 11.224 shows the interconnection points between the microprocessor and the A/D converter for a typical interface application.

Fig. 11.225 shows the interconnection between the microprocessor and the TL0808 and T0809 and ADC0808 and ADC0809 A/D converters. The SN74393 dual binary counter is used to generate a 500 kHz clock for the A/D converter. Any binary counter may be used instead of the SN74393 or any clock signal within the A/D converter clock frequency specification may be used. The microprocessor-generated A/D control and address signals are not masked by an addressing scheme; however, if additional I/O devices are placed on the data bus, masking would be necessary.

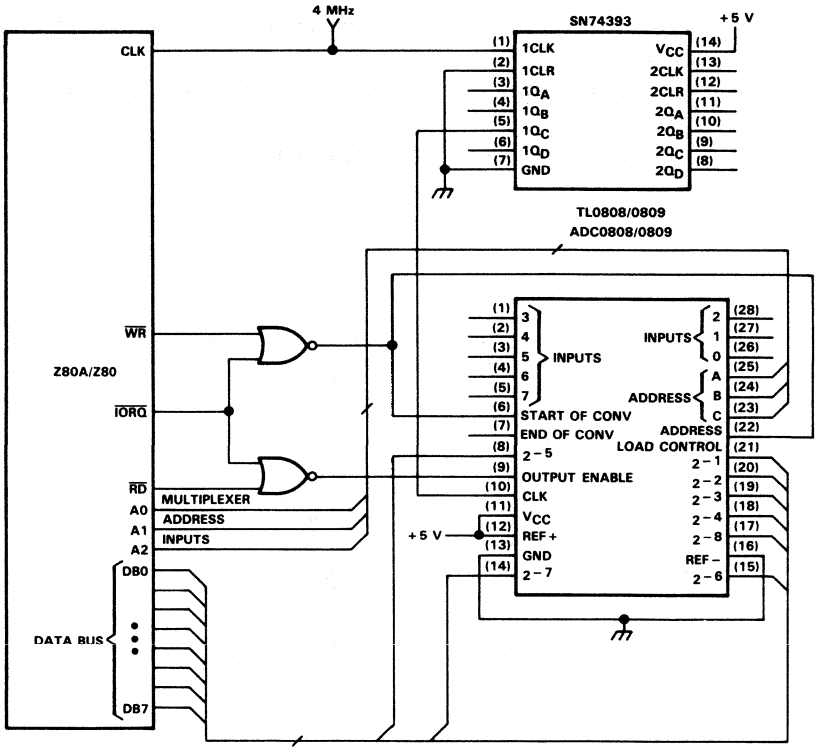


Fig. 11.225 Circuit diagram for Z80A/Z80 to TL0808/0809 ADC0808/0809 interface

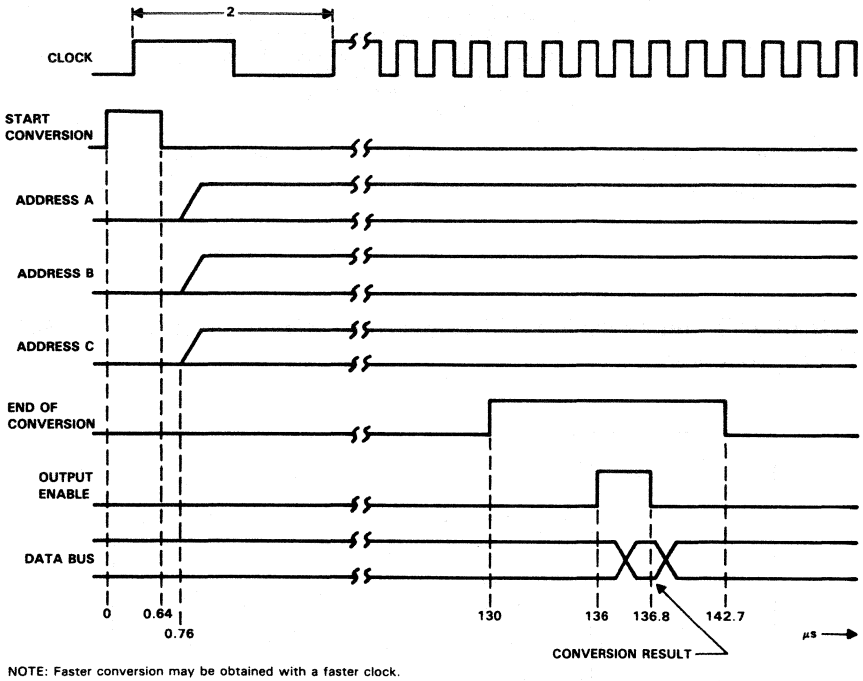
Timing Diagram

Fig. 11.225 is the timing diagram for the interface. With a 500 kHz A/D clock, the conversion time for the TL0808, TL0809, ADC0808, or ADC0809 devices is 130 μs with the EOC signal going low about 6 μs after activation of the output enable signal.

Software

The associated program listing presents the software routines for this interface. The coding is minimal due to the microprocessor-compatibility of the A/D converters. This code is written so that it may be easily incorporated into a subroutine if desired. In allowing time for conversion,

the designer may use any delay which is convenient, including a timer function, or the End of Conversion (EOC) signal on the A/D converter to initiate a microprocessor interrupt routine that reads the conversion result.



NOTE: Faster conversion may be obtained with a faster clock.

Fig. 11.226 Timing diagram for Z80/Z80A to TL0808/0809 ADC0808/0809 interface

```
                        ;Interface Software for Z80A/Z80-TL0808 / TL0809 ICs
                        ;                              -ADC0808 / ADC0809 ICs
                        ;
                        ;
                        ;
        FF 00   READ:   EQU FFH         ;Interface read address; this
                                        ;address could be anything
                                        ;unless it is used to drive
                                        ;an address decode circuit,
                                        ;which addresses the A/D IC
                        ;
0000    0E 00   START:  LD C,00H        ;Prepare to address analog
                                        ;channel 0; 01 would address
                                        ;channel 1, etc.
                        ;
0002    ED 49   LOD0808: OUT (C),C      ;Start conversion of analog
                                        ;channel per C register
```

```
                        ;
                        ;Before reading the conversion result, a slightly
                        ;greater than 130 microsecond delay (with a 500 kHz)
                        ;A/D clock), to allow for conversion time, can be
                        ;selected in any way which is convenient for the
                        ;designer.
                        ;                    OR
                        ;The microprocessor can continue to perform main
                        ;program software and the conversion result can be
                        ;retrieved by an interrupt routine, which is initiated
                        ;by the End of Conversion (EOC) signal on pin 7 of the
                        ;ADC0808/0809 and TL0808/0809 ICs
                        ;
                        ;
0004   DB FF                    IN A, (READ)      ;Read conversion result
                                                  ;into the A register
0006                            END
```

ADC0808/9 and TL0808/9 A/D Converter Interface to Rockwell 6502 Microprocessors

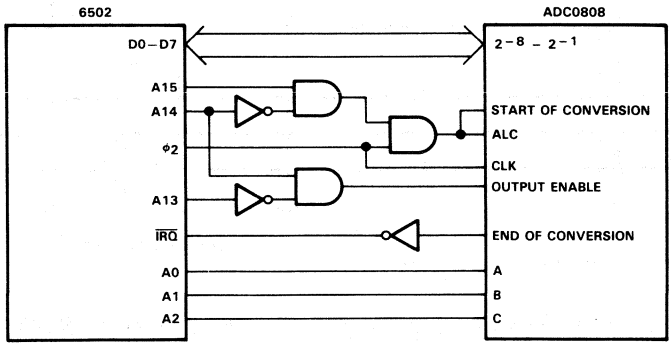


Fig. 11.227 6502 to ADC0809 interface circuit diagram

This application shows the circuit configuration and the associated software that can be used to operate the ADC0808 and ADC0809 A/D converters with the Rockwell 6502 microprocessor. The interface circuit diagram is shown in Fig. 11.227. A data conversion cycle is initiated by performing an STA instruction. This generates a start conversion pulse that clears the EOC line. The address in memory where the accumulator stores its contents determines which channel is selected for conversion. The conversion requires approximately 65 clock cycles. The clock frequency may vary from 10 kHz the accuracy may decrease slightly. The circuit timing diagram is shown in Fig. 11.228.

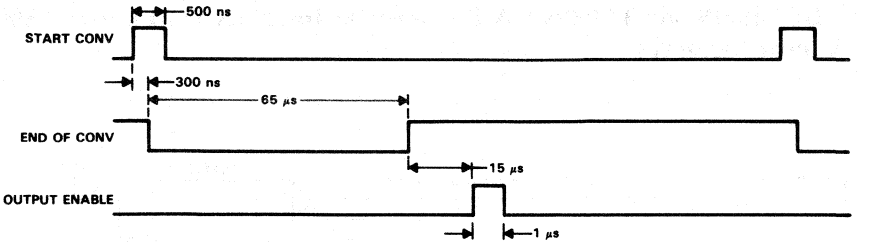


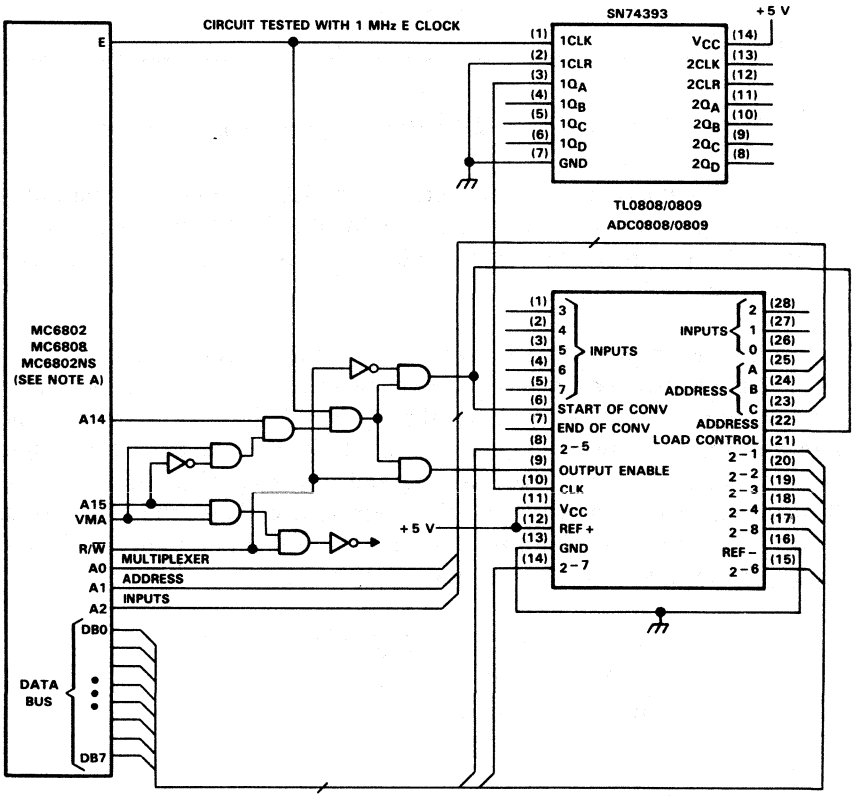
Fig. 11.228 6502 to ADC0809 interface timing diagram

Upon completion of the conversion, the EOC signal is set and may be used to generate an interrupt signal for the Rockwell 6502. Also, a software delay loop may be used to achieve the proper delay until conversion is complete.

If the interrupt scheme is chosen, the interrupt service routine must either start another conversion cycle that will clear the EOC signal or disable the interrupts. If this is not done, the Rockwell 6502 will remain in the interrupt service routine loop. It may be desirable to add a flip-flop and an AND gate to control the enabling and disabling of the EOC signal. Structuring of the interrupt and the address decoding scheme are both flexible and should be optimized for the particular application. A software listing for an interrupt service routine follows.

```
;               Register Assignments for ADC0808 and ADC0809
;               Interface to the Rockwell 6502
ADC0808         .EQU 4000H
WRITE0          .EQU 8000H
WRITE1          .EQU 8001H
  .               .   .
  .               .   .
  .               .   .
WRITE7          .EQU 8007H
;
;
MAIN:             .
                  .
                  .
                STA WRITE0          ;Start conversion, clear interrupt
                  .
                  .
                  .


ISR:            PHA                 ;Save accumulator
                LDA ADC0808         ;Read conversion results into acc.
                STA DATA            ;Store conversion results
                PLA                 ;Restore accumulator
                PLP                 ;Pull status from stack
                SEI                 ;Disable interrupts
                PHP                 ;Push status back on stack
                RTI                 ;Return to main program
```

ADC0808/9 and TL0808/9 A/D Converter Interface to Motorola 6800 Microprocessors



NOTE A: Refer to Table 11-27 for information about other Motorola microprocessors.

Fig. 11.229 Circuit diagram for ADC0808/0809 and TL0808/0809 interface

This application illustrates a circuit configuration and the software that can be used to operate this family of A/D converters with the Motorola 6800 family of microprocessors. Fig. 11.224 shows a typical interface application. Fig. 11.229 shows the interconnection between the Motorola 6802 microprocessor and the TL0808 or TL0809 A/D converter.

Table 11.27 Considerations for using other microprocessors

MICROPROCESSOR	E CLOCK OR E CLOCK EQUIVALENT
6800	ϕ_2 is equivalent to the 6802/6809 E pin
6802/6808/6802ns	See Fig. 11.229
6809	See Fig. 11.229
6809E/68HC09E	See Fig. 11.229 and "Motorola 8-bit Microprocessor and Peripheral Data Book" for clock generator – pg. 3-277

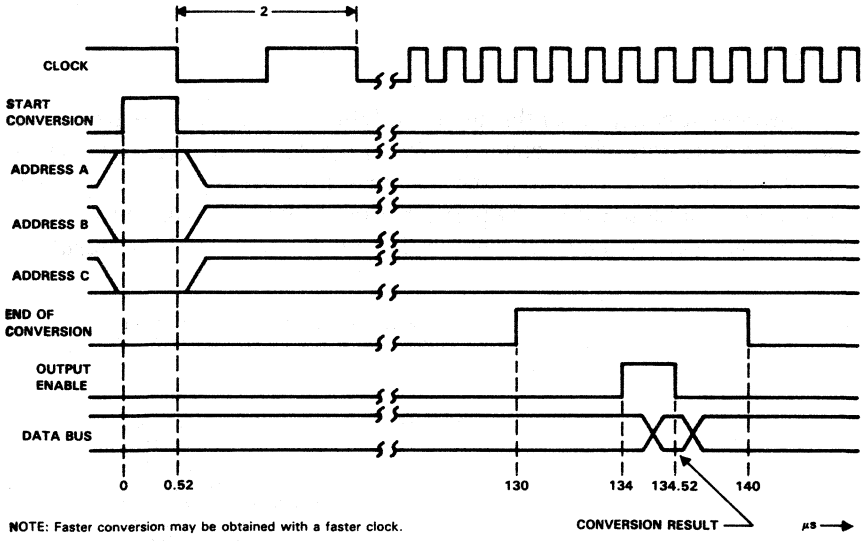


Fig. 11.230 Timing diagram for ADC0808/0809 and TL0808/0809 interface

The SN74393 dual binary counter is used to generate a 500 kHz clock for the A/D converter. Any binary counter may be used in place of the SN74393 or any clock signal within the A/D converter clock frequency specification can be used. The 6800 E clock or E clock equivalent, if within the clock frequency specification, may also be connected directly to the clock input of the A/D converter. When using the Motorola microprocessors with clock frequencies greater than 1 MHz, the designer must check to see that the A/D converter write and read timing specifications are met. The microprocessor-generated A/D control and address signals are masked

by the 4000H-7FFFH addressing range. A more efficient addressing scheme can easily be implemented if desired. Table 11.27 provides information for adapting the circuit of Fig. 11.229 for use with other members of the 6800 family of microprocessors.

Timing Diagram

Fig. 11.230 is the timing diagram for this interface. With a 500 kHz A/D clock, the conversion time for the TL0808/TL0809 and ADC0808/ADC0809 A/D converters is 130 μs. The EOC signal falls approximately 6 μs after activation of the OUTPUT ENABLE signal.

Software

The software listing for this interface follows. The software is minimal due to the microprocessor-compatibility of the A/D converter devices. In allowing time for conversion, the designer can use any delay which is convenient, including either a timer function or the EOC signal to initiate the microprocessor interrupt routine that reads the conversion result.

```
                        ;                Software for Motorola 680X-TL0808 / TL0809 ICs
                        ;                                      -ADC0808 / ADC0809 ICs
                        ;
                        ;
                        ;
        40 00           READ:    EQU 4000H      ;Interface read address; this
                                                ;address is selected to ensure
                                                ;that the read operation reads
                                                ;the conversion result byte
                                                ;and does not read the program
                                                ;memory or on-board RAM (if
                                                ;the microprocessor is the
                                                ;6802) by mistake
                        ;
0000    B7 40 00        START:   STAA 4000H     ;Address analog channel 0 and
                                                ;start conversion; 4001 would
                                                ;address channel 1, etc.
                        ;
                        ;Before reading the conversion result, a slightly
                        ;greater than 130 microsecond delay (with a 500 kHz
                        ;A/D clock), to allow for conversion time, can be
                        ;selected in any way which is convenient for the
                        ;designer.
                        ;                OR
                        ;The microprocessor can continue to perform main
                        ;program software and the conversion result can be
                        ;retrieved by an interrupt routine, which is initiated
                        ;by the End of Conversion (EOC) signal on pin 7 of the
                        ;ADC0808/0809 and TL0808/0809 ICs
                        ;
                        ;
0003    B6 40 00                 LDAA READ      ;Read conversion result
                                                ;into the A accumulator
0006                             END
```

ADC0831, ADC0832, ADC0834, ADC0838 8-BIT A/D CONVERTERS WITH SINGLE DIFFERENTIAL, 2, 4, 8-MULTIPLEXER, CHANNEL OPTIONS AND SERIAL CONTROL

The ADC0831, ADC0832, ADC0834, and ADC0838 A/D converters are a family of A/D converters that may be easily used with a microprocessor or operated stand-alone. There is no zero or full-scale adjust. An on-chip shunt regulator on the ADC0834 and ADC0838 allows operation with high-voltage supplies. A conversion time of 32 μs is possible at a clock frequency of 250 kHz. The input voltage range is 0 to 5 V with a single 5 V supply. The systematic placing of data lines, V_{CC}, and ground terminals shown in Fig. 11.231 allows efficient PC board layout.

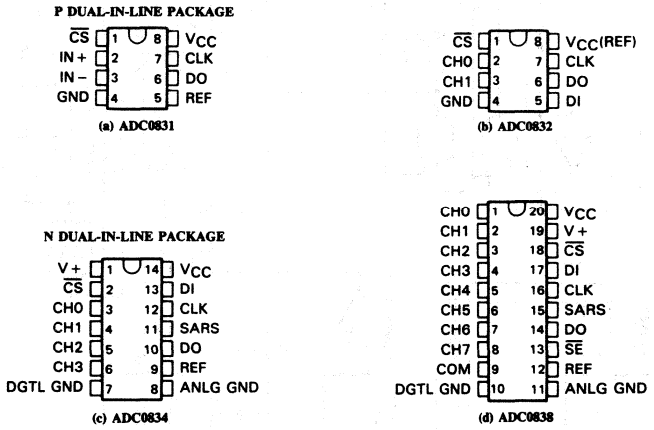


Fig. 11.231 ADC0831, ADC0832, ADC0834, ADC0838 pinouts (top view)

Description

The ADC0831, ADC0832, ADC0834, and ADC0838 are 8-bit successive-approximation analog-to-digital converters each with a serial input/output and configurable input multiplexer with up to 8 channels on the ADC0838. The serial input/output can be used with standard shift registers or microprocessors. See Fig. 11.232 for a functional block diagram of the ADC0831 and ADC0832 and Fig. 11.233 for the functional block diagram of the ADC0834 and ADC0838.

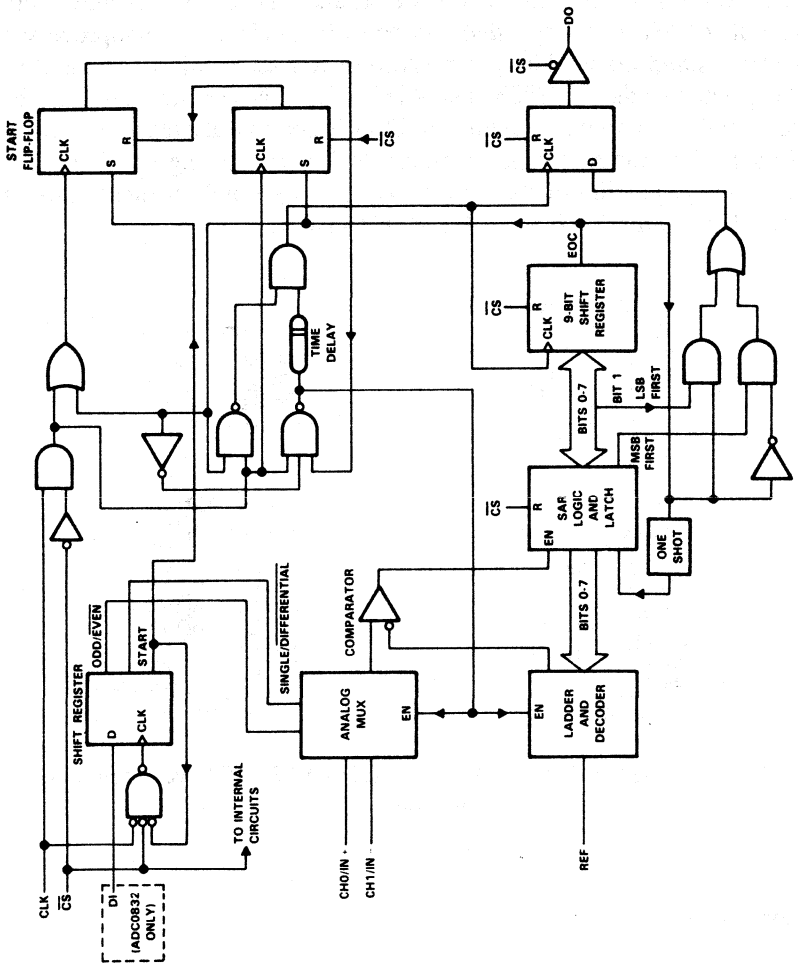
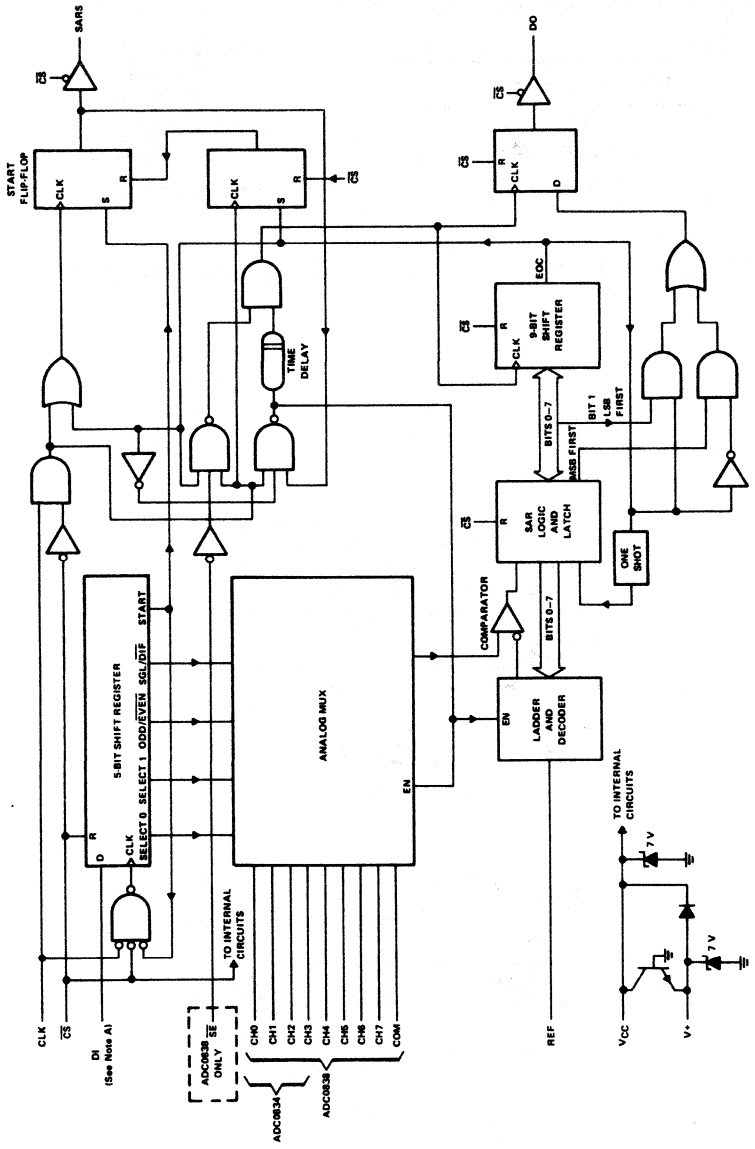


Fig. 11.232 ADC0831A-B, ADC0832A-B functional block diagram



NOTE A: For the ADC0834, DI is input directly to the D input of SELECT 1, SELECT 0 is forced to a high.

Fig. 11.233 ADC0834A-B, ADC0838A-B functional block diagram

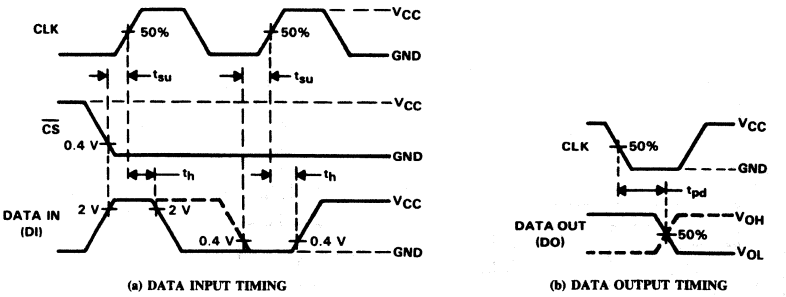


Fig. 11.234 ADC0832 data input and output timing diagram

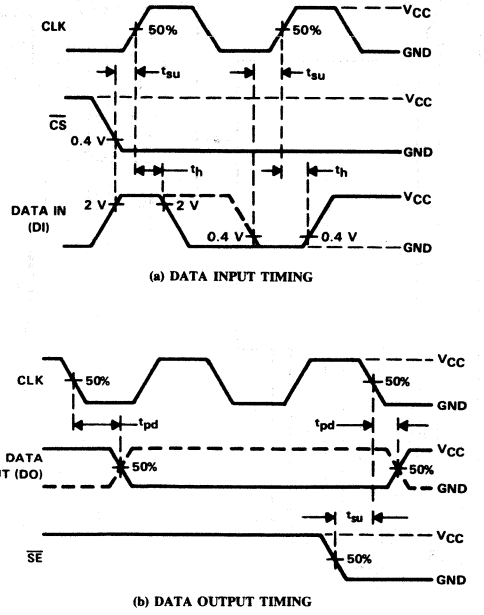


Fig. 11.235 ADC0834, ADC0838 Data Input and Output Timing Diagram

The 2-, 4-, or 8-channel multiplexers are software configured for single-ended or differential inputs as well as channel assignment. Figs. 11.234 and 11.235 show the ADC0832 and ADC0834–838 input and data output timing diagrams and Fig. 11.236 shows the ADC0831 sequence of operation. The operating sequence for the ADC0832 is illustrated in Fig. 11.237 while the ADC0834 sequence of operation is shown in Fig. 11.238. The ADC0838 sequence of operation is shown in Fig. 11.239.

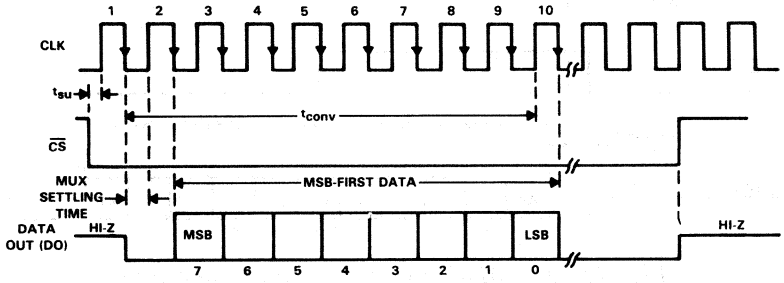


Fig. 11.236 ADC0831 sequence of operation

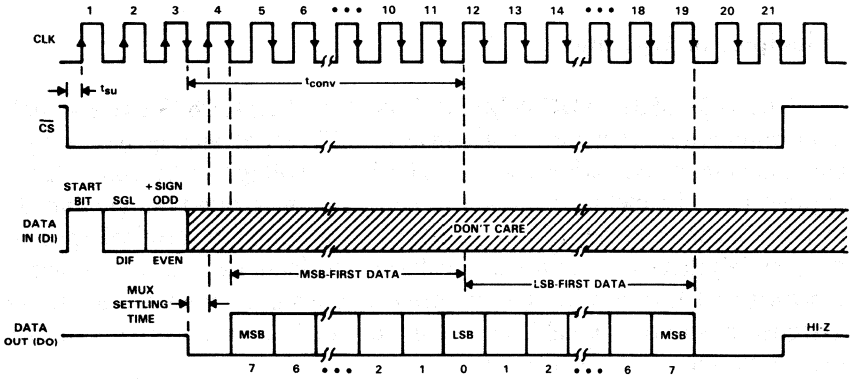


Fig. 11.237 ADC0832 sequence of operation

The differential analog voltage input increases the common-mode rejection and eliminates the offset due to the analog zero input voltage value. In addition, the voltage reference input can be adjusted to allow the encoding of any smaller analog voltage span to the full 8 bits of resolution.

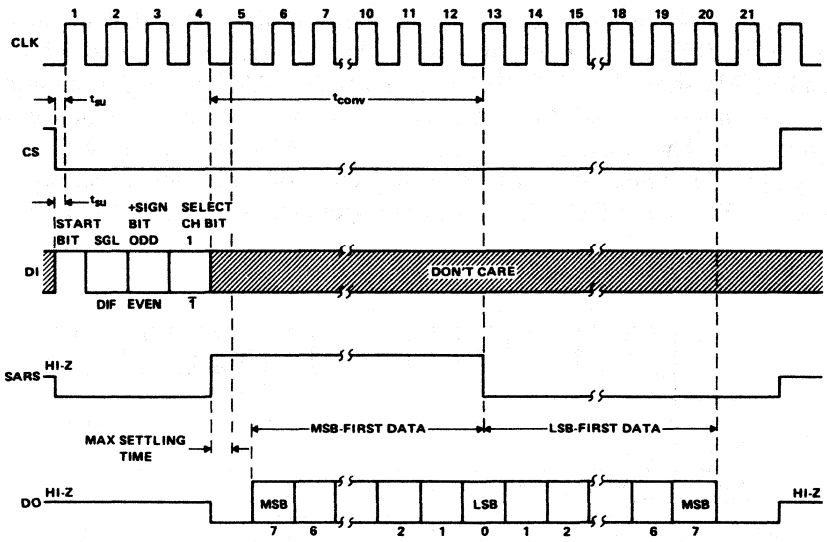


Fig. 11.238 ADC0834 sequence of operation

Principles of Operation

The ADC0831, ADC0832, ADC0834, and ADC0838 use a sample data comparator structure that converts differential analog inputs by a successive-approximation routine. The input voltage to be converted is applied to a channel terminal and is compared to ground (single-ended), to an adjacent channel (differential), or to a common terminal (pseudo differential) that can be an arbitary voltage. The input terminals are assigned a positive (+) or negative (−) polarity. If the signal input applied to the assigned positive terminal is less than the signal on the negative terminal, the converter output is all zeros.

Channel selection and input configuration are under software control using a serial data link from the controlling processor. A serial communication format allows more functions to be included in a converter package with no increase in pin count. In addition, it eliminates the transmission of low-level analog signals by facilitating the remote location of the converter at the analog sensor. This process returns noise-free digital data to the processor.

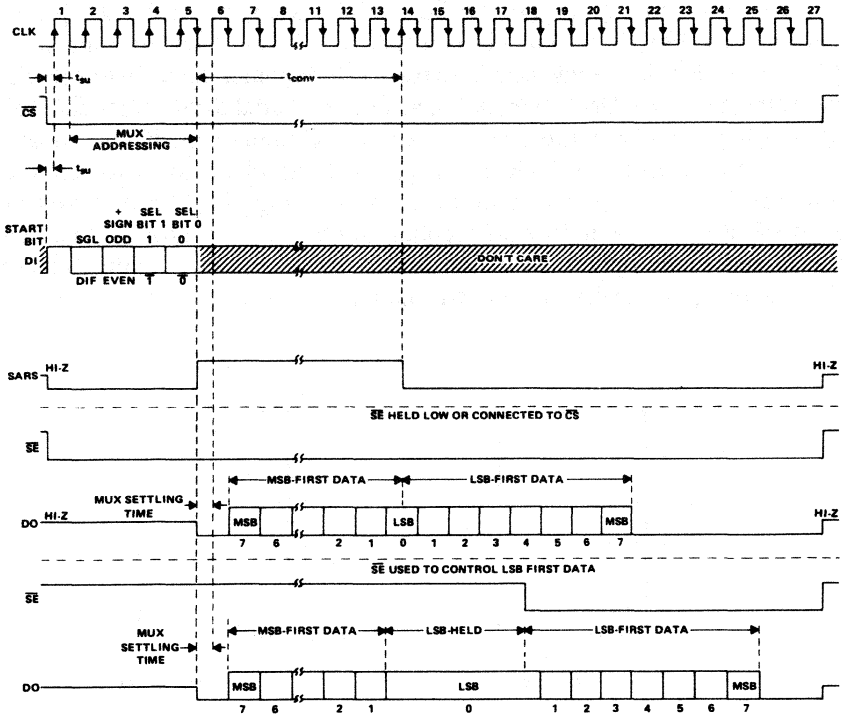


Fig. 11.239 ADC0838 sequence of operation

A particular input configuration is assigned during the multiplexer addressing sequence. The multiplexer address is shifted into the converter through the data input (DI) line. (The ADC0831, unlike the ADC0832, ADC0834, and ADC0838, contains only one differential input channel having a fixed polarity assignment and does not have multiplexer addressing.) The multiplexer address selects the analog inputs to be enabled and determines whether the input is single-ended or differential. When the input is differential, the polarity of the channel input is assigned. Differential inputs are assigned to adjacent channel pairs. For example, channel 0 and channel 1 may be selected as a differential pair. However, these channels cannot act differentially with any other channel. In addition to selecting the differential mode, the polarity may also be selected. Either channel of the channel pair may be designated as negative or positive.

As shown in Fig. 11.240 the ADC0838 has several input multiplexer options. In Fig. 11.240(a), it is used for 8 single-ended inputs. In Fig. 11.240(b), it is being used for a pseudo differential input. In this mode, the voltage on the COM(−) input is negative compared to any other channel. This voltage can be any reference potential common to all channel inputs. This feature is useful in single-supply applications where all analog circuits are biased to a potential other than ground. Fig. 11.240(c) shows an ADC0838 set up with four differential inputs. Fig. 11.240(d) illustrates the mixed mode option. Selection of this mode allows combinations of differential pairs and single-ended inputs.

