Signetics

Integrated Fuse Logic 1981

FOREWORD

Signetics Integrated Fuse Logic elements combine into single-chip dense arrays of gates, buffers, and flip-flops interconnected via fusible Ni-Cr links. IFL, in effect, lifts circuit connections from the printed circuit board and integrates them on chip where they can be selectively blown by the user with standard PROM programming equipment.

The flexible architecture of Signetics IFL elements allows a "firmware" approach to the synthesis of complex logic functions which result in distinct design advantages. Specifically, most random logic designs using discrete TTL elements can be condensed into fewer IC packages, dramatically reducing overall system cost. Also, since IFL devices can be customized or edited in the field without retooling, your products can benefit from shorter development cycles, custom design flexibility, and quick recovery from design errors.

With the ability to manipulate a flexible logic system quick to debug and adapt to changes in architecture, you gain a competitive edge, not only by compacting in a system more functions, speed, and cost advantages, but by getting your products to the market ahead of your competitors.

Revised November 1981

Contents

DEFINITION OF TERMS

Data Sheet Identification	Product Status	Definition
Preview	Formative or In Design	This data sheet contains the design specifications for product development. Specifications may change in any manner without notice.
Advance Information	Sampling or Pre-Production	This data sheet contains advance information and specifications are subject to change without notice.
Preliminary	First Production	This data sheet contains preliminary data and supplementary data will be published at a later date. Signetics reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.
No Identification Noted	Full Production	This data sheet contains final specifications. Signetics reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.

FIELD PROGRAMMABLE LOGIC ARRAY (16X48X8) 82S100 (T.S.)/82S101 (O.C.)

INTEGRATED FUSE LOGIC
SERIES 28

DESCRIPTION

The 82S100 (tri-state outputs) and the 82S101 (open collector outputs) are Bipolar Programmable Logic Arrays, containing 48 product terms (AND terms), and 8 sum terms (OR terms). Each OR term controls an output function which can be programmed either true active-high (Fp), or true active-low ($\overline{F}_p$). The true state of each output function is activated by any logical combination of 16-input variables, or their complements, up to 48 terms. Both devices are field programmable, which means that custom patterns are immediately available by following the fusing procedure outlined in this data sheet.

The 82S100 and 82S101 are fully TTL compatible, and include chip-enable control for expansion of input variables, and output inhibit. They feature either open collector or tri-state outputs for ease of expansion of product terms and application in bus-organized systems.

Both devices are available in commercial and military temperature ranges. For the commercial temperature range (0°C to +75°C) specify N82S100/101, F or N, and for the military temperature range (−55°C to +125° C) specify S82S100/101, F or G, I, R.

LOGIC FUNCTION

Typical Product Term:
$P_0 = I_0 \bullet I_1 \bullet \overline{I_2} \bullet I_5 \bullet \overline{I_{13}}$

Typical Output Functions: @ $\overline{CE}$ = 0:
$F_0 = (P_0 + P_1 + P_2)$ @ L = Closed
$F_0 = (\overline{P_0} \bullet \overline{P_1} \bullet \overline{P_2})$ @ L = Open

NOTE

For each of the 8 outputs, either the function Fp (active-high) or $\overline{F}p$ (active low) is available, but not both. The required function polarity is programmed via link (L).

FEATURES

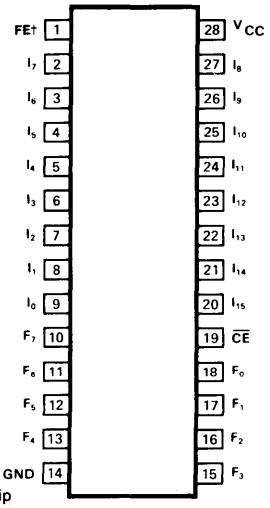
- **Field programmable (Ni-Cr link)**
- **Input variables: 16**
- **Output functions: 8**
- **Product terms: 48**
- **Address access time:**
  S82S100/101—80ns Max
  N82S100/101—50ns Max
- **Power dissipation: 600mW typ**
- **Input loading:**
  S82S100/101: −150μA Max
  N82S100/101: −100μA Max
- **Chip enable input**
- **Output option:**
  82S100: Tri-state
  82S101: Open collector
- **Output disable function:**
  Tri-state—Hi-Z
  Open collector—Hi

APPLICATIONS

- CRT display systems
- Random logic
- Code conversion
- Peripheral controllers
- Function generators
- Look-up and decision tables
- Microprogramming
- Address mapping
- Character generators
- Sequential controllers
- Data security encoders
- Fault detectors
- Frequency synthesizers

PIN CONFIGURATION

F,G,I,N,R PACKAGES

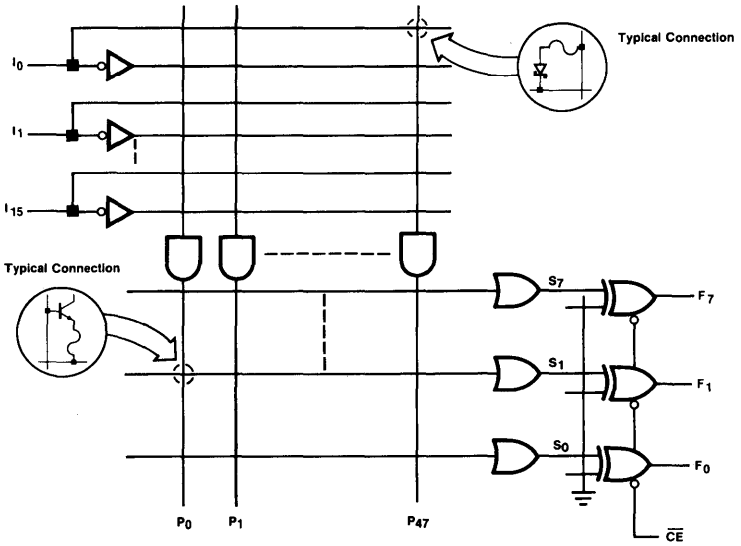


F = Cerdip
G = Leadless I = Ceramic N = Plastic
†Open or grounded during normal operation

TRUTH TABLE

MODE	Pn	$\overline{CE}$	Sr ? f(Pn)	Fp	$\overline{F}_p$
Disabled (82S101)	X	1	X	1	1
Disabled (82S100)				Hi-Z	Hi-Z
Read	1	0	Yes	1	0
	0	0		0	1
	X	0	No	0	1

LOGIC DIAGRAM

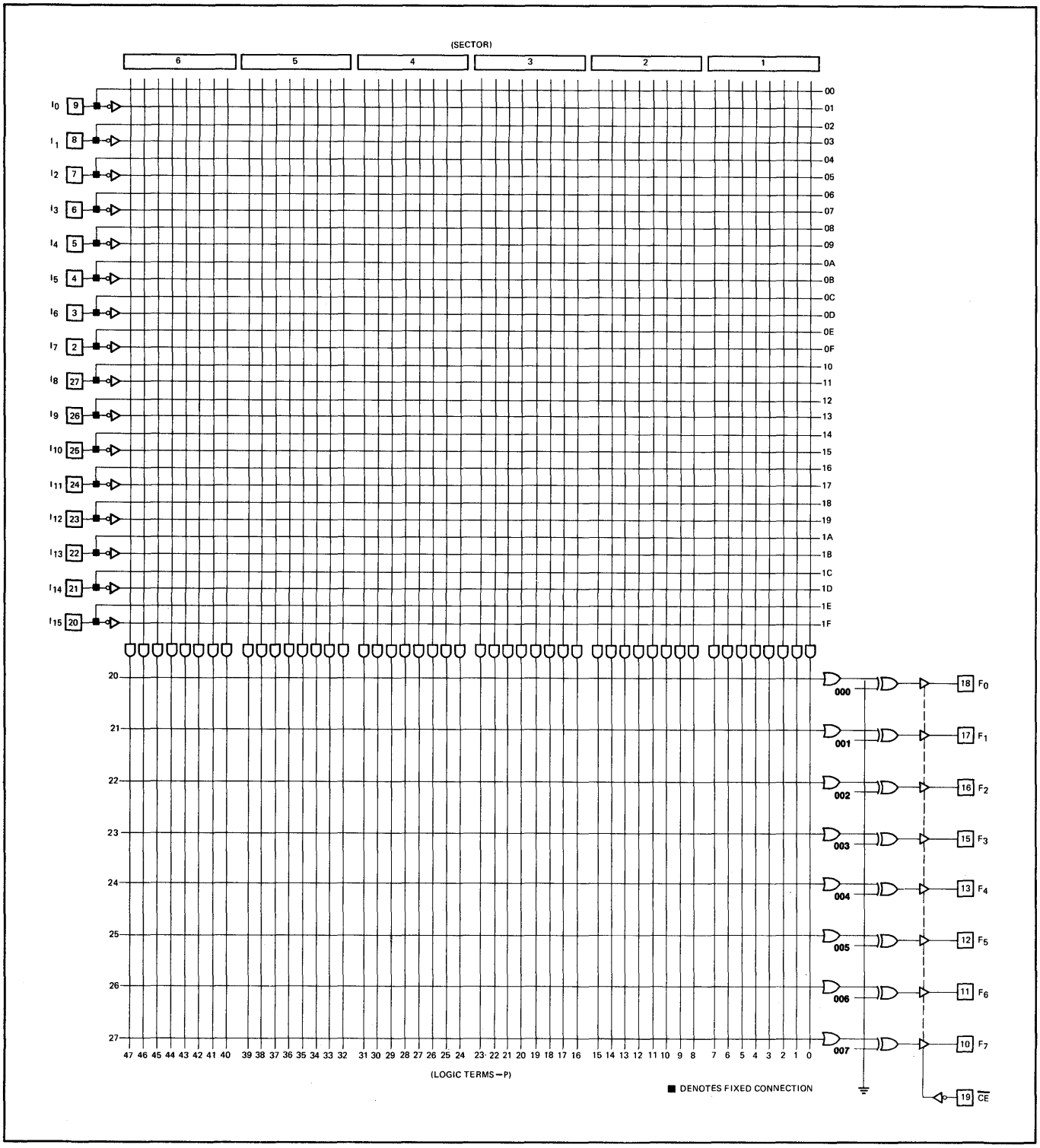


Notes
1. Fusible Ni-Cr links are initially intact at all array cross-points.
2. ■ = fixed connection.

FIELD PROGRAMMABLE LOGIC ARRAY (16X48X8) 82S100 (T.S.)82S101 (O.C.)

INTEGRATED FUSE LOGIC
SERIES 28

FPLA LOGIC DIAGRAM



FIELD PROGRAMMABLE LOGIC ARRAY (16X48X8) 82S100 (T.S.)/82S101 (O.C.)

INTEGRATED FUSE LOGIC
SERIES 28

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	-30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			°C
T_A	Operating			
	N82S100/101	0	+75	
	S82S100/101	-55	+125	
T_{STG}	Storage	-65	+150	

THERMAL RATINGS

TEMPERATURE	MILITARY	COMMERCIAL
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHARACTERISTICS

N82S100/101: $0° \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S100/101: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TEST CONDITIONS	N82S100/101 Min	N82S100/101 Typ[2]	N82S100/101 Max	S82S100/101 Min	S82S100/101 Typ[2]	S82S100/101 Max	UNIT
	Input voltage[3]								V
V_{IH}	**High**	V_{CC} = Max	2			2			
V_{IL}	**Low**	V_{CC} = Min			0.85			0.8	
V_{IC}	**Clamp[3,4]**	V_{CC} = Min, I_{IN} = -18mA		**-0.8**	-1.2		**-0.8**	-1.2	
	Output voltage	V_{CC} = Min							V
V_{OH}	**High (82S100)[3,5]**	I_{OH} = -2mA	2.4			2.4			
V_{OL}	**Low[3,6]**	I_{OL} = 9.6mA		**0.35**	0.45		**0.35**	0.50	
	Input current								µA
I_{IH}	High	V_{IN} = 5.5V		**<1**	25		**<1**	50	
I_{IL}	Low	V_{IN} = 0.45V		**-10**	-100		**-10**	-150	
	Output current	$\overline{CE}$ = High, V_{CC} = Max							
I_{OLK}	**Leakage[7]**	V_{OUT} = 5.5V		**1**	40		**1**	60	µA
$I_{O(OFF)}$	**Hi-Z state (82S100)[7]**	V_{OUT} = 5.5V		**1**	40		**1**	60	µA
		V_{OUT} = 0.45V		**-1**	-40		**-1**	-60	
I_{OS}	**Short circuit (82S100)[4,8]**	$\overline{CE}$ = Low, V_{OUT} = 0V	-20		-70	-15		-85	mA
I_{CC}	**V_{CC} supply current[9]**	V_{CC} = Max		**120**	170		**120**	180	mA
	Capacitance[7]	$\overline{CE}$ = High, V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_{OUT}	Output	V_{OUT} = 2.0V		17			17		

AC ELECTRICAL CHARACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1k\Omega$, $C_L = 30pF$
N82S100/101: $0°C \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S100/101: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TO	FROM	N82S100/101 Min	N82S100/101 Typ[2]	N82S100/101 Max	S82S100/101 Min	S82S100/101 Typ[2]	S82S100/101 Max	UNIT
	Progagation delay									ns
T_{IA}	Input	Output	Input		**35**	50		**35**	80	
T_{CE}	Chip enable	Output	Chip enable		**15**	30		**15**	50	
	Disable time									ns
T_{CD}	Chip disable	Output	Chip enable		**15**	30		**15**	50	

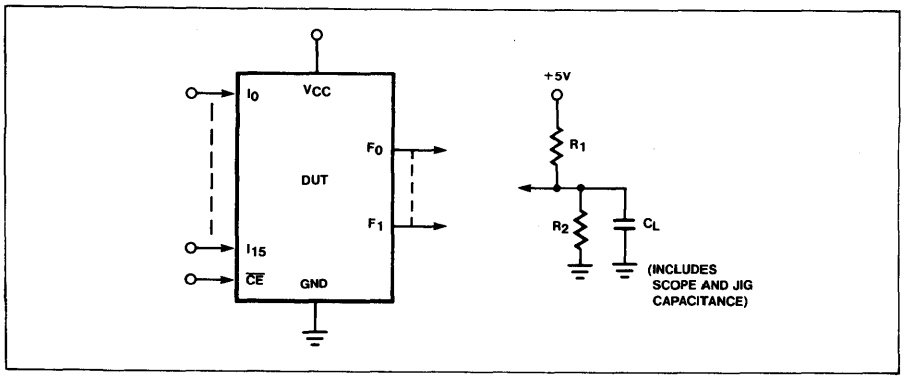
NOTES on following page.

FIELD PROGRAMMABLE LOGIC ARRAY (16X48X8) 82S100 (T.S.)82S101 (O.C.)

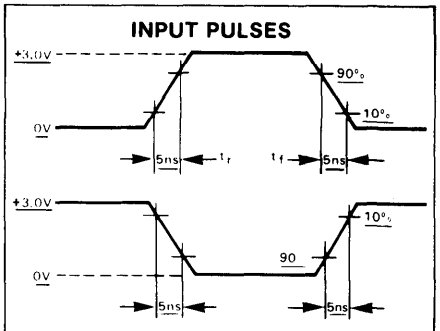
INTEGRATED FUSE LOGIC
SERIES 28

NOTES
1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions above those indicated in the operation of the device specifications is not implied.
2. All voltage values are at V_{CC} = 5V, T_A = 25°C.
3. All voltage values are with respect to network ground terminal.
4. Test one at a time.
5. Measured with V_{IL} applied to $\overline{CE}$ and a logic high stored.
6. Measured with a programmed logic condition for which the output test is at a low logic level. Output sink current is applied thru a resistor to V_{CC}.
7. Measured with V_{IH} applied to $\overline{CE}$.
8. Duration of short circuit should not exceed 1 second.
9. I_{CC} is measured with the chip enable input grounded, all other inputs at 4.5V and the outputs open.

TEST LOAD CIRCUIT

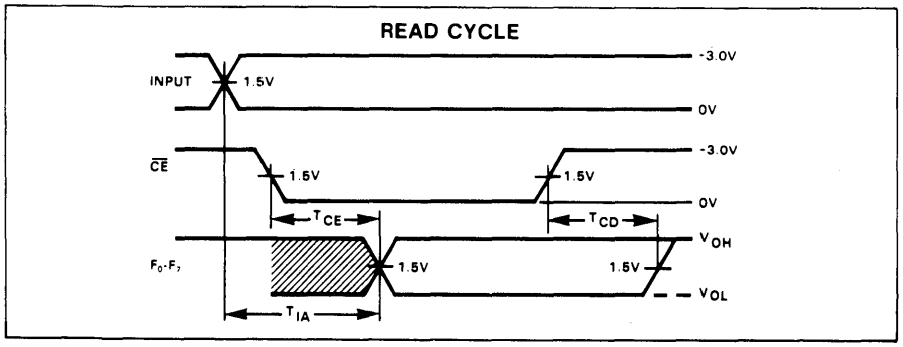


VOLTAGE WAVEFORM



Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs.

TIMING DIAGRAM



TIMING DEFINITIONS

T_{CE} **Delay between beginning of Chip Enable low (with Input valid) and when Data Output becomes valid.**

T_{CD} Delay between when Chip Enable becomes high and Data Output is in off state (Hi-Z or high).

T_{IA} Delay between beginning of valid Input (with Chip Enable low) and when Data Output becomes valid.

VIRGIN DEVICE

The 82S100/101 are shipped in an unprogrammed state, characterized by:

1. All internal Ni-Cr links are intact.
2. Each product term (P-term) contains both true and complement values of every input variable I_m (P-terms always logically "false").
3. The "OR" Matrix contains all 48-P-terms.
4. The polarity of each output is set to active high (Fp function).
5. All outputs are at a low logic level.

RECOMMENDED PROGRAMMING PROCEDURE

To program each of 8 Boolean logic functions of 16 true or complement variables, including up to 48 P-terms, follow the Program/Verify procedures for the "AND" matrix, "OR" matrix, and output polarity outlined below. To maximize recovery from programming errors, leave all links in unused device areas intact.

SET-UP

Terminate all device outputs with a 10K resistor to +5V. Set GND (pin 14) to 0V.

Output Polarity

PROGRAM ACTIVE LOW ($\overline{Fp}$ FUNCTION)

Program output polarity before programming "AND" matrix and "OR" matrix. Program 1 output at the time. (L) links of unused outputs are not required to be fused.

1. Set FE (pin 1) to V_{FEL}.
2. Set V_{CC} (pin 28) to V_{CCL}.
3. Set $\overline{CE}$ (pin 19), and I_0 through I_{15} to V_{IH}.
4. Apply V_{OPH} to the appropriate output, and remove after a period t_p.
5. Repeat step 4 to program other outputs.

VERIFY OUTPUT POLARITY

1. Set FE (pin 1) to V_{FEL}; set V_{CC} (pin 28) to V_{CCS}.
2. Enable the chip by setting $\overline{CE}$ (pin 19) to V_{IL}.
3. Address a non-existent P-term by applying V_{IH} to all inputs I_0 through I_{15}.
4. Verify output polarity by sensing the logic state of outputs F_0 through F_7. All outputs at a high logic level are programmed active low (F_p function), while all outputs at a low logic level are programmed active high ($\overline{F_p}$ function).
5. Return V_{CC} to V_{CCP} or V_{CCL}.

"AND" Matrix

PROGRAM INPUT VARIABLE

Program one input at the time and one P-term at the time. All input variable links of unused P-terms are not required to be fused. However, unused input variables must be programmed as Don't Care for all programmed P-terms.

1. Set FE (pin 1) to V_{FEL}, and V_{CC} (pin 28) to V_{CCP}.
2. Disable all device outputs by setting $\overline{CE}$ (pin 19) to V_{IH}.
3. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.
4. Address the P-term to be programmed (No. 0 through 47) by forcing the corresponding binary code on outputs F_0 through F_5 with F_0 as LSB. Use standard TTL logic levels V_{OHF} and V_{OLF}.
5a. If the P-term contains neither I_0 nor $\overline{I_0}$ (input is a Don't Care), fuse both I_0 and $\overline{I_0}$ links by executing both steps 5b and 5c, before continuing with step 7.
5b. If the P-term contains I_0, set to fuse the $\overline{I_0}$ link by lowering the input voltage at I_0 from V_{IX} to V_{IH}. Execute step 6.
5c. If the P-term contains $\overline{I_0}$, set to fuse the I_0 link by lowering the input voltage at I_0 from V_{IX} to V_{IL}. Execute step 6.
6a. After t_D delay, raise FE (pin 1) from V_{FEL} to V_{FEH}.
6b. After t_D delay, pulse the $\overline{CE}$ input from V_{IH} to V_{IX} for a period t_p.
6c. After t_D delay, return FE input to V_{FEL}.
7. Disable programmed input by returning I_0 to V_{IX}.
8. Repeat steps 5 through 7 for all other input variables.
9. Repeat steps 4 through 8 for all other P-terms.
10. Remove V_{IX} from all input variables.

VERIFY INPUT VARIABLE

1. Set FE (pin 1) to V_{FEL}; set V_{CC} (pin 28) to V_{CCP}.
2. Enable F_7 output by setting $\overline{CE}$ to V_{IX}.
3. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.
4. Address the P-term to be verified (No. 0 through 47) by forcing the corresponding binary code on outputs F_0 through F_5.
5. Interrogate input variable I_0 as follows:
   A. Lower the input voltage at I_0 from V_{IX} to V_{IH}, and sense the logic state of output F_7.
   B. Lower the input voltage at I_0 from V_{IH} to V_{IL}, and sense the logic state output F_7.

The state of I_0 contained in the P-term is determined in accordance with the following truth table:

I_0	F_7	INPUT VARIABLE STATE CONTAINED IN P-TERM
0 1	1 0	$\overline{I_0}$
0 1	0 1	I_0
0 1	1 1	Don't Care
0 1	0 0	(I_0), $(\overline{I_0})$

Note that 2 tests are required to uniquely determine the state of the input variable contained in the P-term.

6. Disable verified input by returning I_0 to V_{IX}.
7. Repeat steps 5 and 6 for all other input variables.
8. Repeat steps 4 through 7 for all other P-terms.
9. Remove V_{IX} from all input variables.

"OR" MATRIX

PROGRAM PRODUCT TERM

Program one output at the time for one P-term at the time. All P_n links in the "OR" matrix corresponding to unused outputs and unused P-terms are not required to be fused.

1. Set FE (pin 1) to V_{FEL}.
2. Disable the chip by setting $\overline{CE}$ (pin 19) to V_{IH}.
3. After t_D delay, set V_{CC} (pin 28) to V_{CCS}, and inputs I_6 through I_{15} to V_{IH}, V_{IL}, or V_{IX}.
4. Address the P-term to be programmed (No. 0 through 47) by applying the corresponding binary code to input variables I_0 through I_5, with I_0 as LSB.
5a. If the P-term is contained in output function F_0 ($F_0 = 1$ or $\overline{F_0} = 0$), got to step 6, (fusing cycle not required).
5b. If the P-term is **not** contained in output function F_0 ($F_0 = 0$ or $\overline{F_0} = 1$), set to fuse the P_n link by forcing output F_0 to V_{OPF}.
6a. After t_D delay, raise FE (pin 1) from V_{FEL} to V_{FEH}.
6b. After t_D delay, pulse the $\overline{CE}$ input from V_{IH} to V_{IX} for a period t_p.
6c. After t_D delay, return FE input to V_{FEL}.
6d. After t_D delay, remove V_{OPF} from output F_0.
7. Repeat steps 5 and 6 for all other output functions.
8. Repeat steps 4 through 7 for all other P-terms.
9. Remove V_{CCS} from V_{CC}.

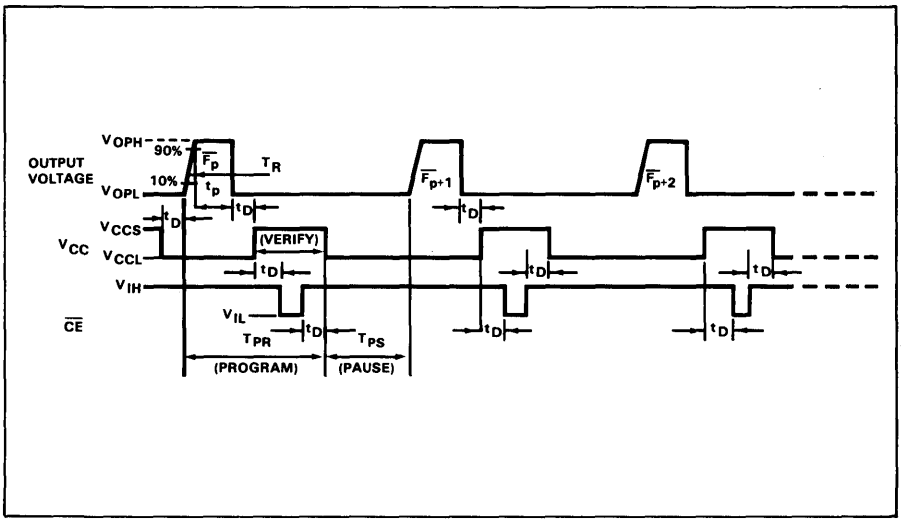
VERIFY PRODUCT TERM

1. Set FE (pin 1) to V_{FEL}.
2. Disable the chip by setting $\overline{CE}$ (pin 19) to V_{IH}.
3. After t_D delay, set V_{CC} (pin 28) to V_{CCS}, and inputs I_0 through I_{15} to V_{IH}, V_{IL}, or V_{IX}.
4. Address the P-term to be verified (No. 0 through 47) by applying the corresponding binary code to input variables I_0 through I_5.
5. After t_D delay, enable the chip by setting $\overline{CE}$ (pin 19) to V_{IL}.
6. To determine the status of the P_n link in the "OR" matrix for each output function F_p or $\overline{F_p}$, sense the state of outputs F_0 through F_7. The status of the link is given by the following truth table:

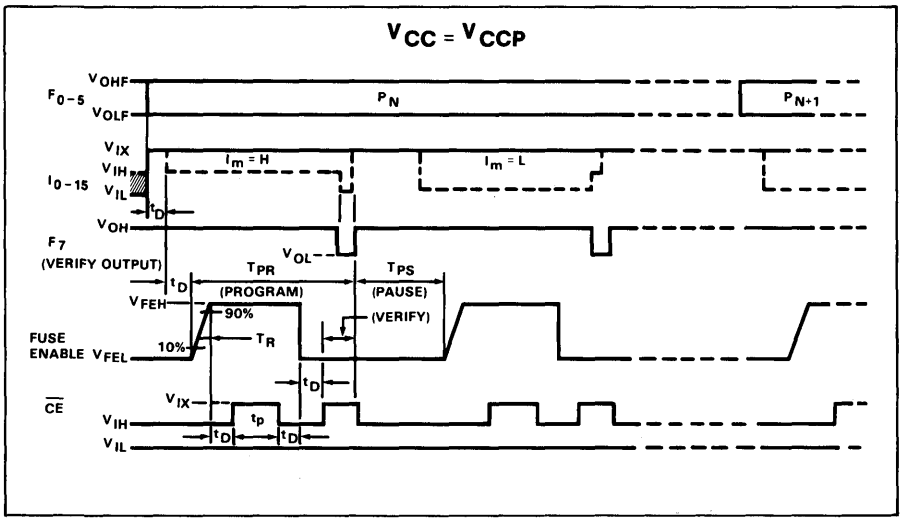
OUTPUT		P-TERM LINK
Active High (F_p)	Active Low ($\overline{F_p}$)	
0 1	1 0	**Fused** **Present**

7. Repeat steps 4 through 6 for all other P-terms.
8. Remove V_{CCS} from V_{CC}.

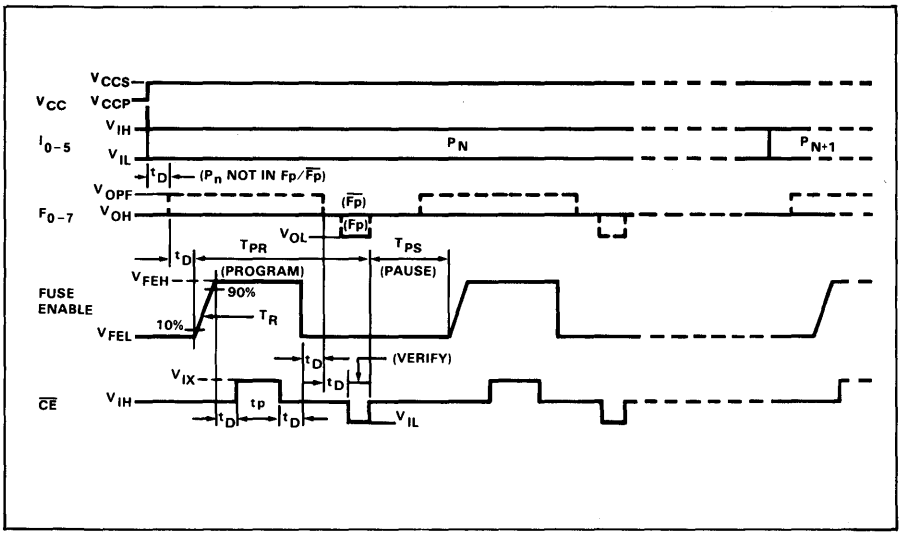
OUTPUT POLARITY PROGRAM-VERIFY SEQUENCE (TYPICAL)



"AND" MATRIX PROGRAM-VERIFY SEQUENCE (TYPICAL)



"OR" MATRIX PROGRAM-VERIFY SEQUENCE (TYPICAL)



PROGRAMMING SYSTEM SPECIFICATIONS[1] (T_A = +25°C)

PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{CCS}	V_{CC} supply (program/verify "OR", verify output polarity)[2]	I_{CCS} = 550mA, min, Transient or steady state	8.25	8.5	8.75	V
V_{CCL}	V_{CC} supply (program output polarity)		0	0.4	0.8	V
I_{CCS}	I_{CC} limit (program "OR")	V_{CCS} = +8.5 ± .25V	550		1,000	mA
	Output voltage					V
V_{OPH}	Program output polarity[3]	I_{OPH} = 300 ± 25mA	16.0	17.0	18.0	
V_{OPL}	Idle		0	0.4	0.8	
I_{OPH}	Output current limit (Program output polarity)	V_{OPH} = +17 ± 1V	275	300	325	mA
	Input voltage					V
V_{IH}	High		2.4		5.5	
V_{IL}	Low		0	0.4	0.8	
	Input current					μA
I_{IH}	High	V_{IH} = +5.5V			50	
I_{IL}	Low	V_{IL} = 0V			-500	
	Forced output voltage					V
V_{OHF}	High		2.4		5.5	
V_{OLF}	Low		0	0.4	0.8	
	Output current					
I_{OHF}	High	V_{OHF} = +5.5V			100	μA
I_{OLF}	Low	V_{OLF} = 0V			-1	mA
V_{IX}	$\overline{CE}$ program enable level		9.5	10	10.5	V
I_{IX1}	Input variables current	V_{IX} = +10V			10	mA
I_{IX2}	$\overline{CE}$ input current	V_{IX} = +10V			10	mA
V_{FEH}	FE supply (program)[3]	I_{FEH} = 300 ± 25mA, Transient or steady state	16.0	17.0	18.0	V
V_{FEL}	FE supply (idle)	I_{FEL} = -1mA, max	1.25	1.5	1.75	V
I_{FEH}	FE supply current limit	V_{FEH} = +17 ± 1V	275	300	325	mA
V_{CCP}	V_{CC} supply (program/verify "AND")	I_{CCP} = 550mA, min, Transient or steady state	4.75	5.0	5.25	V
I_{CCP}	I_{CC} limit (program "AND")	V_{CCP} = +5.0 ± .25V	550		1,000	mA
V_{OPF}	Forced output (program)		9.5	10	10.5	V
I_{OPF}	Output current (program)				10	mA
T_R	Output pulse rise time	10% to 90%	10		50	μs
t_P	$\overline{CE}$ programming pulse width		0.3	0.4	0.5	ms[5]
t_D	Pulse sequence delay		10			μs
T_{PR}	Programming time			0.6		ms
$\frac{T_{PR}}{T_{PR} + T_{PS}}$	Programming duty cycle				50	%
F_L	Fusing attempts per link				2	cycle
V_S	Verify threshold[4]		1.4	1.5	1.6	V

NOTES

1. These are specifications which a Programming System must satisy in order to be qualified by Signetics.
2. Bypass V_{CC} to GND with a 0.01μf capacitor to reduce voltage spikes.
3. Care should be taken to ensure that the voltage is maintained during the entire fusing cycle. The recommended supply is a constant current source clamped at the specified voltage limit.
4. V_S is the sensing threshold of the FPLA output voltage for a programmed link. It normally constitutes the reference voltage applied to a comparator circuit to verify a successful fusing attempt.
5. These are new limits resulting from device improvements, and which supersede, but do not obsolete the performance requirements of previously manufactured programming equipment.