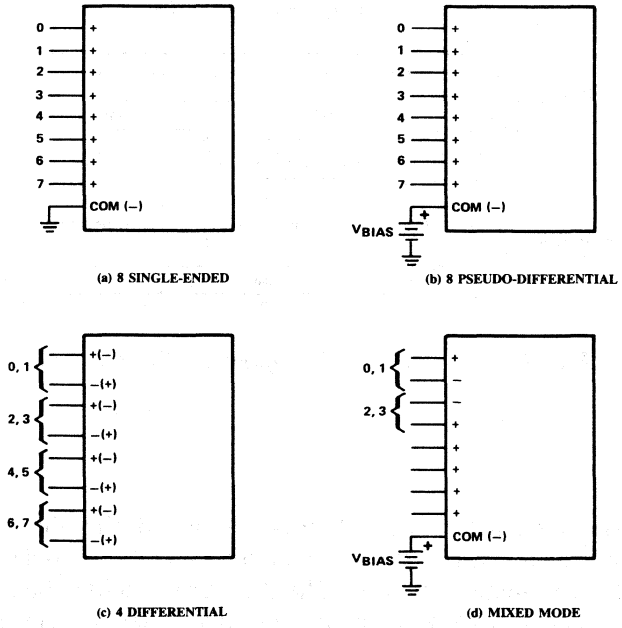


Fig. 11.240 Analog input multiplexer options for ADC0838

A conversion is initiated by setting the chip select ($\overline{CS}$) input low, which enables all of the logic circuits. The $\overline{CS}$ input must be held low for the complete conversion process. On each low-to-high transition of the clock input received from the processor, the data on the DI input is clocked into the multiplexer address shift register. The first logic high on the input

is the start bit. A 2-to-4 bit assignment word follows the start bit. On each successive low-to-high transition of the clock input, the start bit and assignment word are shifted through the shift register. When the start bit has been shifted into the start location of the multiplexer register, the input channel has been selected and conversion starts. The SAR status output (SARS) goes high to indicate that a conversion is in progress and the DI input to the multiplexer shift register is disabled for the duration of the conversion.

An interval of one clock period is automatically inserted to allow for the selected multiplexer channel to settle. The data output (DO) comes out of the high-impedance state and provides a leading low for this one clock period of multiplexer settling time. The SAR comparator compares successive outputs from the resistive-ladder with the incoming analog signal. The comparator output indicates whether the analog input is greater or less than the resistive-ladder output. This data is parallel loaded in to a 9-bit shift register that immediately outputs an 8-bit serial data word to the DO output with the most significant bit (MSB) first. After eight clock periods, the conversion is complete and the SAR status (SARS) output goes low. When $\overline{CS}$ goes high, all internal registers are cleared. At this time, the output circuits go to three state. If another conversion is desired, the CS line must make a high-to-low transition followed by address information.

In the ADC0831, only MSB-first data is output. The ADC0832 and ADC0834 output the LSB-first data after the MSB-first data stream. In the ADC0838, the programmer has the option of selecting MSB-first data or LSB-first data. To output LSB-first data, the shift enable ($\overline{SE}$) control input must go low. Data stored in the 9-bit shift register is now output with LSB first. The DI and DO pins can be tied together and controlled by a bidirectional processor I/O bit received on a single wire. This is possible because the DI input is examined only during the multiplexer addressing interval and the DO output is still in a high-impedance state.

ADC0831/0832/0834/0838 A/D Converter Interface to Zilog Z80A and Z80 Microprocessors

The ADC0831, ADC0832, ADC0834, and ADC0838 devices are 8-bit successive-approximation analog-to-digital converters with serial input/output and configurable input multiplexers with up to 8 channels. These A/D converters are designed to easily communicate with microprocessors in a serial fashion. The hardware configurations and the

associated software presented can be used to operate the ZILOG Z80A and Z80 microprocessors with the ADC0830 converters. The timing diagrams show the interaction between the microprocessor and the A/D converter.

The ADC0832, ADC0834, and ADC0838 A/D converters can be software configured in either the single-ended or differential input mode. Additionally, the differential ± inputs can be interchanged through software manipulation.

The serial interface features:

1. Low cost
2. Minimum circuitry
3. Fast conversion and communication between the A/D converter and the microprocessor
4. Remote control advantages of serial A/D converters.

Circuitry–ADC0832, ADC0834, and ADC0838

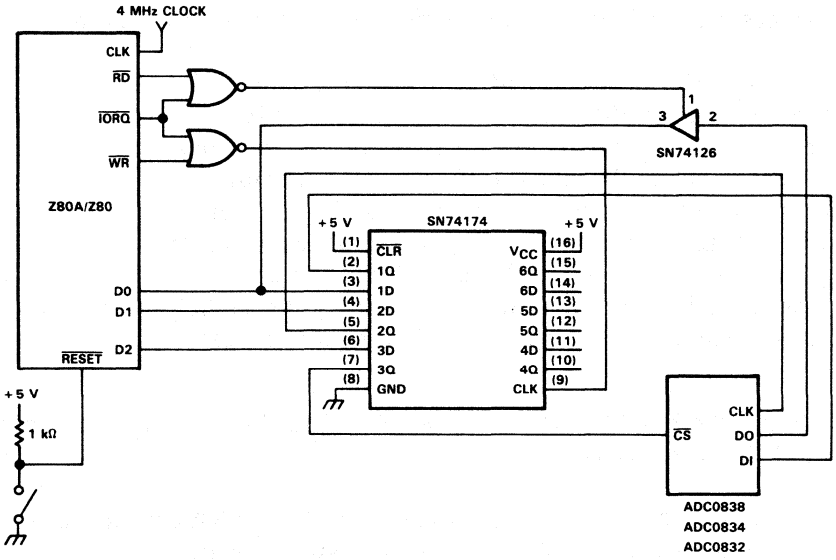


Fig. 11.241 Z80A/Z80 to ADC0838, ADC0834, ADC0832 interface circuit diagram

Fig. 11.241 shows the interconnection between the microprocessor and the ADC0832, ADC0834, and ADC0838 A/D converters. The interconnection is identical for all three A/D converters. The microprocessor DO pin can be used to transmit to and receive digital data between the ADC083X DI and DO pins, respectively. The SN74126 3-state buffer output is in the high-impedance state except during a microprocessor read operation. The SN74174 quad D-type flip flop is used to synchronize and slow down the write/read communication between the microprocessor and ADC083X so the ADC083X timing specifications are satisfied. Rather than NOR gates, OR gates may be used to activate the flip-flops and 3-state buffer on the positive transitions of the write WR and read RD strobes, instead of on the negative transitions. However, if OR gates are used, a SN74125 buffer must be used in place of the SN74126.

Timing Diagram—ADC0838 Device

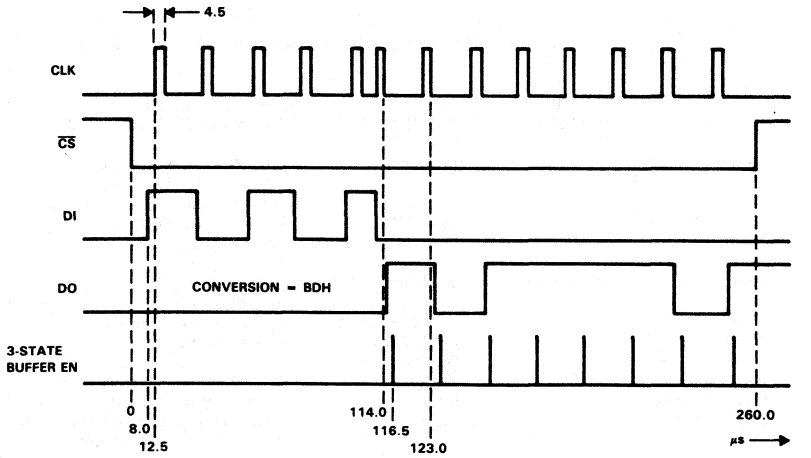


Fig. 11.242 Timing diagram for Z80A/Z80 to ADC0838 interface (microprocessor clock = 4 MHz)

Fig. 11.242 is the timing diagram for the Z80A and Z80 to ADC0838 interface. Addressing the analog channel, performing conversion, and retrieving the conversion result requires 260 μs. The timing diagrams for the ADC0832 and ADC0834 converters are similar except fewer input bits are transmitted to the A/D converter. The 3-state buffer enable strobes occur 2.5 μs after the negative transition of the clock CLK signal so the conversion result bits have sufficient time to set up on the DO line before being read by the microprocessor.

Software–ADC0832, ADC0834, and ADC0838 Devices

The following software listing presents the interface software for these A/D converters. The software is written so it can be readily applied to any of the three A/D converters. Also, the software can be easily incorporated into a subroutine so the designer can access the software quickly. The software differences for these A/D converters are as follows:

	ADC0832	ADC0834	ADC0838
Send Select Bit 1	No	Yes	Yes
Send Select Bit 0	No	No	Yes

The above differences are accommodated by initializing the C and B registers correctly before accessing the software. The software listing shows the interface software routines and Tables 11.28 through 11.33 list the mux addressing to accommodate the differences.

```
                        ; Software for Z80/Z80A to ADC0838, ADC0834, ADC0832 Interface
                        ;
                        ;
        FF 00   WRITE:  EQU FFH         ;Interface write address
        FF 00   READ:   EQU FFH         ;Interface read address
                        ;
0000    06 05   START:  LD B,05H        ;ADC0838; Input bit counter = 5
                                        ;ADC0834; Input bit counter = 4
                                        ;ADC0832; Input bit counter = 3
0002    0E 15           LD C,15H        ;Load input address in C.
                                        ;C0 is sent first
                        ;
0004    3E 04   ADC83X  LD A,04H        ;CS(bar)(A2) = 1,CLK(A1) = 0,
                                        ;D1/D0(A0) = 0
0006    D3 FF           OUT (WRITE),A   ;Write these values to A/D
0008    3E 00           LD A,00H        ;Lower CS(bar)
000A    D3 F            OUT (WRITE),A
000C    CB 1F   ADC83X1: RR A           ;Prepare to load next input bit in A
000E    CB 19           RR C            ;Put next input bit in carry
0010    CB 17           RL A            ;Load next input bit in A0 and realign
                                        ;A1 & A2 so they are correct
0012    D3 FF           OUT (WRITE),A   ;Set up next bit on D1 line
0014    F6 02           OR 02H          ;Raise CLK line and clock in,
0016    D3 FF           OUT (WRITE),A   ;next bit
0018    E6 FD           AND FDH         ;Lower CLK line
001A    D3 FF           OUT (WRITE),A
001C    10 EE           DJNZ ADC83X1    ;If B>0; branch
001E    06 08           LD B,08H        ;Conversion bit counter = 8
0020    3E 02   ADC83X0: LD A,02H       ;Raise CLK & keep CS(bar) = 0
0022    D3 FF           OUT (WRITE),A
0024    3E 00           LD A,00H        ;Lower CLK line and clock out,
0026    D3 FF           OUT (WRITE),A   ;next conversion bit
0028    DB FF           IN A,(READ)     ;Put next conversion bit in A0
002A    CB 1F           RR A            ;Put conversion bit in carry
002C    CB 11           RL C            ;Put conversion bit in C0 and
                                        ;shift other conversion bits in C
002E    10 F0           DJNZ ADC83X0    ;If B>0; branch
0030    F6 04           OR 04H          ;Raise CS(bar)
0032    D3 FF           OUT (WRITE),A
0034                    END             ;Conversion result in register C
```

ADC0832 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 1)

Table 11.28 Single-ended MUX mode

START BIT	MUX ADDRESS		CHANNEL NO.		PUT DATA INTO REGISTER C
	SGL/$\overline{\text{DIF}}$	ODD/SIGN	0	1	
1	1	0	+		#03H
1	1	1		+	#0BH

Table 11.29 Differential MUX mode

START BIT	MUX ADDRESS		CHANNEL NO.		PUT DATA INTO REGISTER C
	SGL/$\overline{\text{DIF}}$	ODD/SIGN	0	1	
1	0	0	+	–	#01H
1	0	1	–	+	#09H

NOTE 1: Internally, Select 0 is low. Select 1 is high, COMMON is internally connected to ANLG GND.

ADC0834 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 2)

Table 11.30 Single-ended MUX mode

START BIT	MUX ADDRESS			CHANNEL NO.				PUT DATA INTO REGISTER C
	SGL/$\overline{\text{BIT}}$	ODD/SIGN	SELECT 1	0	1	2	3	
1	1	0	0	+				#03H
1	1	0	1			+		#0BH
1	1	1	0		+			#07H
1	1	1	1				+	#0FH

Table 11.31 Differential MUX mode

START BIT	MUX ADDRESS			CHANNEL NO.				PUT DATA INTO REGISTER C
	SGL/$\overline{\text{BIT}}$	ODD/SIGN	SELECT 1	0	1	2	3	
1	0	0	0	+	–			#01H
1	0	0	1			+	–	#09H
1	0	1	0	–	+			#05H
1	0	1	1			–	+	#0DH

NOTE 2: Internally, Select 0 is high, COMMON is internally connected to ANLG GND.

ADC0838 MUX ADDRESSING (5-BIT SHIFT REGISTER)

Table 11.32 Single-ended MUX mode

START BIT	MUX ADDRESS				ANALOG SINGLE-ENDED CHANNEL NO.									PUT DATA INTO REGISTER C
	SGL/$\overline{\text{DIF}}$	ODD/SIGN	SELECT 1	0	0	1	2	3	4	5	6	7	COM	
1	1	0	0	0	+								–	#03H
1	1	0	0	1			+						–	#13H
1	1	0	1	0					+				–	#0BH
1	1	0	1	1							+		–	#1BH
1	1	1	0	0		+							–	#07H
1	1	1	0	1				+					–	#17H
1	1	1	1	0						+			–	#0FH
1	1	1	1	1								+	–	#1FH

Table 11.33 Differential MUX mode

START BIT	MUX ADDRESS			ANALOG DIFFERENTIAL CHANNEL-PAIR NO.								PUT DATA INTO REGISTER C
	SGL/DIF	ODD/SIGN	SELECT	0		1		2		3		
			1 0	0	1	2	3	4	5	6	7	
1	0	0	0 0	+	−							#011H
1	0	0	0 1			+	−					#11H
1	0	0	1 0					+	−			#09H
1	0	0	1 1							+	−	#19H
1	0	1	0 0	−	+							#05H
1	0	1	0 1			−	+					#15H
1	0	1	1 0					−	+			#0DH
1	0	1	1 1							−	+	#13H

Circuitry – ADC0831 Device

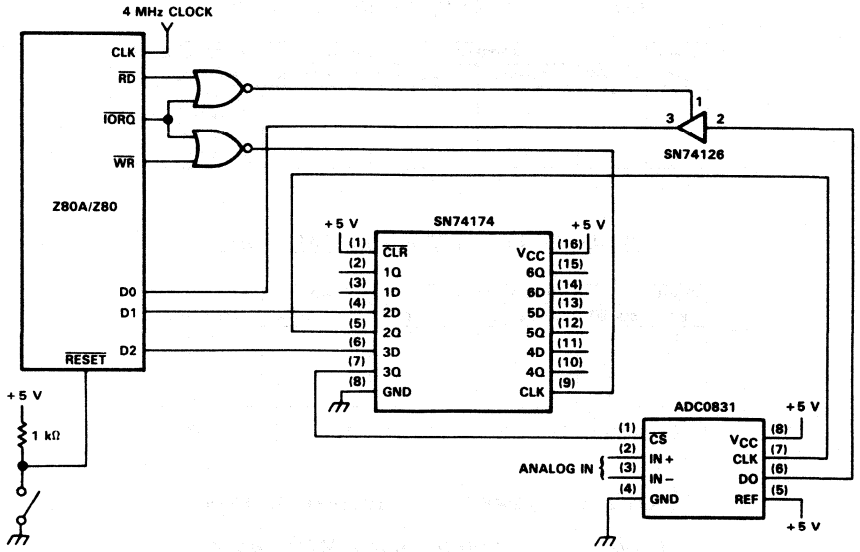


Fig. 11.243 Z80A/Z80 to ADC0831 interface circuit diagram

Fig. 11.243 shows the interconnection between the microprocessor and the ADC0831 converter. The circuitry is basically the same as for the ADC0832, ADC0834, and ADC0838 devices except that the ADC0831 device does not have a DI line.

Timing Diagram–ADC0831 Devices

Fig. 11.244 is the timing diagram for the Z80A and Z80-ADC0831 interface. Performing conversion and retrieving the conversion result requires 181 μs. The 3-state buffer enable strobes occur 2.5 μs after the negative transition of the clock CLK signal so the conversion result bits have sufficient time to set up on the DO line before these bits are read by the microprocessor.

Software–ADC0831 Device

The software listing for the ADC0831 follows. The software can be easily incorporated into a subroutine so the designer can access the software quickly. An initial A/D clock cycle must occur before the eight subsequent clock cycles extract the conversion result bits from the A/D converter.

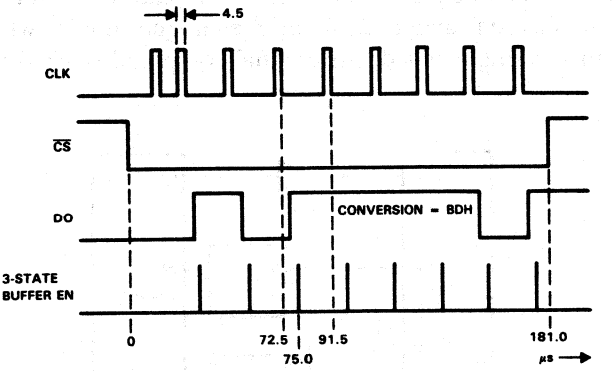


Fig. 11.244 Timing diagram for Z80A/Z80 to ADC0831 interface (microprocessor clock = 4 MHz)

```
                    ; Software for Z80/Z80A to ADC0831 Interface
                    ;
                    ;
        FF 00       WRITE:    EQU FFH          ;Interface write address
        FF 00       READ:     EQU FFH          ;Interface read address
                    ;
0000    3E 04       START:    LD A,04H         ;CS(bar)(A2) = 1,CLK(A1) = 40,
0002    D3 FF                 OUT (WRITE),A    ;Write these values to A/D
0004    3E 00                 LD A,00H         ;Lower CS(bar)
0006    D3 FF                 OUT (WRITE),A
0008    3E 02                 LD A,02H         ;Raise CLK initially
000A    D3 FF                 OUT (WRITE),A
000C    3E 00                 LD A,00H         ;Lower CLK initially
000E    D3 FF                 OUT (WRITE),A
0010    06 08                 LD B,08H         ;Conversion bit counter = 8
0012    3E 02       ADC8310:  LD A,00H         ;Raise CLK & keep CS(bar) = 0
0014    D3 FF                 OUT (WRITE),A
0016    3E 00                 LD A,00H         ;Lower CLK line and clock out,
0018    D3 FF                 OUT (WRITE),A    ;next conversion bit
001A    DB FF                 IN A,(READ)      ;Put next conversion bit in A0
001C    CB 1F                 RR A             ;Put conversion bit in carry
001E    CB 11                 RL C             ;Put conversion bit in C0 and
                                               ;shift other conversion bits in C
0020    10 F0                 DJNZ ADC8310     ;If B>0; branch
0022    3E 04                 LD A,04H         ;Raise CS(bar)
0024    D3 FF                 OUT (WRITE),A
0026                          END              ;Conversion result in register C
```

ADC0831/0832/0838 A/D Converter Interface to Rockwell 6502 Microprocessors

The ADC083X family of successive-approximation A/D converters features 8-bit resolution, microprocessor-compatible control logic, serial data communication, and 1, 2, 4, or 8 analogue inputs, which may be operated in the single-ended, differential, or pseudo-differential mode.

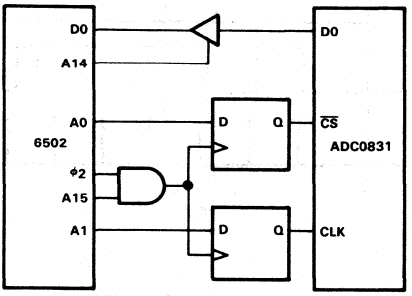


Fig. 11.245 6502 to ADC0831 interface circuit diagram

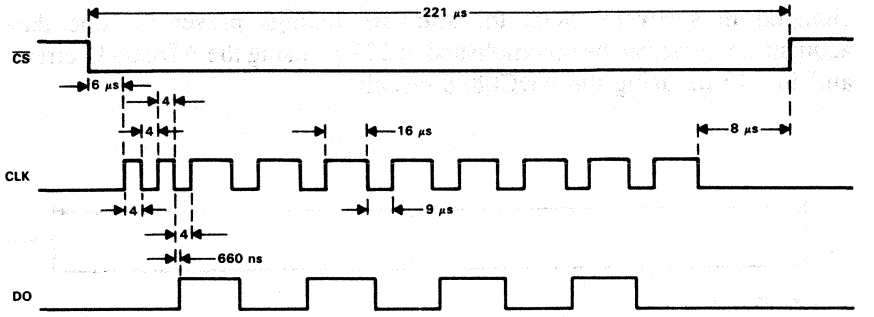


Fig. 11.246 6502 to ADC0831 interface timing diagram

A circuit for an ADC0831 device interface to a 6502 microprocessor is shown in Fig. 11.245. The timing diagram for this circuit is shown in Fig. 11.246 and the interface software is shown as a listing on page 11–266. The interface circuit operates under complete control of the software. The system clock $\phi 2$ is used as the clocking signal to latch data from the address bus into the D-type flip-flops. By writing to the correct address locations, the CS and CLK signals are generated. The address decoding scheme is quite simple; therefore, a few additional gates may be required to provide proper address decoding in more complex applications. Since the ADC0831 device has only a single analog input, no multiplexer address is required which simplifies the timing.

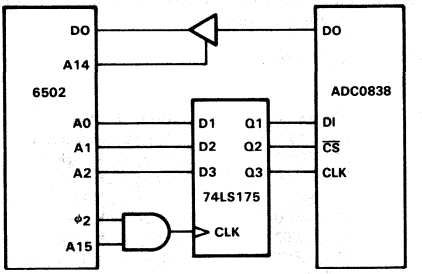


Fig. 11.247 6502 to ADC0838 interface circuit diagram

A similar circuit for a 6502 to ADC0838 interface is presented in Fig. 11.247. The timing diagram is shown in Fig. 11.248 and the interface software listing follows. The operation of this circuit is basically the same as the above circuits with the addition of the DI input for the analog multiplexer address. Operation of the ADC0838 is the same as the ADC0831 converter with the exception of the multiplexer address which can be

changed in software. With the software listings presented, one data acquisition cycle can be accomplished in 221 μs using the ADC0831 circuit and in 345 μs using the ADC0838 circuit.

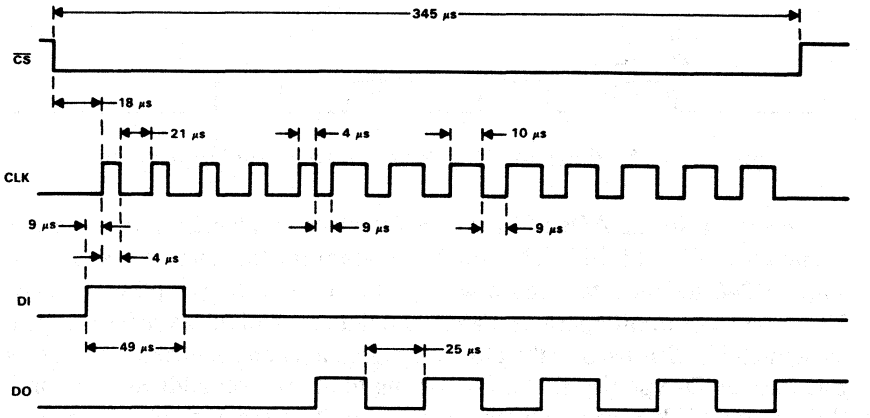


Fig. 11.248 6502 to ADC0838 interface timing diagram

```
; Software Listing for ADC0831, ADC0832, and ADC0838 to 6502 Interface
;                Register Assignments
;
ADC0831         .EQU 4000H
;
;
START:          STA $8000           ; Bring /CS low
                LDY #$08            ; Initialize bit counter
                STA $8002           ; Bring CLK high
                STA $8000           ; Bring CLK low
                STA $8002           ; Bring CLK high
                STA $8000           ; Bring CLK low
LOOP:           STA $8002           ; Bring CLK high
                LDA ADC0831         ; Read bit into accumulator
                ROR A               ; Rotate new bit into carry
                TXA                 ; Transfer result into accumulator
                ROL A               ; Rotate new bit into result
                TAX                 ; Put result back into X register
                STA $8000           ; Bring CLK low
                DEY                 ; Decrement bit counter
                BNE LOOP            ; Go back and get another bit
                STA $8001           ; Bring /CS high

;               Register Assignments
;
MUXADDRESS      .EQU 0000H
ADC0838         .EQU 4000H
;
```

```
;
START:      LDA #$C0            ;
            STA MUXADDRESS      ; Initialize muxaddress
            STA $8000           ; Bring /CS low
            LDY #$05            ; Initialize counter
LOOP1:      ROL MUXADDRESS      ; Rotate address bit into carry
            BCS SET             ; Is address bit set?
            STA $8000           ; Set up a low to be clocked in
            STA $8004           ; Bring CLK high write high to D1
            STA $8000           ; Bring CLK low
            JMP CONTINUE        ; Skip to continue
SET:        STA $8001           ; Set up a high to be clocked in
            STA $8005           ; Bring CLK high, write high to D1
            STA $8001           ; Bring CLK low
CONTINUE:   DEY                 ; Decrement Counter
            BNE LOOP1           ; Go back to clock out next bit
            STA $8004           ; Bring CLK high
            STA $8000           ; Bring CLK low
            LDY #$08            ; Initialize counter
LOOP2:      STA $8004           ; Bring CLK high
            LDA ADC0838         ; Read bit in from ADC0838
            ROR A               ; Rotate new bit into carry
            TXA                 ; Transfer result into accumulator
            ROL A               ; Rotate new bit into result
            TAX                 ; Replace result back in X register
            STA $8000           ; Bring CLK low
            DEY                 ; Decrement counter
            BNE LOOP2           ; Go back to clock next result bit
            STA $8002           ; Bring /CS high
```

ADC0831/08324/0834/0838 A/D Converter Interface to Motorola 6805 Microcomputers

This application describes the circuit configuration and the associated software that can be used to operate the 6805 single-chip microcomputer with the ADC083X converters. The converters are 8-bit successive-approximation devices with serial input/output and a configurable input multiplexer with up to eight channels. These A/D converters are designed to communicate easily with microcomputers in a serial mode. Timing diagrams are presented that describe the interaction between the microcomputer and the A/D converter.

The ADC0832, ADC0834, and ADC0838 devices may be software configured in either the single-ended or differential input mode. Also, the differential inputs can be interchanged through software manipulation. This interface features:

1. Low-cost A/D devices
2. Direct connection between the A/D converter and the microcomputer
3. Fast conversion and communication between the A/D converter and the microcomputer
4. Remote control advantage of serial A/D converters.

Circuitry–ADC0832, ADC0834, and ADC0838 Devices

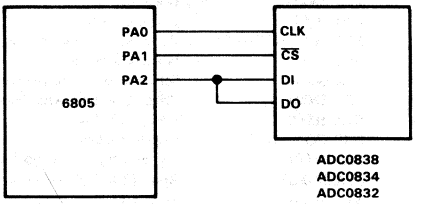


Fig. 11.249 6805 to ADC0832, ADC0834 and ADC0838 interface circuit diagram

Fig. 11.249 shows the interconnection betwen the microprocessor and the ADC0832, ADC0834, and ADC0838 A/D converters. This interconnection is identical for all three A/D converters. A microcomputer port pin can be saved by connecting the A/D converter's DI and DO pins to only one microcomputer pin. However, the designer must be careful to define this port pin as an input pin immediately after the initial input bits have been transmitted to the A/D converter. In so doing, the designer will ensure that the pin will be configured as an input when the data conversion mode bits are received by the microcomputer.

Timing Diagram–ADC0838 Device

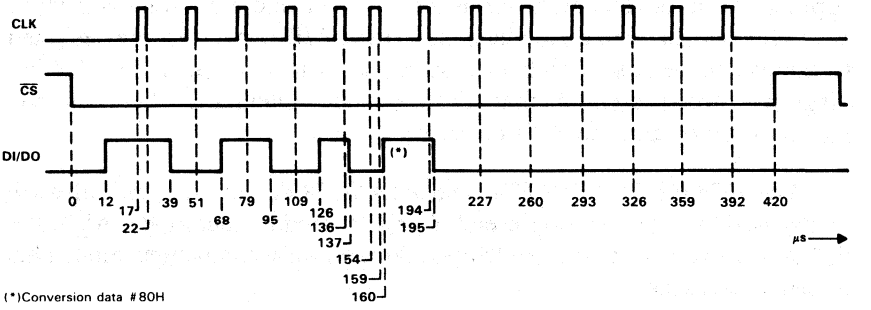


Fig. 11.250 Timing diagram for 6805 to ADC0838 interface

Fig. 11.250 is the timing diagram for the 6805 to ADC0838 interface. Addressing the analog channel, performing conversion, and retrieving the conversion result required 420 μs. The timing diagrams for the ADC0832

and ADC0834 devices are similar except fewer input bits are transmitted to the A/D converter. The CLC instruction in the software listing provides sufficient delay to allow the A/D converters to set up the conversion result bits on the DO line before they are read by the microcomputer.

Software–ADC0832, ADC0834, and ADC0838 Devices

The software listing describes the software routines for these A/D converters. The software is written so it can be readily applied to any of the three A/D converters as a subroutine so the designer can access the software quickly.

The software differences for these devices are listed below:

	ADC0832	ADC0834	ADC0838
Send Select Bit 1	No	Yes	Yes
Send Select Bit 0	No	No	Yes

The above differences are accommodated by initializing the accumulator and index register correctly before accessing the subroutine. The software listing and Tables 11.34 through 11.39 describe how to accommodate these differences.

```
                        ; Software for ADC0832, ADC0834, and ADC0838 to 6805 Interface
                        ;
        ORG  $100H      ; Initialize the starting address for the software
        LDA  #07H       ; Initialize Port A, I/O Pins
        STA  DDR A      ; Configures Port A.0 to A.2 as output pins
        LDA  #A8H       ; For MUX addressing, select appropriate START BIT,
                        ; SGL/DIF(bar), ODD/SIGN, SELECT BIT 1, SELECT BIT 0
                        ; & Load into Acc. Designer can select any MUX
                        ; addressing mode by changing the immediate data.
                        ; See Table 1 to 6 to select the desired MUX
                        ; addressing mode)
        LDX  #05H       ; Set bit counter to 5 or 4 or 3. Designer can
                        ; select any A/D chip out of ADC0838, ADC0834,
                        ; ADC0832.
                        ; Set the immediate date as follows:
                        ;  ADC0838 → Set bit counter to 5
                        ;  ADC0834 → Set bit counter to 4
                        ;  ADC0832 → Set bit counter to 3
        BSR  S83X       ; Load conversion mode bits into Data Input on
                        ; the A/D & Acquire Conversion result into Acc
                        ; from Data out.
                        ; Subroutine S83X
S83X    BCLR 0, Port A  ; Lower CLK
        BCLR 1, Port A  ; Lower Chip Select
S83XRO  ROL  A          ; Shift Conversion Mode bit into C
        BCS  S83XBC     ; If Carry is set ; BRANCH
        BCLR 2, Port A  ; Set 083X DI line to 0
        JMP  S83XJM     ; Go & Raise CLK
S83XBC  BSET 2, Port A  ; Set 083 DI line to 1
S83XJM  BSET 0, Port A  ; Raise CLK
```

```
        BCLR 0, Port A          ; Lower CLK
        DECX                    ; Decrement Counter
        BNE  S83XRO             ; Do 5 or 4 or 3 times
        BCLR 2, DDR A           ; Configures Port A.2 as an input pin
        LDX  RO, #08H           ; Set bit counter to 8
S83XI   BSET 0, Port A          ; Raise CLK
        BCLR 0, Port A          ; Lower CLK
        CLC                     ; Initialize C = 0
        BRCLR 2, Port A, S83X   ; If 083X Data out = 0; BRANCH
        SEC                     ; 083X Data out = 1; Set C = 1
S83XB   LDA  $20H               ; Get Serial Buffer
        ROL  A                  ; Shift Data out Bit into Serial Buffer
        STA  $20H               ; Store Serial Buffer
        DECX                    ; Decrement Counter
        BNE  S83XI              ; Do 8 times
        BSET 1, Port A          ; Raise Chip Select
        LDA  $20H               ; Conversion Data is in Accumulator
        RTS                     ; Return from Subroutine
        END                     ;
```

ADC0832 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 1)

Table 11.34 Single-ended MUX mode

START	MUX ADDRESS		CHANNEL NO.		PUT DATA
BIT	SGL/$\overline{DIF}$	ODD/SIGN	0	1	INTO A_{CC}
1	1	0	+		#C0H
1	1	1		+	#D0H

Table 11.35 Differential MUX mode

START	MUX ADDRESS		CHANNEL NO.		PUT DATA
BIT	SGL/$\overline{DIF}$	ODD/SIGN	0	1	INTO A_{CC}
1	0	0	+	–	#80H
1	0	1	–	+	#90H

NOTE 1: Internally, Select 0 is low. Select 1 is high, COMMON is internally connected to ANLG GND.

ADC0834 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 2)

Table 11.36 Single-ended MUX mode

START	MUX ADDRESS			CHANNEL NO.				PUT DATA
BIT	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	INTO A_{CC}
1	1	0	0	+				#C0H
1	1	0	1			+		#D0H
1	1	1	0		+			#E0H
1	1	1	1				+	#F0H

Table 11.37 Differential MUX mode

START	MUX ADDRESS			CHANNEL NO.				PUT DATA
BIT	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	INTO A_{CC}
1	0	0	0	+	–			#80H
1	0	0	1			+	–	#90H
1	0	1	0	–	+			#A0H
1	0	1	1			–	+	#B0H

NOTE 2: Internally, Select 0 is high, COMMON is internally connected to ANLG GND.

ADC0838 MUX ADDRESSING (5-BIT SHIFT REGISTER)

Table 11.38 Single-ended MUX mode

START BIT	MUX ADDRESS				ANALOG SINGLE-ENDED CHANNEL NO.								PUT DATA INTO A_{CC}
	SGL/$\overline{DIF}$	ODD/SIGN	SELECT 1	SELECT 0	0	1	2	3	4	5	6	7	COM	
1	1	0	0	0	+								–	#C0H
1	1	0	0	1			+						–	#C8H
1	1	0	1	0					+				–	#D0H
1	1	0	1	1							+		–	#D8H
1	1	1	0	0		+							–	#E0H
1	1	1	0	1				+					–	#E8H
1	1	1	1	0						+			–	#F0H
1	1	1	1	1								+	–	#F8H

Table 11.39 Differential MUX mode

START BIT	MUX ADDRESS				ANALOG DIFFERENTIAL CHANNEL-PAIR NO.								PUT DATA
	SGL/$\overline{DIF}$	ODD/SIGN	SELECT 1	SELECT 0	0	1	2	3	4	5	6	7	A_{CC}
1	0	0	0	0	+	–							#80H
1	0	0	0	1			+	–					#88H
1	0	0	1	0					+	–			#90H
1	0	0	1	1							+	–	#98H
1	0	1	0	0	–	+							#A0H
1	0	1	0	1			–	+					#A8H
1	0	1	1	0					–	+			#B0H
1	0	1	1	1							–	+	#C8H

Circuitry – ADC0831 Device

Fig. 11.251 shows the interconnection between the microcomputer and the ADC0831 converter. To assure that the conversion result bits can be read by the microcomputer, the port pin which is assigned to the DO line of the A/D converter must be configured as an input.