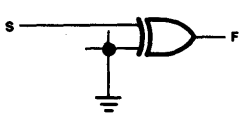
LOGIC PROGRAMMING

The FPLA can be programmed by means of Logic programming equipment.

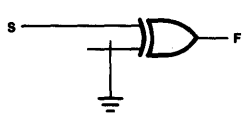
With Logic programming, the AND/OR/EX-OR gate input connections necessary to implement the desired logic function are coded directly from logic equations using the Program Table on the following page.

In this Table the logic state or action of variables I, P, and F, associated with each Sum Term S_r, is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

EX-OR ARRAY-(F)

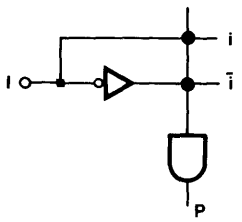


ACTIVE LEVEL	CODE
HIGH	H

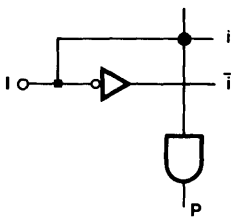


ACTIVE LEVEL	CODE
LOW	L

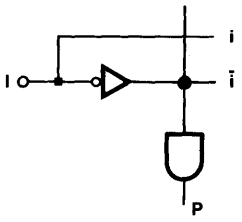
"AND" ARRAY - (I)



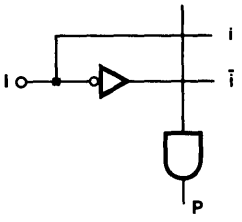
STATE	CODE
INACTIVE 1,2	O



STATE	CODE
I	H

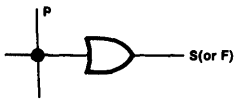


STATE	CODE
ī	L

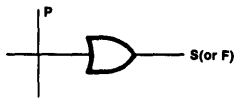


STATE	CODE
DON'T CARE	–

"OR" ARRAY - (F)



Pn STATUS	CODE
ACTIVE	A



Pn STATUS	CODE
INACTIVE	•

NOTES
1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates P_n.
2. Any gate P_n will be unconditionally inhibited if any one of its (I) link pairs is left intact.

FIELD PROGRAMMABLE LOGIC ARRAY (16X48X8) 82S100 (T.S.)82S101 (O.C.)

INTEGRATED FUSE LOGIC
SERIES 28

FPLA PROGRAM TABLE (Logic)

PROGRAM TABLE ENTRIES

INPUT VARIABLE			OUTPUT FUNCTION	OUTPUT ACTIVE LEVEL	
Im	$\overline{Im}$	Don't Care	Prod. Term Present in F_P	Active High	Active Low
H	L	– (dash)	A	H	L

NOTE: Enter (–) for unused inputs of used P-terms

NOTES:
1. Entries independent of output polarity
2. Enter (A) for unused outputs of used P-terms

NOTES:
1. Polarity programmed once only
2. Enter (H) for all unused outputs

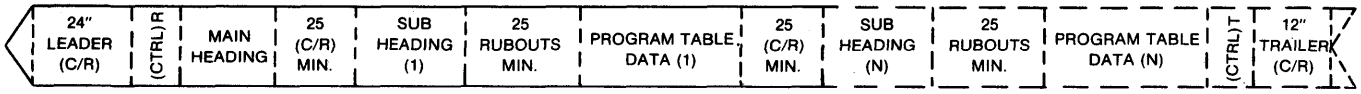
CUSTOMER NAME _____
PURCHASE ORDER # _____ CF (XXXX)
SIGNETICS DEVICE # _____
CUSTOMER SYMBOLIZED PART # _____
TOTAL NUMBER OF PARTS _____
PROGRAM TABLE # _____ REV ____ DATE ____

NO	PRODUCT TERM[1] INPUT VARIABLE[1]								ACTIVE LEVEL[1] OUTPUT FUNCTION[1]							
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																
27																
28																
29																
30																
31																
32																
33																
34																
35																
36																
37																
38																
39																
40																
41																
42																
43																
44																
45																
46																
47																
PIN NO.	20	21	22	23	24	25	26	27	2	3	4	5	6	7	8	9
									10	11	12	13	15	16	17	18
VARIABLE NAME																

TWX TAPE CODING

The FPLA Program Table can be sent to Signetics in ASCII code format via airmail using any type of 8-level tape (paper, mylar, fanfold, etc.), or via TWX: just dial (910) 339-9283, tell the operator to turn the paper puncher on, and acknowledge. At the end of transmission instruct the operator to send tape to Signetics Order Entry.

A number of Program Tables can be sequentially assembled on a continuous tape as follows, however limit tape length to a roll of 1.75 inch inside diameter, and 4.25 inch outside diameter:



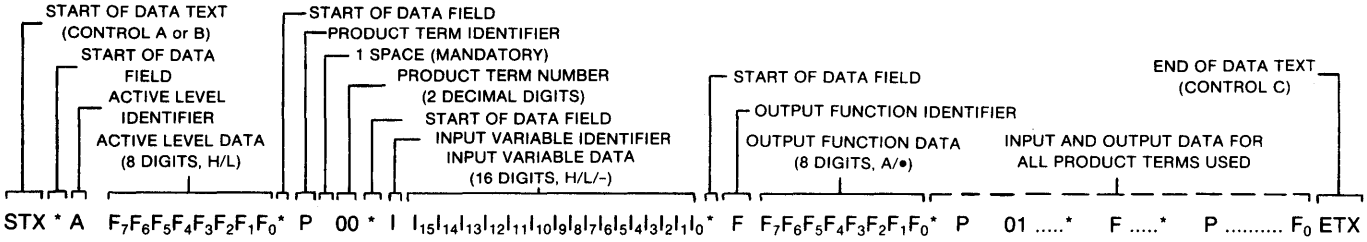
A. The MAIN HEADING at the beginning of tape includes the following information, with each entry preceded by a ($) character, whether used or not:

1. Customer Name _____
2. Customer TWX No. _____
3. Date _____
4. Purchase Order No. _____
5. Number of Program Tables _____
6. Total Number of Parts _____

B. Each SUB HEADING should contain specific information pertinent to each Program Table as follows, with each entry preceded by a ($) character, whether used or not:

1. Signetics Device No. _____
2. Program Table No. _____
3. Revision _____
4. Date _____
5. Customer Symbolized Part No. _____
6. Number of Parts _____

C. Program Table data blocks in Logic format are initiated with an STX character, and terminated with an ETX character. The body of the data consists of Output Active Level, Product Term, and Output Function information separated by appropriate identifiers in accordance with the following sequence:



Entries for the 3 Data Fields are determined in accordance with the following Table:

INPUT VARIABLE		
I_m	$\overline{I_m}$	Don't Care
H	L	— (dash)

NOTE

Enter (—) for unused inputs of used P-terms.

OUTPUT FUNCTION	
Product term present in Fp	Product term not present in Fp
A	• (period)

NOTES

1. Entries independent of output polarity.
2. Enter (A) for unused outputs of used P-terms.

OUTPUT ACTIVE LEVEL	
Active high	Active low
H	L

NOTES

1. Polarity programmed once only.
2. Enter (H) for all unused outputs.

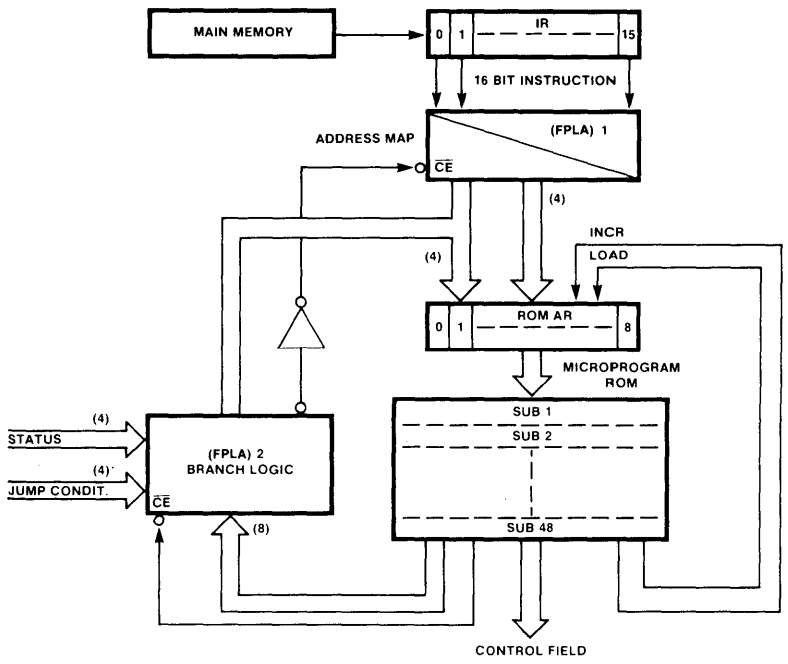
Although the Product Term data are shown entered in sequence, this is not necessary. It is possible to input only one Product Term, if desired. Unused Product Terms require no entry. ETX signalling end of Program Table may occur with less than the maximum number of Product Terms entered.

NOTES

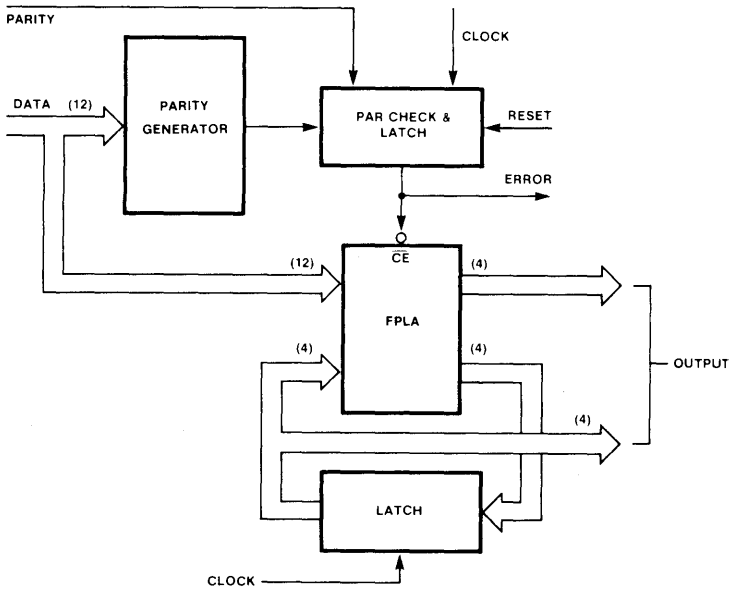
1. Corrections to any entry can be made by backspace and rubout. However, limit consecutive rubouts to less than 25.
2. Terms can be re-entered any number of times. The last entry for a particular P-Term will be interpreted as valid data.
3. To facilitate an orderly Teletype print out, carriage returns, line feeds, spaces, rubouts etc. may be interspersed between data groups.
4. Comments are allowed between data fields, provided that an asterisk (*) is not used in any Heading or Comment entry.

TYPICAL APPLICATIONS

SUBROUTINE ADDRESS MAP



SEQUENTIAL CONTROLLER



FIELD PROGRAMMABLE GATE ARRAY (16X9) 82S102 (O.C.)/82S103 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

DESCRIPTION

The 82S102 and 82S103 are Bipolar programmable AND/NAND gate array, containing 9 gates sharing 16 common inputs. On-chip input buffers enable the user to individually program for each gate either the True (I_m), Complement ($\overline{I_m}$), or Don't Care (X) logic state of each input. In addition, the polarity of each gate output is individually programmable to implement either AND or NAND logic functions.

Alternately, if desired, OR/NOR logic functions can also be realized by programming for each gate the complement of its input variables, and output (DeMorgan theorem).

Both devices are field-programmable, which means that custom patterns are immediately available by following the fusing procedure outlined in this data sheet.

The 82S102 and 82S103 include chip-enable control for output strobing and inhibit. They feature either open collector or tri-state outputs for ease of expansion of input variables and application in bus-organized systems.

Both devices are available in the commercial and military temperature ranges. For the commercial range (0°C to +75°C) specify N82S102/103, F or N, and for the military range (-55°C to +125°C) specify S82S102/103, F, G, I, and R.

FEATURES

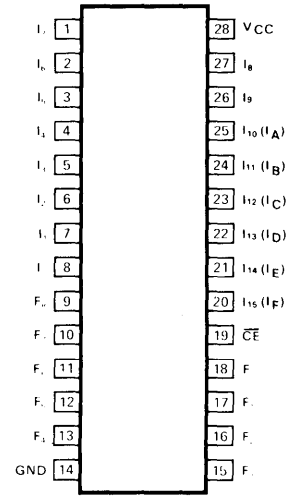
- **Field programmable (Ni-Cr link)**
- **16 input variables**
- **9 output functions**
- **Chip enable input**
- **I/O propagation delay:**
  N82S102/103: 35ns max
  S82S102/103: 50ns max
- **Power dissipation: 600mW typ**
- **Input loading:**
  N82S102/103: -100μA max
  S82S102/103: -150μA max
- **Output options:**
  82S102: Open collector
  82S103: Tri-state
- **Output disable function:**
  82S102: Hi
  82S103: Hi-Z
- **Fully TTL compatible**

APPLICATIONS

- **Random logic**
- **Address decoders**
- **Code detectors**
- **Peripheral selectors**
- **Fault monitors**
- **Machine state decoders**

PIN CONFIGURATION

F,G,I,N,R PACKAGES



F = Cerdip
G = Leadless
I = Ceramic
N = Plastic
R = Flat pack

LOGIC FUNCTION

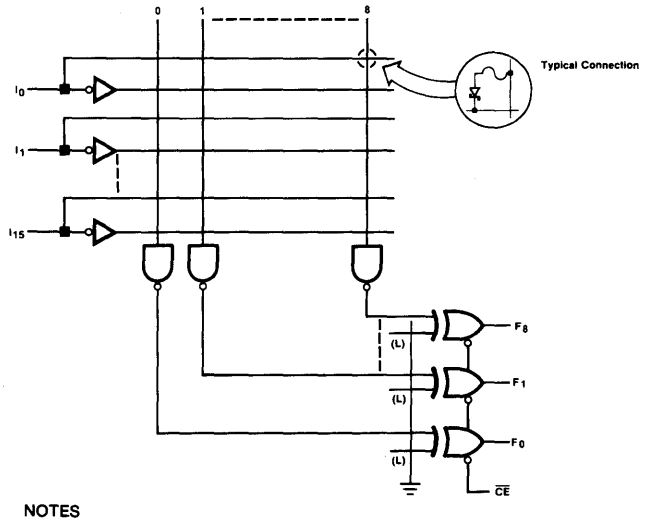
Typical Output Functions @ $\overline{CE}$ = 0:

At L = Open:
$F_0 = (I_0 \bullet I_1 \bullet I_2 \bullet \ldots \overline{I_m})$

At L = Closed:
$F_0 = (\overline{I_0} + \overline{I_1} + \overline{I_2} + \ldots I_m)$
m = 0, 1, 2, ... 15

NOTES
For each of the 9 outputs, either the function Fp (active high) or $\overline{Fp}$ (active low) is available but not both. The required function polarity is programmed via link (L).