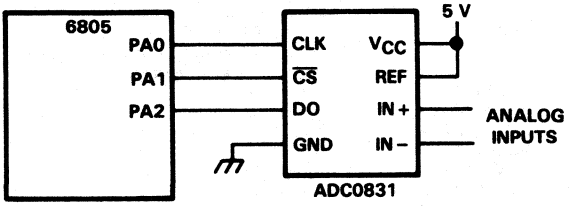


Fig. 11.251 6805 to ADC0831 interface circuit diagram

Timing Diagram–ADC0831 Device

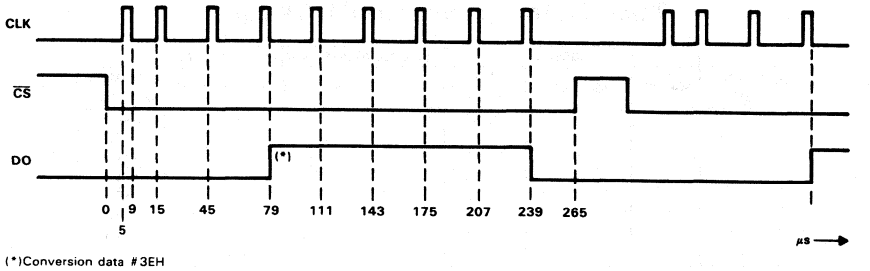


Fig. 11.252 Timing diagram for 6805 to ADC0831 interface

The timing diagram for the 6805 to ADC0831 interface is shown in Fig. 11.252. Performing conversion and retrieving the conversion result requires 265 μs. The CLC instruction in the software listing provides sufficient delay to allow the A/D converters to set up the conversion result bits on the DO line before they are read by the microcomputer.

Software–ADC0831 Device

The following software listing presents the routines for this converter. The software is written as a subroutine so the designer can access the software quickly. An initial A/D clock cycle must occur before the eight subsequent clock cycles can be used to extract the conversion result bits from the A/D converter.

```
                                ; Software for 6805 to ADC0831 Interface
                                ;
        ORG  $100H              ; Initialize the starting address for the software
        BSR  S831               ; Branch to the Subroutine
                                ; Load 9 clocks to CLK input on the A/D
                                ; Acquire the Conversion Result into Accumulator.
                                ;
                                ; Subroutine S831
S831    LDA  #03H               ; Initialize Port A, I/O pins
        STA  DDR A              ; Configures Port A.0 and A.1 as output pins
                                ; and Port A.2 as an input pin
        BCLR 0, Port A          ; Lower CLK
        BCLR 1, Port A          ; Lower Chip Select
        BSET 0, Port A          ; Raise CLK
        BCLR 0, Port A          ; Lower CLK
        LDX  #08H               ; Set bit counter to 8
S831I   BSET 0, Port A          ; Raise CLK
        BCLR 0, Port A          ; Lower CLK
        CLC                     ; Initialize C=0
        BRCLR 2, Port A, S831B  ; If 0831 Data out=0; BRANCH
```

```
        SEC             ; 0831 Data out = 1; Set C = 1
S831B   LDA  $20H       ; Get Serial Buffer
        ROL  A          ; Shift Data out Bit into Serial Buffer
        STA  $20H       ; Store Serial Buffer
        DECX            ; Decrement Counter
        BNE  S831I      ; Do 8 times
        BSET 1, Port A  ; Raise Chip Select
        LDA  $20H       ; Conversion Data is in Accumulator
        RTS             ; Return from Subroutine
        END             ;
```

ADC0831/0832/0834/0838 A/D Converter Interface to Motorola 6800, 6802, 6809 and 6809E Microprocessors

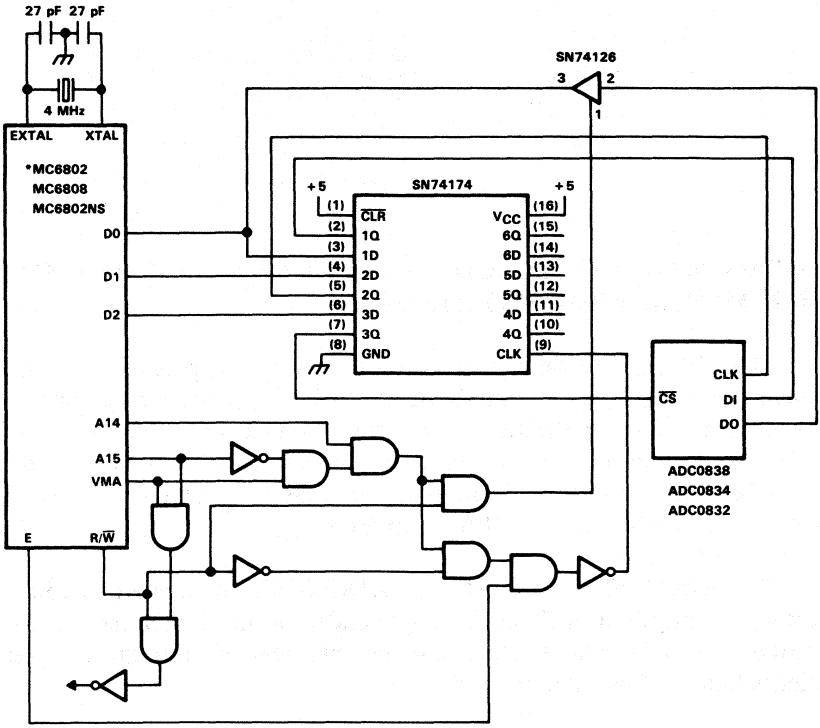
The ADC083X family of 8-bit successive-approximation A/D converters is designed to communicate easily with microprocessors in a serial mode. This application shows the circuit configurations and the software that can be used to operate this family of microprocessors with the ADC083X converters. Timing diagrams show the interaction between the microprocessor and the A/D converter.

The ADC0832, ADC0834, and ADC0838 A/D converters can be software configured in either the single-ended or differential input mode. Further, the differential inputs can be interchanged through software manipulation. This interface features:

1. Low-cost A/D converter
2. Minimum circuitry
3. Fast conversion and communication between the A/D converter and the microprocessor
4. Remote control advantages of serial A/D converters.

Circuitry—ADC0832, ADC0834, and ADC0838 Devices

Fig. 11.253 shows the interconnection between the microprocessor and the ADC0832, ADC0834, and ADC0838 A/D converters. The interconnection is identical for all three converters. The microprocessor DO pin can be used to transmit to and receive digital data from the ADC083X DI and DO pins, respectively. The SN74126 3-state buffer output is in the high-impedance state except during a microprocessor read operation. The SN74174 quad D-type flip-flop is used to synchronize and slow down the write/read communication between the microprocessor and ADC083X converter family so the timing requirements are satisfied. The configuration of AND gates and inverters assures that the flip-flops receive the A/D input data on the negative edge of the E clock. Thus, the flip-



*See Table 11.40 for information about other Motorola microprocessors.

Fig. 11.253 Circuit diagram for the ADC0832, ADC0834, and ADC0838 interface

flops will receive the correct data. Also, this configuration enables the 3-state buffer and allows the microprocessor to read the A/D DO line when the read/write and address lines are correctly activated. Table 11.40 provides information for adapting the circuit of Fig. 11.253 for use with other members of the 6800 family of microprocessors.

Table 11.40 Adapting Fig. 11.253 and Fig. 255 to other 6800 family microprocessors

MICROPROCESSOR	CRYSTAL OSCILLATOR AND E PIN
6800	ϕ_2 is equivalent to the 6802/6809 E pin. See Fig. 11.253 and Fig. 11.255
6802/6808/6802NS	See Fig. 11.253 and Fig. 11.255
6809	See Fig. 11.253 and Fig. 11.255
6809E/68HC09E	See Fig. 11.253 and MC6809 Data Sheet for Clock Generator information

Timing Diagram–ADC0838 Device

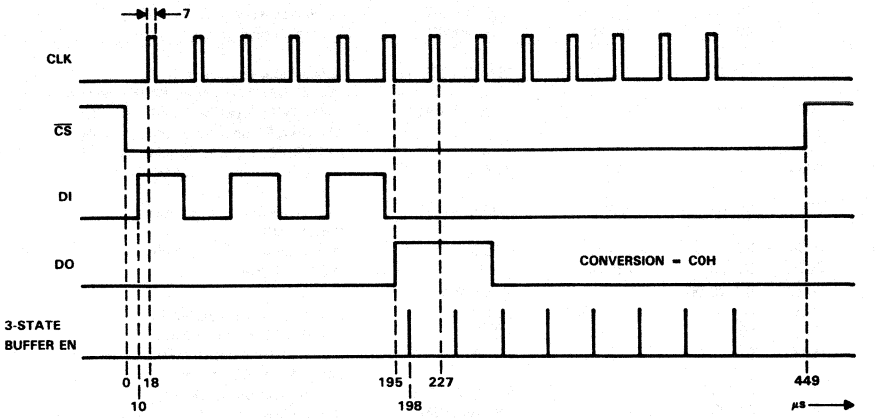


Fig. 11.254 Timing diagram for 6802–ADC0838 interface (1 MHz microprocessor clock cycle)

Fig. 11.254 presents the timing diagram for the 6802 to ADC0838 interface. Addressing the analog channel, performing conversion, and retrieving the conversion result requires 449 μs. The timing diagrams for the ADC0832 and ADC0834 converters are similar except fewer input bits are transmitted to the A/D converter. The 3-state buffer enable strobes occur 3 μs after the negative transition of the CLK signal so the conversion result bits have sufficient time to set up on the DO line before these bits are read by the microprocessor.

Software–ADC0832, ADC0834, and ADC0838 Devices

The software listing for these converters follows this discussion. The software is written for any of the three A/D converters. Also, the software can be easily incorporated into a subroutine so the designer can access the software quickly. The software differences for these A/D converters are as follows:

	ADC0832	ADC0834	ADC0838
Send Select Bit 1	No	Yes	Yes
Send Select Bit 0	No	No	Yes

The above differences are accommodated by correctly clearing the B accumulator and a byte of RAM that is used as a bit counter before accessing the software. The software listing and Tables 11.41 through 11.46 describe how to accommodate these differences.

```
                        ; Software for 680X Family to ADC0838, ADC0834, ADC0832 Interface
                        ;
                        ;
        40 00           WRITE:      EQU 4000H       ;Interface write address
        40 00           READ:       EQU 4000H       ;Interface read address
                        ;
0000    C6 15           START:      LDAB #15H       ;Load input address in B, See
                                                    ;Tables 11-25 to 11-30, B0 is sent first
0002    86 0                        LDAA #10H       ;ADC0838; Input bit counter = 10H
                                                    ;ADC0834; Input bit counter = 08H
                                                    ;ADC0832; Input bit counter = 04H
0004    97 00           STAA        0000H           ;Store bit counter in RAM
                        ;
0006    86 04           ADC83X      LDAA #04H       ;CS(bar)(A2) = 1, CLK(A1) = 0,
                                                    ;DI/D0(A0) = 0
0008    B7 40 00        STAA        WRITE           ;Write these values to A/D
000B    86 00           LDAA        #00H            ;Lower CS(bar)
000D    B7 40 00                    STAA WRITE
0010    46              ADC83XI:    RORA            ;Prepare to load next bit in A
0011    56                          RORB            ;Put next input bit in carry
0012    49                          ROLA            ;Load next input bit in A0 and re-
                                                    ;align A1 & A2 so they are correct
0013    B7 40 00                    STAA WRITE      ;Set up next bit on DI line
0016    8A 02                       ORAA #02H       ;Raise CLK line and clock in,
0018    B7 40 00                    STAA WRITE      ;next bit
001B    84 FD                       ANDA #FDH       ;Lower CLK line
001D    B7 40 00                    STAA WRITE
0020    76 00 00                    ROR 0000H       ;Rotate bit counter
0023    24 EB                       BCC ADC83XI     ;If carry = 0; branch
0025    86 80                       LDAA #80H       ;Conversion bit counter = 8
0027    97 00                       STAA 000H       ;Store bit counter in RAM
0029    86 2            ADC83XO:    LDAA #02H       ;Raise CLK & keep CS(bar) = 0
002B    B7 40 00                    STAA WRITE
002E    86 00                       LDAA #00H       ;Lower CLK line and clock out,
0030    B7 40 00                    STAA WRITE      ;next conversion bit
0033    B6 40 00                    LDAA READ       ;Put next conversion bit in A0
0036    46                          RORA            ;Put conversion bit in carry
0037    59                          ROLB            ;Put conversion bit in B0 and
                                                    ;shift other conversion bits in B
0038    76 00 00                    ROR 0000H       ;Rotate bit counter
003B    24 EC                       BCC ADC83XO     ;If carry = 0; branch
003D    86 04                       LDAA #04H       ;Raise CS(bar)
003F    B7 40 00                    STAA WRITE
0042                                END             ;Conversion result is in B accumulator
```

ADC0832 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 1)

Table 11.41 Single-ended MUX mode

START BIT	MUX ADDRESS		CHANNEL NO.		PUT DATA INTO ACCUMULATOR B
	SGL/$\overline{DIF}$	ODD/SIGN	0	1	
1	1	0	+		#03H
1	1	1		+	#0BH

Table 11.42 Differential MUX mode

START BIT	MUX ADDRESS		CHANNEL NO.		PUT DATA INTO
	SGL/$\overline{DIF}$	ODD/SIGN	0	1	ACCUMULATOR B
1	0	0	+	−	#01H
1	0	1	−	+	#05H

NOTE 1: Internally, Select 0 is low. Select 1 is high, COMMON is internally connected to ANLG GND.

ADC0834 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 2)

Table 11.43 Single-ended MUX mode

START BIT	MUX ADDRESS			CHANNEL NO.				PUT DATA INTO
	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	ACCUMULATOR B
1	1	0	0	+				#03H
1	1	0	1			+		#0BH
1	1	1	0		+			#07H
1	1	1	1				+	#0FH

Table 11.44 Differential MUX mode

START BIT	MUX ADDRESS			CHANNEL NO.				PUT DATA INTO
	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	ACCUMULATOR B
1	0	0	0	+	−			#01H
1	0	0	1			+	−	#09H
1	0	1	0	−	+			#05H
1	0	1	1			−	+	#0DH

NOTE 2: Internally, Select 0 is high, COMMON is internally connected to ANLG GND.

ADC0838 MUX ADDRESSING (5-BIT SHIFT REGISTER)

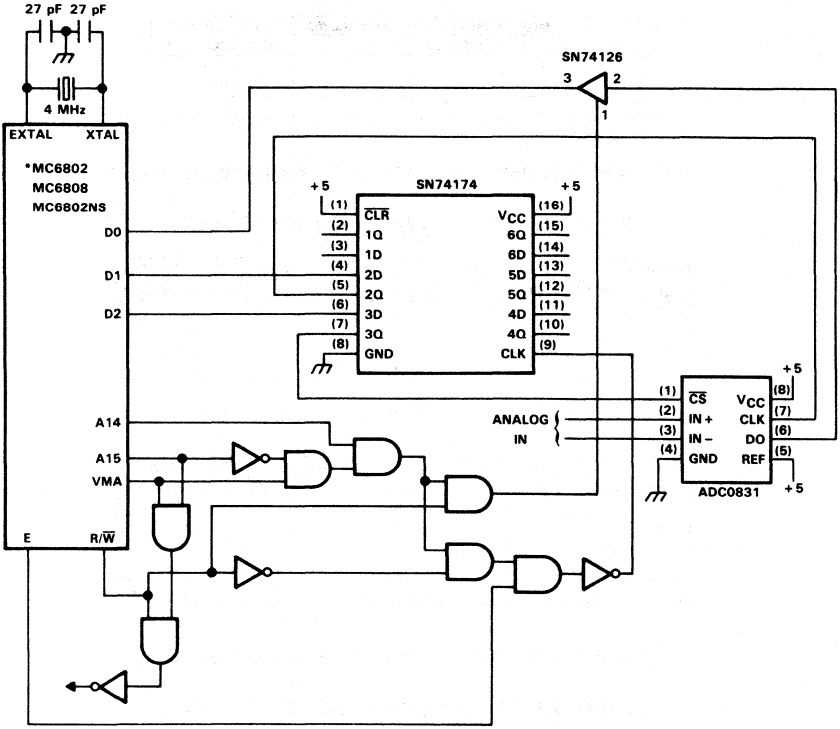
Table 11.45 Single-ended MUX mode

START BIT	MUX ADDRESS			ANALOG SINGLE-ENDED CHANNEL NO.									PUT DATA INTO ACCUMULATOR B
	SGL/$\overline{DIF}$	ODD/SIGN	SELECT 1 0	0	1	2	3	4	5	6	7	COM	
1	1	0	0 0	+								−	#03H
1	1	0	0 1			+						−	#13H
1	1	0	1 0					+				−	#0BH
1	1	0	1 1							+		−	#1BH
1	1	1	0 0		+							−	#07H
1	1	1	0 1				+					−	#17H
1	1	1	1 0						+			−	#0FH
1	1	1	1 1								+	−	#1FH

Table 11.46 Differential MUX mode

START BIT	MUX ADDRESS			ANALOG DIFFERENTIAL CHANNEL-PAIR NO.								PUT DATA INTO ACCUMULATOR B
	SGL/$\overline{DIF}$	ODD/SIGN	SELECT 1 0	0	1	2	3	4	5	6	7	
1	0	0	0 0	+	−							#01H
1	0	0	0 1			+	−					#11H
1	0	0	1 0					+	−			#09H
1	0	0	1 1							+	−	#19H
1	0	1	0 0	−	+							#05H
1	0	1	0 1			−	+					#15H
1	0	1	1 0					−	+			#0DH
1	0	1	1 1							−	+	#13H

Circuitry–ADC0831 Device



*See Table 11.40 for information about other Motorola microprocessors.

Fig. 11.255 Circuit diagram for the ADC0831 interface

Fig. 11.255 shows the interconnection between the microprocessor and the ADC0831 converter. The circuitry is basically the same as for the ADC0832, ADC0834 and ADC0838 interface except the ADC0831 does not have a DI line. Table 11.40 provides information that will help the designer adapt the circuit of Fig. 11.255 to other members of the 6800 family of microprocessors.

Timing Diagram–ADC0831 Device

Fig. 11.256 shows the timing diagram for the 6802 to ADC0831 interface. Performing conversion and retrieving the conversion result requires 283 μs. The 3-state buffer enable strobes occur 3 μs after the negative transition of the CLK signal so the conversion result bits have sufficient time to set up on the DO line before these bits are read by the microprocessor.

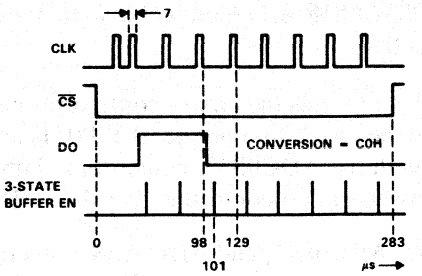


Fig. 11.256 Timing diagram for 6802 to ADC0831 interface (1 MHz microprocessor clock cycle)

Software–ADC0831 Device

The following software listing provides the software routines for this converter. The software can be easily incorporated into a subroutine so the designer can access the software quickly. An initial A/D clock cycle must occur before the eight subsequent clock cycles can be used to extract the conversion result bits from the A/D converter.

```
                        ; Software for 680X Family to ADC0831 Interface
                        ;
                        ;
        40 00           WRITE:    EQU 4000H       ;Interface write address
        40 00           READ:     EQU 4000H       ;Interface read address
                        ;
0000    86 04           START:    LDAA #04H       ;CS(bar)(A2) = 1, CLK(A1) = 0
0002    B7 40 00                  STAA WRITE      ;Write these values to A/D
0005    86 00                     LDAA #00H       ;Lower CS(bar)
0007    B7 40 00                  STAA WRITE
000A    86 02                     LDAA #02H       ;Raise CLK initially
000C    B7 40 00                  STAA WRITE
000F    86 00                     LDAA #00H       ;Lower CLK initially
0011    B7 40 00                  STAA WRITE
0014    86 80                     LDAA #80H       ;Conversion bit counter = 8
0016    97 00                     STAA 0000H      ;Store bit counter in RAM
0018    86 02           ADC8310:  LDAA #02H       ;Raise CLK and keep CS(bar) = 0
001A    B7 40 00                  STAA WRITE
001D    86 00                     LDAA #00H       ;Lower CLK line and clock out,
001F    B7 40 00                  STAA WRITE      ;next conversion bit
0022    B6 40 00                  LDAA READ       ;Put next conversion bit in A0
0025    46                        RORA            ;Put conversion bit in carry
0026    59                        ROLB            ;Put conversion bit in B0 and
                                                  ;shift other conversion bits in B
0027    76 00 00                  ROR 0000H       ;Rotate bit counter
002A    24 EC                     BCC ADC8310     ;If carry = 0; branch
002C    86 04                     LDAA #04H       ;Raise CS(bar)
002E    B7 40 00                  STAA WRITE
0031                              END             ;Conversion result is in B accumulator
```

ADC0831/0832/0834/0838 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers

This application presents the circuit configurations and the associated software that can be used to operate the 8051 and 8052 family of microcontrollers with the ADC083X converters. Timing diagrams show the interaction between microcontroller and the A/D converter.

The ADC0832, ADC0834, and ADC0838 converters can be software configured in either the single-ended or differential input mode. Also, the differential ± inputs may be interchanged through software manipulation. This serial interface features:

1. Low-cost A/D converter
2. Direct connection between the A/D converter and the microcontroller
3. Fast conversion and communication between the A/D converter and the microcontroller
4. Remote control advantages of serial A/D converters.

The 8051 and 8052 microcontroller family consists of the following:

8031AH	8052AH
8032AH	80C51
8051AH	8751H

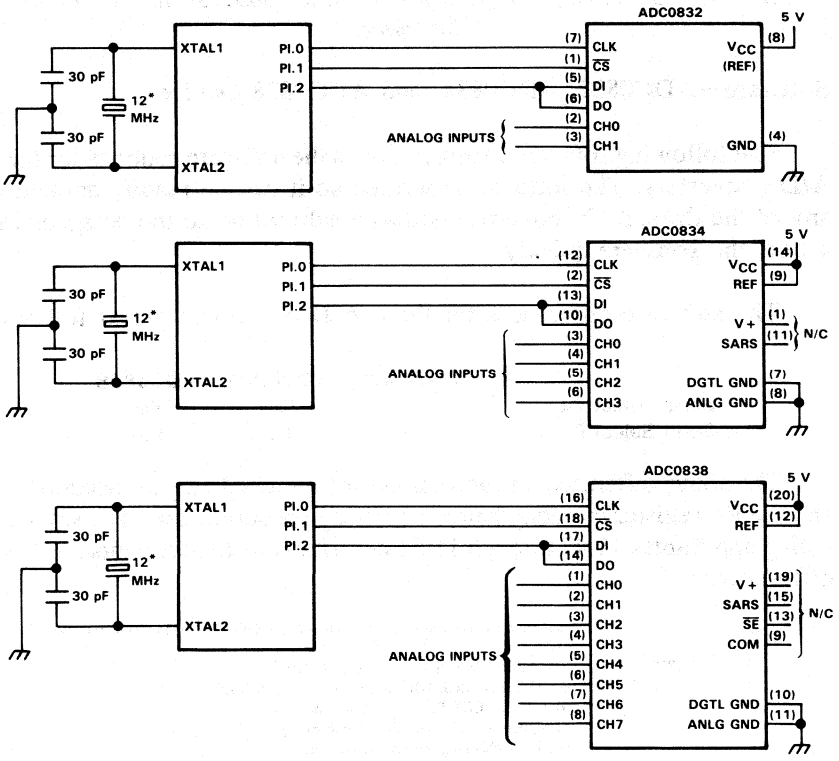
Circuitry–ADC0832, ADC0834, and ADC0838 Devices

Fig. 11.257 shows the interface between the microprocessor and the ADC0832, ADC0834, and ADC0838 A/D converters. The interconnection is identical for these converters. A microcontroller port pin can be saved by connecting the A/D converter's DI and DO pins to one microprocessor port pin. However, the designer must be careful to define this port pin as an input pin immediately after the initial input bits have been transmitted to the A/D converter. By doing this, the designer will ensure that the pin will be configured as an input when the data conversion bits are received by the microcontroller.

Timing Diagram–ADC0838 Device

Fig. 11.258 presents the timing diagram for the 8051 and 8052 to ADC0838 interface. Addressing the analog channel, performing conversion, and retrieving the conversion result requires only 150 μs. The timing

diagrams for the ADC0834 and ADC0832 converters are similar except fewer input bits are transmitted to the A/D converter. The CLR instruction in the software listing provides sufficient delay to allow the A/D converter to set up the conversion result bits on the DO line before they are read by the microcontroller.



*11 MHz for 8048/49 Family

Intel 8051/52 Family: 8031AH, 8051AH, 8032AH, 8052AH, 8751AH, 80C51

Intel 8048/49 Family: 8048AH, 8748H, 8035AHL, 8049AH, 8749H, 8039AHL, 8050AH, 8040AHL, 80C49

Fig. 11.257 The Intel 8051/52 or 8048/49 to ADC0832, ADC0834, and ADC0838 interface circuit diagram

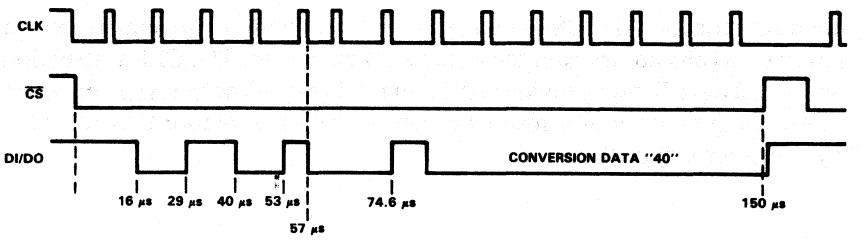


Fig. 11.258 Timing diagram for the Intel 8051/52 to ADC0838 interface

Software–ADC0832, ADC0834, and ADC0838 Devices

The following software listing provides the software routines for these A/D converters. The software is written so it can be readily applied to any of the three A/D converters and as a subroutine so the designer can access the software quickly.

The software differences for these A/D converters are as follows:

	ADC0832	ADC0834	ADC0838
Send Select Bit 1	No	Yes	Yes
Send Select Bit 0	No	No	Yes

The above differences are accommodated by initializing the accumulator and the R6 register correctly before accessing the subroutine. The software listing and Tables 11.47 through 11.52 describe how to accommodate these differences.

```
                    ; Software for Intel 8051/52 or 8048/49 to ADC0832, 0834, 0838 Interface
                    ;
MOV A, #A8H         ; For MUX addressing, select appropriate
                    ; START BIT, SGL/DIF(bar), ODD/SIGN, SELECT
                    ; BIT 1, SELECT BIT 0 and Load into Acc.
                    ; Designer can select any MUX addressing
                    ; mode by changing the immediate data
                    ; (See Table 1 to 6 to select the desired
                    ; MUX addressing mode)
MOV R6, #05H        ; Set bit counter to 5
                    ; Designer can select any A/D chip out of
                    ; ADC0838, ADC0834, ADC0832
                    ; Set the immediate data as follows:
                    ;  ADC0838 → Set bit counter to 5
                    ;  ADC0834 → Set bit counter to 4
                    ;  ADC0832 → Set bit counter to 3
ACALL S83X          ; Load conversion mode bits into Data Input
                    ; on the A/D chip & Acquire Conversion
                    ; Result into Accumulator from Data Output
                    ;
                    ; Subroutine S83X
CLR P1.0            ; Lower CLK
```

```
S83X    CLR P1.1          ; Lower Chip Select
S83XRL  RLC A             ; Shift Conversion Mode bit into C
        JC S83XJC         ; If Carry is set ; BRANCH
        ANL P1, #FBH      ; Set 083X DI line to 0
        SJMP S83XSJ       ; Go & Raise CLK
S83XJC  ORL P1, #04H      ; Set 083X DI line to 1
S83XSJ  NOP               ; Delay to Set up Mode bit
        CPL P1.0          ; Raise CLK
        NOP               ; Delay to slow CLK
        CLR P1.0          ; Lower CLK
        DJNZ R6, 83XRL    ; Do 5, 4, or 3 times
        ORL P1, #04H      ; Configure P1.2 as an input pin
        MOV R0, #08H      ; Set bit counter to 8
S83XI   CPL P1.0          ; Raise CLK
        NOP               ; Delay to slow CLK
        CLR P1.0          ; Lower CLK
        CLR C             ; Initialize C = 0
        JNB P1.2, S83XJ   ; If 083X Data out = 0; BRANCH
                          ; This pin must be in input mode

        CPL C             ; 083X Data out = 1; Set C = 1
S83XJ   MOV A,R1          ; Get Serial Buffer
        RLC A             ; Shift Data out Bit into Serial Buffer
        MOV R1,A          ; Store Serial Buffer
        DJNZ R0, S83XI    ; Do 8 times
        CPL P1.1          ; Raise Chip Select
        MOV A,R1          ; Conversion Data is in Accumulator
        RET               ;
        END;
```

ADC0832 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 1)

Table 11.47 Single-ended MUX mode

START	MUX ADDRESS		CHANNEL NO.		PUT DATA
BIT	SGL/$\overline{DIF}$	ODD/SIGN	0	1	INTO A_{CC}
1	1	0	+		#C0H
1	1	1		+	#D0H

Table 11.48 Differential MUX mode

START	MUX ADDRESS		CHANNEL NO.		PUT DATA
BIT	SGL/$\overline{DIF}$	ODD/SIGN	0	1	INTO A_{CC}
1	0	0	+	−	#80H
1	0	1	−	+	#A0H

NOTE 1: Internally, Select 0 is low. Select 1 is high, COMMON is internally connected to ANLG GND.

ADC0834 MUX ADDRESSING (5-BIT SHIFT REGISTER) (See Note 2)

Table 11.49 Single-ended MUX mode

START	MUX ADDRESS			CHANNEL NO.				PUT DATA
BIT	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	INTO A_{CC}
1	1	0	0	+				#C0H
1	1	0	1			+		#D0H
1	1	1	0		+			#E0H
1	1	1	1				+	#F0H

Table 11.50 Differential MUX mode

START BIT	MUX ADDRESS			CHANNEL NO.				PUT DATA
	SGL/$\overline{BIT}$	ODD/SIGN	SELECT 1	0	1	2	3	INTO A_{CC}
1	0	0	0	+	–			#80H
1	0	0	1			+	–	#90H
1	0	1	0	–	+			#A0H
1	0	1	1			–	+	#B0H

NOTE 2: Internally, Select 0 is high, COMMON is internally connected to ANLG GND.

ADC0838 MUX ADDRESSING (5-BIT SHIFT REGISTER)

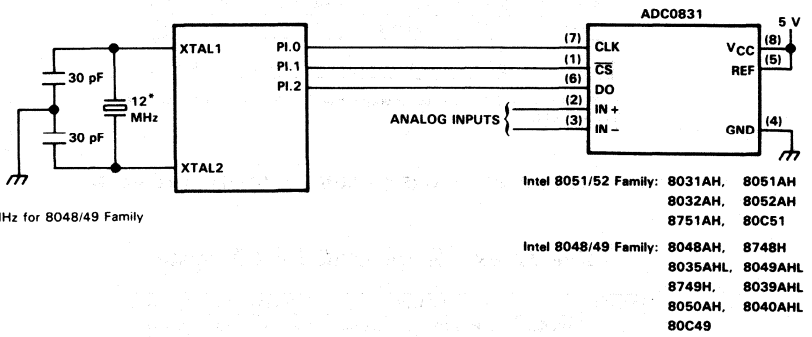
Table 11.51 Single-ended MUX mode

START BIT	MUX ADDRESS		SELECT		ANALOG SINGLE-ENDED CHANNEL NO.								PUT DATA
	SGL/$\overline{DIF}$	ODD/SIGN	1	0	0	1	2	3	4	5	6	7	COM	INTO A_{CC}
1	1	0	0	0	+								–	#C0H
1	1	0	0	1			+						–	#C8H
1	1	0	1	0					+				–	#D0H
1	1	0	1	1							+		–	#D8H
1	1	1	0	0		+							–	#E0H
1	1	1	0	1				+					–	#E8H
1	1	1	1	0						+			–	#F0H
1	1	1	1	1								+	–	#F8H

Table 11.52 Differential MUX mode

START BIT	MUX ADDRESS		SELECT		ANALOG DIFFERENTIAL CHANNEL-PAIR NO.								PUT DATA
	SGL/$\overline{DIF}$	ODD/SIGN	1	0	0	1	2	3	4	5	6	7	A_{CC}
1	0	0	0	0	+	–							#80H
1	0	0	0	1			+	–					#88H
1	0	0	1	0					+	–			#90H
1	0	0	1	1							+	–	#98H
1	0	1	0	0	–	+							#A0H
1	0	1	0	1			–	+					#A8H
1	0	1	1	0					–	+			#B0H
1	0	1	1	1							–	+	#B8H

Circuitry–ADC0831 Device



*11 MHz for 8048/49 Family

Fig. 11.259 the Intel 8051/52 or 8048/49 to ADC0831 interface circuit diagram

Fig. 11.259 shows the interconnection between the microcontroller and the ADC0831 converter. To assure that the conversion result bits can be read by the microcontroller, the designer must configure the microcontroller port pin which is assigned to the DO line of the A/D converter as an input.

Timing Diagram–ADC0831 Device

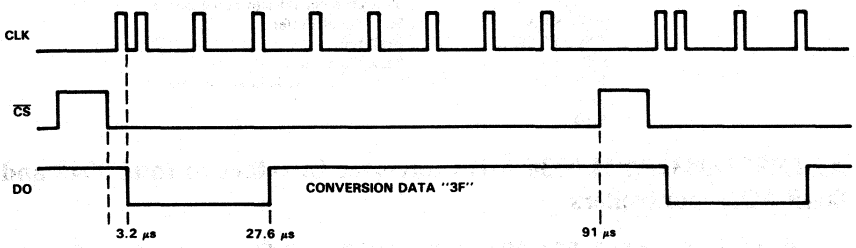


Fig. 11.260 Timing diagram for the Intel 8051/52 to ADC0831 interface

Fig. 11.260 shows the timing diagram for the 8051 and 8052 to ADC0831 interface. Performing conversion and retrieving the conversion result requires only 91 μs. The CLR C instruction in the software listing provides sufficient delay to allow the A/D converter to set up the conversion result bits on the D line before they are read by the microcontroller.