LOGIC DIAGRAM

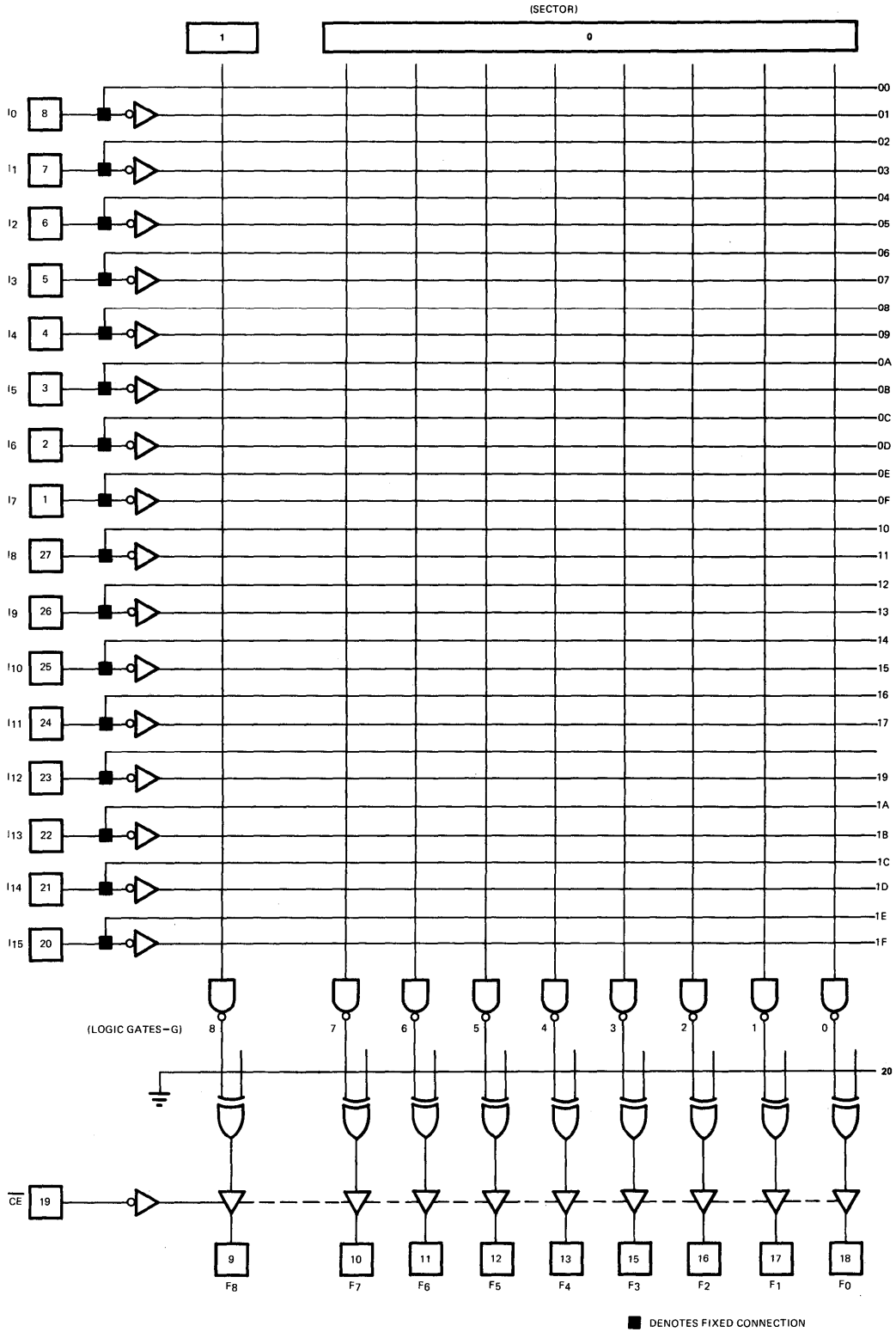


NOTES
1. Fusible Ni-Cr links are initially intact at all array cross-points.
2. ■ = fixed connection.

ABSOLUTE MAXIMUM RATINGS

PARAMETER		RATING	UNIT
V_{CC}	Supply voltage	+7	Vdc
V_{IN}	Input voltage	+5.5	Vdc
	Output voltage		Vdc
V_{OH}	High (82S102)	+5.5	
V_O	Off-state (82S103)	+5.5	
I_{IN}	Input current	±30	mA
I_{OUT}	Output current	+100	mA
	Temperature range		°C
T_A	Operating		
	N82S102/103	0 to +75	
	S82S102/103	-55 to +125	
T_{STG}	Storage	-65 to +150	

FPGA LOGIC DIAGRAM



DC ELECTRICAL CHARACTERISTICS

N82S102/103: $0°C \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S102/103: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER[1]		TEST CONDITIONS	N82S102/103			S82S102/103			UNIT
			Min	Typ[2]	Max	Min	Typ[2]	Max	
	Input voltage								V
V_{IL}	Low[1]	V_{CC} = Min			0.85			0.8	
V_{IH}	High[1]	V_{CC} = Max	2.0			2.0			
V_{IC}	Clamp[1,3]	V_{CC} = Min, I_{IN} = -18mA		-0.8	-1.2		-0.8	-1.2	
	Output voltage	V_{CC} = Min							V
V_{OL}	Low[1,4]	I_{OL} = 9.6mA		0.35	0.45		0.35	0.50	
V_{OH}	High (82S103)[1,5]	I_{OH} = -2mA	2.4			2.4			
	Input current								μA
I_{IL}	Low	V_{IN} = 0.45V		-10	-100		-10	-150	
I_{IH}	High	V_{IN} = 5.5V		<1	25		<1	50	
	Output current	V_{CC} = Max							
I_{OLK}	Leakage (82S102)[6]	V_{OUT} = 5.5V		1	40		1	60	μA
$I_{O(OFF)}$	Hi-Z state (82S103)[6]	V_{OUT} = 5.5V		1	40		1	60	μA
		V_{OUT} = 0.45V		-1	-40		-1	-60	
I_{OS}	Short circuit (82S103)[3,7]	V_{OUT} = 0V	-20		-70	-15		-85	mA
I_{CC}	V_{CC} supply current[8]	V_{CC} = Max		120	170		120	180	mA
	Capacitance	V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_{OUT}	Output[6]	V_{OUT} = 2.0V		15			15		

AC ELECTRICAL CHARACTERISTICS

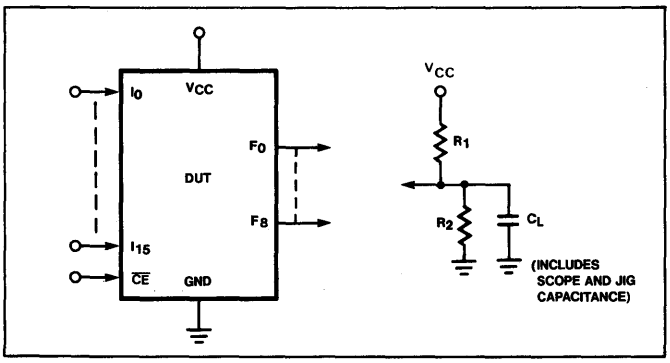
$R_1 = 470\Omega$, $R_2 = 1k\Omega$, C_L = 30pF
N82S102/103: $0°C \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S102/103: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TO	FROM	N82S102/103			S82S103/103			UNIT
				Min	Typ[2]	Max	Min	Typ[2]	Max	
	Progagation delay									ns
T_{IA}	Input	Output	Input		20	35		20	55	
T_{CE}	Chip enable	Output	Chip enable		15	30		15	45	
	Disable time									ns
T_{CD}	Chip disable	Output	Chip enable		15	30		15	45	

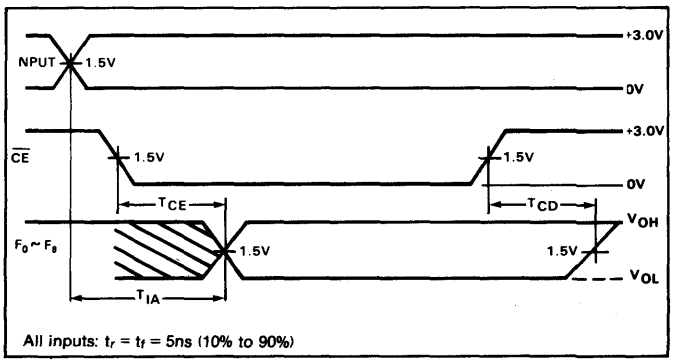
NOTES

1. All voltage values are with respect to network ground terminal.
2. All typical values are at V_{CC} = 5V, T_A = 25°C.
3. Test each output one at a time.
4. Measured with a programmed logic condition for which the output under test is at a low logic level. Output sink current is supplied through a resistor to V_{CC}.
5. Measured with V_{IL} applied to $\overline{CE}$ and a logic high at the output.
6. Measured with V_{IH} applied to $\overline{CE}$.
7. Duration of short circuit should not exceed 1 second.
8. I_{CC} is measured with the chip enable input grounded, all other inputs at 4.5V and the outputs open.

TEST LOAD CIRCUIT



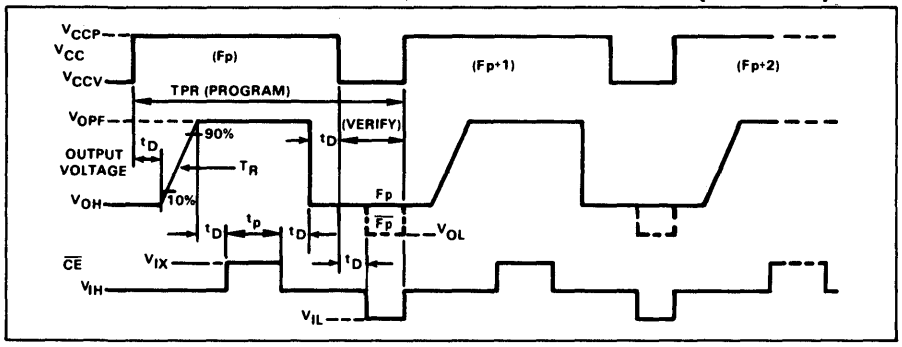
VOLTAGE WAVEFORM



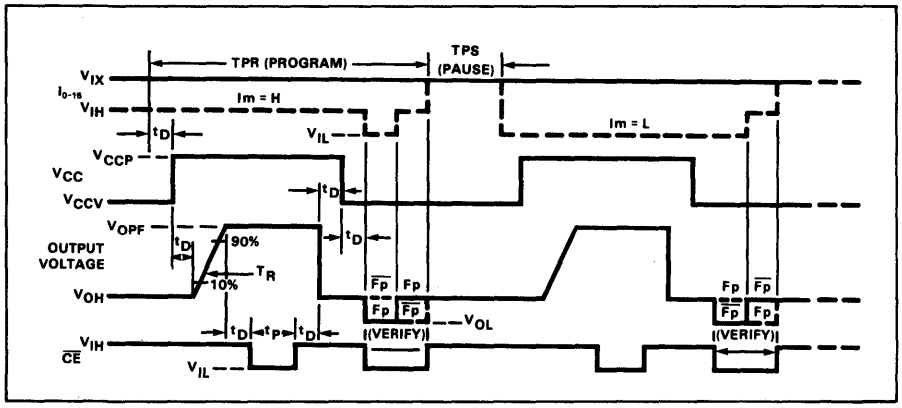
FIELD PROGRAMMABLE GATE ARRAY (16X9) 82S102 (O.C.)/82S103 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

OUTPUT POLARITY PROGRAM-VERIFY SEQUENCE (TYPICAL)



INPUT MATRIX PROGRAM-VERIFY SEQUENCE (TYPICAL)



VIRGIN DEVICE

The 82S102/103 are shipped in an unprogrammed state, characterized by:

1. All internal Ni-Cr links are intact.
2. Each gate contains both true and complement values of every input variable I_m (logic Null state).
3. The polarity of each output is set to active low ($\overline{F_p}$ function).
4. All outputs are at a high logic level.

RECOMMENDED PROGRAMMING PROCEDURE

To program each of 9 Boolean logic functions of 16 True, Complement, or Don't Care input variables follow the program/verify procedures for the Input Matrix and Output Polarity outlined below. To maximize recovery from programming errors, leave all links of unused gates intact.

SET-UP

Terminate all device outputs with a 10KΩ resistor to +5V.

Output Polarity

PROGRAM ACTIVE HIGH (Fp FUNCTION)

Program output polarity before programming inputs (for convenience). Program one output at a time. (L) links of unused outputs are not required to be fused.

1. Set GND (pin 14) to 0V, and V_{CC} (pin 28) to V_{CCV}.
2. Disable all device outputs by setting $\overline{CE}$ (pin 19) to V_{IH}.
3. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.

A. Raise V_{CC} (pin 28) from V_{CCV} to V_{CCP}.
B. After t_D delay, force output to be programmed to V_{OPF}.
C. After t_D delay, pulse the $\overline{CE}$ input from V_{IH} to V_{IX} for a period t_p.
D. After t_D delay, remove V_{OPF} voltage source from output being programmed.
E. After t_D delay, return V_{CC} (pin 28) to V_{CCV}, and verify.
F. Repeat steps A through E for any other output.

VERIFY OUTPUT POLARITY

1. Set GND (pin 14) to 0V, and V_{CC} (pin 28) to V_{CCV}.
2. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.

A. After t_D delay, set the $\overline{CE}$ input to V_{IL}.
B. Verify output polarity by sensing the logic state of outputs F_0 through F_8. All outputs at a low logic level are programmed active low ($\overline{F_p}$ function), while all outputs at a high logic level are programmed active high (F_p function).

Input Matrix

PROGRAM INPUT VARIABLE

Program one input at a time for one gate at a time. Input variable links of unused gates are not required to be fused. However, unused input variables must be programmed at Don't Care for all used gates.

1. Set GND (pin 14) to 0V, and V_{CC} (pin 28) to V_{CCV}.
2. Disable all device outputs by setting $\overline{CE}$ (pin 19) to V_{IH}.
3. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.

A-1. If a gate contains nether I_0 nor $\overline{I_0}$ (input is a Don't Care), fuse both links by executing both steps A-2 and A-3, before continuing with step C.
A-2. If a gate contains I_0, set to fuse link by lowering the input voltage at I_0 from V_{IX} to V_{IL}. Execute step B.
A-3. If a gate contains $\overline{I_0}$, set to fuse link by lowering the input voltage at I_0 from V_{IX} to V_{IL}. Execute step B.
B-1. After t_D delay, raise V_{CC} from V_{CCV} to V_{CCP}.
B-2. After t_D delay, force output of gate to be programmed to V_{OPF}.
B-3. After t_D delay, pulse the $\overline{CE}$ input from V_{IH} to V_{IL} for a period t_p.
B-4. After t_D delay, remove V_{OPF} voltage source from output of gate being programmed.
B-5. After t_D delay, return V_{CC} (pin 28) to V_{CCV}, and verify.
C. Disable programmed input by returning I_0 to V_{IX}.
D. Repeat steps A through C for all other input variables.
E. Repeat steps A through D for all other gates to be programmed.
F. Remove V_{IX} from all input variables.

VERIFY INPUT VARIABLE

Unambiguous verification of the logic state programmed for the inputs of each gate requires prior knowledge of its programmed output polarity. Therefore, the output polarity verify procedure must precede input variable verify.

1. Set GND (pin 14) to 0V, and V_{CC} (pin 28) to V_{CCV}.
2. Enable all outputs by setting $\overline{CE}$ (pin 19) to V_{IL}.
3. Disable all input variables by applying V_{IX} to inputs I_0 through I_{15}.

A. Interrogate input variable I_0 as follows: Lower the input voltage to I_0 from V_{IX} to V_{IL}, and sense the logic state of outputs F_{0-8}.

Raise the input voltage to I_0 from V_{IL} to V_{IH} and sense the logic state of outputs F_{0-8}.

The state of I_0 contained in each gate is determined in accordance with the given truth table. Note that 2 tests are required to uniquely determine the state of the input variable contained in each gate.

B. Disable verified input by returning I_0 to V_{IX}.

C. Repeat steps A and B for all other input variables.

D. Remove V_{IX} from all input variables.

TRUTH TABLE FOR INPUT VERIFICATION

I_0	Fp	$\overline{Fp}$	INPUT VARIABLE STATE
0	1	0	$\overline{I_0}$
1	0	1	
0	0	1	I_0
1	1	0	
0	1	0	Don't care
1	1	0	
0	0	1	(I_0), ($\overline{I_0}$)
1	0	1	

PROGRAMMING SYSTEMS SPECIFICATIONS[1] $T_A = 25°C$

	PARAMETER	TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
	V_{CC} supply					V
V_{CCP}	Program[2]	I_{CCP} = 550mA, min Transient or steady state	8.25	8.5	8.75	
V_{CCV}	Verify		4.75	5.0	5.25	
I_{CCP}	I_{CC} limit (program)	V_{CCP} = +8.5 ± .25V, Transient or steady state	550		1,000	mA
V_{OPF}	Forced output voltage[3] (program)	I_{OP} = 300 ± 25mA, Transient or steady state	16.0	17.0	18.0	V
I_{OPF}	Output current (program)	V_{OP} = +17 ± 1V, Transient or steady state	275	300	325	mA
	Input voltage					V
V_{IH}	High		2.4		5.5	
V_{IL}	Low		0	0.4	0.8	
	Input current					μA
I_{IH}	High	V_{IH} = +5.5V			50	
I_{IL}	Low	V_{IL} = 0V			-500	
V_{IX}	$\overline{CE}$ program enable level		9.5	10	10.5	V
I_{IX1}	Input variables current	V_{IX} = +10V			10.0	mA
I_{IX2}	$\overline{CE}$ input current	V_{IX} = +10V			10.0	mA
T_R	Output pulse rise time	10% to 90%	10		50	μs
t_P	$\overline{CE}$ programming pulse width		0.3	0.4	0.5	ms
t_D	Pulse sequence delay		10			μs
T_{PR}	Programming time			0.6		ms
$\frac{T_{PR}}{T_{PR}+T_{PS}}$	Programming duty cycle				100	%
F_L	Fusing attempts per link				2	cycle
V_S	Verify threshold[4]		1.4	1.5	1.6	V

NOTES

1. These are specifications which a Programming System must satisfy in order to be qualified by Signetics.
2. Bypass V_{CC} to GND with a 0.01μF capacitor to reduce voltage spikes.
3. Care should be taken to ensure that the voltage is maintained during the entire fusing cycle. The recommended supply is a constant current source clamped at the specified voltage limit.
4. V_S is the sensing threshold of a gate output voltage for a programmed link. It normally constitutes the reference voltage applied to a comparator circuit to verify a successful fusing attempt.

FIELD PROGRAMMABLE GATE ARRAY (16X9) 82S102 (O.C.)/82S103 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

LOGIC PROGRAMMING

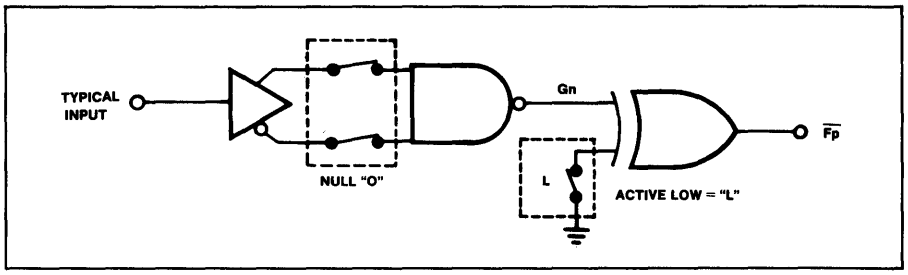
In a virgin device all Ni-Cr links are intact. The initial programmed state of each gate is shown in the Typical Gate illustration.

The FPGA can be programmed by means of Logic programming equipment.

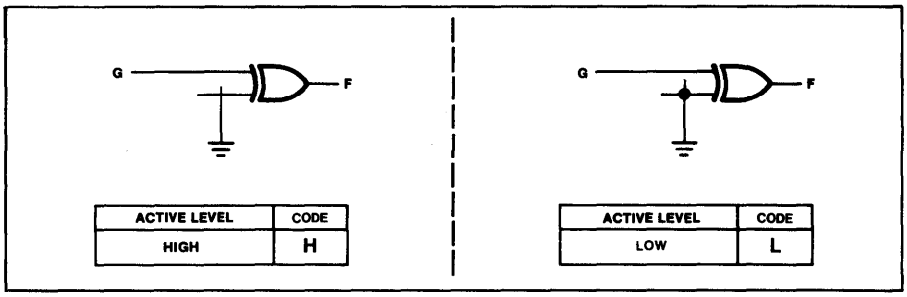
With Logic programming, the AND/EX-OR gate input connections necessary to implement the desired logic function are coded directly from logic equations using the Program Table on the following page.

In this table, the logic state or action of variables I and F associated with each gate G_n is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

TYPICAL GATE



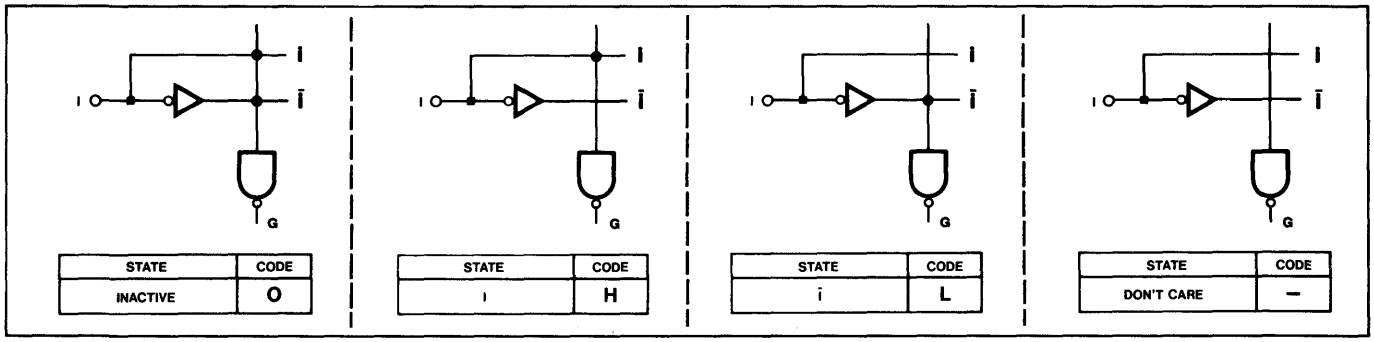
EX-OR ARRAY - (F)



ACTIVE LEVEL	CODE
HIGH	H

ACTIVE LEVEL	CODE
LOW	L

"AND" ARRAY - (I)



STATE	CODE
INACTIVE	O

STATE	CODE
I	H

STATE	CODE
$\bar{I}$	L

STATE	CODE
DON'T CARE	–

NOTES
1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates G_n.
2. Any gate G_n will be unconditionally inhibited if any one of its (I) link pairs is left intact.

FIELD PROGRAMMABLE GATE ARRAY (16X9) 82S102 (O.C.)/82S103 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPGA PROGRAM TABLE (Logic)

CUSTOMER NAME _____	*THIS PORTION TO BE COMPLETED BY SIGNETICS*
PURCHASE ORDER # _____	*CF (XXXX)* _____
SIGNETICS DEVICE # _____	*CUSTOMER SYMBOLIZED PART #* _____
TOTAL NUMBER OF PARTS _____	*DATE RECEIVED* _____
PROGRAM TABLE # _____	*COMMENTS* _____

F_0 (18) _____ =

F_1 (17) _____ =

F_2 (16) _____ =

F_3 (15) _____ =

F_4 (13) _____ =

F_5 (12) _____ =

F_6 (11) _____ =

F_7 (10) _____ =

F_8 (9) _____ =

GATE		INPUT VARIABLE															
ACTIVE LEVEL		I_{15}	I_{14}	I_{13}	I_{12}	I_{11}	I_{10}	I_9	I_8	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0
F_0	0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F_1	16	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
F_2	32	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
F_3	48	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
F_4	64	79	78	77	76	75	74	73	72	71	70	69	68	67	66	65	64
F_5	80	95	94	93	92	91	90	89	88	87	86	85	84	83	82	81	80
F_6	96	111	110	109	108	107	106	105	104	103	102	101	100	99	98	97	96
F_7	112	127	126	125	124	123	122	121	120	119	118	117	116	115	114	113	112
F_8	128	143	142	141	140	139	138	137	136	135	134	133	132	131	130	129	128
PIN NO.		20	21	22	23	24	25	26	27	1	2	3	4	5	6	7	8
VARIABLE NAME																	

Active—High = H
Action—Low = L

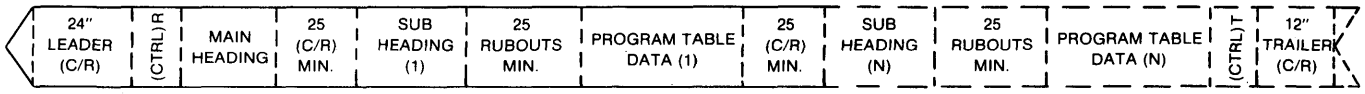
I_m = H I_m = L Don't Care = -

NOTES
1. The number in each cell in the table denotes its address for programmers with a decimal address display.

TWX TAPE CODING

The FPGA Program Table can be sent to Signetics in ASCII code format via airmail using any type of 8-level tape (paper, mylar, fanfold, etc.), or via TWX: just dial (910) 339-9283, tell the operator to turn the paper puncher on, and acknowledge. At the end of transmission instruct the operator to send tape to Signetics Order Entry.

A number of Program Tables can be sequentially assembled on a continuous tape as follows, however limit tape length to a roll of 1.75 inch inside diameter, and 4.25 inch outside diameter:



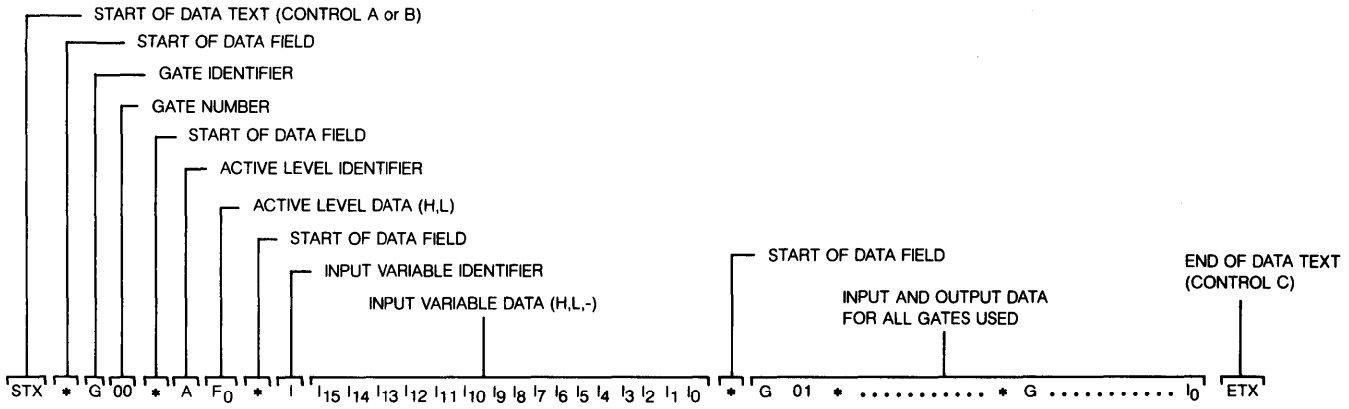
A. The MAIN HEADING at the beginning of tape includes the following information, with each entry preceded by a ($) character, whether used or not:

1. Customer Name _____
2. Customer TWX No. _____
3. Date _____
4. Purchase Order No. _____
5. Number of Program Tables _____
6. Total Number of Parts _____

B. Each SUB HEADING should contain specific information pertinent to each Program Table as follows, with each entry preceded by a ($) character, whether used or not:

1. Signetics Device No. _____
2. Program Table No. _____
3. Revision _____
4. Date _____
5. Customer Symbolized Part No. _____
6. Number of Parts _____

C. Program Table data blocks in Logic format are initiated with an STX character, and terminated with an ETX character. The body of the data consists of Output Active Level and AND gates information separated by appropriate identifiers in accordance with the following sequence:



Entries for the 2 Data Fields are determined in accordance with the following Table:

INPUT VARIABLE		
I_M	$\overline{I_M}$	Don't care
H	L	— (dash)

OUTPUT ACTIVE LEVEL	
Active high	Active low
H	L

NOTE
Enter (—) for unused inputs of used gates.

NOTES
1. Polarity programmed once only.
2. Enter (L) for all unused outputs.

Although AND Gate data are shown entered in sequence, this is not necessary. It is possible to input only one Gate if desired. Unused Gates require no entry. ETX signalling end of Program Table may occur with less than the maximum number of AND Gates entered.

NOTES
1. Corrections to any entry can be made by backspace and rubout. However, limit consecutive rubouts to less than 25.
2. Terms can be re-entered any number of times. The last entry for a particular P-Term will be interpreted as valid data.
3. To facilitate an orderly Teletype print out, carriage returns, line feeds, spaces, rubouts etc. may be interspersed between data groups.
4. Comments are allowed between data fields, provided that an asterisk (*) is not used in any Heading or Comment entry.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

DESCRIPTION

The 82S104 (open collector outputs) and the 82S105 (tri-state outputs) are bipolar, programmable state machines of the Mealy type. They contain logic AND-OR gate arrays with user programmable connections which control the inputs of on-chip State and Output registers. These consist respectively of 6 Q_P, and 8 Q_F edge triggered, clocked S/R flip-flops, with an asynchronous Preset option. All flip-flops are unconditionally preset to "1" during power turn on.