Software–ADC0831 Device

The following software listing provides the software routines for this A/D converter. The software is written as a subroutine so the designer can access the software quickly. An initial A/D clock cycle must occur before the eight subsequent clock cycles can be used to extract the conversion result bits from the A/D converter.

```
                                ; Software for Intel 8051/52 Interface to ADC0831
                                ;
                                ;
                                ;
                                ;
        ACALL S831              ; Load 9 clocks to CLK input on the A/D
                                ; chip & Acquire Conversion result
                                ; into Accumulator
                                ;
                                ; Subroutine S831
S831    ORL P1,04H              ; Configure P1.2 as an input pin
        CLR P1.0                ; Lower CLK
        CLR P1.1                ; Lower Chip Select
```

```
        CPL P1.0        ; Raise CLK
        NOP             ; Delay to slow CLK
        CLR P1.0        ; Lower CLK
        MOV R0, #8H     ; Set bit counter to 8
S831I   CPL P1.0        ; Raise CLK
        NOP             ; Delay to slow CLK
        CLR P1.0        ; Lower CLK
        CLR C           ; Initialize C = 0
        JNB P1.2,S831J  ; If 0831 Data out = 0 ; BRANCH
                        ; This pin must be in input mode
        CPL C           ; 0831 Data out = 1 ; Set C = 1
S831J   MOV A,R1        ; Get Serial Buffer
        RLC A           ; Shift Data out Bit into Serial Buffer
        MOV R1,A        ; Store Serial Buffer
        DJNZ R0,S831I   ; Do 8 times
        CPL P1.1        ; Raise Chip Select
        MOV A,R1        ; Conversion Data is in Accumulator
        RET             ;
        END             ;
```

ADC0831/0832/0834/0838 A/D Converter Interface to Intel 8048 and 8049 Microcontrollers

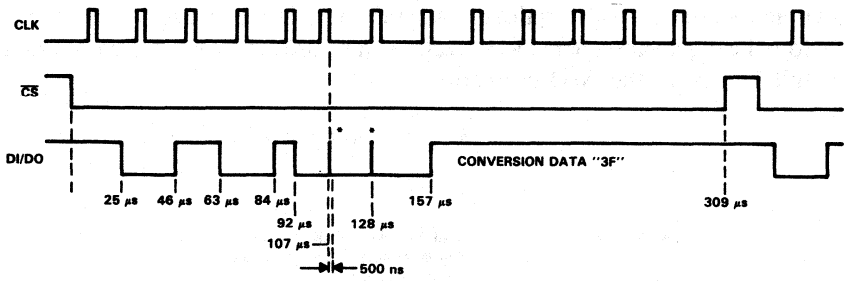
Refer to the **ADC0831/0832/0834/0838 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers** application for additional information regarding these devices. The 8048 and 8049 microcontroller family consists of the following:

8035AHL	8045AH	8748AH
8039AHL	8049AH	8749H
8040AHL	8050AH	80C49

Circuitry–ADC0838, ADC0834, and ADC0832

See Fig. 11.257 in the previous application for circuitry also used with this family of converters.

Timing Diagram–ADC0838 Device



*Designer may see some spikes on the Data Output that do not effect the reading of the conversion results, since the microprocessor reads the DO pin after the spikes occur (see description of timing diagram).

Fig. 11.261 Timing diagram for the Intel 8048/49 to ADC0838 interface

Fig. 11.261 presents the timing diagram for the 8048 and 8049 to ADC0838 interface. Addressing the analog channel, performing conversion, and retrieving the conversion result requires 309 μs. The timing diagrams for the ADC0834 and ADC0832 converters are similar except fewer input bits are transmitted to the A/D converter.

The NOP and CLR C instructions in the software listing provide sufficient delay to allow the A/D converter to set up the conversion result bits on the DO line before they are read by the microcontroller. Also, these instructions guarantee that occasional spikes on the DI/DO line during clock edges will not cause erroneous readings. These spikes appeared during our testing and may have been due to our test circut layout. However, these spikes are harmless because they appear during clock edges and the delay time of the NOP and CLR C instructions is longer than the delay time of the spikes. Therefore, the DO line can be read accurately. If the designers feel uncomfortable with these spikes, they may connect the DI/DO line to a microcontroller port other than the port to which the A/D clock line is connected. For example, the A/D converter chip select and clock lines might be connected to port 1 of the microcontroller, while the DI/DO line might be connected to port 2. This use of two different ports eliminates the occasional spike action.

Software – ADC0838, ADC0834, and ADC0832 Devices

See the **ADC0831/0832/0834/0838 A/D Converter interface to Intel 8051 and 8052 Microcontrollers** application for more information on these A/D converters. When reading that application, substitute the software listing for this application.

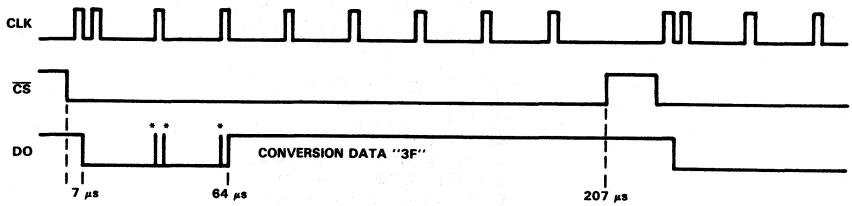
```
                    ; Software for Intel 8051 and 8052 to ADC8032,
                      ADC8034 and ADC8038 Interface
MOV A, #A8H         ; For MUX addressing, select appropriate
                    ; START BIT, SGL/DIF(bar), ODD/SIGN, SELECT
                    ; BIT 1, SELECT BIT 0 and Load into Acc.
                    ; Designer can select any MUX addressing

                    ; mode by changing the immediate data
                    ; (See Table 1 to 6 to select the desired
                    ; MUX addressing mode)
MOV R6, #05H        ; Set bit counter to 5
                    ; Designer can select any A/D chip out of
                    ; ADC0838, ADC0834, ADC0832
                    ; Set the immediate data as follows:
                    ; ADC0838 → Set bit counter to 5
                    ; ADC0834 → Set bit counter to 4
                    ; ADC0832 → Set bit counter to 3
CALL S83X           ; Load conversion mode bits into Data Input
                    ; on the A/D chip & Acquire Conversion
                    ; Result into Accumulator from Data Output
                    ;
```

```
                                ;
                                ; Subroutine S83X
          ANL P1,#FEH           ; Lower CLK
S83X      ANL P1,#FDH           ; Lower Chip Select
S83XRL    RLC A                 ; Shift Conversion Mode bit into C
          JC S83XJC             ; If Carry is set: BRANCH
          ANL P1,#FBH           ; Set 83X DI line to 0
          JMP S83XSJ            ; Go & Raise CLK
S83XJC    ORL P1,#04H           ; Set 83X DI line to 1
S83XSJ    NOP                   ; Delay to Set up Mode bit
          ORL P1,#01H           ; Raise CLK
          NOP                   ; Delay to slow CLK
          ANL P1,#FEH           ; Lower CLK
          DJNZ R6,S83XRL        ; Do 5, 4, or 3 times
          MOV R0,#08H           ; Set bit counter to 8
S83XI     ORL P1,#01H           ; Raise CLK
          NOP                   ; Delay to slow CLK
          ANL P1,#FEH           ; Lower CLK
          NOP                   ; To guarantee that occasional spike on DI/DO
                                ; at clock edges do not cause erroneous readings
          CLR C                 ; Initialize C=0
          IN A,P1               ; Get Port 1
          CPL A                 ; Complement Accumulator
          JB2 S83XJ             ; If 083X Data out=1: BRANCH
          CPL C                 ; 083X Data out=1: Set C=1
S83XJ     MOVA,R1               ; Get Serial Buffer
          RLC A                 ; Shift Data out Bit into Serial Buffer
          MOV R1,A              ; Store Serial Buffer
          DJNZ R0,S83XI         ; Do 8 times
          ORL P1,#02H           ; Raise Chip Select
          MOV A,R1              ; Conversion Data is in Accumulator
          RET                   ;
          END                   ;
```

Circuitry–ADC0831 Device

See the ADC0831/0832/0834/0838 A/D Converter Interface to Intel 8051 and 8052 Microcontrollers application.

Timing Diagram–ADC0831 Device



*Designer may see some spikes on the Data Output that do not effect the reading of the conversion results, since the microporcessor reads the DO pin after the spikes occur (see description of timing diagram).

Fig. 11.262 Timing diagram for the Intel 8048/49 to ADC0831 interface

Fig. 11.262 shows the timing diagram for the 8048 and 8049 to ADC0831 interface. Performing conversion and retrieving the conversion result requires only 207 μs. The CLR instruction in the software listing provides sufficient delay to allow the A/D converter to set up the conversion result bits on the DO line before they are read by the microprocessor.

Software–ADC0831 Device

The following software listing provides the software routines for this A/D converter. The software is written as a subroutine so the designer can access the software quickly. An initial A/D clock cycle must occur before the eight subsequent clock cycles can be used to extract the conversion result bits from the A/D converter.

```
                        ; Software for Intel 8048/49 Interface to ADC0831
                        ;
                        ;
                        ;
        CALL S831       ; Load 9 clocks to CLK input on the A/D
                        ; chip & Acquire Conversion result
                        ; into Accumulator
                        ;
                        ; Subroutine S831
S831    ORL P1,#04H     ; Configure P1.2 as an input pin
        ANL P1,#FEH     ; Lower CLK
        ANL P1,#FDH     ; Lower Chip Select
        ORL P1,#01H     ; Raise CLK
        NOP             ; Delay to slow CLK
        ANL P1,#FEH     ; Lower CLK
        MOV R0,#08H     ; Set bit counter to 8
S831I   ORL P1,#01H     ; Raise CLK
        NOP             ; Delay to slow CLK
        ANL P1,#FEH     ; Lower CLK
        NOP             ; To guarantee that occasional spikes on
                        ; DI/DO line at clock edges do not cause
                        ; erroneous readings
        CLR C           ; Initialize C=0
        IN A,P1         ; Get Port 1
        CPL A           ; Complement Accumulator
        JB2 S831J       ; If 0831 Data out=1 ; BRANCH
        CPL C           ; 0831 Data out=1 ; Set C=1
S831J   MOV A,R1        ; Get Serial Buffer
        RLC A           ; Shift Data out Bit into Serial Buffer
        MOV R1,A        ; Store Serial Buffer
        DJNZ R0,S831I   ; Do 8 times
        ORL P1,#02H     ; Raise Chip Select
        MOV A,R1        ; Conversion Data is in Accumulator
        RET             ;
        END             ;
```

Section 12

Peripheral Drivers

INTRODUCTION

Peripheral drivers are general-purpose integrated circuits that can be used to interface between TTL, MOS, and CMOS logic levels and higher voltage, higher current components. These include lamps, relays, solenoids, data transmission lines, and motors (Fig. 12.1).

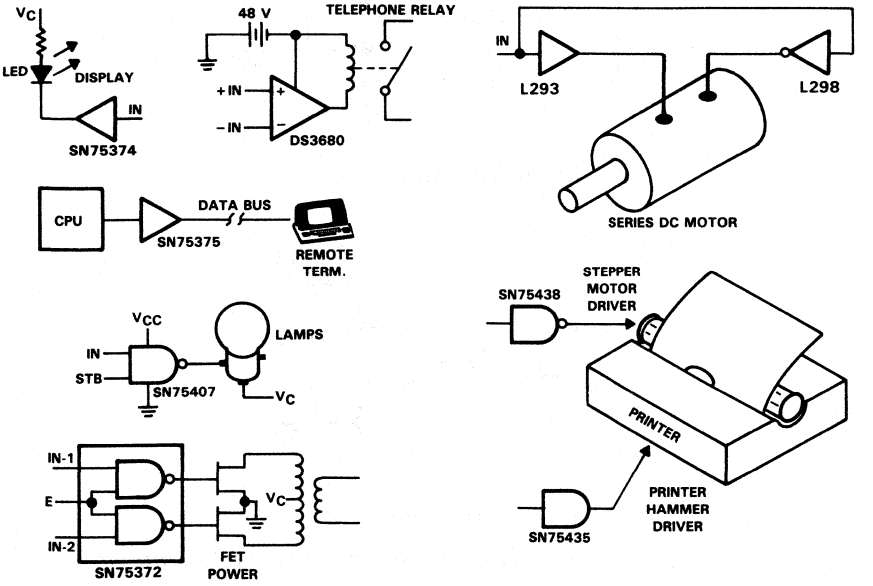


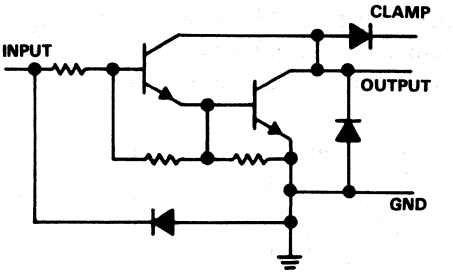
Fig. 12.1 Typical uses for peripheral drivers.

DEVICE CONSIDERATIONS

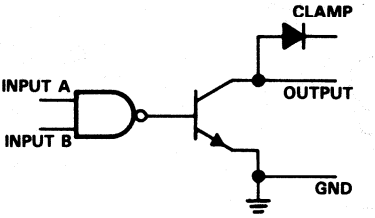
Basic Configurations

Integrated circuit devices have been developed to allow direct, single IC interfacing from logic levels to components requiring additional drive power. These peripheral drivers usually include output transistors to provide the required drive capability, preceded by logic level shifting

circuitry. Level shifting is accomplished by integrated resistors and diodes or logic gates. Fig. 12.1 illustrates two basic configurations of peripheral drivers. One uses resistor and/or diode level shifting [Fig. 12.2(a)], and the other uses logic gate level shifting [Fig. 12.2(b)].



(a) Resistor/diode input



(b) Gate input

Fig. 12.2 Basic peripheral driver input configurations

Typical Requirements

Power

Peripheral drivers are used in situations which require them to control a significant amount of power. In practice they are not able to achieve the ideal of zero internal power dissipation and as a result most IC peripheral driver packages are designed to handle at least 1 W. Packages with copper lead frames are often used to improve power handling capability. The small dual-in-line 8-pin packages with a copper

lead frame will typically handle over 1·4 W at 25°C. The 14 and 16-pin packages with copper lead frames are rated at 2 W.

Voltage

Voltage capability ranges from a minimum of 15 V to 100 V. Care must be exercised in the selection of drivers for switching applications. The switching voltage output limits are generally less than the maximum dc standoff voltage [$V_{(BR)CER}$]. For example, a typical peripheral driver with a standoff voltage rating 30 V may not be suitable for use as a 24-V relay driver, even if an output clamp diode is used. The high level output voltage after switching (V_{OH}) is typically 20 V for this device. The output voltage swing of 24 V or 25 V would result in secondary breakdown and device latch-up. This is a destructive condition which could result in device failure. The correct device for this application would have a V_{OH} rating of 30 V. Peripheral devices that have at their inputs internal control gates require a 5 V supply in addition to any higher supply required for the output device.

Current

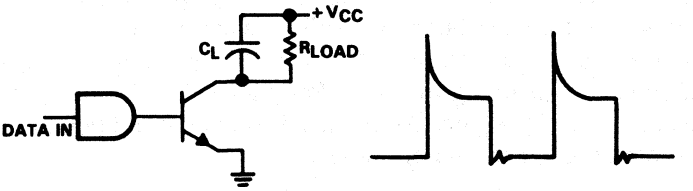
Peripheral drivers are designed to operate at output current levels of 100 mA to 2 A. As with output voltage, device output current selection should be done with care. Most devices will have a continuous output current rating and peak current rating. Peak current ratings are specified for a maximum on time of 10 ms and a duty cycle of 50% or less. NOTE: The peak current level of integrated circuit drivers should not be exceeded, no matter how short the time or how low the duty cycle. Although average power dissipation may be within limits if the on times and duty cycles are very short, chip surface metal migration may occur with any current above the rated peak level. Metal migration results in destruction of the chip surface metal. This is associated with the output emitter contacts, and eventually causes device failure. For example, short current spikes associated with charging a capacitive load could seem to have no immediate effect on the device. However, if the peak current level is exceeded for even a short time, a small amount of deterioration occurs and the device will exhibit long-term failure.

Speed

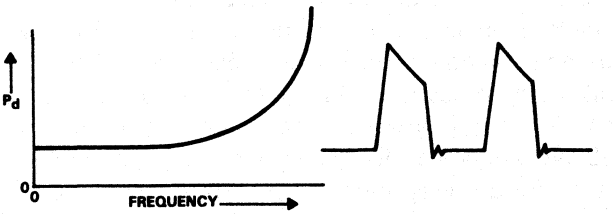
Peripheral drivers can be used as switches and are therefore operating in a dc or very low frequency mode. Other applications, such

as memory clock drivers require speeds as high as 10 MHz. At low or high data rates or dc operation, the device must never exceed its voltage, current or power limits.

High-speed operation can, in some applications, result in excessive power dissipation. High-speed power dissipation can (and should) be limited to improve operating reliability. Turn-on transients are one form of excessive power that can be controlled to some extent. Fig. 12.3 illustrates the effects of transients on power dissipation as frequences are increased.



(a) Peak surge adds 80% to average DC power



(b) Peak surge adds to average DC power

Fig. 12.3 Transient effects on power dissipation

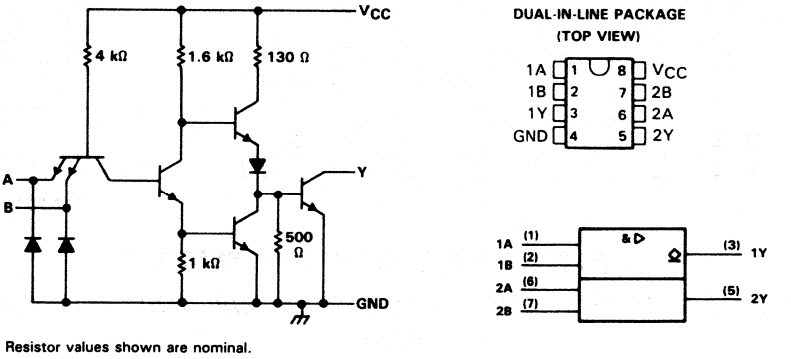
Logic

Generally these devices are used to interface between logic level signals and circuits requiring more drive power. Most peripheral driver inputs are compatible with TTL voltage levels. The input resistance of some devices is high allowing compatibility with low level CMOS, MOS, and low power Schottky TTL, as well as standard TTL.

PERIPHERAL DRIVER DEVICES AND APPLICATIONS

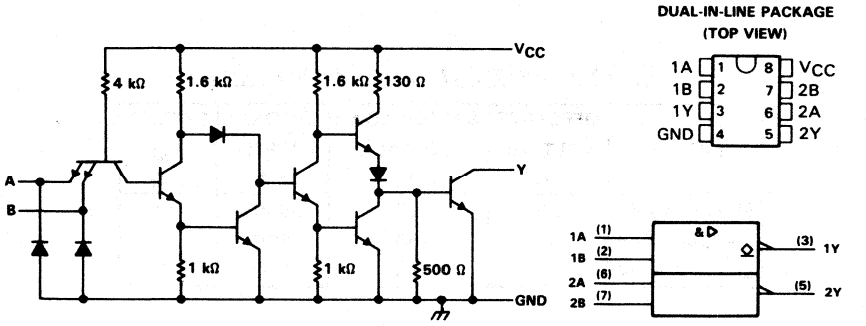
SN75451, SN75461 and SN75471

A wide variety of peripheral drivers are available for today's design engineer. The need for different voltages, currents and speeds has resulted in the development of several families of peripheral drivers. The first of these, with the same block diagram and basic function, include the SN75451, SN75461 and SN75471 series. Figs. 12.4 and 12.5 list the basic schematics, logic symbols and package pinouts for these four families.



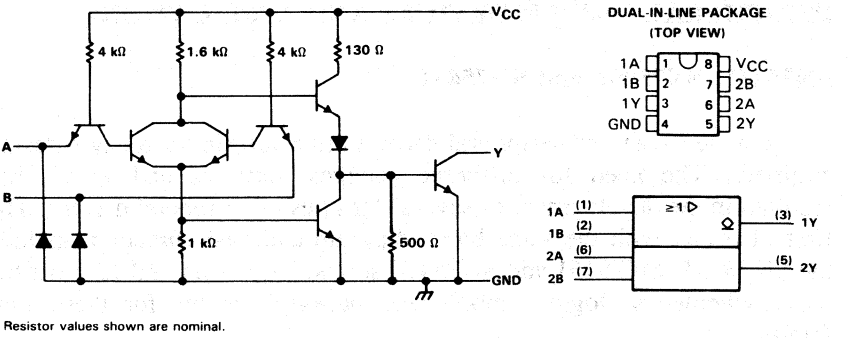
Resistor values shown are nominal.

(a) SN55/75451, SN55/75461, SN55/75471



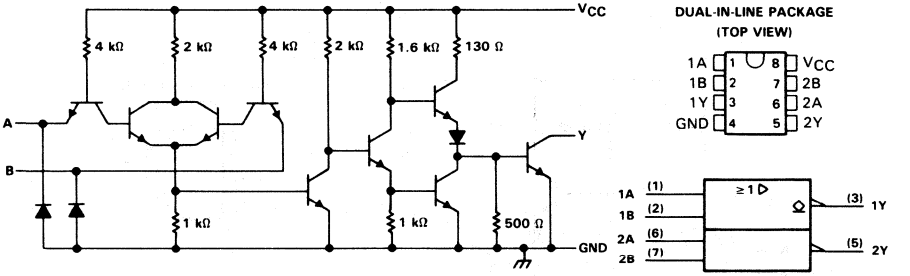
(b) SN55/75452, SN55/75462, SN55/75472

Fig. 12.4 SN75/451/461 and 471 series (AND, NAND) schematics and diagrams



Resistor values shown are nominal.

(a) SN55/75453 , SN55/75463, SN55/75473



(b) SN55/75454, SN55/75464, SN55/75474

Fig. 12.5 SN75431/451/461 and 471 series (OR, NOR) schematics and diagrams

Table 12.1 SN75451 series device features

FAMILY	OFFSTATE VOLTAGE (V)	SWITCH LOAD VOLTAGE (V)	OUTPUT CURRENT (mA)	TYPICAL t_{pd} (ns)
SN75451-454	30	20	300	20
SN75461-464	40	30	300	35
SN75471-474	70	55	300	35

Features and comparisons of the SN754X1 series are listed in Table 12.1.

Driving Tungsten Filament, or Equivalent Incandescent Lamps

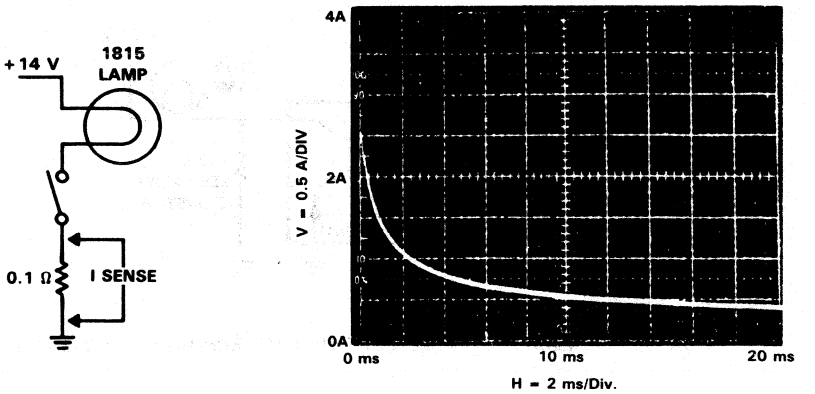


Fig. 12.6 Results of lamp surge current test

Incandescant lamps have a characteristic that can overstress monolithic integrated circuit drivers. The resistance of the typical incandescent bulb varies significantly with changes in filament temperature. For example, a typical general-purpose bulb, type 1815, is rated at 14 V and 200 mA operating current. It might be assumed that a monolithic driver rated at 300-mA continuous and 500-mA peak surge would be adequate for driving a lamp requiring only 200-mA drive current. The problem becomes evident if the turn-on surge current is monitored. Type 1815 bulbs have a (cold) turn-on peak surge current of up to 10 times their steady-state value, possibly as high as 2·7 A. Fig. 12.6 illustrates a test of surge current for a type 1815 bulb. To overcome the problem of high turn-on surge current the designer has two alternatives: Select a driver with a 3 A surge capability, or do something to limit the surge current.

Using SN75451B to Drive Lamps

Several methods can be employed to limit surge currents when using peripheral drivers. One method uses keep-alive resistors as shown in Fig. 12.7. These resistors maintain the off-state current at about 50% of the normal steady-state level and result in a standby light intensity of about 10% of normal. Steady-state current for the particular bulbs being used was 250 mA and the warm-up current was 120 mA. With the keep-

alive resistor, the surge current is limited to only 500 mA or twice the normal steady-state value.

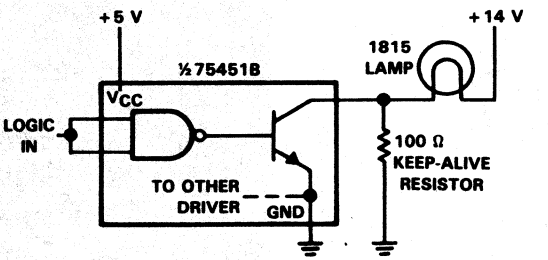


Fig. 12.7 Limiting surge current with keep-alive resistors

SN75446 and SN75476 Series

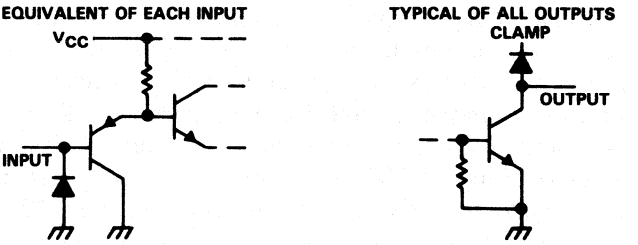


Fig. 12.8 Schematics of device inputs and outputs

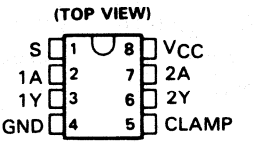


Fig. 12.9 SN75446 and SN75476 series package pinouts

The SN75446-9 and SN75476-9 series are dual drivers with PNP input transistors, illustrated in Fig. 12.8, resulting in compatibility with TTL, MOS and low level CMOS logic. Built-in clamp diodes allow clamping of inductive loads without external diodes. The SN75446-9 and the SN75476-9 series are characterised to provide 350 mA and 300 mA

continuous output current respectively both with 70-V standoff and 55-V inductive switching capability. These devices are especially suited for driving inductive loads such as relays, solenoids, printer hammers and motors. Fig. 12.9 shows the package and functional similarities between these devices and their interchangability.

Incandescent Lamp Driving with SN75447

A dual stage circuit that will provide consistent performance is the delayed turn-on configuration shown in Fig. 12.10.

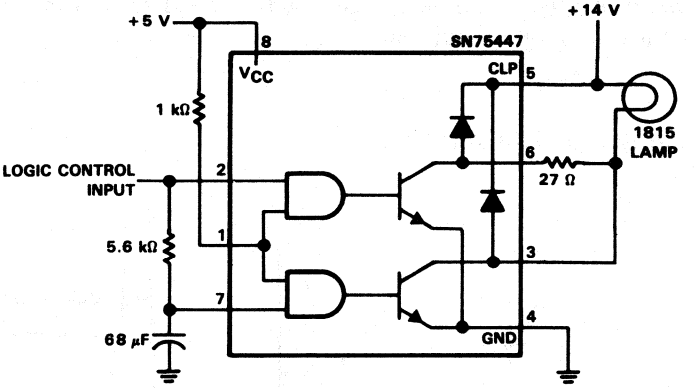


Fig. 12.10 Two stage, full power delayed, surge limiting circuit.

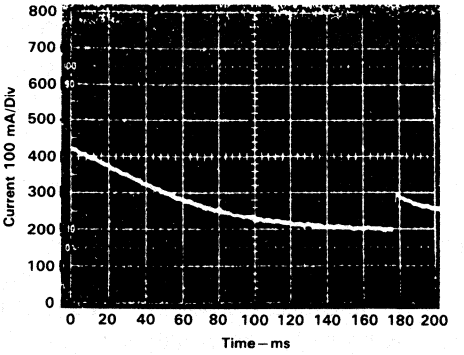
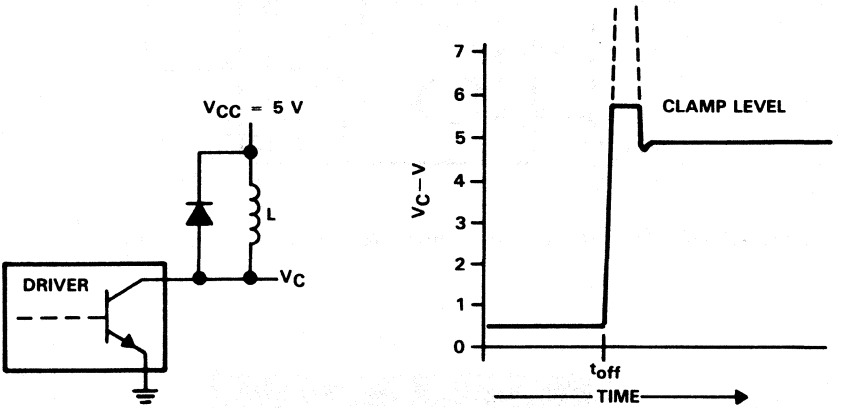


Fig. 12.11. Delayed turn-on surge current vs time.

In this application, limited lamp current is allowed for a period of 150 to 200 ms while the lamp warms up and increases its resistance. After this delay full power is applied, but the bulb resistance is higher and only a small surge current is experienced. A 27-Ω resistor is used in series with the initial switch, limiting the peak current to under 500 mA. The second driver channel has an RC delay network at its input consisting of a 5.6-kΩ resistor and a 68-μF capacitor. The resulting RC time constant is about 350 ms. For a logic input voltage of 3.5 V, with the gate input threshold level of 1.4 V, the second channel will be turned on when its input voltage reaches 1.4 V or 40% of the logic "1" level. A 40% amplitude level will occur at 0.5 RC time constants. Thus the delayed full power turn-on will occur at about 180 ms as illustrated in Fig. 12.11. In this application, the initial surge current is limited to 425 mA and the delayed turn-on surge is only 300 mA.

Relay and Solenoid Drivers



(a) Basic diode clamping *(b) Clamped output transition*

Fig. 12.12 Diode clamping of driver V_C

When peripheral drivers are used to drive inductive loads such as relays and solenoids, special attention should be given to the device's switching voltage, as well as current and power capabilities. Most peripheral drivers have their maximum switching voltage specified and care must be taken to avoid exceeding this parameter. Often clamp diodes are used to prevent excessively high voltages at the driver output when driving inductive loads. Fig. 12.12 illustrates the basic application and

the typical waveform of a diode clamped output. Many peripheral drivers now have built-in inductive clamp diodes.

SN75447 Print Hammer Driver Application

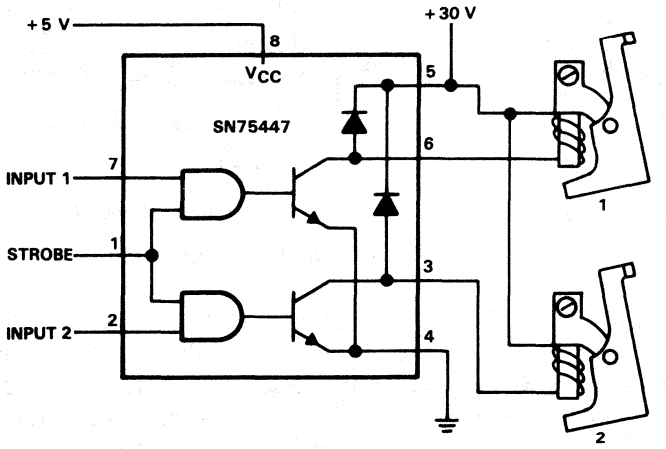
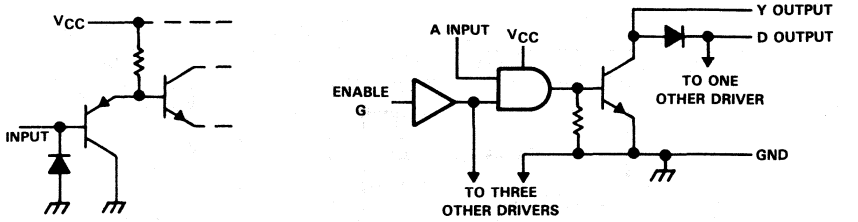


Fig. 12.13 SN75447 dual hammer driver

In this application, the hammer solenoids require operation from a 30-V supply with peak driver currents of 300 mA. More than one hammer may be actuated at a time. The SN75447 peripheral driver was selected for this application (Fig. 12.13). This dual driver will switch up to 50 V (after conducting 300 mA) without latching or breaking down. In addition, the output transient suppression diodes are included in the driver. Since both outputs of the dual driver may be on at the same time, its power handling capabilities must be compared with actual worst case operating conditions. Maximum 5 V supply power dissipation for the SN75447 is equal to 5.25 V times 18 mA (the maximum specified chip supply voltage and current) or 94.5 mW. Output power will be the product of the worst case V_{OL} and the peak I_{OL} expected. For this device the output power dissipation, with both outputs on, will be 2(300 mA)(0.65 V) or 390 mW. Power handling capability for this device is 1380 mW at 25°C. The package has a derating factor of 11.1 mW/°C which yields an 880-mW power capability at the maximum operating temperature of 70°C. Since the total (chip supply power of

94.5 mW and the output power of 390 mW) is 484.5 mW, it is well within the 880 mW capability at 70°C, making the SN75447 a good choice.

SN75436, SN75437A and SN75438



(a) Equivalent of each input *(b) Functional block diagram*

Fig. 12.14 Basic device schematics

The SN75436, SN75437A and SN75438 are quad, gate input, peripheral drivers. They are designed for driving loads requiring relatively high power (15 to 35 W). Each device features four open-collector output drivers with common enables. These devices are designed for use as relay drivers, printer hammer or other types of solenoid drivers, lamp drivers, motor drivers, data line drivers, and memory drivers. The basic device schematic diagrams, Fig. 12.14 show some of the features of this series. PNP transistors provide high-impedance inputs for TTL and CMOS compatibility. Low power logic control circuitry results in less than 26 mW standby power. Open-collector output transistors provide low resistance saturated outputs resulting in low V_{SAT} levels. The outputs of these devices do not have

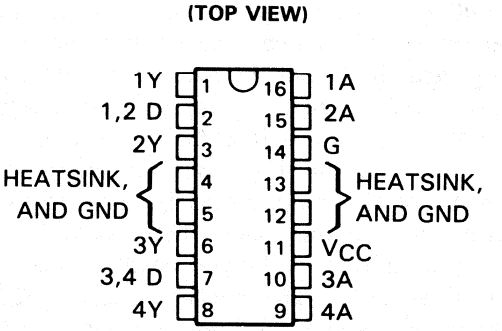


Fig. 12.15 Package pinout

Table 12.2 Device selection guide

Feature	436	437A	438
Output current	500 mA	500 mA	1000 mA
Maximum V_{SAT}	0.5 V	0.5 V	1.0 V
Max switching voltage	50 V	35 V	35 V

spurious transients during power-up or power-down sequencing. The device package, Fig. 12.15, provides four heat sink pins to help conduct heat from the device. As a result of this, and their copper leads, the SN75436, SN75437A and SN75438 packages are rated for 2 W continuous power dissipation at 25° C or less (free-air operating temperature). Inputs and outputs are conveniently located on opposite sides of the package for easy PC-board layout and assembly. Table 12.2 compares the maximum output current, output saturation voltage, and switching voltage for these three devices.