The AND array combines 16 external inputs $I_{0\text{-}15}$ with 6 internal inputs $P_{0\text{-}5}$ fed back from the State register to form up to 48 Transition terms (AND terms). All Transition terms can include True, False, or Don't Care states of the controlling variables, and are merged in the OR array to issue next-state and next-output commands to their respective registers on the Low to High transition of the Clock pulse. Both True and Complement Transition terms can be generated by optional use of the internal input variable (C) from the Complement array. Also, if desired, the Preset input can be converted to $\overline{\text{Output-Enable}}$ function, as an additional user programmable option.

Both devices are available in commercial and military temperature ranges. For the commercial temperature range (0°C to +75°C) specify N82S104/105, F or N, and for the military temperature range (−55°C to +125°C) specify S82S104/105, F, I, G or R.

FEATURES

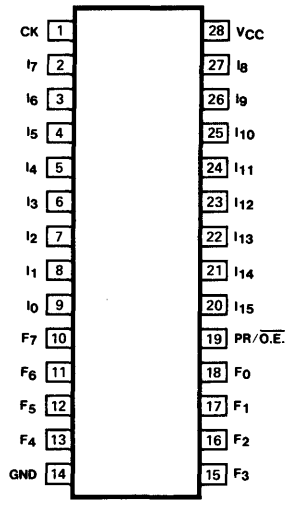
- Field programmable (Ni-Cr link)
- 16 input variables
- 8 output functions
- 48 transition terms
- 6-BIT state register
- 8-BIT output register
- Transition complement array
- Positive edge trigger clock
- Programmable asynchronous preset or output enable
- Power-on preset to all "1" of internal registers
- 90ns maximum I/O delay
- 650mW power dissipation (typical)
- TTL compatible
- Single +5V supply

APPLICATIONS

- Interface protocols
- Sequence detectors
- Peripheral controllers
- Timing generators
- Sequential circuits
- Elevator controllers
- Security locking systems

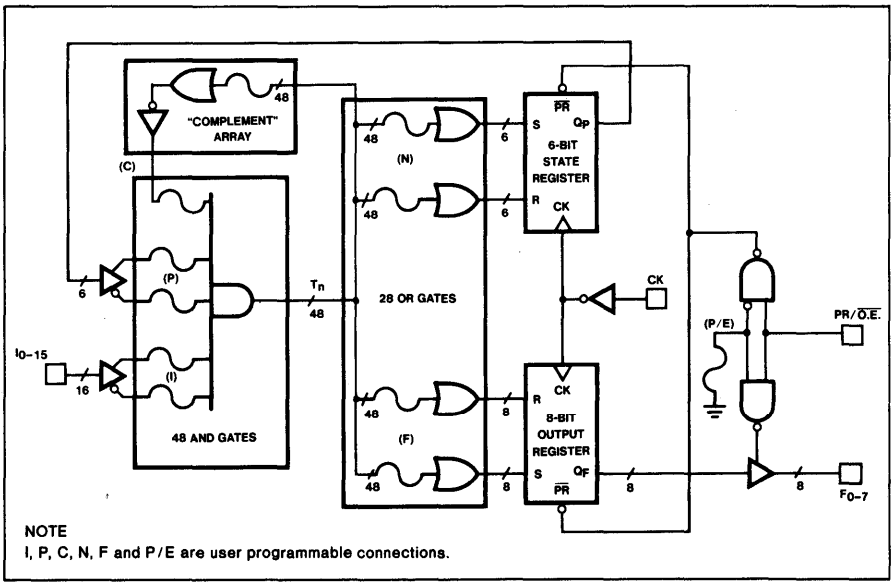
PIN CONFIGURATION

F,G,I,N,R PACKAGES



F = Cerdip I = Ceramic R = Flat Pack
N = Plastic G = Leadless

BLOCK DIAGRAM



NOTE
I, P, C, N, F and P/E are user programmable connections.

TRUTH TABLE (Output Control)

I_0	INPUT OPTION		F_N
	PR	$\overline{\text{O.E.}}$	
*	H		H
+10V	L		Q_P
X	L		Q_F
*		H	H/Hi-Z
+10V		L	Q_P
X		L	Q_F

NOTES
1. Positive Logic:
   $S/R = T_0 + T_1 + T_2 + \ldots + T_{47}$
   $T_n = C\ (I_0\ I_1\ I_2 \ldots)\ (P_0\ \overline{P_1 \ldots P_5})$
2. Either Preset (active-High) or $\overline{\text{Output Enable}}$ (active-Low) are available, but not both. The desired function is a user programmable option.
3. ↑ denotes transition from Low to High level.
4. R = S = High is an illegal input condition.
5. * = H/L/+10V
6. X = Don't Care (≤5.5V)

TRUTH TABLE (All flip-flops)

V_{CC}	INPUT OPTION		CK	S	R	STATE REGISTER	OUTPUT REGISTER
	PR	$\overline{\text{O.E.}}$				Q_P	Q_F
+5V	H		X	X	X	H	H
	L	X	↑	L	L	Q_P	Q_F
	L	X	↑	L	H	L	L
	L	X	↑	H	L	H	H
	L	X	↑	H	H	INDET.	INDET.
↑	X	X	X	X	X	H	H

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

PIN DESIGNATION

PIN NO.	SYMBOL	NAME AND FUNCTION	POLARITY
1	CK	**CLOCK** The clock input to the State and Output Registers. A Low-to-High transition on this line is necessary to update the contents of both registers.	Active-High
[2-8, 20-27]	I_{1-15}	**LOGIC INPUTS** The 15 external inputs to the AND array used to program jump conditions between machine states, as determined by a given logic sequence.	Active-High
9	I_0	**LOGIC/DIAGNOSTIC INPUT** A 16th external logic input to the AND array, as above, when exercised with standard TTL levels. When I_0 is held at +10V, device outputs F_{0-5} reflect the contents of State Register bits P_{0-5}. The contents of the Output Register remain unaltered.	Active-High
[10-13, 15-18]	F_{0-7}	**LOGIC/DIAGNOSTIC OUTPUTS** Eight device outputs which normally reflect the contents of Output Register bits Q_{0-5}, when enabled. When I_0 is held at +10V, $F_{0-5} = (P_{0-5})$, and $F_{6,7}$ = Logic "I".	Active-High
19	PR/$\overline{O.E.}$	**PRESET OR $\overline{\text{OUTPUT ENABLE}}$ INPUT** A user programmable function:	
		• **PRESET** Provides an asynchronous preset to logic "1" of all State and Output Register bits. Preset overrides Clock, and when held High, clocking is inhibited and F_{0-7} are High. Normal clocking resumes with the first full clock pulse following a High-to-Low clock transition, after Preset goes Low.	Active-High
		• **$\overline{\text{OUTPUT ENABLE}}$** Provides an output enable function to buffers F_{0-7} from the Output Register.	Active-Low

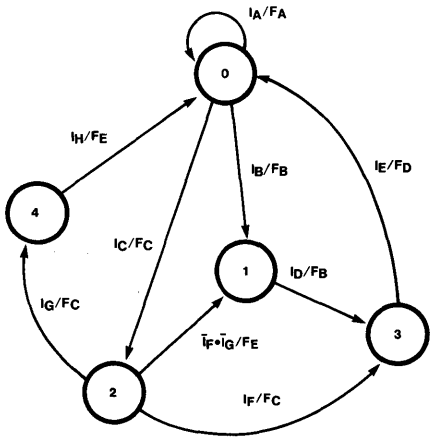
PROGRAMMABLE VARIABLES

The FPLS can be programmed with any clocked sequence expressed in terms of "control" variables, which are coupled to on-chip gates and flip-flops by means of programmable connections through Ni-Cr fusible links:

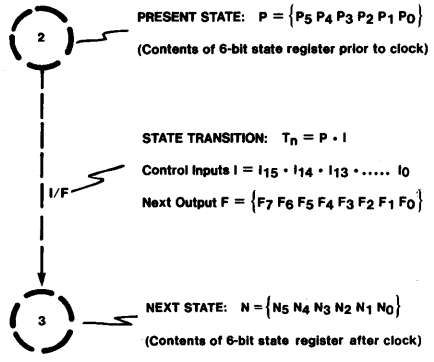
(I) - External control inputs for state jump conditions.
(P) - Present state, prior to clock.
(C) - Complement variable for activating and generating the complement of programmed jump conditions.

(T_n) - Transition "AND" terms including I, P.
(N) - Next state, following clock.
(F) - Next logic output, following clock.
(P/E) - Asynchronous preset, or output enable function.

TYPICAL STATE DIAGRAM



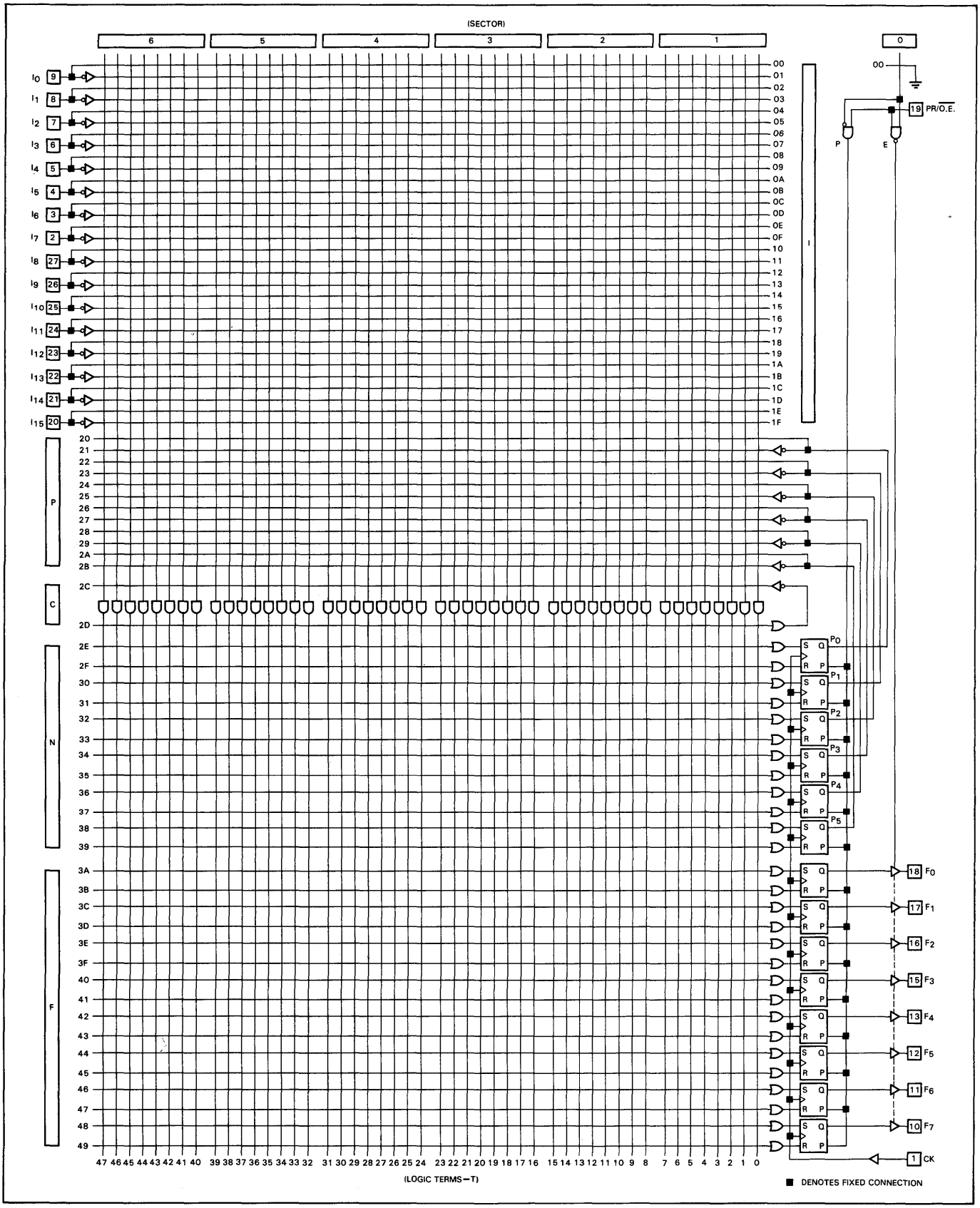
TYPICAL STATE TRANSITION



FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPLS LOGIC DIAGRAM



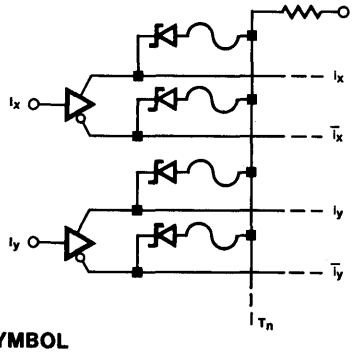
FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

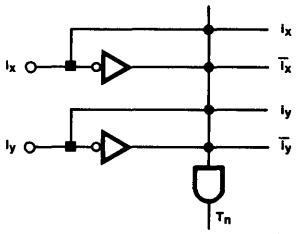
PROGRAMMABLE CONNECTIONS (■ Denotes fixed connection)

"AND" ARRAY (TYPICAL)

CONNECTION

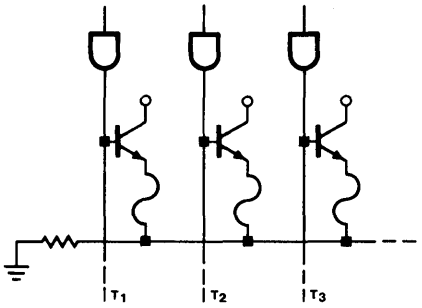


SYMBOL

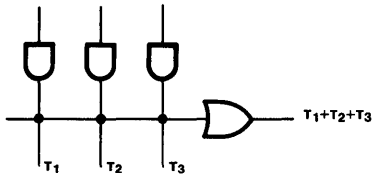


"OR" ARRAY (TYPICAL)

CONNECTION

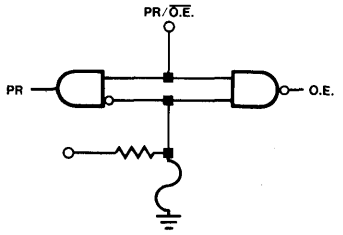


SYMBOL

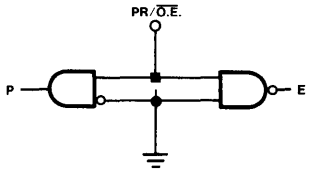


PR/$\overline{O.E.}$ OPTION

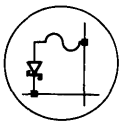
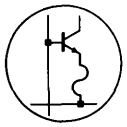
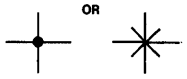
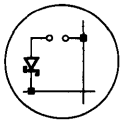
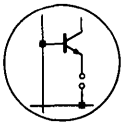
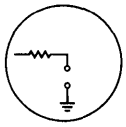
CONNECTION



SYMBOL



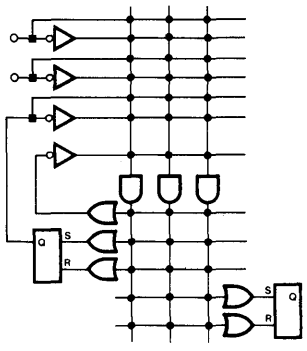
PROGRAMMING LEGEND

AND	OR	PR/$\overline{O.E.}$	LINK	SYMBOL
			CLOSED	
			OPEN	

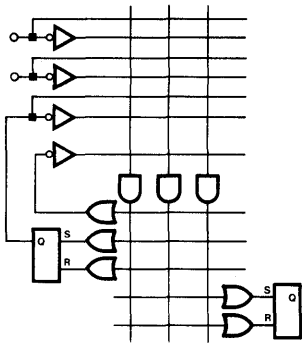
VIRGIN DEVICE

The FPLS is shipped with all internal links intact, so that a "dot" connection exists at all cross points in all arrays. For clarity, unprogrammed arrays are initially shown blank. The desired functional logic diagram is obtained by placing "dot" connections in used device areas where links remain intact. (Since both True and Complement input links of all AND gates are initially closed, all gates are disabled, preventing clocking. For testing purposes, clocking can occur via a factory programmed Test Array.