SN75435 Quad Driver

The SN75435 consists of four peripheral drivers, each with up to 20 W output drive capability. It features (Figs. 12.16 and 12.17) four open-collector drivers with a common enable input that, when taken low, disables all four outputs (Table 12.3). Output on resistance is less than 1 Ω at an output current of 500 mA. The standard 2-W DIP (Fig. 12.16) is used for this device. Output clamp diodes for transient protection are built in. The SN75435 is also free of spurious transients during power-up and power-down sequencing. Device standby power is less than 53 mW allowing cool operation and good long-term reliability.

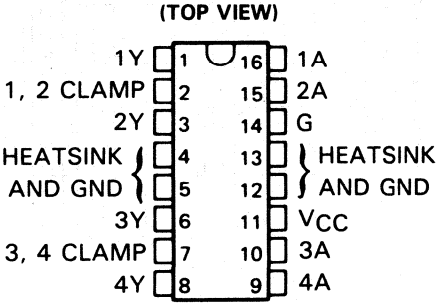


Fig. 12.16 SN75435 package pinout

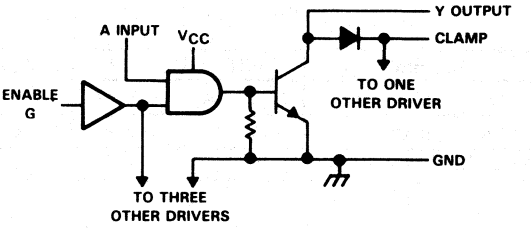


Fig. 12.17 SN75435 basic logic diagram

Table 12.3. SN75435 function table

INPUTS					OUTPUTS			
1A	2A	3A	4A	G	1Y	2Y	3Y	4Y
L	L	L	L	X	H	H	H	H
X	X	X	X	L	H	H	H	H
H	L	L	L	H	L	H	H	H
L	H	L	L	H	H	L	H	H
L	L	H	L	H	H	H	L	H
L	L	L	H	H	H	H	H	L
H	H	H	H	H	L	L	L	L

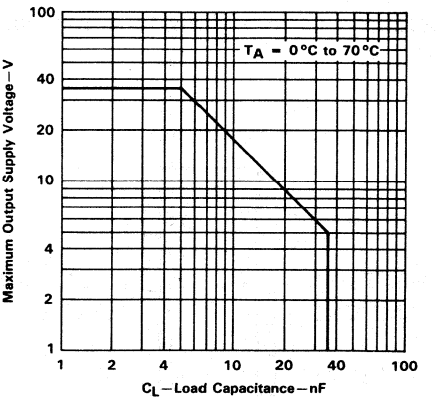


Fig. 12.18 Recommended maximum supply voltage vs load capacitance

One unique feature is that each driver output is protected against load shorts with its own latching over-current shutdown circuitry. The output will be turned off whenever a load short is detected. A short on one output does not affect the other three drivers. The latch for shutdown will hold the output off until the input or enable pin is taken low and then high again. A delay circuit is incorporated in the over-current shutdown to allow for a load capacitance of 5 nF at 35 V. Fig. 12.18 illustrates the recommended maximum supply voltage versus load capacitance.

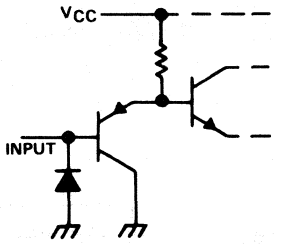


Fig. 12.19 SN75435 equivalent schematic of each input

The SN75435, Fig. 12.19, has high-impedance PNP inputs to provide both TTL and low level CMOS compatibility. Inputs are also diode clamped for negative voltage input transient protection. Although very well suited for driving solenoids, relays, memory systems, and LED circuits, this device is particularly well suited for driving lamps and motors.

SN75440 Quad Driver

The SN75440 quadruple peripheral driver is designed for use in systems requiring high current, high voltage, and high load power. The package Fig. 12.20 allows easy heat sinking and will provide 2-W power handling capability at an air temperature of 25°C. The device standby power is only 21 mW. Each device has four noninverting open-collector outputs with 600 mA sink capability and will switch inductive loads with a supply voltage of 35 V. The SN75440 also features an enable input control on pin 14, Fig. 12.20, for enabling or disabling all four outputs. The open-collector outputs, Fig. 12.21, are diode protected for transient protection and have a low on-state voltage. The outputs are free from spurious transitions (glitching) during power-up or power-down sequencing.

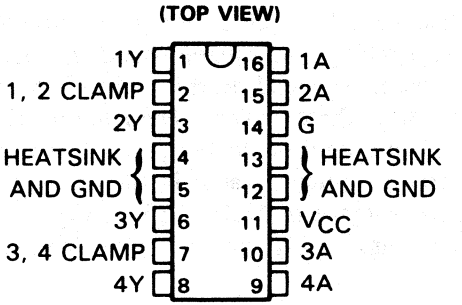


Fig. 12.20 SN75440 package pinout

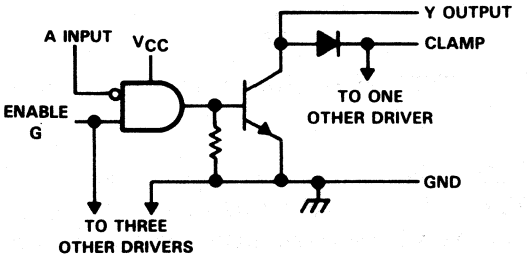


Fig. 12.21 SN75440 logic diagram (each driver)

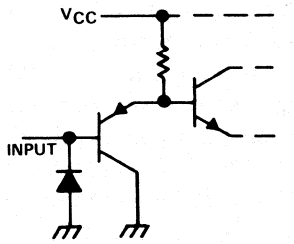


Fig. 12.22 SN75440 equivalent schematic of each input

Table 12.4 SN75440 logic functions

INPUTS		OUTPUT
A	G	Y
L	H	L
H	X	H
X	L	H

H = high level, L = low level,
X = irrelevant

PNP inputs, Fig. 12.22, have low level input currents of less than 10 μA. Functional relationships between the input and output logic functions are given in Table 12.4. Applications include driving relays, lamps, solenoids, motors, LEDs, data transmission lines, printer hammers, and other systems with high drive power requirements.

SN75372 Dual and SN75374 Quad Power FET Drivers

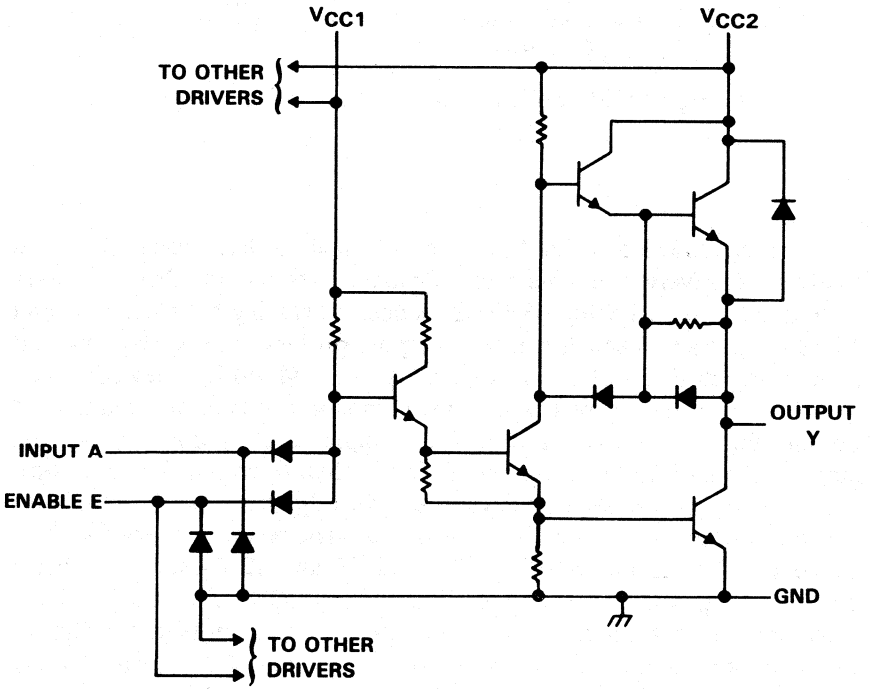


Fig. 12.23 SN75372 schematic (each driver)

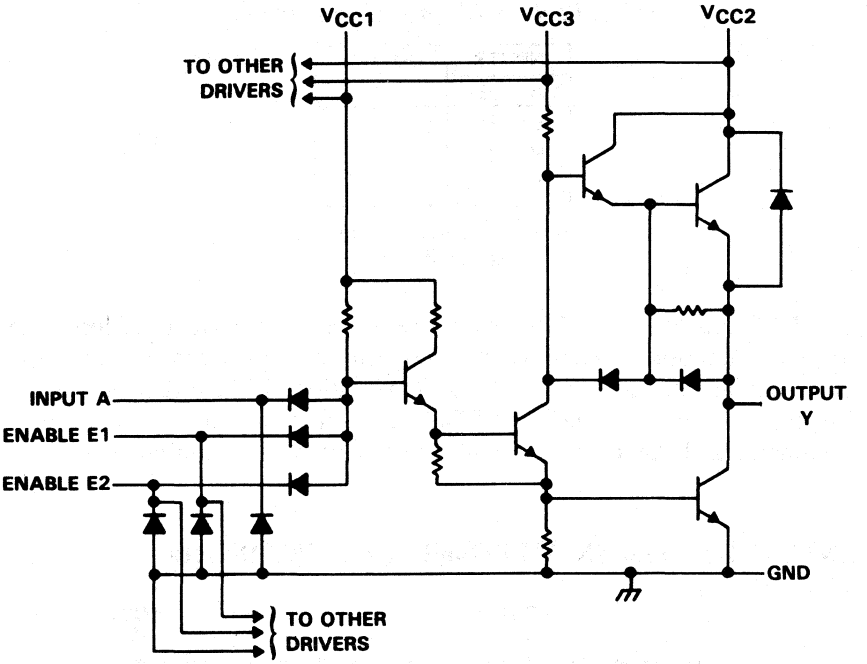


Fig. 12.24 SN75374 schematic (each driver)

The SN75372 and SN75374 are designed to drive capacitive type loads at relatively high data rates. Primary uses include driving power FET devices in switching applications and interfacing between TTL and MOS or CMOS. Their totem-pole outputs provide high speed sink and source capability suited for driving power MOSFET devices. The SN75372 is a dual driver with TTL compatible inputs and totem-pole outputs that can source, from a V_{CC2} supply level of up to 24 V, and sink currents of 100 mA minimum. Even when operating into a 390-pF load device propagation delays are typically less than 35 ns with transition times accounting for 25 ns of that. The result is a very adequate speed for driving power FETs. Figs. 12.23 and 12.24 show the basic circuit configurations for the SN75372 and SN75374. Both devices have input and output transient protection diodes. These devices have transient overdrive protection to minimize power dissipation. The typical standby power is 22 mW for the dual SN75372 and 38 mW for the quad SN75374.

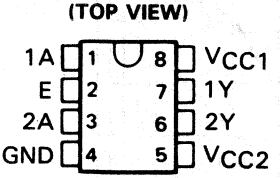


Fig. 12.25
SN75372 package pinout

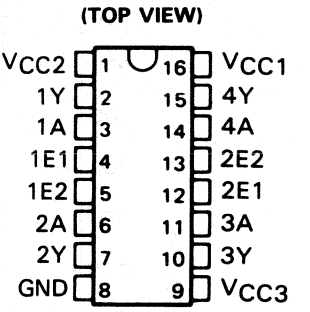


Fig. 12.26
SN75374 package pinout

The dual SN75372 comes in the 8-pin DIP package, Fig. 12.25, and has a common enable. The quad SN75374 comes in a 16-pin, Fig. 12.26, and has 2 enables for each pair of drivers. An additional feature of the SN75374, shown in both Figs. 12.24 and 12.26, is the availability of the pre-driver supply rail (V_{CC3} on pin 9). With V_{CC3} at 3 to 4 V above the value of V_{CC2}, it is possible to drive the output level very close to the V_{CC2} rail.

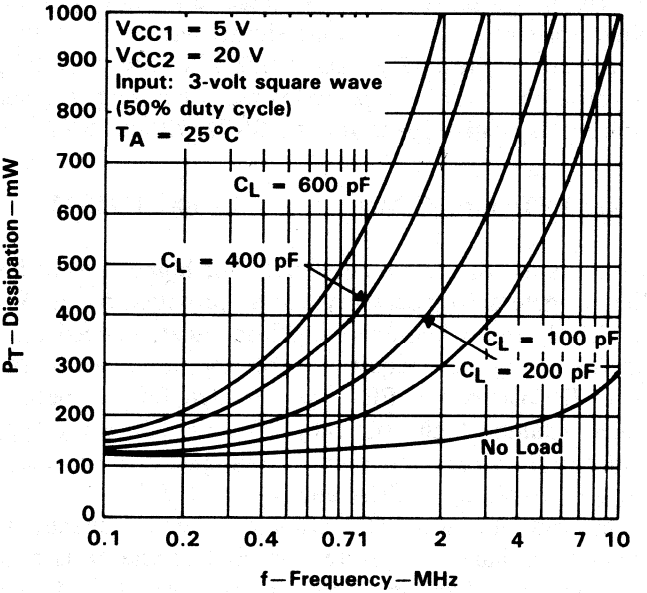


Fig. 12.27 Total dissipation both SN75372 drivers vs frequency

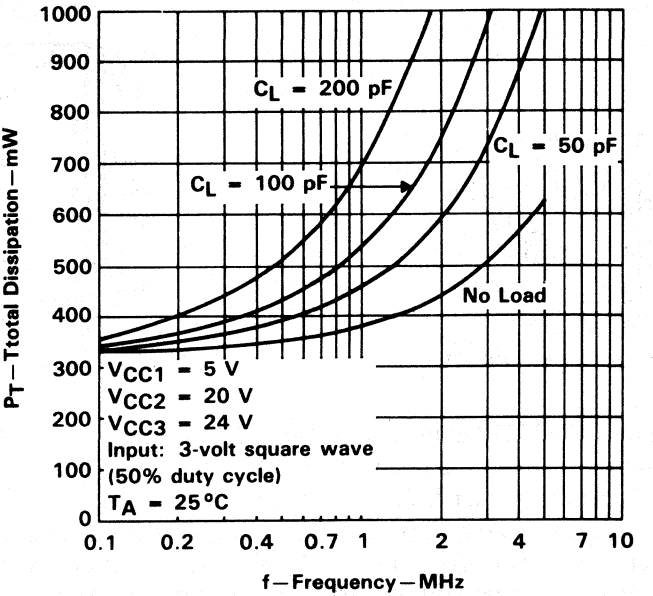


Fig. 12.28 Total dissipation all SN75374 drivers vs frequency

When driving capacitive loads at relatively high data rates, the package power dissipation will become significant and it is desirable to know what to expect. Figs. 12.27 and 12.28 illustrate the total package power dissipation that may be expected versus operating frequency with several different capacitive loads for the SN75372 and SN75374.

Driving Power FETs for DC Motor Control with SN75372

When motors use discrete power FETs for higher power levels the higher values of gate drive necessary can be provided by the SN75372. This is particularly true when pulse-width-modulated speed control and high speed switching are required. For the application shown in Fig. 12.29 an IRF630 power FET was selected for its speed, low on resistance and resulting high efficiency. The IRF630 has a typical on-state resistance of 0.25Ω and a maximum total gate turn-on charge Qg of 30 nC at V_{GS} = 10 V, I_D = 12 A and V_{DS} = 0.8 V.

A characteristic of power FETs is a wide effective channel resulting in a large gate capacitance. A discrete amount of charge is required to turn a device on for a given drain voltage and current, and gate-source voltage (V_{GS}). Unfortunately during switching, it is necessary to feed

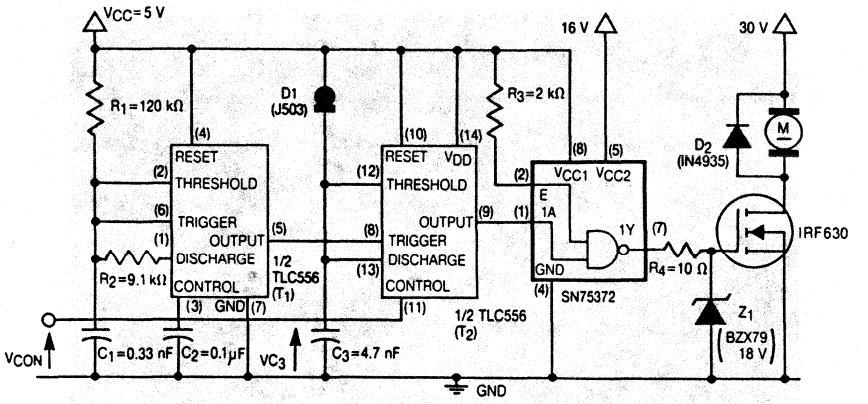


Fig. 12.29 PWM control circuit.

much more dynamic charge into the gate than required statically to achieve device turn-on. This extra dynamic charge, comprehended within total gate charge Qg, depends on the increase in effective gate capacitance due to the amplification of drain to gate capacitance C_{DG}. This is often called "Miller effect" and results from the effective load impedance during transitions and device gain.

During FET switch-on the gate-source voltage V_{GS} passes through three distinct regions that relate to the effective input capacitance, Fig. 12.30. Prior to time t_1 the FET is off and it's input voltage is dominated by the linear gate-source capacitance C_{GS}. During the period before t_1, the gate voltage passes through the threshold $V_{GS(TH)}$ towards a gate turn-on voltage $V_{GS(ON)}$ which can support substantial drain current.

At and beyond time t_1, when the gate-source turn-on voltage $V_{GS(ON)}$ is reached and the FET is operating in it's linear region, the effective input capacitance increases quickly due to the "Miller" action on C_{DG} resulting from drain-source voltage V_{DS} falling rapidly.

After time t_2, V_{GS} takes to time t_3 to attain its final value when C_{GS} and C_{DG} become fully charged. V_{GS} is limited by external circuitry to below the gate breakdown value. Assuming constant current gate drive, the time up to t_2 represents the total amount of charge necessary to turn the FET on, good design dictates overdrive which is achieved by t_3.

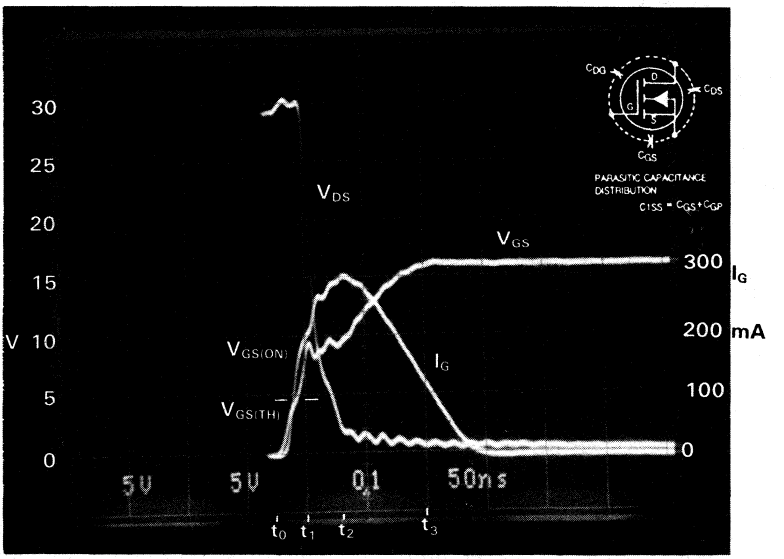


Fig. 12.30 Photo of V_{DS}, V_{GS} and I_G.

The ringing visable on the waveforms of Fig. 12.30, is due to parasitics in the circuit excited by the fast edges present.

Practically, the waveform of the current supplying this charge to the effective gate capacitance during switching is dependent on the driver characteristics and drive capability. In this application the waveform of FET gate input current I_G, Fig. 12.30, is a triangle of peak value 300 mA, this is within the maximum output current capability of SN75372 of 500 mA.

Pulse-width-modulated motor control, Fig. 12.29, is implemented using a dual LinCMOS timer TLC556, and may be part of an overall feedback control system. Timer T1 is configured as an astable oscillator operating at a frequency of 34 kHz. PWM duty cycle and hence motor speed is adjusted by varying control voltage V_{CON}.

Timer T1's output mark-space ratio is determined by the charge path R_1C_1 and the discharge path R_2C_1. The falling edge from timer T1 forms the trigger pulse for timer T2. The current regulator diode D1 delivers 0.56 mA that charges C_3 linearly towards threshold level V_{CON}. This provides a linear relationship, by means of PWM, between V_{CON} and

motor drive on-time, waveforms showing operation are shown in Fig. 12.31.

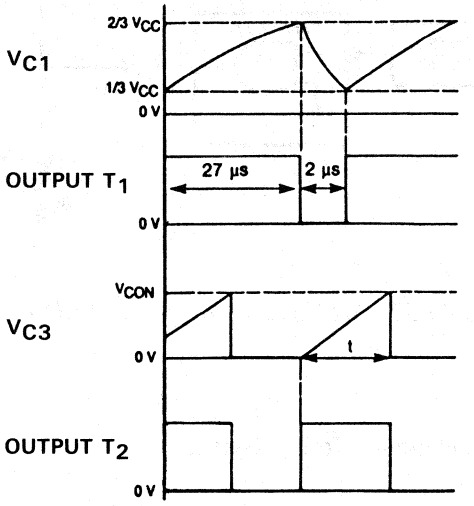


Fig. 12.31 Control Circuit Waveforms.

Operating frequency is given by the expression:

$$f = \frac{1}{[(R_1 + R_2)C_1 \times 0.693]}$$

Capacitor C_3's value must allow it to be charged up to the maximum value of V_{CON} within the mark period of timer T1:
$t = (C_3 \times V_{CON})/I_1$. Power supply sequencing should ensure V_{CC1} is applied before $V_{CC2/3}$.

Driving LED Devices with the SN75372

The SN75372 has the power to drive directly either a local or remotely located LED indicator. As shown in Fig. 12.32 it may be connected from the SN75372 output either to the supply voltage or ground. In calculating the values for current limiting resistors, SN75372 output voltages, either V_{OH} or V_{OL} and LED forward drop should be taken into consideration.

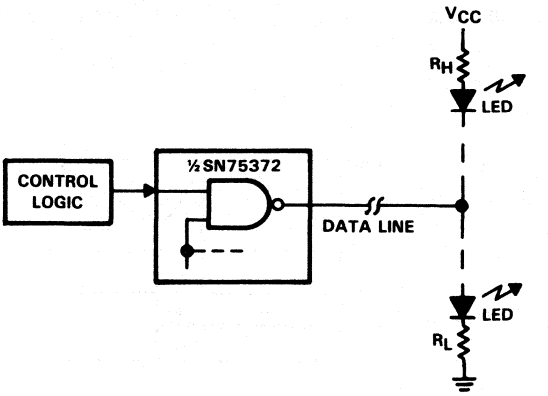


Fig. 12.32 SN75372 as a remote LED driver

DS3680 Quad Telephone Relay Driver

The DS3680 quad relay driver is a monolithic integrated circuit designed to interface from TTL to telephone relay systems or other −48 V systems. It is capable of sourcing 50 mA from standard −52 V battery power. To reduce the effects of noise and IR drop between logic ground and battery ground, these drivers are designed to operate with a common-mode input range of ±20 volts referred to battery ground. Each driver in the package has common-mode input voltage independent of the other drivers. High input impedance with low input current (typically less than 100 μA) results in minimum loading of the driving circuit. Built-in driver output clamp diodes eliminate the need for external networks to limit high voltage kickback levels present when switching inductive

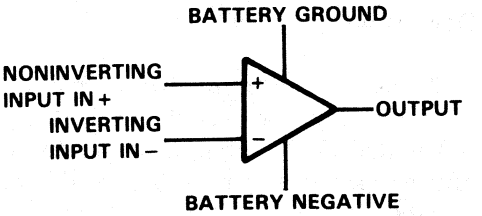


Fig. 12.33 DS3680 logic diagram (each driver)

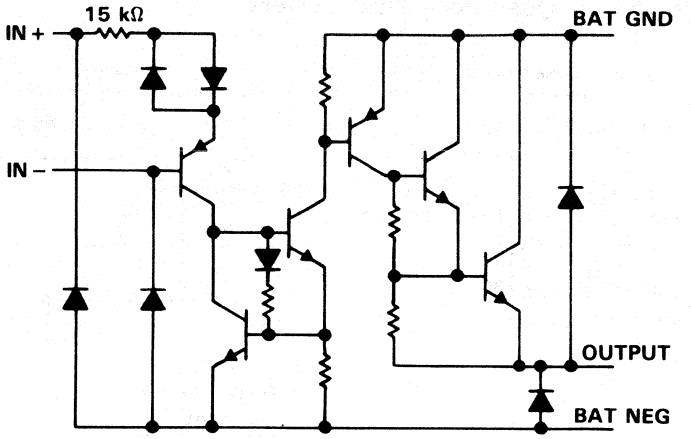


Fig. 12.34 Schematic diagram (each driver)

loads. A fail-safe feature incorporated in the DS3680 ensures that if either input is open, the driver output will be off. Figs. 12.33 and 12.34 illustrate each driver's logic symbol and schematic diagram respectively. Fig. 12.35 shows the convenience of the package pinout with inputs on one side and outputs on the other. Ground is located on a corner pin to implement easy board layouts.

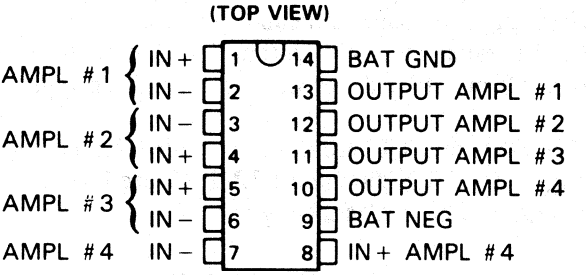


Fig. 12.35 DS3680 package pinout

DARLINGTON ARRAY DRIVERS

Some peripheral drivers are basically Darlington transistor arrays designed to have logic compatible inputs. They are often used in high current applications where the control logic is provided externally. The following devices are of this type.

ULN2064 Series Quad Peripheral Drivers

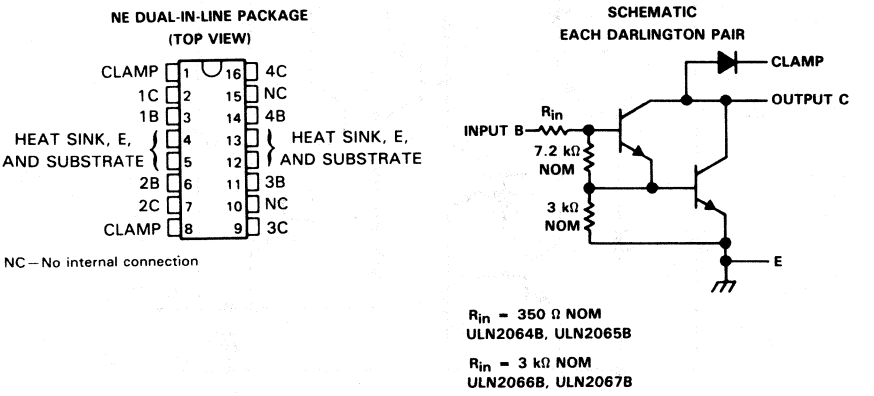


Fig. 12.36 ULN2064 series package pinout and schematic

The ULN2064, ULN2065, ULN2066, and ULN2067 are quad high current, high voltage Darlington switches. Each device has four Darlington transistor drivers with common-cathode clamp diodes for switching inductive loads (Fig. 12.36). Each of the drivers has 0.5 A output current capability, and their inputs and outputs may be paralleled for even higher current handling. Connected as common emitter circuits, these devices provide sink current drive for switching a variety of applications including relays, printer hammers, lamps, display circuits, memory circuits, and data transmission lines. The NE packages (Fig. 12.36) are rated at 2 W. Output loads may be operated from voltages up to 50 or 80 V, depending on the device type.

The ULN2064, and ULN2065 are intended for use with TTL and 5-V MOS logic. The ULN2066 and ULN2067 are intended for use with PMOS and higher voltage CMOS logic.

SN75068/ULN2068 and SN75069/ULN2069 Quad Darlington Switches

The SN75068/ULN2068 and SN75069/ULN2069 are quad high voltage, high current Darlington drivers (Fig. 12.37). Their outputs have common cathode clamp diodes for switching inductive loads. A third transistor at the input acts as a preamplifier providing high current gain and input compatibility with low power TTL and 5-V CMOS signals. With a maximum input current of only 250 μA at a V_{in} of 2.4 V, these

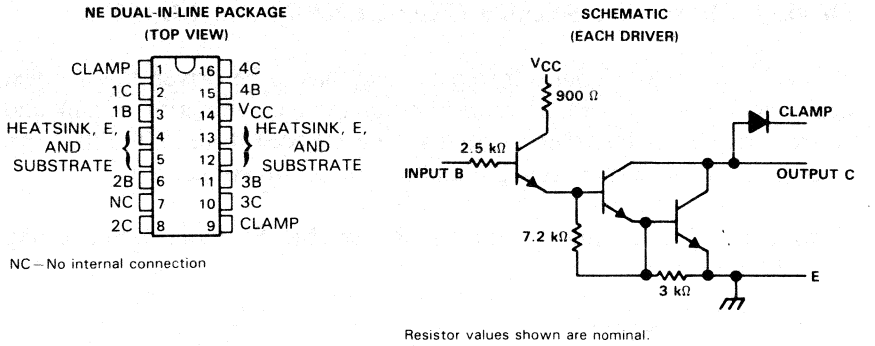


Fig. 12.37 SN75068, SN75069, ULN2068, and ULN2069 package pinout and schematic

devices may be operated directly from low power sources such as CMOS microprocessors or computers. Their outputs can sink up to 1.5 A and switch voltages of up to 50 V for the 068 devices and 80 V with the 069 devices. The recommended V_{CC} for the preamp is 5 V.

ULN2074 and ULN2075 Quad Darlington Sink or Source Drivers

The ULN2074 and ULN2075 are quad, high current, high voltage Darlington transistor switches (Fig. 12.38). They feature output voltage operation up to 80 V and output current capabilities of up to 1.5 A. These devices are unique in that they feature uncommitted collectors and emitters allowing sink or source applications. Inputs with respect to the emitters are compatible with TTL logic levels. These devices are particularly useful in H-drive applications because of the ability to perform both sink and source functions.

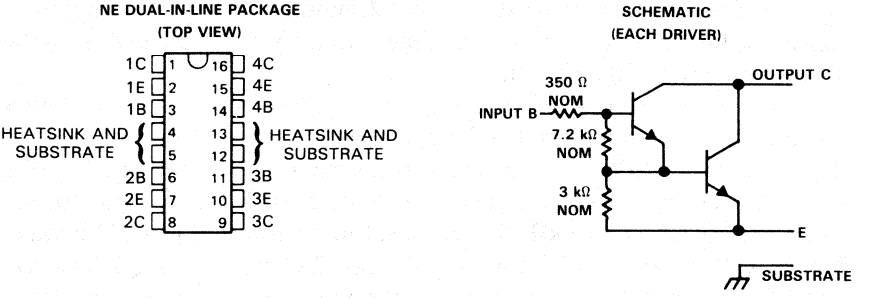
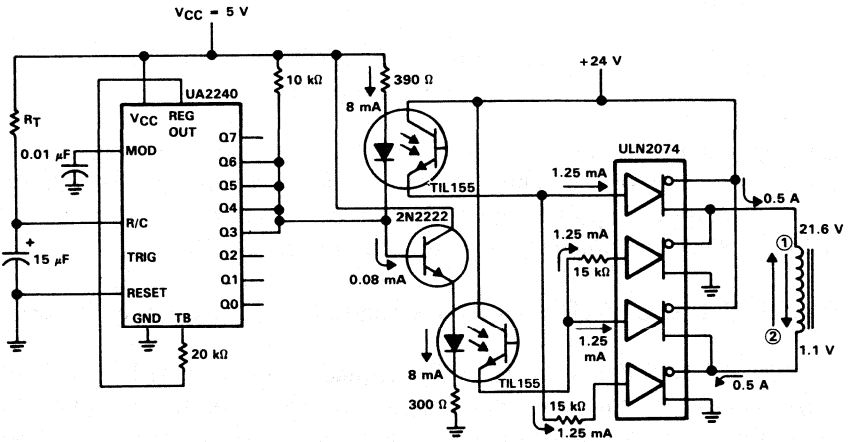


Fig. 12.38 ULN2074 and ULN2075 package pinout and schematic

Driving a Reversible Solenoid Using ULN2074 and ULN2075

When a bidirectional solenoid or relay has to be driven this is most easily accomplished with drivers either having a combination of sink and source outputs or totem-pole outputs. The ULN2074 and ULN2075 are examples of the former suitable and are for bidirectional drive.

Opto Isolated and Time Controlled Reversable Solenoid Drive Using ULN2074



R_T = 937.5 kΩ for exactly 15 μF to generate 1 hour in 256 steps or t_{Bo} = 14.0625 seconds.
Solenoid position ① ~ 56 min
Solenoid position ② ~ 4 min

Fig. 12.39 Opto isolated and timer controlled reversible solenoid driver

In this application, it is necessary to provide reversible solenoid drive with one position actuated for about 56 minutes out of an hour time period and the other position actuated for about 4 minutes. This is a continuously repeating function that requires the timer circuitry to be isolated from the solenoid supply (Fig. 12.39).

Timing is accomplished using a μA2240 programmable timer connected in the astable mode with a time base of about 14 s. A time base (t_b) of exactly 14.0625 s will result in a system cycle time of 256 × t_b or 3600 s (1 hr). The timing network consists of a 15 μF capacitor C_T and an R_T of about 940 kΩ. R_T will need to be adjusted to yield a value for t_b as close to 14.0625 s as is practical. Timer outputs Q3, Q4, Q5 and Q6 are wire-ORed to yield the proper output sequence. The result for an accurate time base will be an output that is low for 56.25 minutes and

high for 3.75 minutes. The total time may be adjusted by R_T but the ratio of time high to time low will remain the same. Opto isolation with TIL155 optocouplers provides 2500 V isolation between the controller and the solenoid circuits. The ULN2074 driver requires two control inputs, one inverted from the other. One TIL155 is used to couple, in-phase, with the drivers for the forward direction. The inverted drive is done using a 2N2222 NPN transistor operating as an emitter follower driver between the timer and the second TIL155 optocoupler. Output drivers for the reverse direction receive their drive signals inverted from those for the forward direction, allowing the ULN2074 to function as a bidirectional solenoid driver. The many applications for a bidirectional solenoid or relay driver include: Fluid Flow Valve Control, Car Electric Door Locks, Relay Switching, and Position Controls.

SN75465 and ULN2001A Series Seven Darlington Transistor Arrays

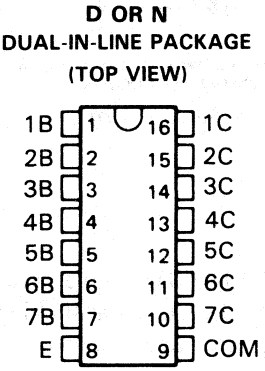


Fig. 12.40 SN75465 and ULN2001A series package pinout

The SN75465 series and the ULN2001A series are high voltage, high current Darlington npn transistor arrays with common cathode clamp diodes for switching inductive loads (Fig. 12.41). Each Darlington pair has a collector current rating of 500 mA. Darlington drivers may also be paralleled for higher current capability when using the (D) small outline and (N) plastic dual-in-line packages (Fig. 12.40) Total package substrate current for the D and N package is 2.5 A. The SN75465 series devices have a 100 V maximum collector-emitter voltage rating while the ULN2001A series have a 50 V maximum.

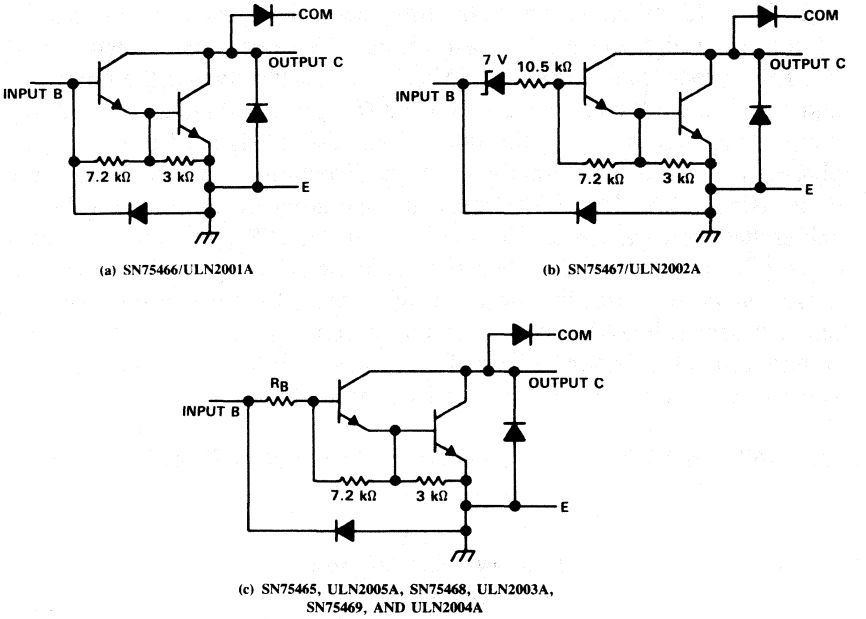


Fig. 12.41 Schematics of darlington pairs

The SN75465 and ULN2005A have 1.05-kΩ series base input resistors and are expecially designed for standard TTL or other sources with TTL equivalent or greater drive current capability.

The SN75466 and ULN2001A are general-purpose arrays with no input resistors or diodes for level shifting. They have a guaranteed minimum h_{FE} of 1000. These devices will work well as switches or linear amplifiers.

The SN75467 and ULN2002A are especially designed for use with 14 V to 25 V input signals, typical of PMOS logic devices. Each driver has a zener diode and resistor in series with its input to limit the input current for high voltage applications.

The SN75468 and ULN2003A have 2.7-kΩ resistors in series with their inputs allowing them to be compatible with TTL and 5-V CMOS logic.

The SN75469 and ULN2004A have 10.5-kΩ resistors in series with their inputs. They operate from medium logic voltage levels of 6 V to 15 V, typical of some CMOS and PMOS logic.

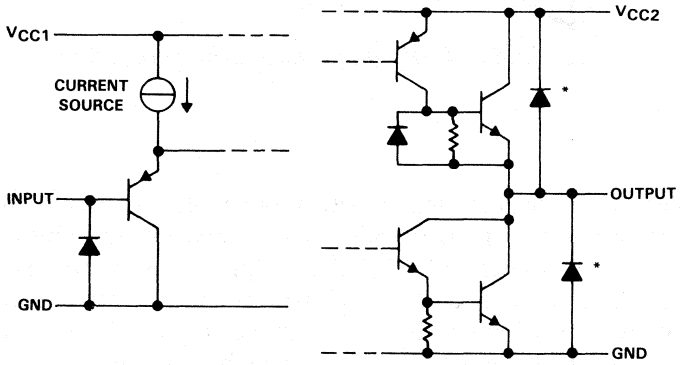
FULL AND HALF-H DRIVERS

L293, L293D, SN754410 and SN754411 Quad Half-H Drivers

The L293, L293D, SN754410 and SN754411 are a family of quadruple high-current half-H drivers that provide bidirectional drive currents at voltages from 4.5 to 36 V. These devices are designed to drive inductive loads such as relays, solenoids, dc and stepping motors, as well as other high-current, high voltage loads in positive-supply applications.

Table 12.5

DEVICE No.	OUTPUT CURRENT I_O	INTERNAL CLAMP DIODES	TEMPERATURE RANGE
L293	1 A	NO	0 – 70°C
L293D	600 mA	YES	0 – 70°C
SN754410	1 A	YES	–40 – 85°C
SN754411	1 A	NO	–40 – 85°C

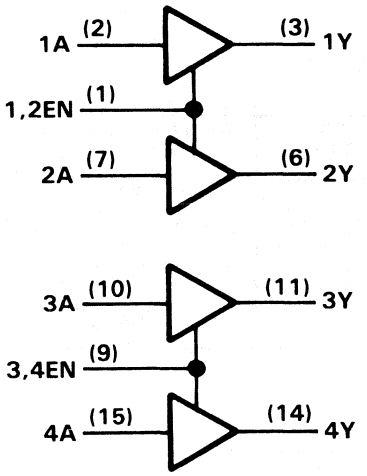


* L293D and SN754410 only

Fig. 12.42 Device inputs and outputs.

The basic differences between individual members of the family are shown in Table 12.5. L293 and SN754411 need external high speed output clamp diodes for inductive transient suppression, while L293D and SN754410 have diodes included. The SN74410/11 devices will operate over an extended temperature range suitable for industrial and automotive environments.

The basic device input and output schematics are shown in Fig. 12.42. All inputs are TTL and CMOS compatible and each output has a complete totem-pole circuit with a Darlington transistor sink and psuedo-Darlington source.



FUNCTION TABLE
(EACH CHANNEL)

INPUTS		OUTPUT
A	EN	Y
H	H	H
L	H	L
X	L	Z

H = high-level
L = low-level
X = irrelevant
Z = high-impedance (off)

Fig. 12.43 Logic diagram and function table

The logic diagram and function table of Fig. 12.43, show that channels are enabled in pairs with channel 1 and 2 enabled by 1,2EN and channels 3 and 4 enabled by 3,4EN. When an enable input is high, the associated channels are enabled and their outputs are active and in phase with their inputs. When the enable input is low, those channels are disabled and their outputs are off in a high-impedance state. With the proper data inputs, each pair of drivers form a full-H (or bridge) reversible drive suitable for motor or solenoid applications.

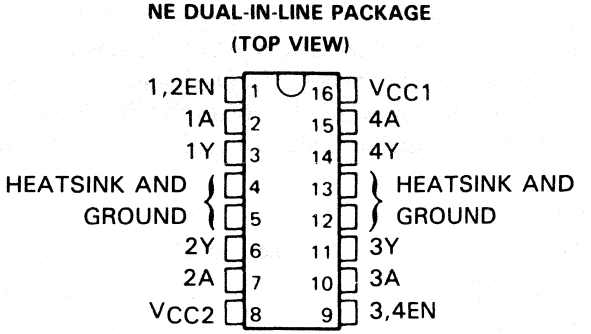


Fig. 12.44 L293 series package pinout

To minimise device power dissipation, a V_{CC1} supply voltage input, separate from V_{CC2} is provided for the logic supply. The "NE" dual-in-line package, Fig. 12.44 with copper leads is used for this device. It employs the four centre heatsink pins to help conduct heat away, for instance, to a copper heatsink area on a printed circuit board. This package is able to dissipate 2 W at a free air temperature of 25°C. In practice duty cycle and application specific thermal considerations have to be taken into account to be able to use the full current rating of the device.

L298 Dual Full-H Driver

The L298 is a dual high-current full-H driver that provides bidirectional drive currents of up to 2 A at voltages from 5 to 45 volts. This device is designed to drive inductive loads such as stepping motors, dc motors, relays, solenoids and other high voltage and high current loads in positive supply applications.

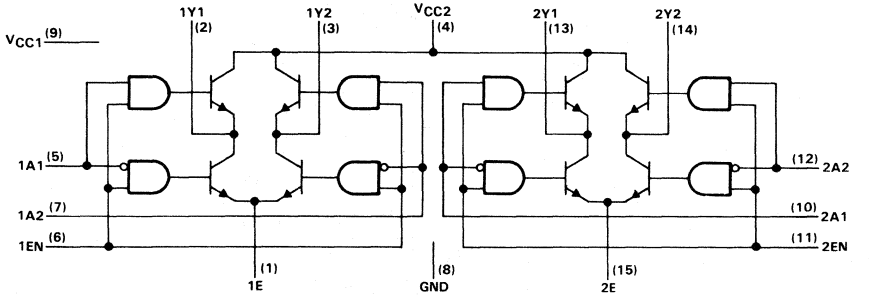


Fig. 12.45 L298 logic diagram

The logic diagram Fig. 12.45 shows the features of the device, inputs are TTL and CMOS compatible, each output has a totem pole driver with a Darlington transistor sink and psuedo-Darlington source. Each full-H driver is enabled separately when the corresponding enable input EN is high as shown by the function table and logic symbol of Fig. 12.46.

FUNCTION TABLE
(EACH CHANNEL)

INPUTS		OUTPUT
A	EN	Y
H	H	H
L	H	L
X	L	Z

H = high-level
L = low-level
X = irrelevant
Z = high-impedance (off)

logic symbol

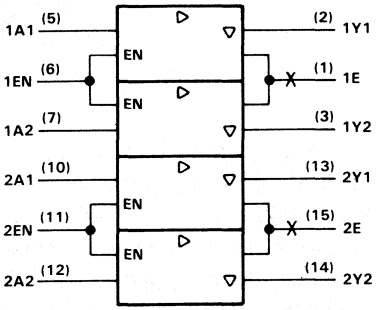
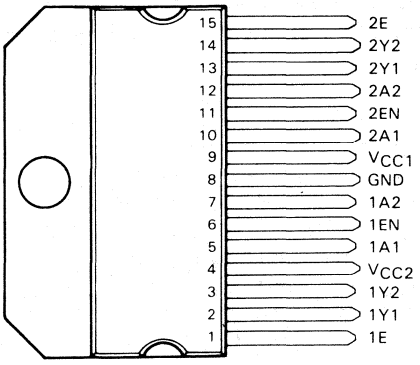


Fig. 12.46 Function table and logic symbol

Each half of the device forms a full-H reversible driver suitable for solenoid or motor applications. The current in each full-H driver can be monitored by connecting a resistor from the sense output terminals 1E or 2E respectively to ground.

KV PACKAGE
(TOP VIEW)



The tab is electrically connected to pin 8.

Fig. 12.47 Single in-line power package

External high-speed output clamp diodes should be used for inductive transient suppression. To minimise device power dissipation, a V_{CC1} supply voltage input, separate from V_{CC2} is provided for the logic supply.

A 15 pin single-in-line (SIP) power package, Fig. 12.47, is used for this device which is able to dissipate 25 W with a case temperature of 75°C. This corresponds to a junction to case thermal resistance of 3 °C/W and allows significant power levels to be handled by a power IC when suitable heat sinking is employed.

L293 and L298 Families Applications

DC Motor Bridge or "H" Drivers

When it is necessary to reverse the direction of a dc motor this can be economically achieved from one supply rail using a bridge or "full-H" drive configuration. This is where each end of the motor winding is switched as illustrated in Fig. 12.48. These switches or drivers must be able to either sink or source motor current. Both the L293 Quad Half-H Driver and the L298 Dual full-H can provide two full-H functions at their respective current levels.

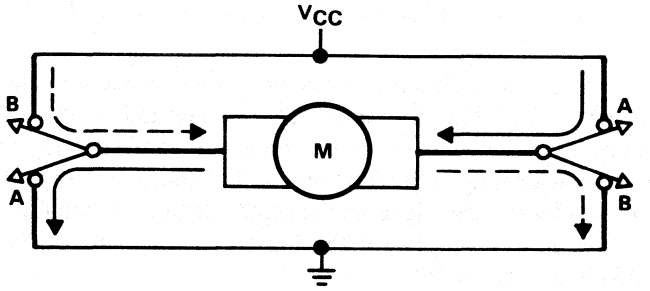


Fig. 12.48 Basic H motor drive configuration.

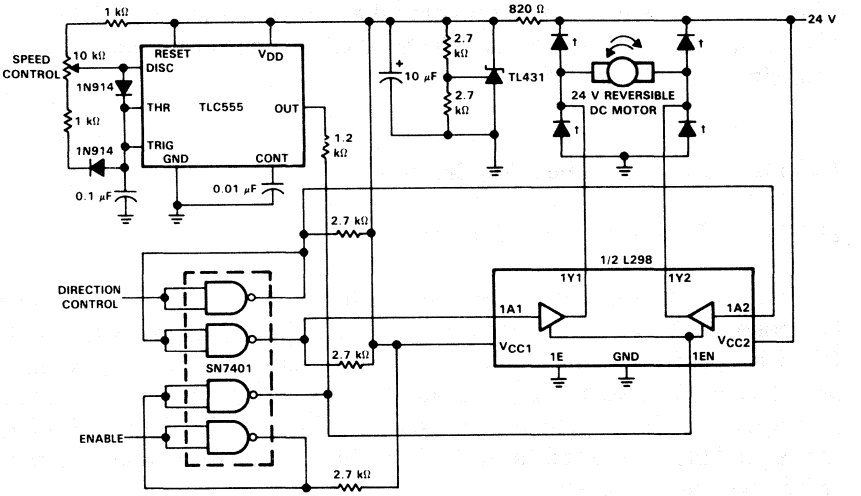
L298 Speed Controlled Bidirectional DC Motor Driver

In a typical application, L298 could be used to provide bidirectional full-H bridge drive for 24 V, 2 A dc motors. Fig. 12.49 illustrates one half of such a circuit.

FUNCTION TABLE

ENABLE	DIRECTION CONTROL	1Y1	1Y2
H	H	source	sink
H	L	sink	source
L	X	disabled	disabled

X = don't care H = high level L = low level



†Diodes are 1N4934 or equivalent.

Fig. 12.49 L298 as a bidirectional motor driver

Speed control is achieved by providing variable duty cycle pulses to the EN input of the L298. The duty cycle, generated by the TLC555, is adjustable between 10% and 90% at an approximate frequency of 1.2 kHz and provides a wide range of motor speeds. The motor enable and direction are determined by the logic level at the inputs detailed in the function table of Fig. 12.49. A 5 V supply for the logic and timer circuit is provided by a TL431 shunt regulator.

Stepper Motor Overview and Drives

Stepper motors are commonly available in three basic types: variable reluctance, hybrid and permanant magnet. These are further divided into bipolar and unipolar types.

All are similar in construction and consist of a rotor, surrounded by a stator which has many individual poles and coils. Rotors also have many poles and, respectively for the three types, are either made from toothed

soft iron, a combination of an axial permanant magnet with alternately toothed soft iron "caps" or a cylindrical permanent magnet with alternate poles magnetised on the outside. Stepping angle, torque and speed are dependent on the motor type, numbers of rotor and stator teeth and details of the motor design.

By applying stepping pulses to the coils the rotor and stator magnetic fields align to cause the rotor to step through an angle determined by motor design and driving mode. In variable-reluctance motors all the fields are solely produced by the stator current while the hybrid and permanent magnet types depend upon interaction between the stator currents and permanent magnet fields.

Unipolar drive uses bifilar windings to reverse the stator flux typically driven by a simple quad darlington array. The disadvantages of the extra windings are reduced torque, due to thinner wire for the same size motor, and higher cost. Bipolar drive alters the direction of stator flux by reversing the current flow in the same winding, typically by means of a full-H driver. This method of drive has become more cost effective with the introduction of integrated circuit full-H drivers.

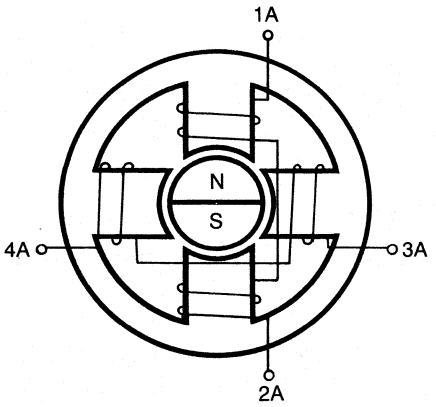


Fig. 12.50 Basic permanent magnet stepper motor.

A basic four pole, two phase, bipolar, permanent magnet motor is shown in Fig. 12.50. This with Fig. 12.51, will be used to illustrate the three common driving modes.

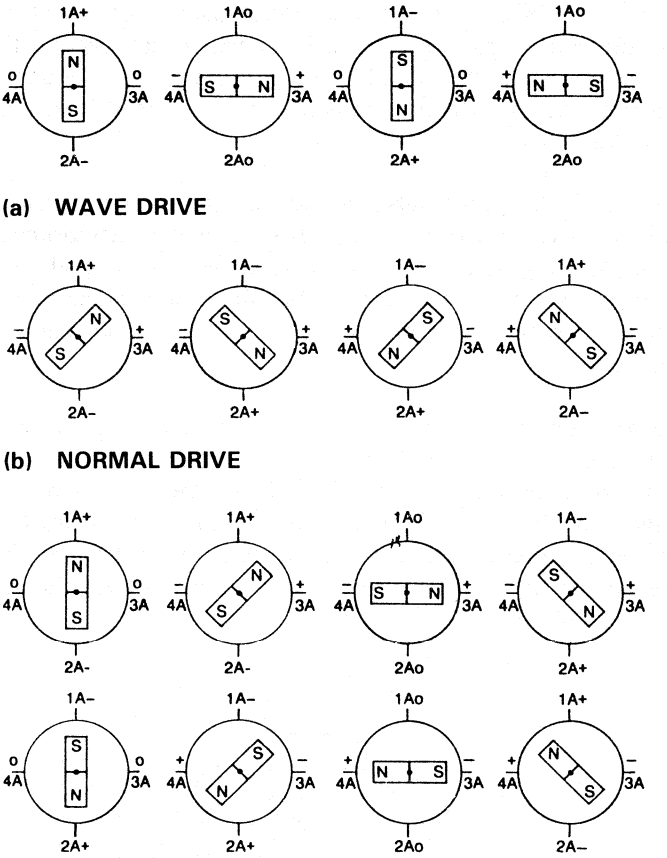
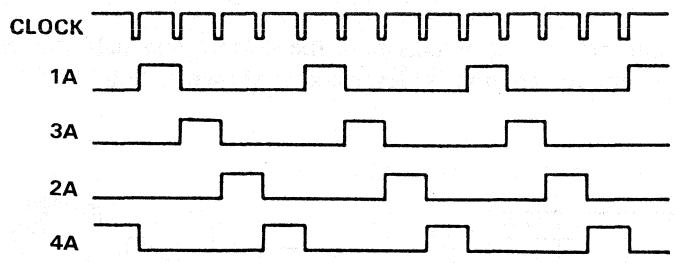


Fig. 12.51 Permanent-magnet rotor alignment.

Wave drive causes the rotor to align itself with the energised coil and the uses drive sequence, Fig. 12.51 (a), 1A2A/3A4A/2A1A/4A3A. This is a full-step sequence with only one-phase-on and uses the lowest operating current.

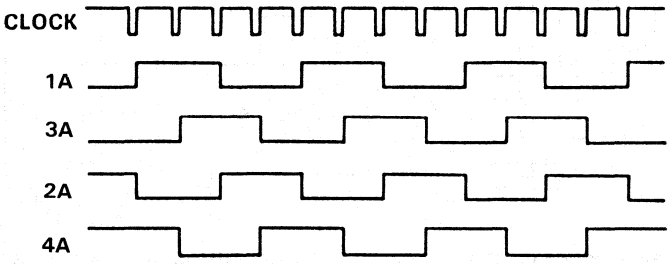
Normal drive energises two adjacent coils simltaneously and causes the rotor to be aligned midway between the two. The drive sequence, Fig. 12.51(b), 1A2A,3A4A/ 2A1A,3A4A/ 2A1A,4A3A/ 1A2A,4A3A provides a high torque two-phase-on drive also giving full steps, in this simple case at 45° to wave drive.

Half step drive is achieved by alternating wave drive and normal drive, Fig. 12.51(c).



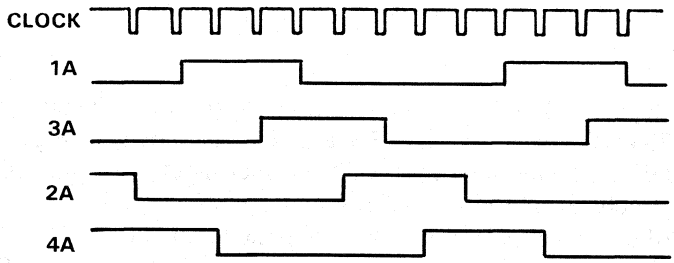
WAVE DRIVE MODE

Provides 1 phase on at a time.



NORMAL DRIVE MODE

Provides, sequentially overlapping, 2 phases on at a time.



HALF STEP MODE

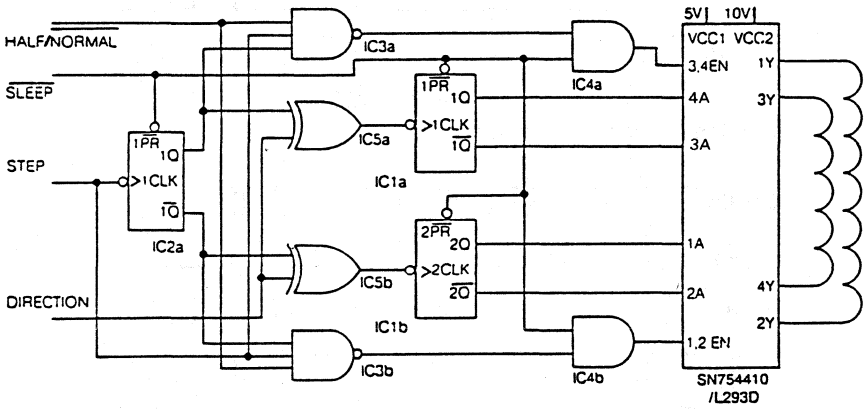
Provides 2 phases on, 1 phase on, 2 phases on, 1 phase on type of drive sequence.

Fig. 12.52 Half-step, wave and normal-drive mode waveforms.

Wave-drive, normal-drive and half-step waveforms for the motor of Fig. 12.50 are shown in Fig. 12.52.

Stepper Motor Drive Using L293D/SN754410

A circuit which implements both the normal and half step stepper motor drive modes previously described is shown in Fig. 12.53.



IC1/2:SN74HC112, IC3:SN74HC10, IC4:SN74HC08, IC5:SN74HC386,

Fig. 12.53 Normal and half step stepper motor control.

Flip-flops, IC1 & 2, generate four quadrature signals at the L293/SN754410 inputs 1A, 2A, 3A and 4A, as shown in Fig. 12.54, with the DIRECTION input low. Half/Normal input uses gates IC3a & b to selectively enable inputs 1,2EN and 3,4EN, Fig. 12.53 to allow selection of either normal-drive or half-step modes at outputs 1Y, 2Y, 3Y and 4Y. The corresponding waveforms are shown in Fig. 12.54. Taking SLEEP input low places the bridge in a high impedance state and initialises the circuit logic, SLEEP input needs to be toggled when the direction is reversed.

Motor speed is determined by the step rate and is set according to motor parameters.

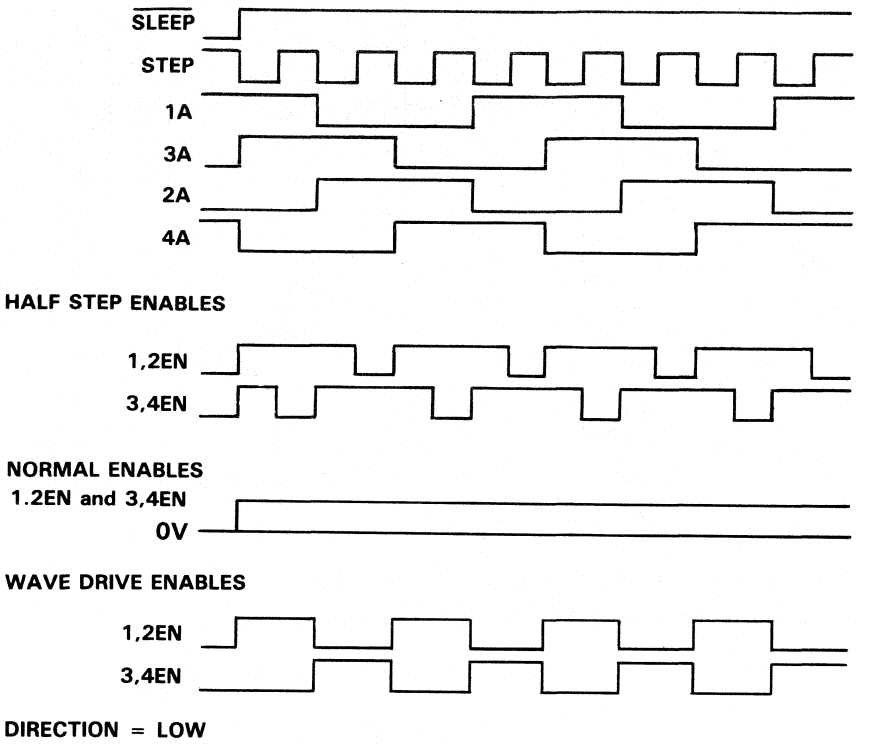
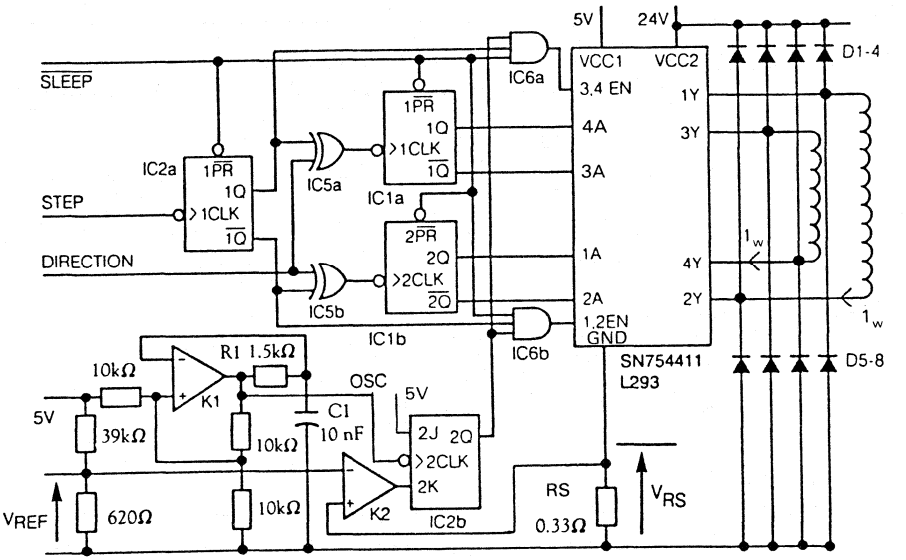


Fig. 12.54 Driving Sequences

Wave Drive with Constant Current Control using L293/SN754411

Stepper motor winding time constant (L/R) causes the motor current to respond slowly to stepped inputs. Consequently during high speed operation the motor current and torque can never reach its rated value. A chopper circuit, Fig. 12.55, will permit the motor to be driven from a high supply voltage, allowing faster current increase.

This circuit, Fig. 12.55, uses the same logic (IC 1,2 & 5) to generate the quadrature signals at L293/SN754411 inputs as the circuit of Fig. 12.53. Wave drive is produced by using IC6a & b to selectively enable inputs 1,2EN and 3,4EN. These enable waveforms, Fig. 12.54, correspond with the wave drive outputs shown in Fig. 12.52. The other inputs of IC6a & b driven from IC2b output are used to modulate the enable inputs to achieve the chopper action. Fig. 12.56 shows a part of this chopper action when a phase is energised.



IC1/2:SN74HC112, 1C3:SN74HC10, 1C4:SN74HC08, IC5:SN74HC386, IC6:SN74HC11, K1/2:TLC3702

Fig. 12.55 Wave drive with constant current control

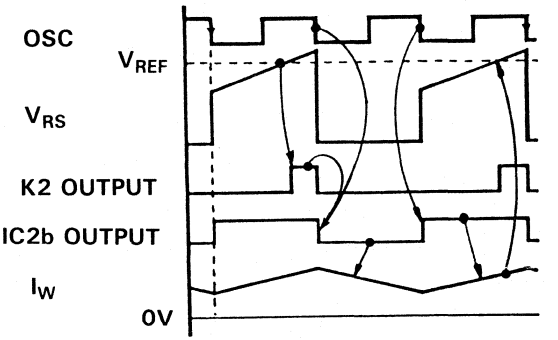


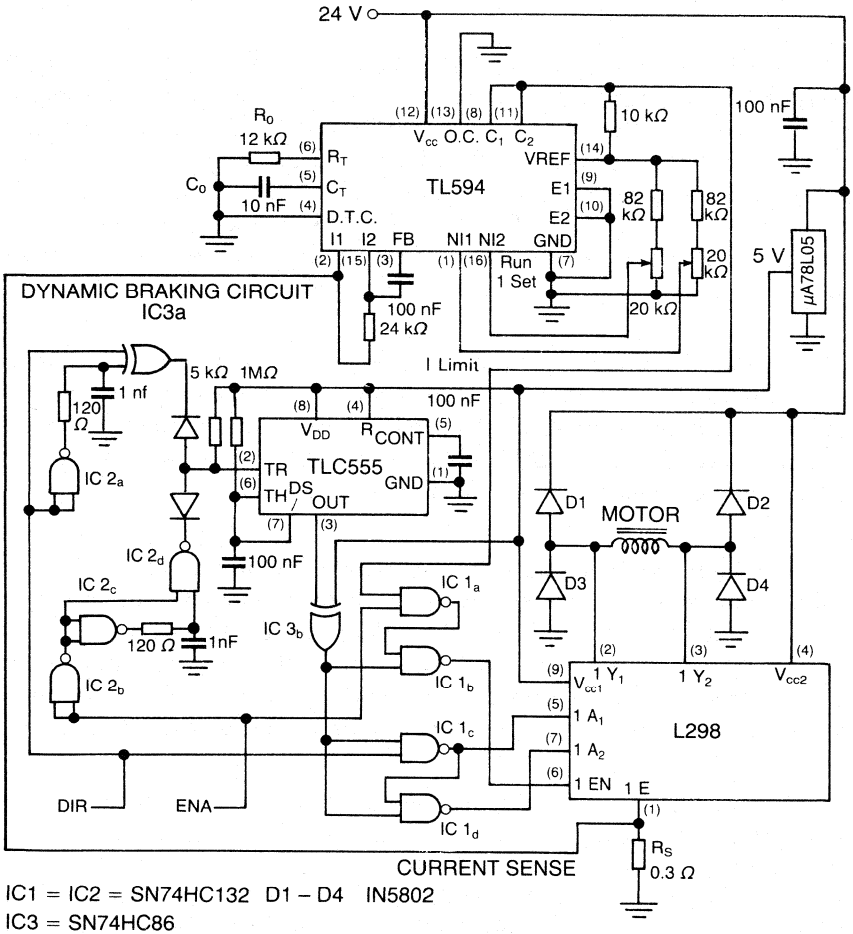
Fig. 12.56 Chopper action

Initially, flip-flop IC2b's output is high allowing control signals from the phase generator to energise the appropriate winding. Current in this winding increases at a rate determined by the L/R time constant and the supply voltage V_{CC2} until voltages V_{RS} and V_{REF} are equal. Comparator K2's output is forced high allowing IC2b's output to go low on the next

falling edge from oscillator circuit K1. This disables the bridge causing the winding current to decay through the clamp diodes. When voltage V_{RS} has fallen below V_{REF}, K2's output is forced low. Upon receipt of the next negative edge from the oscillator IC2b's output is clocked high allowing phase signals to enable the bridge.

Audible noise is avoided by choosing the chopper frequency, $0{\cdot}36/(R1 \times C1)$, to be greater than 20 kHz. Peak motor current I_{pk} is set by V_{REF} and RS.

L298 PWM Servo Motor control with Dynamic Braking



IC1 = IC2 = SN74HC132 D1 – D4 IN5802
IC3 = SN74HC86

Fig. 12.57 L298 PWM servo-motor drive

Constant servo motor torque with dynamic braking and enable and direction control are provided by the L298 and surrounding circuit, Fig. 12.57. Applications where constant torque is found useful are constant tension reel drives, tape drives with a constant torque region at start-up and rotary machine tools.

Motor torque T_T, from the relationship $T_T = K_T I_A$ between motor current I_A and motor constant K_T N m/A, will be constant for constant motor current. This is achieved using feedback, proportional to motor current, to control a pulse width modulator (PWM) that drives the motor through one of the full bridge circuits of L298.

Motor current through the L298 driver is monitored across a small resistor RS to ground. The difference or error signal between this signal and the preset voltage V_P is integrated by the TL594's error amplifier which then further produces a PWM signal at the TL594's output. This PWM signal is applied in the correct phase to the L298 enable pin 1EN via the NAND logic gates IC1a & b. The PWM frequency is set by R_O and C_O to 20 times the reciprocal of motor time constant which allows the motor current to be averaged to a substantially constant value.

Dynamic braking is provided at motor direction reversal and motor stop (dis-Enable). With a direction change a momentary stop allows inductive energy to be dissipated and reduces inertial backlash before motor reversal. The enable signal stops the motor with minimum mechanical overshoot for fast positioning response.

For high side braking, the brake operates by turning both the upper driver transistors on, and effectively shorts the motor via a driver output transistor and a clamp diode (which pair depends on motor direction) and the V_{CC2} supply rail. Low side braking would need Schottky diode clamps from output to ground to prevent a momentary negative motor voltage, when both lower transistors are turned-on, damaging the L298.

The one-shot, a TLC555 in monostable mode, generates a braking pulse of approximately five times the motor time constant, about 100 ms. Its output is inverted by EXOR, IC3b, to provide inputs for NAND gates IC1a, b & c. This inhibits PWM drive through IC1b, and causes a logic high to appear at L298 inputs 1A1, 1A2 and 1EN during the braking period. The TLC555 is triggered via a diode OR, on both direction input

changes using IC2a & 3a, and on Enable high-low transition using IC2b, c & d.

It should be noted that a 24 V motor cannot be controlled at maximum rated torque with PWM. Under these conditions, the controlled torque must be less than the rated torque at the manufacturer's recommended supply.