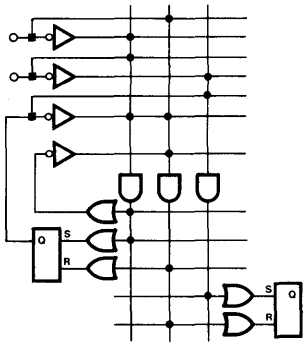
VIRGIN (PARTIAL)



SYMBOL



PROGRAMMED (TYPICAL)



FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPLS ARCHITECTURE

The 82S104/105 Logic Sequencer is a programmable state machine of the Mealy type, in which the output is a function of the present state and the present input.

With the FPLS a user can program any logic sequence expressed as a series of jumps between stable states, triggered by a valid input condition (I) at clock time (t). All stable states are arbitrarily assigned and stored in the State Register. The logic output of the machine is also programmable, and is stored in the Output Register.

CLOCKED SEQUENCE

A synchronous logic sequence can be represented as a group of circles interconnected with arrows. The circles represent stable states, labeled with an arbitrary numerical code (binary, hex, etc.) corresponding to discrete states of a suitable register. The arrows represent state transitions, labeled with symbols denoting the jump condition and the required *change* in output. The number of states in the sequence depends on the length and complexity of the desired algorithm.

STATE JUMPS

The state from which a jump originates is *referred* as the present state (P), and the state to which a jump terminates is defined as the next state (N). A state jump always causes a change in state, but may or may not cause a change in machine output (F).

State jumps can occur only via "transition terms" T_n. These are logical AND functions of the clock (t), the present state (P), and a valid input (I). Since the clock is actually applied to the State Register, $T_n = I \bullet P$. When T_n is "true", a control signal is generated and used at clock time (t) to force the contents of the State Register from (P) to (N), and to change the contents of the Output Register (if necessary). The simple state jump below, involving 2 inputs, 1 state bit, and 1 output bit, illustrates the equivalence of discrete and programmable logic implementations.

FPLS LOGIC STRUCTURE

The FPLS consists of programmable AND and OR gate arrays which control the Set and Reset inputs of a State Register, as well as monitor its output via an internal feedback path. The arrays also control an independent Output Register, added to store output commands generated during state transitions, and to hold the output constant during state sequences involving no output changes. If desired, any number of bits of the Output Register can be used to extend the width of the State Register, via external feedback.

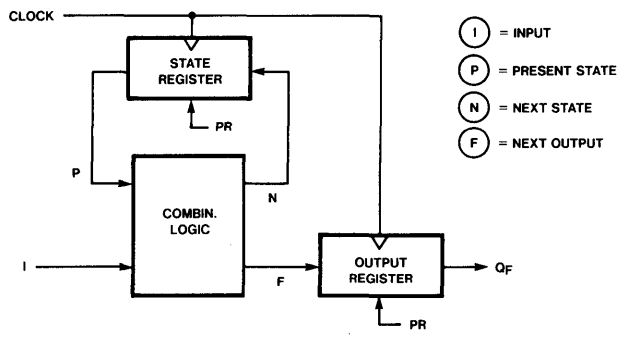


Figure 1: Basic architecture of 82S104/105 FPLS. I, P, N, and F are multi-line paths denoting groups of binary variables programmed by the user.

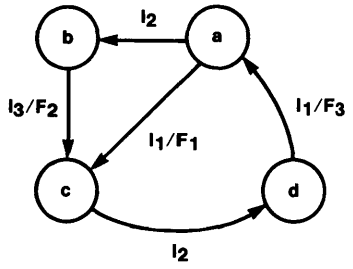


Figure 2: Typical state diagram. I_{1-3} are jump conditions which must be satisfied before any transitions take place. F_r are changes in output triggered by I_m, and stored in the Output Register. State transitions a→b and c→d involve no output change.

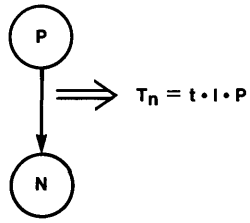


Figure 3: Typical state transition between any two states of Figure 2. The arrow connecting the two states gives rise to a transition term T_n. I is the jump condition.

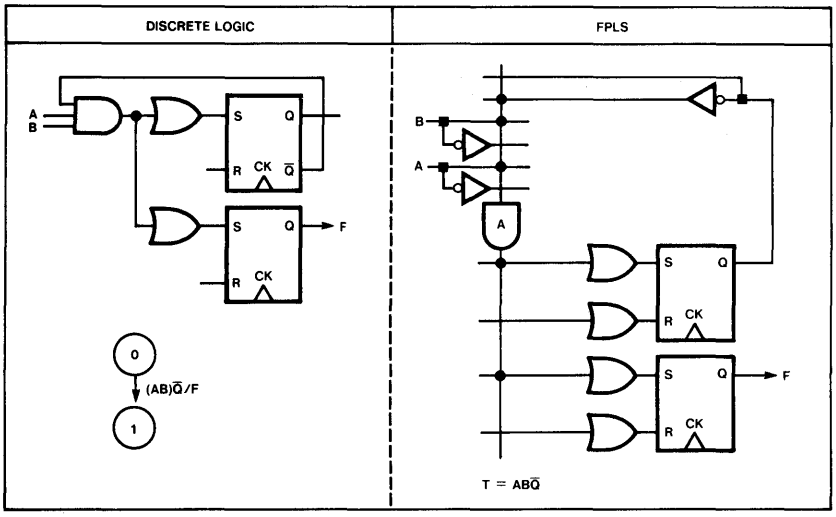


Figure 4: Typical state jump from state (0) to state (1), if inputs A=B="1". The jump also forces F="1", as required.

SIMPLIFIED LOGIC DIAGRAM OF 82S104/105 FPLS.

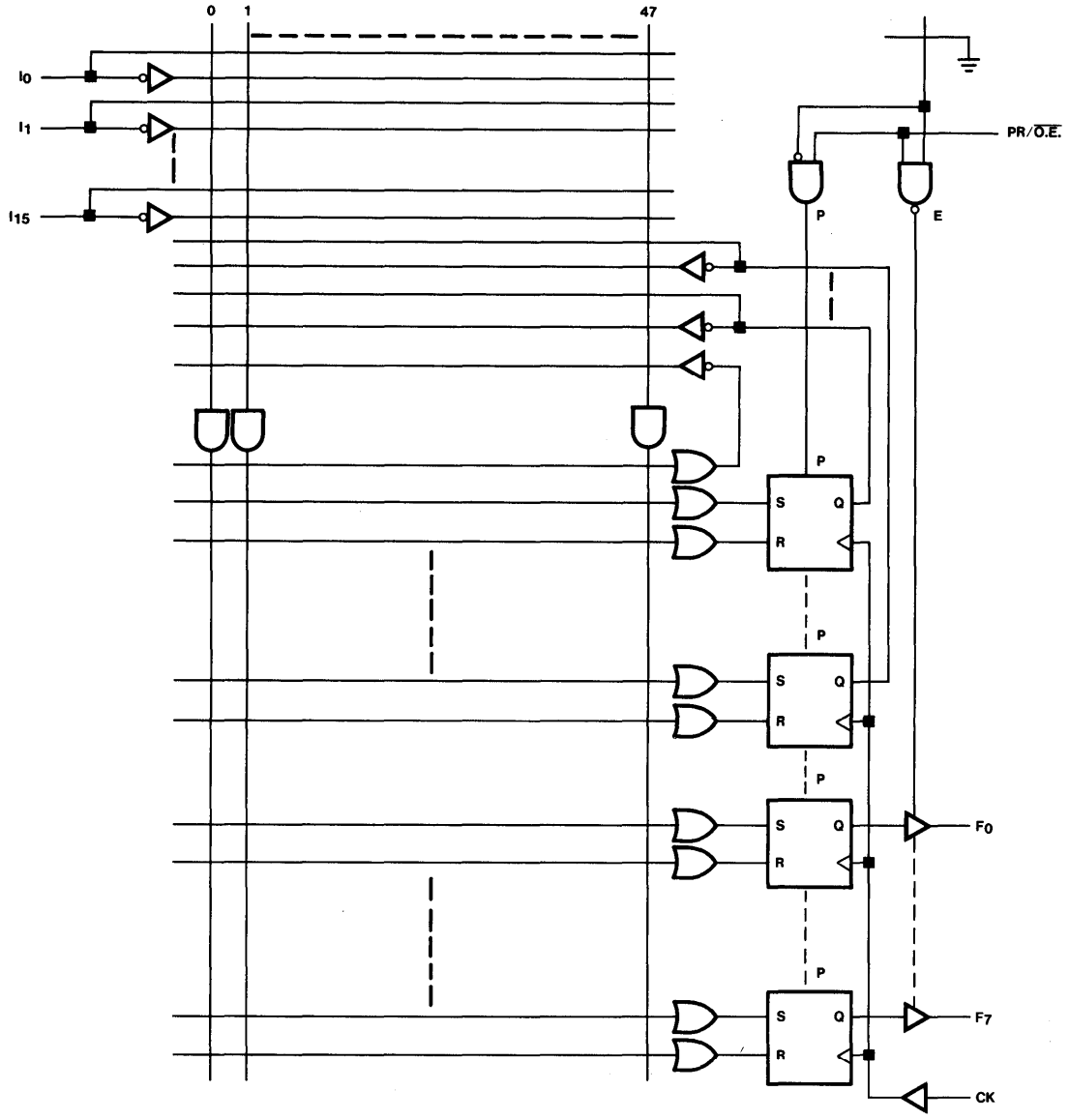


Figure 5

INPUT BUFFERS

16 external inputs (I_m) and 6 internal inputs (P_s), fed back from the state register, are combined in the AND array through two sets of True/Complement (T/C) buffers. There are a total of 22 T/C buffers, all connected to multi-input AND gates via fusible links which are initially intact.

Selective fusing of these links allows coupling either True, Complement, or Don't Care values of (I_m) and (P_s).

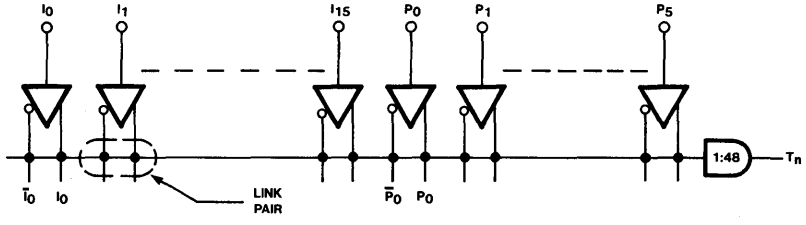


Figure 6: Typical AND gate coupled to (I) and (P) inputs. If at least one link *pair* remains intact, T_n is unconditionally forced Low.

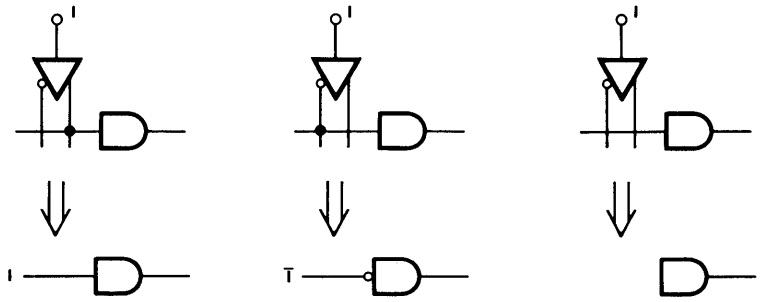


Figure 7: Choice of input polarity coupling to a typical AND gate. With both links open, (I) is logically Don't Care.

"AND" ARRAY

State jumps and output changes are triggered at clock time by valid transition terms T_n. These are logical AND functions of the present state (P) and the present input (I).

The FPLS AND array contains a total of 48 AND gates. Each gate has 45 inputs—44 connected to 22 T/C input buffers, and 1 dedicated to the Complement Array. The outputs of all AND gates are propagated through the OR array, and used at clock time (t) to force the contents of the State Register from (P) to (N). They are also used to control the Output Register, so that the FPLS 8-bit output F_r is a function of the inputs and present state.

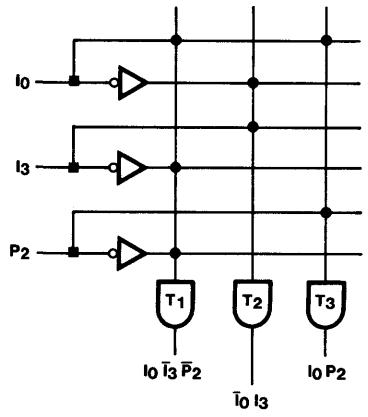


Figure 8: Typical transition terms involving arbitrary inputs and state variables. All remaining gate inputs are programmed Don't Care. Note that T_2 output is state independent.

"OR" ARRAY

In general, a clocked sequence will consist of several stable states and transitions, as determined by the complexity of the desired algorithm. All state and output changes in the state diagram imply changes in the contents of state and output registers.

Thus, each flip-flop in both registers may need to be conditionally set or reset several times with T_n commands. This is accomplished by selectively ORing through a programmable OR array all AND gate outputs T_n necessary to activate the proper flip-flop control inputs.

The FPLS OR array consists of 14 pairs of OR gates, controlling the S/R inputs of 14 state and output register stages, and a single OR gate for the Complement Array. All gates have 48 inputs for connecting to all 48 AND gates.

COMPLEMENT ARRAY

The COMPLEMENT array provides an asynchronous feed back path from the OR array back to the AND array.

This structure enables the FPLS to perform both direct and complement sequential state jumps with a minimum of transition (AND) terms.

Typically direct jumps, such as T_1 and T_2 in Figure 11 require only a single AND gate each.

But a complement jump such as T_3 generally requires many AND gates if implemented as a direct jump. However, by using the complement array, the logic requirements for this type of jump can be handled with just one more gate from the AND array.

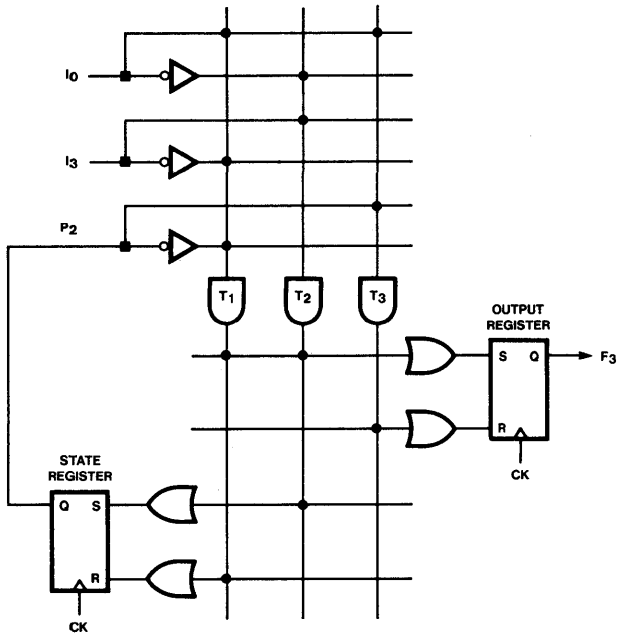


Figure 9: Typical OR array gating of transition terms $T_{1,2,3}$ controlling arbitrary state and output register stages.