TLP609 Dual Flux-Regulating Actuator

Intelligent power IC TLP609, is a dual flux-regulating driver for switching double-ended inductive actuators with currents of up to 2.5 A per channel at supply voltages from 30 to 60 volts. It is designed to drive and control the magnetic flux in printheads, solenoids, stepper motors, relays and other loads whose inductance value value varies during operation.

The TLP609 performs the function of flux-regulation under the control of standard TTL or CMOS logic signals for two independent channels. Flux is monitored in the load inductor and controlled using chopper mode operation. Constant flux regulation is maintained by varying the load current to compensate for reluctance variation and load dynamics during system operation.

Historical solutions to the problem of maintaining constant magnetic load flux in the presence of varying reluctance, mutual inductance and power supplies have resulted in less than optimum performance. These solutions include: tightly controlling either load current, voltage or excitation time. A better solution is to regulate magnetic flux in the load circuit, this can be achieved by use of a control loop which has as its input an error signal which comprehends changes in the load magnetic flux.

A typical example of a variable inductance load is an impact matrix printhead which operates by storing energy in a spring permanent-magnet system. A coil in its magnetic circuit ideally supplies exactly enough flux to cancel the effect of the magnet and release the spring to operate a printwire. With the old solutions, multiple printwire interactions cause variable printwire throw lengths and impact forces giving inconsistent results.

Principles of operation

As is known the total amount of flux, $N\Phi$, in the magnetic field surrounding an inductor is related to the current by the proportionality called inductance:

$$L = \frac{\mu N^2 A}{l} \quad \ldots(1)$$

Inductance is often assumed to be constant but in practice this is only true for limited operating conditions. Turns N, usually remain constant but current and temperature changes cause shifts in permeability, μ, and magnetic paths involving motion have area A and path length l, that vary. For constant flux a control mechanism has to regulate the inductor current to comprehend this changing and non-linear inductance.

Faraday's law says that the voltage across a pure inductor is proportional to the rate of change of flux linkages.

$$v(t) = N\frac{d\Phi}{dt} \left(= L\frac{di}{dt} \right) \quad \ldots(2)$$

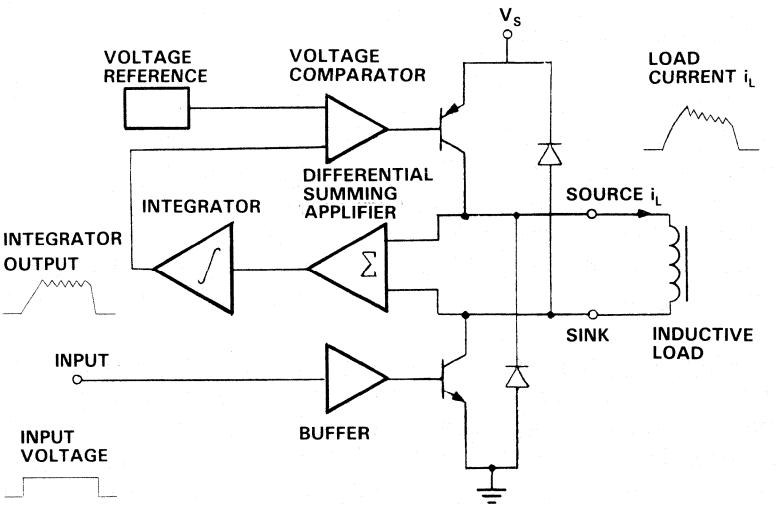


Fig. 12.58 TLP609 simplified functional block diagram.

Conversely flux is proportional to the integral of the voltage across an inductor, the output of such an integration process has the dimensions of volt-seconds (Webers).

$$\Phi = \frac{1}{N}\int_0^t v(t)\,dt \text{ volt-seconds (Webers)} \quad ...(3)$$

This simple derivation ignores the effect of circuit resistance which is assumed to be low.

The foundation of flux regulation is found in the above equation for Φ, and is illustrated by the simplified block diagram of Fig. 12.58. The integral of the load voltage which represents the flux level is compared to a reference value at the comparator input. The difference is used as an error signal to control a switching regulator formed by the comparator and source driver transistor output.

TLP609 Operation

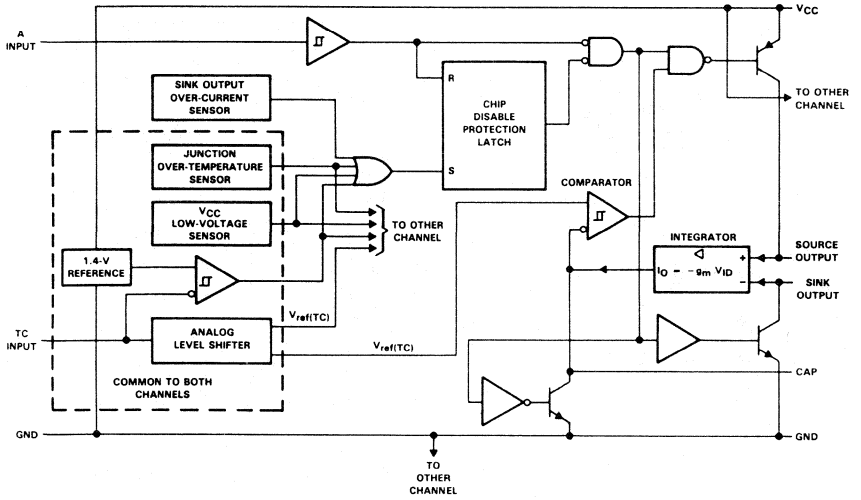


Fig. 12.59 TLP609 functional block diagram

The TLP609 functional block diagram is shown in Fig. 12.59. Each channel, Fig. 12.60, has a separate sink and source outputs for driving each end of the inductive load. The internal feedback path, which consists

of a differential transconductance amplifier and comparator, provides flux regulation by chopper mode operation at the source output.

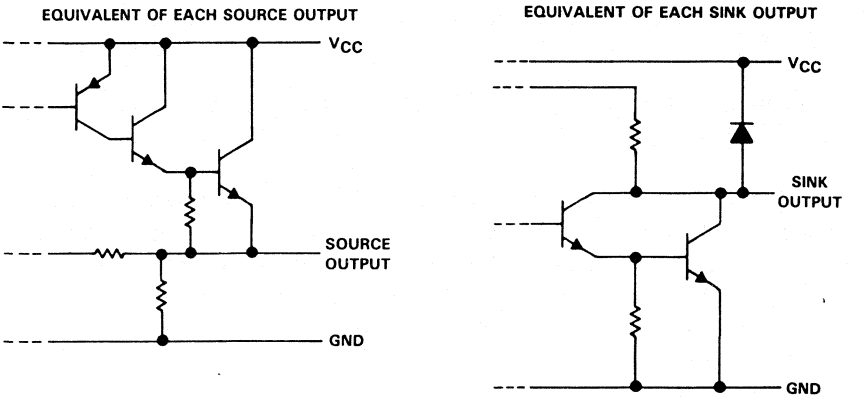


Fig. 12.60 Schematics of source and sink driver outputs

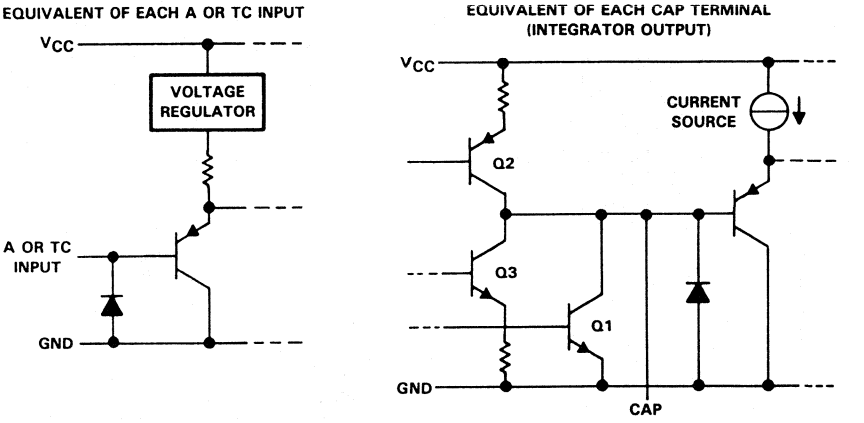


Fig. 12.61 Schematics of A & TC inputs and CAP output

The differential transconductance amplifier forms an integrator with an external capacitor connected from the CAP terminal to GND. The current provided at the capacitor terminal CAP, and therefore the integral of the voltage referenced to GND, Fig. 12.61, is proportional to the differential voltage between the sink and source outputs.

The feedback path controls the state of the source driver output from the output of the differential voltage comparator. The comparator's inverting input is connected to the CAP terminal and the non-inverting input is connected to a voltage $V_{ref(TC)}$, which is referenced to GND. $V_{ref(TC)}$ is proportional to the Threshhold Control (TC) voltage $V_{I(TC)}$ which is referenced to GND. The comparator hysteresis controls the charge and discharge voltage excursions at the CAP terminal and thus controls the on and off time of the source output drive chopper action.

Sink outputs have on-chip clamp diodes which suppress inductive transients. External high-speed clamp diodes must be used on the source output to provide the clamped load recirculation path during the off-time portion of the source chopper cycle and inductive transient protection.

DRIVER FUNCTION TABLE (EACH CHANNEL)

INPUTS			OUTPUTS		OPERATING MODE	COMMENTS
A	TC	CAP	SOURCE	SINK		
X	≤0.8 V	X	OFF	OFF	Disabled	TC acts as a digital input referenced to GND
H	≥2 V	X	OFF	OFF	Active	TC acts as an analog input referenced to GND (See Note 1)
L	≥2 V	$<V_{T+}$	ON	ON		
L	≥2 V	$>V_{T-}$	OFF	ON		

INTEGRATOR FUNCTION TABLE (EACH CHANNEL)

VOLTAGE INPUTS		CAP VOLTAGE	DIFFERENTIAL VOLTAGE $V_{O(SOURCE)} - V_{O(SINK)}$	CAP TERMINAL (See schematic)	INTEGRATOR MODE OF OPERATION
A	TC				
X	≤0.8 V	X	X	Q1 Sinking	Reset (Disabled)
H	≥2 V	X	X	Q1 Sinking	Reset
L	≥2 V	X	≥300 mV	Q2 Sourcing	Charge
L	≥2 V	$>V_{T-}$	≤ −300 mV	Q3 Sinking	Discharge

H = high level; L = low level; X = irrelevant

NOTE 1: The TC input has an operating range from 0 to 6 volts, but its effect on the CAP terminal is linear from approximately 2 V to 6 V. Best linearity is achieved within the recommended operating linear range.

Table 12.6 Driver and integrator function tables

Driver and integrator function tables, Table 12.6, describe in greater detail the individual operation of each input and output of Figs. 12.60 and 12.61.

The TLP609 features built-in thermal protection and a sink over-current sensor to prevent damage to the device. The outputs are disabled under low-V_{CC} supply voltage conditions to prevent transient output turn-on during power-up or power-down. The TC input is a combined threshold and logic input that disables the outputs when the $V_{I(TC)}$ voltage is less than 0.8 V. This permits an external RC time delay at the TC input, during logic system power-up, to allow logic at input A to stabilize without causing undesired output turn-on. When a fault condition is detected by any one of these four detection features, the RS latch for each channel is set. The fault condition must be removed and the A input taken high before the RS latch will reset, reactivating the channel.

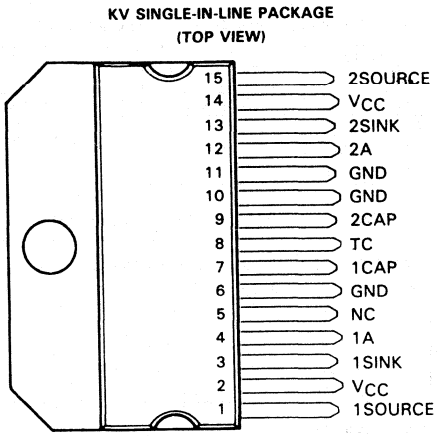


Fig. 12.62 TLP609 package pinouts

The TLP609 is packaged, Fig. 12.62, in a 15 pin SIP power package which can dissipate 20 W at a case temperature of 90°C. TLP609 is characterised to operate over a temperature range of −20°C to 85°C.

Dot-Matrix Printhead-Driving with TLP609

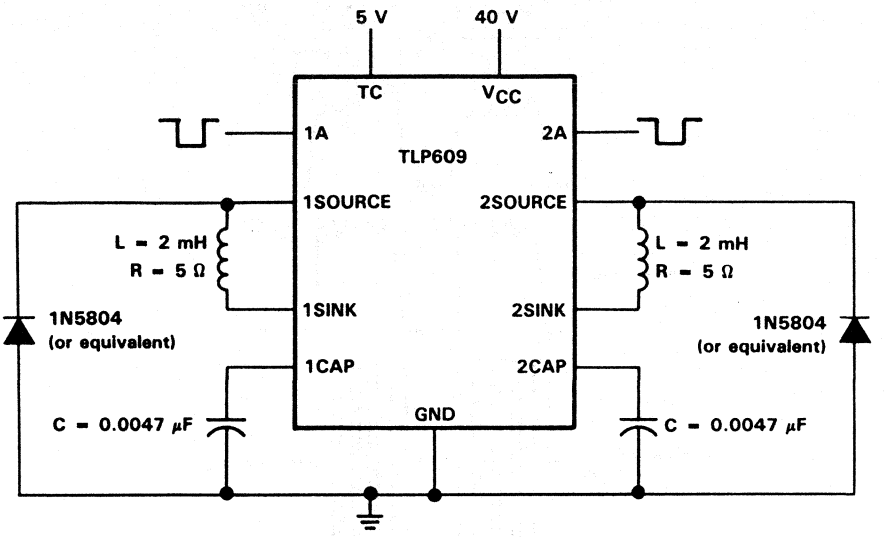


Fig. 12.63 Typical dot-matrix printhead-driver application

An application of the TLC609 Dual-Flux regulator is driving inductive dot-matrix printheads, Fig. 12.63.

The circuit waveforms, Fig. 12.64, illustrate the controlled changes of load current to compensate for a change in load inductance. A constant magnetic flux is maintained from a corresponding constant 'CAP' feedback voltage waveforms as described above.

To use the TLP609 in a given application requires selecting an integrator capacitor C and Threshold Control input voltage level $V_{I(TC)}$ along with the number of chop cycles to obtain the desired flux regulation.

If the required flux level is known then the charge on the integration capacitor C may be found from (3) and the transconductance gm:

$$Q = C \times V_{CAP} = gm \times N \times \Phi \quad ...(4)$$

where $gm = I_{CAP}/(V_{SOURCE} - V_{SINK})$ in Siemens S, (A/V).

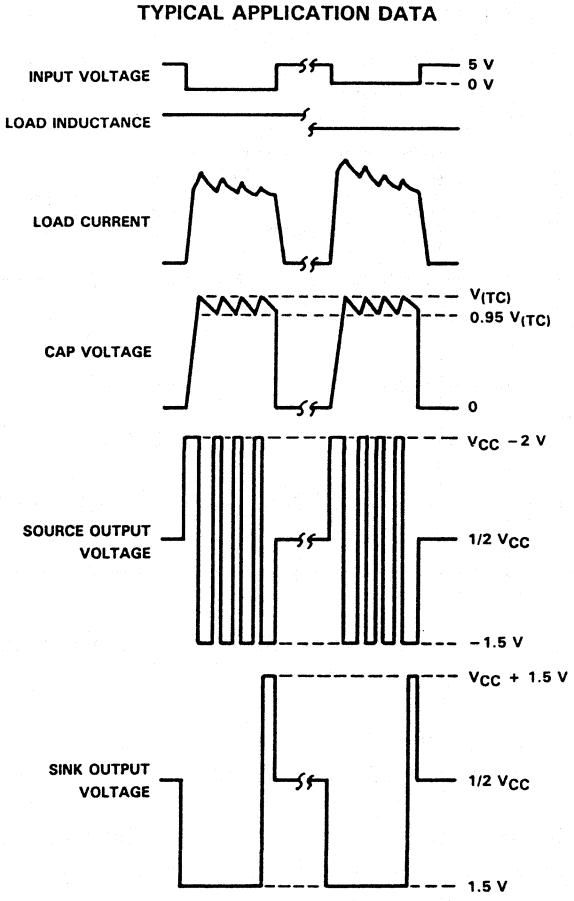


Fig. 12.64 Representative waveforms

Usually the required flux level is not known and must be determined as described below.

The charge on the capacitor C is the time integral of the CAP terminal output current $i_{CAP}(t)$:

$$Q = C \times V_{CAP} = \int_0^t i_{CAP}(t)\,dt \quad \ldots(5)$$

Current $i_{CAP}(t) = gm \times V_L(t)$ where $V_L(t) = V_{SOURCE}(t) - V_{SINK}(t)$ is the voltage across the TLP609 load. Substituting in (5):

$$C \times V_{CAP} = \int_0^t gm \times V_L(t)\, dt \quad ...(6)$$

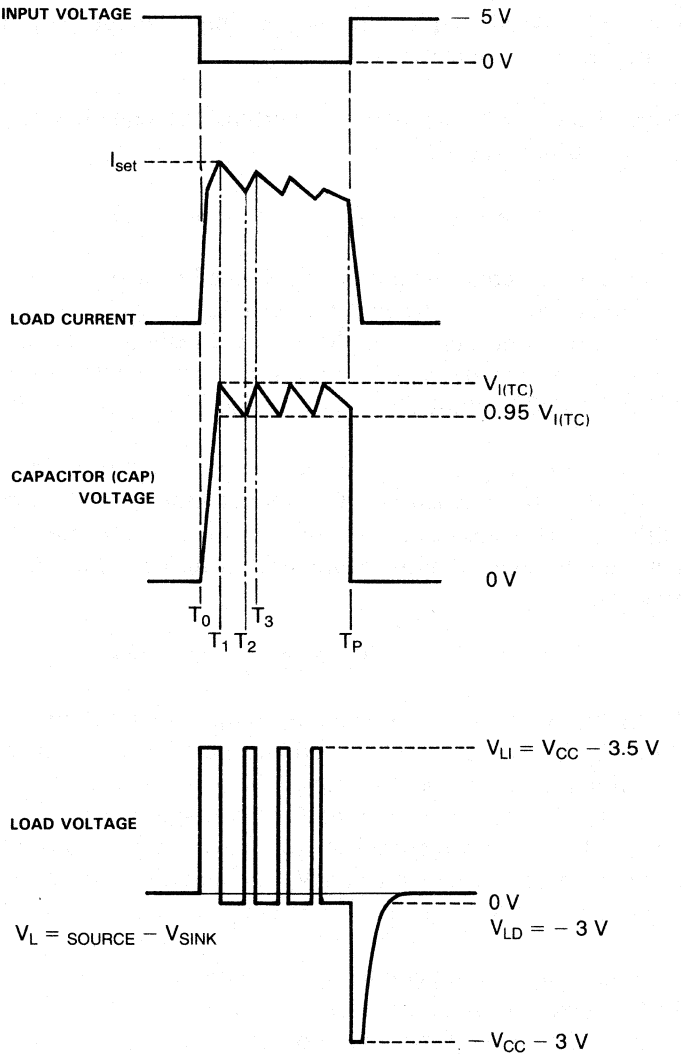


Fig. 12.65 Expanded view of inductive load waveforms.

An expanded view of the waveforms of Fig. 12.64 is shown in Fig. 12.65 giving more details of operation and parameters used in this description. Flux regulation occurs during time T_O to T_P (T_{OP}) when input A is low. In most applications the switching times of the source and sink outputs can be neglected in comparison with the input pulse period. During the regulation period, as shown on the load voltage waveform, the voltage across the inductive load essentially consists of two fixed voltage values of V_L. With load current increasing, voltage V_{LI}, and load current decreasing, voltage V_{LD}.

Using these two voltages of V_L permits the equation of (6) to be solved over an interval Δt:

$$C \times V_{CAP} = V_L \times gm \times \Delta t \quad ...(7)$$

This equation applies within the piecewise linear regions on the CAP voltage waveform below the upper threshold, $V_{CAP} = V_{I(TC)}$, and the increasing waveform and above the lower threshold voltage, $V_{CAP} = 0{\cdot}95\ V_{I(TC)}$, for the decreasing waveform trip points. At these trip points, the TLP609 comparator switches the source output chopper.

The ratio of $V_{I(TC)}$ to V_{CAP} at the comparator threshold trip points is pre-adjusted to compensate for manufacturing variances in the TLP609's g_m. This adjustment causes the integration time to be more constant unit to unit. It also permits substituting $V_{I(TC)}$ for V_{CAP} in equation (7):

$$C \times V_{I(TC)} = V_L \times gm \times \Delta t \quad ...(8)$$

When this $C \times V_{I(TC)}$ product is found using (8) the integrating capacitor C, the voltage VI(TC) and the number of chop cycles and timing characteristics can be determined. This is achieved by substituting T_{01} ($T_1 - T_0$) for t after which time CAP voltage = $V_{I(TC)}$ as shown in Fig. 12.65. Time T_{01} is required for the inductive load current to reach the optimum level, I_{set} under given V_{LI} voltage drive conditions. I_{set} is known or can be determined experimentally for a given inductive load.

Experimental determination of excitation time, T_{01}, to reach the desired I_{set} current is usually necessary because inductance is most often a non-linear function of current. For instance, in this stored energy printhead example, I_{set} is the current value at which the magnetic field is cancelled for only one pin firing.

Equation (8) now becomes:

$$C \times V_{I(TC)} = V_{LI} \times gm \times T_{01} \quad ...(9)$$

Now that the $C \times V_{I(TC)}$ product can be determined using (9) integrating capacitor C is chosen such that $V_{I(TC)}$ is within the linear threshold control voltage range of 3 to 6 volts as specified in the TLP609 data sheet. Voltage $V_{I(TC)}$ can be adjusted by the user to control the flux in a linear manner.

After the initial pulse rise time T_{01}, the alternating falling and rising voltage waveforms at the CAP terminal are identical within the duration of the A input pulse width (T_{0P}). Chopping action occurs during time T_{1P}. After the $C \times V_{I(TC)}$ product is determined, the time of rising and falling voltage waveforms at the CAP terminal, and number of chop cycles, may be computed from the following equations that refer to Fig. 12.65:

$$\text{Number of chops} = \frac{(T_{0P} - T_{01})}{(T_{12} + T_{23})} \quad ...(10)$$

Where:

T_{01} has either been determined empirically or, if flux level was originally specified, from:

$$T_{01} = \frac{C \times V_{I(TC)}}{gm \times V_{LI}} \quad ...(11)$$

$$T_{12} = \frac{C \times V_{I(TC)} \times (0.95 - 1)}{gm \times V_{LD}} \quad ...(12)$$

$$T_{23} = \frac{C \times V_{I(TC)} \times (1 - 0.95)}{gm \times V_{LI}} \quad ...(13)$$

The 5% hysteresis value in $V_{I(TC)}$ CAP voltage, shown in Fig. 12.65 is included in equations (12) & (13).

Design Example

By substituting values into equations (9) to (13) for the example printhead application shown in Figs. 12.63 and 12.65 the component values and circuit conditions can be calculated.

Suppose the following are the design parameters:

V_{CC} = 40 V — system supply.
T_{0P} = 500 μs — system parameter.
I_{set} = 1.5 A — load characteristic, known or determined empirically.
T_{01} = 170 μs — load chracteristic, known or determined empirically.
V_{LI} = 36.5 V — TLP609 max source output voltage.
V_{LD} = −3 V — TLP609 and external diode characteristic.
g_m = 3 μS — TLP609 transconductance.

Using equation (9) $C \times V_{I(TC)} = 1.89 \times 10^{-8}$ ampere-secs, if C is chosen to be 4.7 nF so that $V_{I(TC)} = 4.01$ V then condition $3\text{ V} < V_{I(TC)} < 6\text{ V}$ is met. $T_{12} = 105\ \mu$s, $T_{23} = 8.6\ \mu$s and the number of chops = 2.9.

Usually it is desirable to have between two and eight chops for proper flux regulation, dependent on the input pulse width, T_{0P}. An excessive number of chops will produce unnecessary switching losses and power dissipation. Two few chops will cause poorer flux regulation. As the comparator hysteresis is fixed at 5%, the primary way to vary the number of chops is to vary the chopping time, T_{1P}, by changing either the supply voltage, the L/R ratio of the inductor of the input pulse width.

The above method and equations can usually be used to obtain reasonable approximations to the desired flux-regulation performance. Mechanical interactions and variation not taken into account in the equations may require that $V_{I(TC)}$ is adjusted empirically for optimum performance.

DRIVING DATA TRANSMISSION LINES

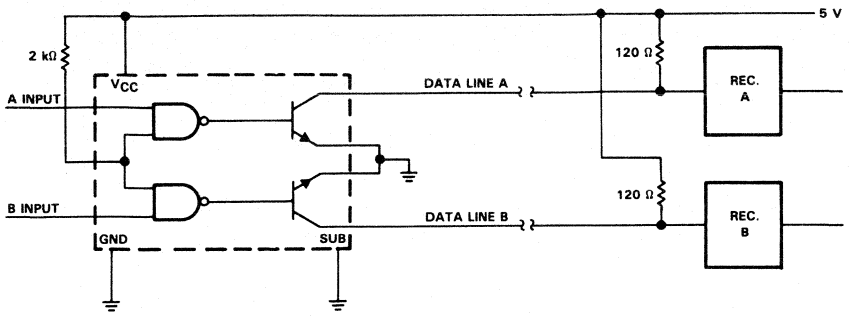


Fig. 12.66 SN75451B as dual sink-mode line driver.

Peripheral drivers' power capabilities sometimes make them suitable for driving data transmission lines. The SN75451B type, for example, allows sink mode line driving as illustrated in Fig. 12.66 where the lines are terminated in their characteristic impedance to the supply rail.

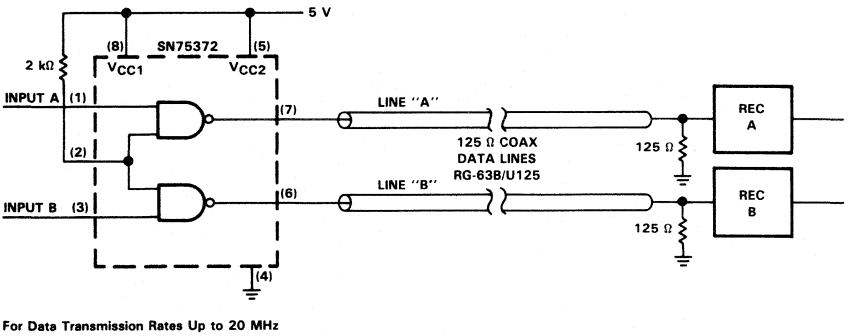


Fig. 12.67 High speed coaxial data line transmission

For higher speed applications totem-pole output drivers such as SN75372 may be more applicable. This device will drive single ended coaxial or twisted pair line at high data rate as illustrated in Fig. 12.67. With transition times less than 12 ns, the SN75372 drivers can transmit data at frequencies up to 20 MHz.

Section 13

HALL-EFFECT DEVICES

THE HALL EFFECT

Introduction

Edwin Hall first noted the effect that bears his name at Johns Hopkins university in 1879. He was investigating the effects of a steady magnetic field on current in a thin gold foil. He observed a small voltage at the edges of a current-carrying gold foil when a magnetic field was applied that was perpendicular to the foil.

The Hall-Effect in Silicon

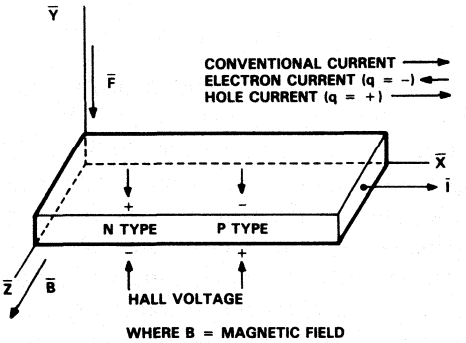


Fig. 13.1 Hall-effect current flow

The Hall-effect occurs as current flows through a semiconductor material in the presence of a magnetic field. As electrons or holes flow through the material, their path depends on the charge density and velocity of the majority carriers as well as the magnetic flux. In Fig. 13.1, if current (I) flows from left to right with magnetic flux (B) in the direction shown, the force (F) applies downward on the majority carriers, whether holes or electrons. Charge carriers collect near the bottom surface of the semiconductor material and generate the Hall voltage. In n-type material, the majority carriers are electrons and the polarity of the Hall voltage becomes negative on the bottom surface with respect to the top. The converse is true for p-type material because holes are the majority carriers. Accurate diffusion of a specific impurity into silicon determines the mobility of the majority carriers and the charge density. The Hall-effect ideally produces

a repeatable Hall voltage that is linearly proportional to the external magnetic field.

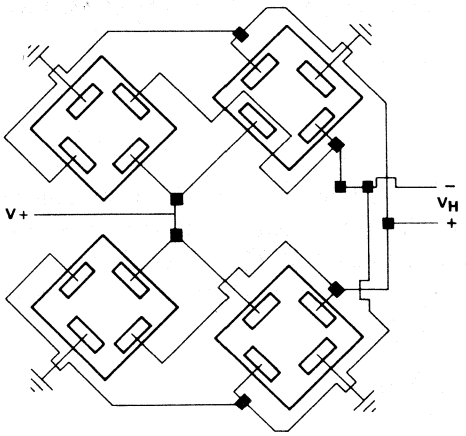


Fig. 13.2 Hall-effect sensor layout

However, in addition to magnetic-field intensity, there are other semiconductor-material factors that govern the Hall voltage, including temperature, mechanical stress, and current. Both mechanical stress and temperature changes affect mobility of the majority carriers. Also changes in current flow cause nonlinear fluctuations in the Hall voltage. A constant-current generator eliminates current changes, and temperature-compensation circuits offset the thermal effects. The nonlinearity caused by mechanical stress is not easily corrected. The most desirable approach for minimizing the effects of mechanical stress utilizes the architecture of the sensor. Fig. 13.2 shows the geometric layout of the orthogonal cross-coupled Hall cell. The cell with four sensors connected in an orthogonal manner, reduces the effect of mechanical stress and also improves sensitivity to magnetic fields. The cross coupling of devices also reduces process-related variations and reduces the offset by a factor of 16.

HALL-EFFECT DEVICES

A Hall-effect device is a circuit consisting of a Hall-effect cell, signal conditioning functions which may include hysteresis, and an output transistor integrated into a monolithic chip. The three basic types of Hall-effect devices are the switch, the latch, and the linear device. The switch and latch are digital devices while the linear Hall device provides a voltage output that is linear with respect to changes in magnetic flux density. The unit of magnetic flux density in the International System of Units is the tesla (T).

The tesla is equal to one weber per square meter. Values expressed in milliteslas may be converted to gauss by mulitplying by ten, e.g., 50 millitesla = 500 gauss.

Switching Devices

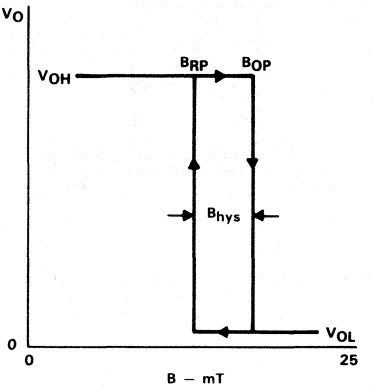


Fig. 13.3 Representative curve of V_O vs B

The typical switching device is used in applications calling for a normally OFF Hall-effect switch. This device turns ON in the presence of a positive magnetic field and turns OFF when the field is removed. A typical transition diagram for this type of device is shown in Fig. 13.3. Note that both the operate point (B_{OP}) and the release point (B_{RP}) are positive values. The hysteresis provides stable switching characteristics. The operating and release values, and the width of the hysteresis, are parameters that should be considered when choosing a device and magnet for a specific application. The functional block diagram for a switching device is shown in Fig. 13.4.

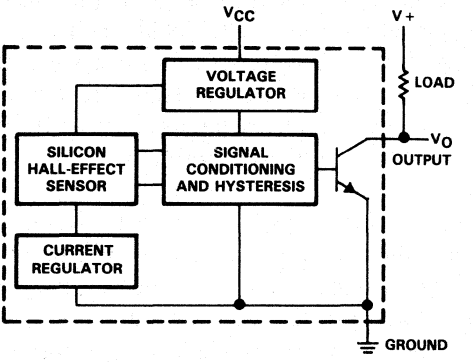


Fig. 13.4 Functional block diagram of Hall-effect switch

Latching Devices

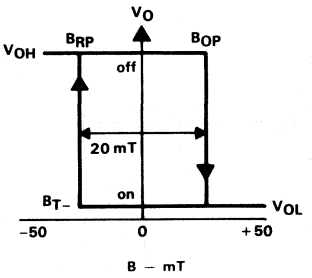


Fig. 13.5 Representative curve of V_O vs B

A Hall-effect latch is a switch that turns on in the presence of a positive magnetic field and off in the presence of a negative magnetic field. The maximum positive field strength required to turn ON a typical latch is 25 mT. The minimum field to turn a typical latch OFF is −25 mT. See Fig. 13.5 for a hysteresis-loop characteristic diagram. The output transistor is turned on when a positive field of sufficient magnitude is present and remains on until a negative magnetic field of sufficient magnitude is present.

Linear Hall-Effect Devices

A linear Hall-effect device may be defined as a magnetic field sensor designed to provide an output voltage change that is linearly proportional to a change in the applied magnetic field. Not all applications involve strictly ON/OFF switch conditions; sometimes you must know the strength of a magnetic field and its polarity. Linear Hall-effect devices contain no hysteresis circuitry, bu their sensitivity (approximately 16 mV/mT) facilities accurate magnetic-field-strength measurement.

You may also utilize such a linear sensor to determine a magnetic field's polarity if you know the device's intercept value (the point at which the sensor's output voltage characteristic crosses the zero magnetic field strength line). An output voltage greater than the intercept value indicates the presence of a north magnetic pole, while a smaller output denotes a south pole.

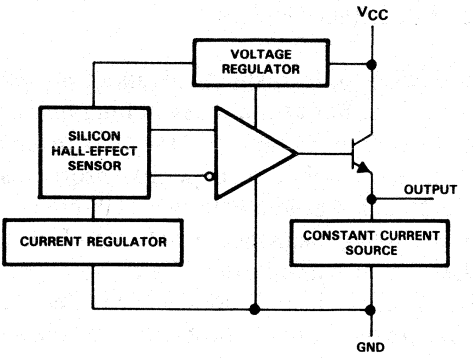


Fig. 13.6 TL3103 Functional block diagram

Fig. 13.6 shows the functional block diagram of a TL3103 linear Hall-effect device. This circuit incorporates a Hall element as the primary sensor along with a voltage reference and a precision amplifier.

Temperature stabilization and internal trimming circuitry provide a device that features high overall sensitivity accuracy with less than 5% error over its operating temperature range. The Hall voltage is amplified to provide a convenient voltage level that is proportional to the magnetic field sensed. The nominal output voltage in the presence of a zero magnetic field is 6 V. The output voltage increases 16 mV/mT with a positive magnetic field and decreases 16 mV/mT with a negative magnetic field as shown in Fig. 13.7.

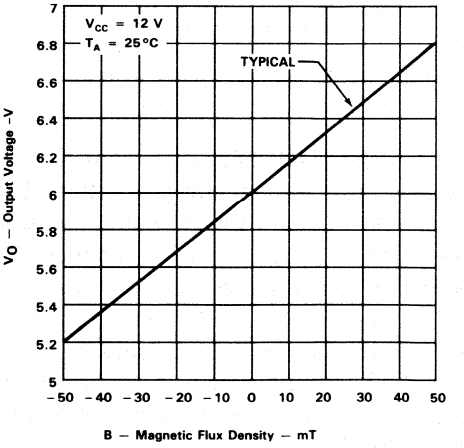


Fig. 13.7 TL3103 V_O vs magnetic flux density

Advantages of Hall-Effect Devices

Hall-effect devices give distinct advantages over mechanical and optoelectronic switches. For example, switching thresholds in Hall-effect devices do not degrade with time, as they do in emitter/sensor pairs. Additionally, while stray light affects photosensors in some applications, stray magnetic fields do not generally trigger Hall-effect sensors. As another advantage, you can isolate magnetically activated Hall-effect devices from environmental hazards such as dirt, dust, light, water, or vapor. You cannot as readily seal mechnical and optical switches and relays against contamination. Where moisture is a problem or where a switch spark might ignite explosive vapors, a specially sealed mechanical switch entails high cost. Further Hall-effect advantages over mechanical switches and relays include no contacts to wear, pit, or weld. No-contact switching implies a low failure rate and no maintenance. An internal hysteresis circuit in Hall-effect switches also eliminates contact bounce, a serious problem where mechanical switches must interface directly to a miroprocessor.

HALL-EFFECT DEVICE SELECTION

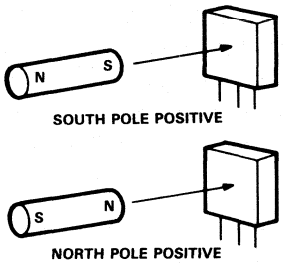


Fig. 13.8 Determining polarity

It is helpful to understand that two conventions are used by manufacturers in specifying the magnetic properties of today's devices. The definitions "into" and "out of" the cell cause confusion because both are defined in terms of the orientation of a bar magnet relative to the cell. If the magnet is perpendicular to the cell and the north pole is closest to the sensor, the field is "into the cell" when using the north-pole positive convention. If the south pole is closest to the sensor, the field is "out of" the cell. (See Fig. 13.8.) The south-pole positive convention reverses this definition. It is important to take note of the orientation of the magnet with respect to the face of the Hall-effect unit when selecting a device for a specific application.

Table 13.1 Electrical and magnetic properties of Hall-effect devices

NORTH POLE POSITIVE DEVICES

DEVICE	TYPE	V_{CC} SUPPLY	V_{OUT}	I_{OUT}	HYSTERESIS (TYPICAL mT)	SENSITIVITY (mT)
TL3101	General-purpose switch	5 V	30 V	20 mA	20	+25/–25
TL3103	Linear	12 V	5.8 to 6.2 V (B = 0)	Sink 0.5 mA Source 2 mA	-----	(Typical) 16 V/T

SOUTH POLE POSITIVE SWITCHES

DEVICE	TYPE	V_{CC} SUPPLY	V_{OUT}	I_{OUT}	HYSTERESIS (TYPICAL mT)	SENSITIVITY (mT)
TL3013	N.O.	4.5 to 40 V	40 V	30 mA	7.5	45/25
TL3019	N.O.	4.5 to 40 V	40 V	30 mA	12	50/12.5
TL3020	N.O.	4.5 to 40 V	40 V	30 mA	5.5	35/5
TL3030	General Purpose	4.5 to 40 V	40 V	30 mA	5	+25/–25
TL3040	N.O.	4.5 to 40 V	40 V	30 mA	5	20/5

N.O. = Normally open.

The TL31xx series uses the north-pole positive convention, as opposed to other similar devices that use the south-pole positive convention. Also, TI second sources UGN30xx parts with the TL30xx series, and these devices are south-pole positive. Therefore, you should check the specifications carefully before selecting a device. Table 13.1 shows a list of Hall-effect devices with respect to their electrical and magnetic properties.

HALL-EFFECT APPLICATIONS

Although the key feature of a silicon Hall-effect device is its ability to sense magnetic fields, applications are not limited to magnetic-field-related uses. You can utilize them to sense virtually any type of movement by incorporating magnetic material in the moving object. While environmental conditions such as moisture and vibration can adversely affect optical and mechanical devices, Hall-effect units are immune to most environmental conditions.

Traditionally, engineers have not used Hall-effect devices because their cost was much higher than opto or mechanical components. The cost of Hall components has dropped significantly in the past five years so this is not a significant factor in most designs. Designers can now consider using Hall sensors in many applications where mechanical or optical sensors have been used.

The following applications demonstrate methods of using Hall-effect sensors in isolated feedback applications and to sense motion or position.

TL3103 Linear Hall-Effect Device in Isolated Sensing Applications

For several years, opto coupler devices have been used for isolated sensing in power supplies. This application demonstrates how the TL3103 Hall-effect device can provide isolated sensing. The TL3103 senses the presence of either a positive or a negative magnetic field with a sensitivity of 16 mV/mT. In the absence of a magnetic field, the TL3103 output voltage is typically 6 V.

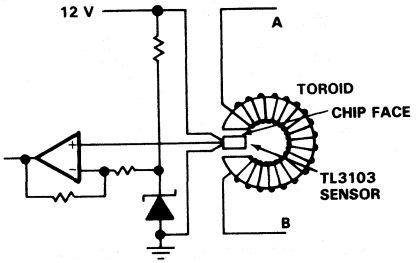


Fig. 13.9 Current measuring circuit

Because the output of the TL3103 varies proportionally to the magnetic flux density, the device can be used with a toroid to sense current or voltage. Current can be measured as shown in Fig. 13.9. When the toroid terminals A and B are connected in series with the circuit to be measured, a change in current changes the magnetic flux density in the toroid gap and causes a voltage change in the TL3103 output. Whether the change is an increase or decrease depends on the direction of current flow through the toroid. The output of the TL3103 can be used to drive an amplifier as shown in Fig. 13.9. The TL3103 output is 6 V in the absence of toroid current and flux. One input of the amplifier is therefore referenced to 6 V, allowing a level shift to 0 V at its output for zero sense current.

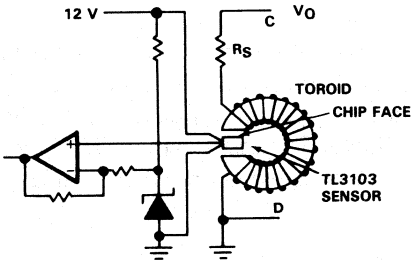


Fig. 13.10 Voltage measuring circuit

A similar arrangement provides an isolated voltage measurement. This is accomplished by connecting a resistor in series with the toroid as illustrated in Fig. 13.10. The voltage to be measured is between terminals C and D. In this configuration, the current in the toroid is determined by the output voltage (Is = Vout/Rs). Thus, the output variations of the TL3103 are proportional to the sensed output voltage variations.

Toroid Design

A variety of soft magnetic materials can be used in manufacturing cores for use with Hall-effect devices. These include manganese-zinc based ferrites, Molypermalloy powder cores, high flux powder cores, and strip wound tape cores.

The choice of the core to be gapped and used with a Hall-effect device depends upon the core characteristics required, cost, and stability over temperature. The effective permeability of the material is a function of the air gap made in the core and the initial permeability of the starting material. Permeability (μ) is defined as the ratio of magnetic flux density (B) in gauss to magnetic field intensity (H) in oersteds. (μ = B/H).

When the gap in a core exceeds a few thousandths of an inch, the effective permeability is determined essentially by the air gap. The magnetic field intensity (H) of a toroid is given by the expression NI/L where:

NI = number of turns (N) x current (I)
L = mean length of the toroid

The expression for the magnetic flux density then becomes:

$$B = (\mu)\ (NI/L)$$

With an air gap, the expression is altered to:

$$Bgap = \mu\ (NI/L) + Kg)\ K$$

μ_o = permeability of air ($12{\cdot}57 \times 10^{-7}$ W/Am)
K = relative permeability of the toroid (μ/μ_o)
g = length of the air gap
W/Am = Webers/amp-meter

As previously discussed, the output of the TL3103 is:

$$V_{sense} = 6 \text{ volts} + (16\text{mV/mT})(\mu_o\ NI/g)$$

This shows how the output of the TL3103 varies with the amp-turns of the toroid. The applications in Figs. 13.9 and 13.10 use an Arnold toroid No. A393163-2 with a 0.165 inch air gap which is sufficient for a LU package. The magnetic flux density in the air gap is:

$$B(\text{gauss}) = 1.92\ NI\ (\text{amp-turns})$$

Thus, the variation in the output of the TL3103 is:

$$V\ \text{sense (mV)} = (1.6)(1.92\ NI)$$

Therefore, the sensitivity of the TL3103 device to the current in the winding is determined by the number of turns in the winding.

V sense	N	I
614 mV	20	10 A
614 mV	200	1 A
614 mV	2000	100 mA

The features of this approach are:

1. Minimum power loss in the sensing element P loss = I^2R in the toroid ($R < 0.01$ for 20 turns)
2. Isolated feedback, no passive connection required.

TL594 Isolated Feedback Power Supply

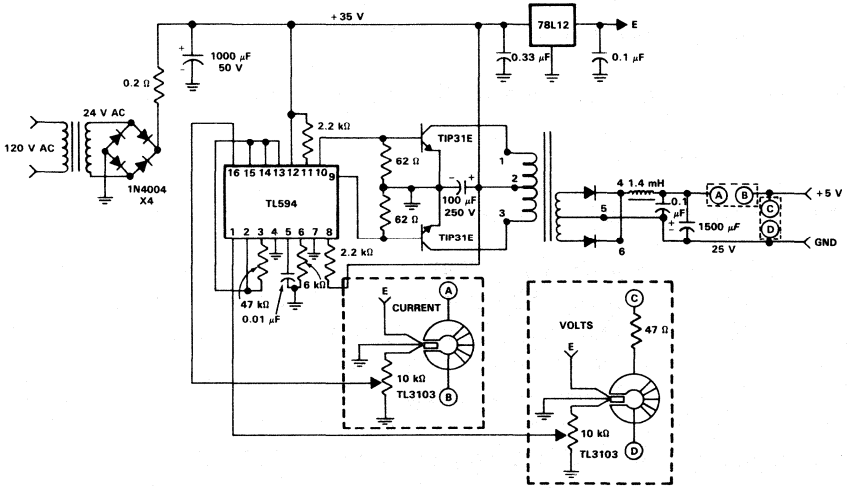


Fig. 13.11 TL594 isolated feedback power supply

Fig. 13.11 is a power supply circuit using the isolated feedback capabilities of the TL3103 for both current and voltage sensing. (See Figs. 13.9 and 13.10). This supply is powered from the ac power line and has an output of 5 V at 1.5 A. Both output voltage and current are sensed and the error voltages are applied to the error amplifiers of the TL594 PWM control IC.

The 24 V transformer produces about 35 V at the 1000 μF filter capacitor. The 20 kHz switching frequency is set by the 6 kΩ resistor and the 0.01 μF capacitor on pins 6 and 5, respectively. The TL594 is set for push-pull operation by tying pin 13 high.

The 5 V reference on pin 14 is tied to pin 15 which is the reference for the current error amplifier. The 5 V reference is also tied to pin 2 which is the reference for the output voltage error amplifier. The output voltage and current limit are set by adjustment of the 10 kΩ pots in the TL3103 error sensing circuits. A pair of TIP31E npn transistors are used as switching transistors in a push-pull circuit. The transformer design information is given in Fig. 13.12.

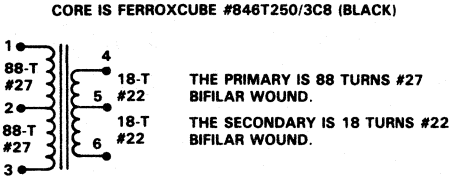


Fig. 13.12 Transformer design data

Tachometer and Direction of Rotation Circuit

In machine and equipment design, some applications require measurement of both the shaft speed and the direction of rotation. Fig. 13.13 shows the circuit of a tachometer which also indicates the direction of rotation.

Tachometer Operation

The flywheel sensor is a TL3101 Hall-effect switch selected for latch operation. The flywheel has two magnets embedded in the outer rim about 45° apart. One magnet has the north pole toward the outside and the other magnet has the south pole toward the outside rim of the flywheel. Due to the magnet spacing, a short ON pulse is produced by the TL301 in one

direction and a long ON pulse in the other direction. A 0−50 μA meter is used to monitor the flywheel speed while the LEDs indicate the direction of rotation.

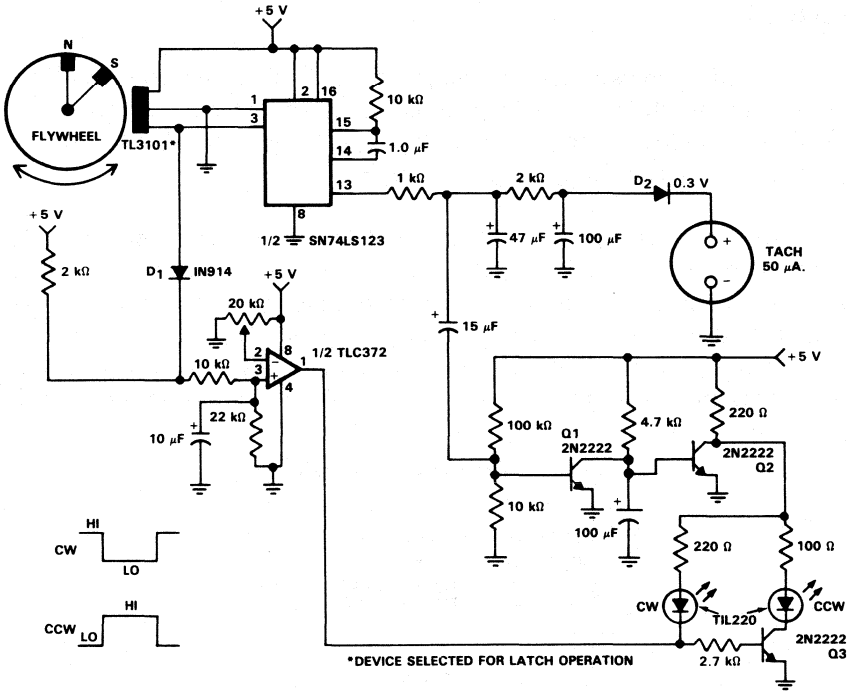


Fig. 13.13 Tachometer and direction of rotation circuit

The output from the TL3101 is applied to pin 3 (CLR 1) of an SN74LS123 one-shot multivibrator. The one-shot output pulse, at pin 13 (Q), goes high with a high input pulse. The output pulse duration is set to about 3 μs with the 1 μF capacitor and the 10 kΩ resistor on pins 14 and 15. The pulses at the output go through a low-pass filter, through diode D2, and to the 0−50 μA meter which is the tachometer. Since the TTL low-level output for the SN74LS123 is still about 0.3 V above ground, the meter would read slightly above zero when the flywheel is not moving. The purpose of the germanium diode, D2, is to produce a 0.3 V drop to correct for the 0.3 V low-level output. The meter may be calibrated in revolutions per minute (rpm), or as the user desires.

Direction of Rotation Circuit

The direction of rotation circuit can be divided into three parts:

1. TLC372 device for input conditioning and reference adjustment
2. Two 2N2222 transistors which apply the V_{CC} to the two LEDs when needed.
3. The two TIL220 LEDs which indicate clockwise (CW) or counterclockwise (CCW) direction of rotation.

The input pulses going to the SN74LS123 one-shot multivibrator are also applied through diode D1 to the noninveting input of the TLC372 device. The inverting input is connected to a 20 kΩ potentiometer, providing an adjustable reference. When the flywheel is rotating in a clockwise direction, a short-high and a long-low pulse train is produced. This causes the output of the TLC372 device to be low, turning on the CW LED and holding the CCW LED off by turning the 2N2222, Q3, off.

In the counterclockwise direction, a short-low and a long-high pulse train is produced. This gives a high output from the TLC372 device which turns the CW LED off and turns on the 2N2222 transistor and the CCW LED. The 20 kΩ reference potentiometer is set between the average voltage level of the high-level pulses and the average voltage level of the low-level pulses. The speed of the flywheel will not make any difference in the calibration of the circuit because the ratio of the high- and low-pulse lengths stays the same.

LED V_{CC} Supply Control

This circuit keeps the LEDs off when the flywheel is not rotating. The circuit consists of two 2N2222 transistor switches, Q1 and Q2, which remove the 5 V supply voltage from the LED circuit when the flywheel is not turning. When the flywheel is stopped, there are no output pulses on the SN74L123 output pin 13. At this time, there is only about 0.5 V bias voltage on the base of transistor Q1, holding it off. This causes the base of transistor Q2 to be high, which turns it on, prohibiting the supply voltage from being applied to the LEDs. With the wheel in motion a train of pulses appears at the SN74LS123 output that passes through the 15 μF capacitor to turn transistor Q1 on. This turns off transistor Q2, allowing the collector to go high. This allows the 5 V supply voltage to activate the LED circuit.

Angle of Rotation Detector

Fig. 13.14(a) shows two TL3103 linear Hall-effect devices used for detecting the angle of rotation. The TL3103s are centered in the gap of a U-shaped permanent magnet. The angle that the south pole makes with the chip face of unit No. 1 is defined as angle θ. Angle θ is set to 0° when the chip face of unit No. 1 is perpendicular to the south pole of the magnet. As the south pole of the magnet sweeps through a 0° to 90° angle, the output of the sensor increases from 0° value of V_{OQ} to a peak value of +Vp at 90°. As the magnet continues to rotate to 180°, the output of the sensor retraces its path to V_{OQ}.

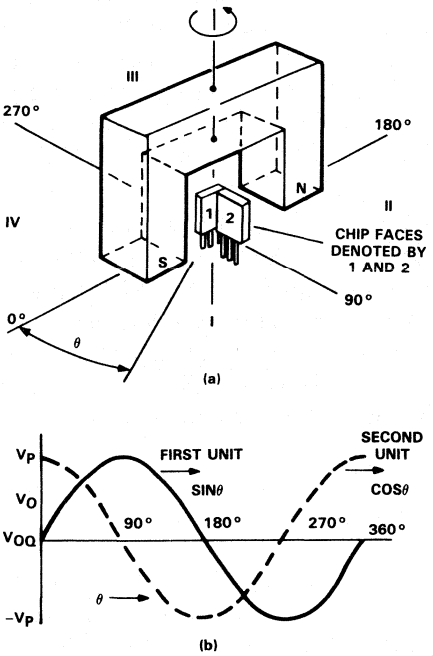


Fig. 13.14 Two linear Hall-effect devices detect angle of rotation

Sensor unit No. 2 decreases from its peak value of +Vp at 0° to a value V_{OQ} at 90°. So, the output of sensor unit No. 1 is a sine function of θ and the output of unit No. 2 is a cosine function of θ as shown in Fig. 13.14(b). Thus, the first sensor yields the angle of rotation and the second sensor indicates the quadrant location.

Hall Effect Compass

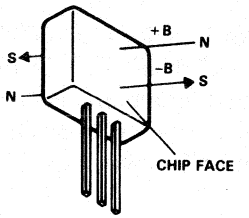


Fig. 13.15 Definition of magnetic flux polarity

The TL3103 linear Hall-effect device may be used as a compass. By definition, the north pole of a magnet is the pole that is attracted by the magnetic north pole of the earth. The north pole of a magnet repels the north-seeking pole of a compass. By convention, lines of flux emanate from the north pole of a magnet and enter the south pole. (See Fig. 13.15.)

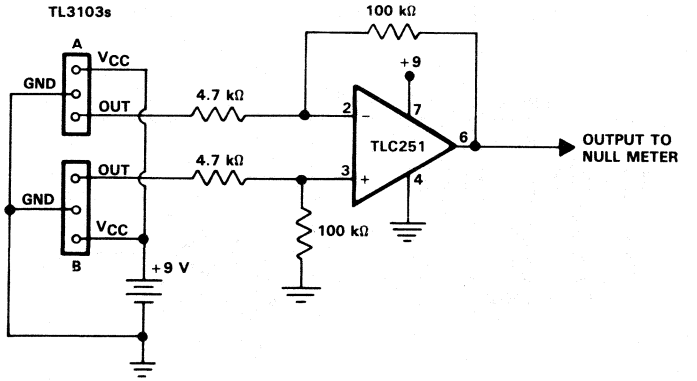


Fig. 13.16 Linear Hall-effect compass

The circuit of the compass is shown in Fig. 13.16. By using two TL3103 devices instead of one, we achieve twice the sensitivity. With each device facing the opposite direction, device A would have a positive output while the output of device B would be negative with respect to the zero magnetic field level. This gives us a differential signal to apply to the TLC251 op amp. The op amp is connected as a difference amplifier with a gain of 20. Its output is applied to a null meter or a bridge balance indicator circuit.

Security Door Alarm

In security systems for buildings, a switch of some type is installed on each door to be monitored. These may be mechanical switches or reed relay type switches operated by a permanent magnet placed in the door. A TL3019 is a normally open, south-pole-positive Hall-effect device, which may be used in this type of application. Fig. 13.17 shows the basic circuit.

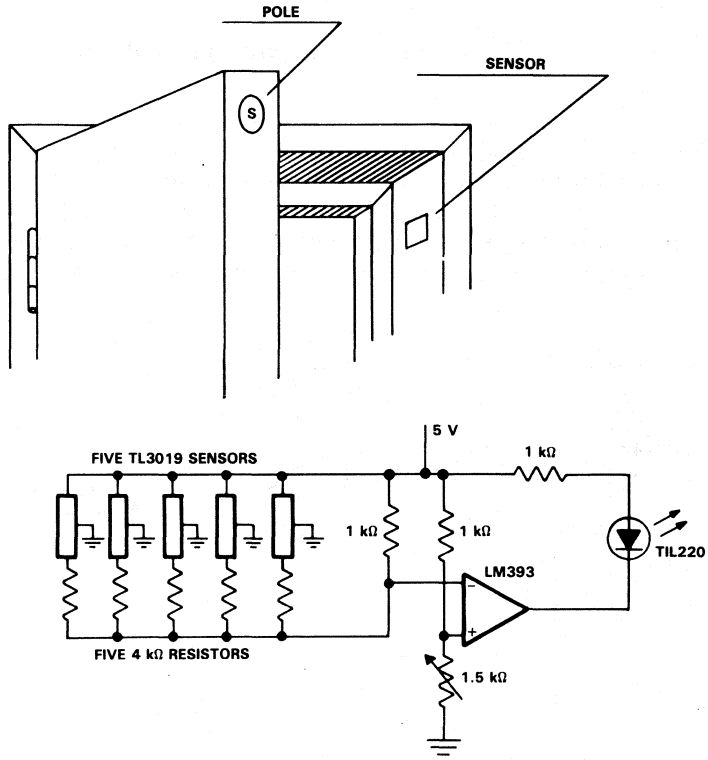


Fig. 13.17 Security door alarm

In operation, the TL3019 device will activate, or go low, when a south pole of a magnet comes near the chip face of the device. The example shows five doors. Each door has a magnet embedded in its edge with the south pole facing the outer surface. At the point where the magnet is positioned with the door closed, a TL3019 sensor is placed in the door jamb. With the door closed, the Hall devices will be in a logic low state. This design has five doors and uses five TL3019 devices. Each TL3019

has a 4 kΩ resistor in series with it and all door sensor and resistor sets are in parallel and connected to the inverting input of an LM393 comparator. With all doors closed, the effective resistance will be about 800 Ω and produce 2.2 V at the inverting input. The noninverting input goes to a voltage divider network which sets the reference voltage. The 1.5 kΩ potentiometer is adjusted so the indicator goes out with all doors closed. This will cause 2.35 V to appear at the noninverting input of the comparator. When a door opens, the voltage at the inverting input will go to 2.5 V which is greater than Vref, and the LED will light.

A large number of doors and windows may be monitored with this type of circuit. Also, it could be expanded to add an audible alarm in addition to the visual LED.

Multiple Position Control System

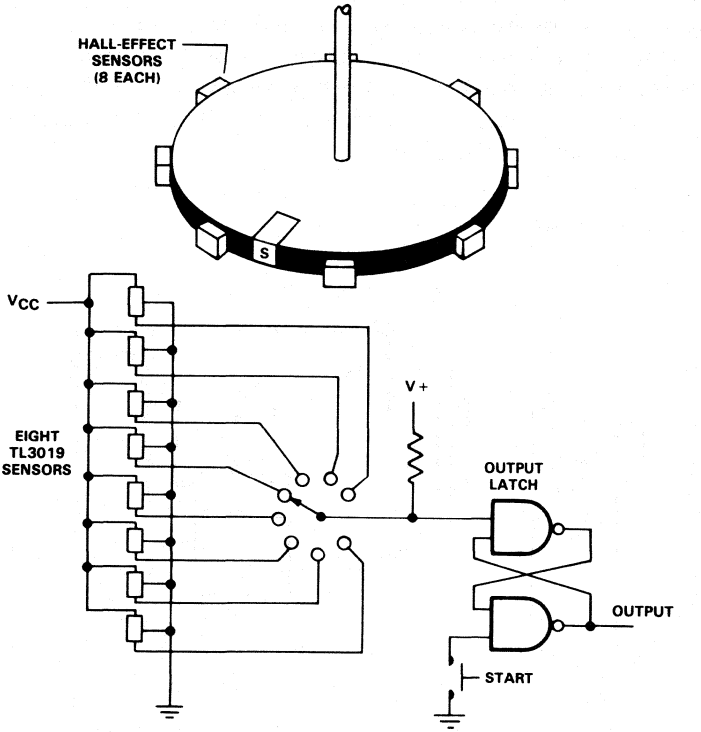


Fig. 13.18 Multiple position sensor system

In machine equipment design, it is sometimes necessary to select an operation or perform steps of another operation in a specific sequence. This application allows a drill press operator to choose a certain size bit, use it, and go to another selection.

Fig. 13.18 shows the circuit of a multiple position sensor system. Eight TL3019 Hall devices are positioned outside the rim of a rotating disc for use in drill-turret control application. Each of the normally OFF switches activates when aligned with the south (S) magnetic pole of the magnet, allowing a computer to stop the turret at any switch location. For example, if the computer chooses drill bit 4, and the turret reaches the bit 4 position, the output of sensor 4 turns on and sets an output latch. This latch output stops the turret motor. When an operator depresses the START pushbutton, the output latch resets and the turret rotates to the next selected drill bit.

There are many variations of this application that may be used in other designs. A comparatively new application is the brushless dc motor. Brushless dc motors are essentially brush-type dc motors turned inside out. Power is fed directly to the armature windings while a permanent magnet field is the rotating member. In this type of motor, a Hall-effect sensor senses the position of the rotating magnet and excites the proper windings using a logic and driver circuit.

Door Open Alarm

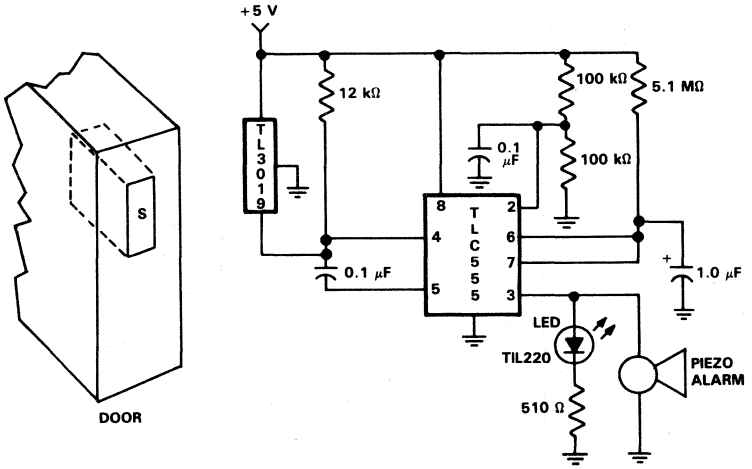


Fig. 13.19 Door open alarm

Door open alarms are used chiefly in automotive, industrial, and appliance applications. This type of circuit can sense the opening of a refrigerator door. When the door opens, a triac could be activated to control the inside light.

Fig. 13.19 shows a door position alarm. When the door is opened, an LED turns on and the piezo alarm sounds for approximately 5 seconds. This circuit use a TL3019 Hall-effect device for the door sensor. This normally open switch is located in the door frame. The magnet is mounted in the door. When the door is in the closed position, the TL3019 output goes to logic low, and remains low until the door is opened.

This design consists of a TLC555 monostable timer circuit. The 1 μF capacitor and 5·1 MΩ resistor on pins 6 and 7 set the monostable RC time constant. These values allow the LED and piezo alarm to remain on about 5 seconds when triggered. One unusual aspect of this circuit is the method of triggering. Usually a 555 timer circuit is triggered by taking the trigger, pin 2, low which produces a high at the output, pin 3. In this configuration with the door in the closed position, the TL3019 output is held low. The trigger, pin 2, is connected to 1/2 the supply voltage V_{CC}. When the door opens, a positive high pulse is applied to control pin 5 through a 0·1 μF capacitor and also to reset pin 4. This starts the timing cycle. Both the piezo alarm and the LED visual indicator are activated.

TI Sales Offices

BELGIQUE/BELGIË
S.A. Texas Instruments Belgium N.V.
11, Avenue Jules Bordetlaan 11,
1140 Bruxelles/Brussel
Tel : (02) 242 30 80
Telex : 61161 TEXBEL

DANMARK
Texas Instruments A/S
Marielundvej 46E
2730 Herlev
Tel : 02 91 74 00
Telefax : 02 91 84 00
Telex : 35123 TEXIN

DEUTSCHLAND
Texas Instruments
Deutschland GmbH.
Haggertystraße 1
8050 Freising
Tel : 0 81 61/80-0 od. Nbst
Telex : 5 26 529 texin d
Btx : *280505#

Kurfürstendamm 195-196
1000 Berlin 15
Tel : 030/8 82 73 65
Telex : 5 26 529 texin d

Düsseldorfer Straße 40
6236 Eschborn 1
Tel : 06196/80 70
Telex : 5 26 529 texin d

Ill. Hagen 43/Kibbelstraße 19
4300 Essen 1
Tel : 0201/24 25-0
Telex 5 26 529 texin d

Kirchhorster Straße 2
3000 Hannover 51
Tel : 0511/64 80 21
Telex : 5 26 529 texin d

Maybachstraße II
7302 Ostfildern 2 (Nellingen)
Tel. 0711/34 03-0
Telex 5 26 529 texin d

EIRE
Texas Instruments Ireland Ltd
7/8 Harcourt Street
Dublin 2
Tel : (01) 78 16 77
Telex : 32626

ESPAÑA
Texas Instruments España S.A.
C/José Lázaro Galdiano No. 6
28036 Madrid
Tel : (1) 458 14 58
Telex : 23634
Fax : (1) 457 94 04

C/Diputacion, 279-3-5
08007 Barcelona
Tel : (3) 317 91 80
Telex : 50436
Fax : (3) 301 84 61

FRANCE
Texas Instruments France
8-10 Avenue Morane Saulnier - B.P. 67
78141 Vélizy Villacoublay cedex
Tel : Standard : (1) 30 70 10 03
Service Technique : (1) 30 70 11 33
Telex : 698707 F

HOLLAND
Texas Instruments Holland B.V.
Hogehilweg 19
Postbus 12995
1100 AZ Amsterdam-Zuidoost
Tel : (020) 5602911
Telex : 12196

ITALIA
Texas Instruments Italia S.p.A.
Divisione Semiconduttori
Viale Europa, 40
20093 Cologno Monzese (Milano)
Tel : (02) 25300 1
Telex : 332633 MITEX I

Via Castello della Magliana, 38
00148 Roma
Tel. (06) 5222651
Telex 610587 ROTEX I
Telefax 5220447

Via Amendola, 17
40100 Bologna
Tel. (051) 554004

NORGE
Texas Instruments Norge A/S
PB 106
Refstad (Sinsenveien 53)
0585 Oslo 5
Tel : (02) 155090

ÖSTERREICH
Texas Instruments Ges.m.b.H.
Hietzinger Kai 101-105
A-1130 Wien
Tel : 0222/9100-0
Telex : 136 796

PORTUGAL
Texas Instruments Equipamento
Electronico (Portugal) LDA.
R. Eng. Frederico Ulrich, 2650
Moreira Da Maia
4470 Maia
Tel : (2) 948 1003
Telex : 22485

SCHWEIZ/SUISSE
Texas Instruments Switzerland AG
Riedstraße 6
CH-8953 Dietikon
Tel : (01) 740 22 20
Telex : 825 260 TEXIN

SUOMI FINLAND
Texas Instruments OY
Ahertajantie 3
P.O. Box 81,
0201 Espoo
Tel : (90) 0-461-422
Telex : 121457

SVERIGE
Texas Instruments
International Trade Corporation
(Sverigefilialen)
Box 30,
S-163 93 Stockholm
Visit address : Isafjordsgatan 7, Kista
Tel : (08) 793 91 70
Telefax : (08) 751 97 15
Telex : 10377 SVENTEX S

UNITED KINGDOM
Texas Instruments Ltd.
Manton Lane,
Bedford,
England, MK41 7PA
Tel : (0234) 270 111
Telex : 82178

Technical Enquiry Service
Tel : (0234) 223000

TEXAS INSTRUMENTS

TI Regional Technology Centres

DEUTSCHLAND
Texas Instruments
Deutschland GmbH.
Haggertystraße 1
8050 Freising
Tel : (08161) 80 40 43

Frankfurt/Main
Düsseldorfer Straße 40
6236 Eschborn
Tel : (0 61 96) 80 74 18

Kirchhorster Straße 2
3000 Hannover 51
Tel : (0511) 64 80 21

Maybachstraße 11
7302 Ostfildern 2 (Nellingen)
Stuttgart
Tel : (0711) 34 03-0

FRANCE
Centre de Technologie
Texas Instruments France
8-10 Avenue Morane Saulnier, B.P. 67
78141 Vélizy Villacoublay cedex
Tel : Standard : (1) 30 70 10 03
Service Technique : (1) 30 70 11 33
Telex : 698707 F

Texas Instruments France
B. P. 5
06270 Villeneuve-Loubet
Tel : 93 22 20 01
Telex : 470127 F

HOLLAND
Texas Instruments Holland B.V.
Hogehilweg 19
Postbus 12995
1100 AZ Amsterdam-Zuidoost
Tel : (020) 5602911
Telex : 12196

ITALIA
Texas Instruments Italia S.p.A.
Divisione Semiconduttori
Viale Europa, 40
20093 Cologno Monzese (Milano)
Tel : (02) 25300 1
Telex : 332633 MITEX I

SVERIGE
Texas Instruments
International Trade Corporation
(Sverigefilialen)
Box 30
S-164 93 Kista
Isafjordsgatan 7
Tel : (08) 752 58 00
Telefax : (08) 751 97 15
Telex : 10377 SVENTEX

UNITED KINGDOM
Texas Instruments Ltd.
Regional Technology Centre
Manton Lane,
Bedford,
England, MK41 7PA
Tel : (0234) 270 111
Telex : 82178

Technical Enquiry Service
Tel : (0234) 223000

S/C.REV.21

Printed in Great Britain