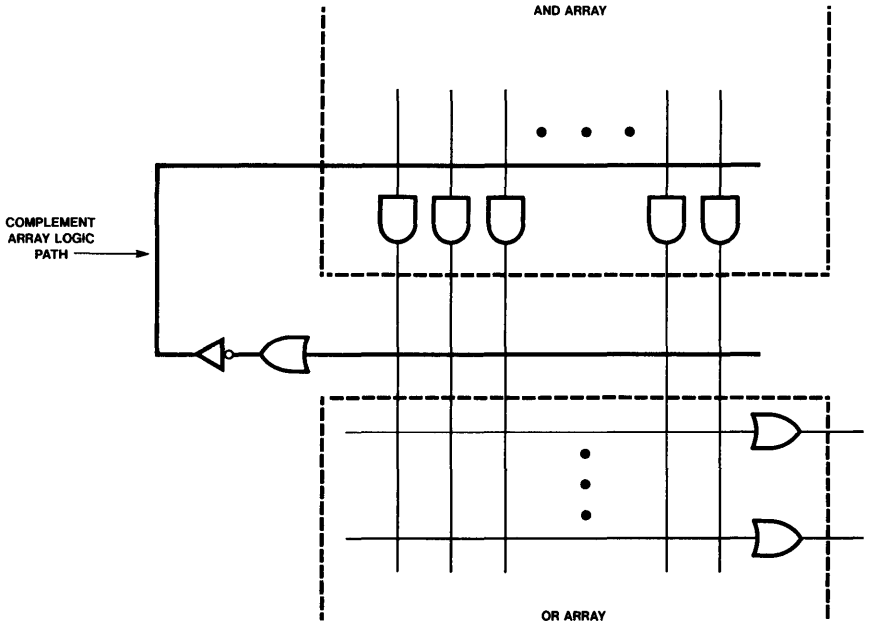
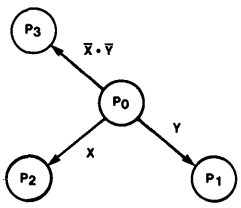


Figure 10: The COMPLEMENT array is logically constructed from a 48-input programmable OR gate followed by an inverter. All AND terms coupled to the OR gate are complemented at the inverter output, and can be fed back as inputs to the AND array.

As indicated in Figure 12, the single complement array gate may be used for many states of the state diagram. This happens because all transition terms linked to the OR gate include the present state as a part of their conditional logic. In any particular state only those transition terms which are a function of that state are enabled; all other terms coupled to different states are disabled and do not influence the output of the complement array. As a general rule of thumb, the complement array can be used as many times as there are states.

(A) TYPICAL STATE SEQUENCE



TRANSITION TERMS

DIRECT

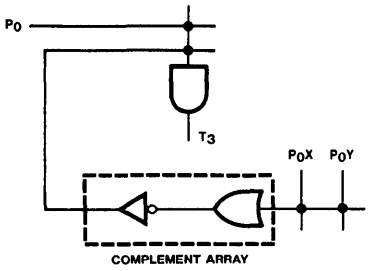
$T_1 = P_0X$

$T_2 = P_0Y$

COMPLEMENT

$T_3 = P_0(\overline{X} \cdot \overline{Y}) = P_0(\overline{T_1 + T_2})$

(B) COMPLEMENT JUMP



$T_3 = P_0(\overline{P_0X + P_0Y})$

$T_3 = P_0[\overline{P_0(X + Y)}]$

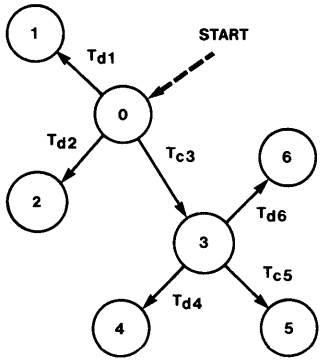
$T_3 = P_0[\overline{P_0} + \overline{(X + Y)}]$

$T_3 = 0 + P_0\overline{(X + Y)}$

$T_3 = P_0(\overline{X} \cdot \overline{Y})$

Figure 11: (A) X and Y specify the conditional logic for direct jump transition terms T_1 and T_2. The complement jump term T_3 is true only when both T_1 and T_2 are false. (B) Note that the complementary logic expression for T_3, $\overline{T_1 + T_2}$, corresponds exactly to the logic structure of the complement array.

(A) STATE DIAGRAM



(B) LOGIC DEFINITION

$T_{d1} = I_0\bar{I}_1P_0$

$T_{d2} = I_2P_0$

$T_{c3} = \overline{(T_{d1} + T_{d2})}P_0 = \overline{(I_0\bar{I}_1 + I_2)}P_0$

$T_{d4} = \bar{I}_2P_3$

$T_{d6} = I_0I_1P_3$

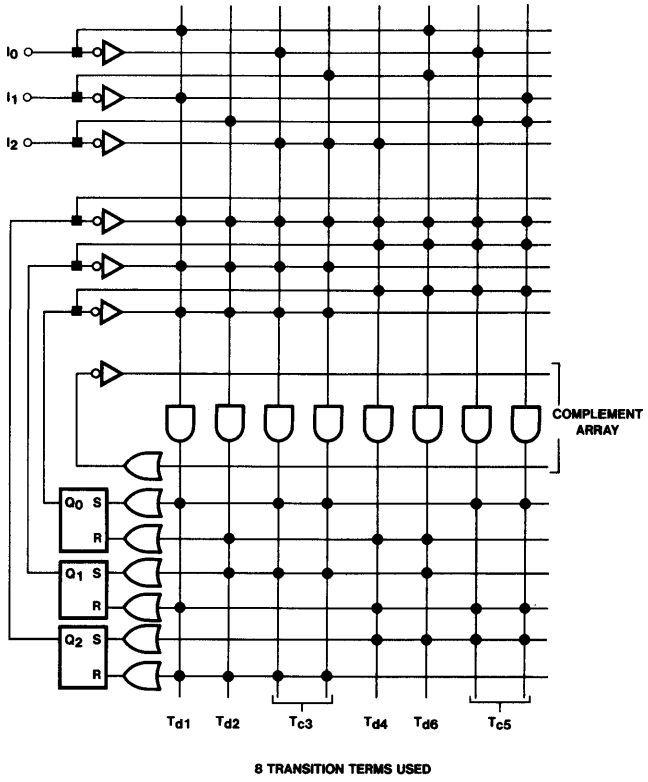
$T_{c5} = \overline{(T_{d4} + T_{d6})}P_3 = \overline{(I_0I_1 + \bar{I}_2)}P_3$

T_{cn} = COMPLEMENT STATE TRANSITION TERM

T_{dn} = DIRECT STATE TRANSITION TERM

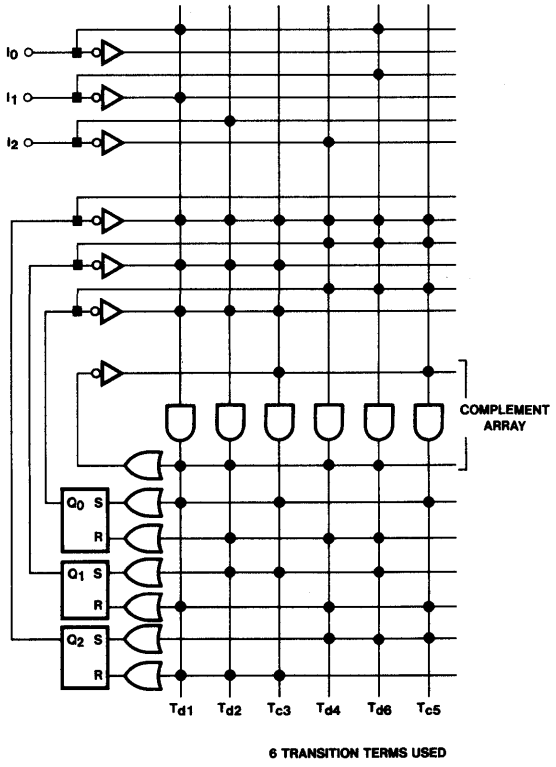
P_s = PRESENT STATE

(C) STATE LOGIC WITHOUT USING THE COMPLEMENT ARRAY



8 TRANSITION TERMS USED

(D) STATE LOGIC USING THE COMPLEMENT ARRAY



6 TRANSITION TERMS USED

Figure 12: Logic reduction with the complement array. The logic state diagram in (a) includes complement jumps T_{C3} and T_{C5} defined in (b). When using the complement array a savings of 2 transition terms results, as shown in (c) and (d).

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

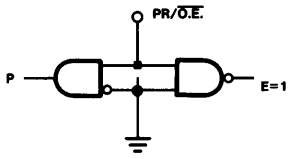
LOGIC PROGRAMMING

The FPLS can be programmed by means of Logic programming equipment.

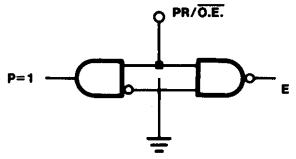
With Logic programming, the AND/OR gate input connections necessary to implement the desired logic function are coded directly from the State Diagram using the Program Table on the following page.

In this Table, the logic state or action of control variables C, I, P, N, and F, associated with each Transition Term T_n, is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

PRESET/$\overline{O.E.}$ OPTION - (P/E)

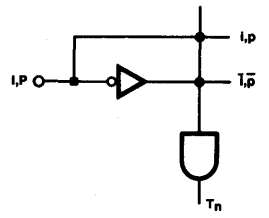


OPTION	CODE
PRESET	H

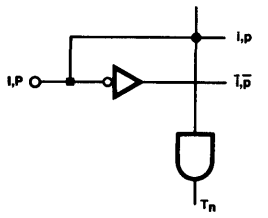


OPTION	CODE
$\overline{O.E.}$	L

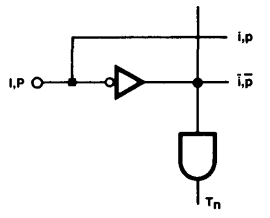
"AND" ARRAY - (I), (P)



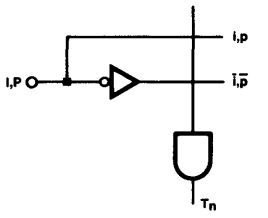
STATE	CODE
INACTIVE 1,2	O



STATE	CODE
I,P	H

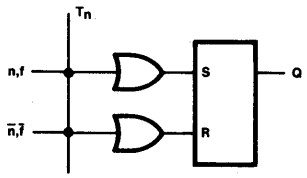


STATE	CODE
$\overline{I,P}$	L

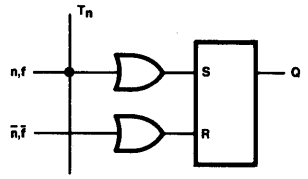


STATE	CODE
DON'T CARE	—

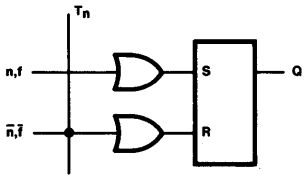
"OR" ARRAY - (N), (F)



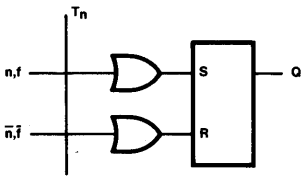
ACTION	CODE
INACTIVE 1,3	O



ACTION	CODE
SET	H

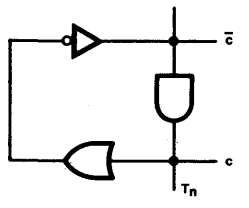


ACTION	CODE
RESET	L

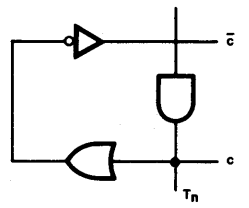


ACTION	CODE
NO CHANGE	—

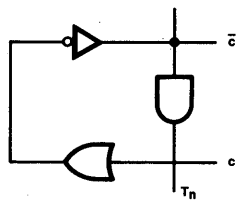
"COMPLEMENT ARRAY" - (C)



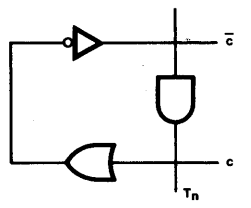
ACTION	CODE
INACTIVE 1,4	O



ACTION	CODE
GENERATE	A



ACTION	CODE
PROPAGATE	•



ACTION	CODE
TRANSPARENT	—

NOTES

1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates T_n.
2. Any gate T_n will be unconditionally inhibited if any one of its I or P link pairs is left intact.
3. To prevent simultaneous Set and Reset flip-flop commands, this state is not allowed for N and F link pairs coupled to active gates T_n (see flip-flop truth tables).
4. To prevent oscillations, this state is not allowed for C link pairs coupled to active gates T_n.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPLS PROGRAM TABLE (Logic)

PROGRAM TABLE ENTRIES:

Cn		Im, Ps		Ns, Fr		P/E	
GENERATE	A	I, P	H	SET	H	PRESET	H
PROPAGATE	●	$\overline{I}$, $\overline{P}$	L	RESET	L	$\overline{O.E.}$	L
TRANSPARENT	—	DON'T CARE	—	NO CHANGE	—		

NOTES

1. The FPLS is shipped with all links initially intact. Thus, a background of "0" for all Terms, and an "H" for the P/E option, exists in the table, shown BLANK instead for clarity.
2. Unused C_n, I_m, and P_s bits are normally programmed Don't Care (—).
3. Unused transition and output Terms can be left blank.
4. Letters in variable fields are used as identifiers by logic type programmers.

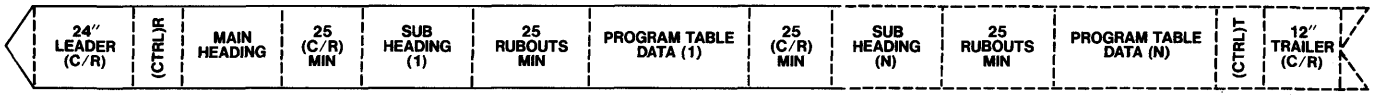
CUSTOMER NAME _____
PURCHASE ORDER # _____ CF (XXXX)
SIGNETICS DEVICE # _____
CUSTOMER SYMBOLIZED PART # _____
TOTAL NUMBER OF PARTS _____ REV ____ DATE ____
PROGRAM TABLE # _____

	TRANSITION TERM																				OPTION (P/E)																
	INPUT VARIABLE (Im)																	PRESENT STATE (Ps)						OUTPUT TERM														
NO.	C_n	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	5	4	3	2	1	0	NEXT STATE (Ns)						OUTPUT FUNCTION (Fr)								
																								5	4	3	2	1	0	7	6	5	4	3	2	1	0
0																																						
1																																						
2																																						
3																																						
4																																						
5																																						
6																																						
7																																						
8																																						
9																																						
10																																						
11																																						
12																																						
13																																						
14																																						
15																																						
16																																						
17																																						
18																																						
19																																						
20																																						
21																																						
22																																						
23																																						
24																																						
25																																						
26																																						
27																																						
28																																						
29																																						
30																																						
31																																						
32																																						
33																																						
34																																						
35																																						
36																																						
37																																						
38																																						
39																																						
40																																						
41																																						
42																																						
43																																						
44																																						
45																																						
46																																						
47																																						
PIN NO.		20	21	22	23	24	25	26	27	2	3	4	5	6	7	8	9													10	11	12	13	15	16	17	18
VARIABLE NAME																																						

TWX TAPE CODING (LOGIC FORMAT)

The FPLS Program Table can be sent to Signetics in ASCII code format via airmail using any type of 8-level tape (paper, mylar, fanfold, etc.), or via TWX: just dial (910) 339-9283, tell the operator to turn the paper puncher on, and acknowledge. At the end of transmission instruct the operator to send tape to Signetics Order Entry.

A number of Program Tables can be sequentially assembled on a continuous tape as follows, however,limit tape length to a roll of 1.75 inch inside diameter and 4.25 inch outside diameter.



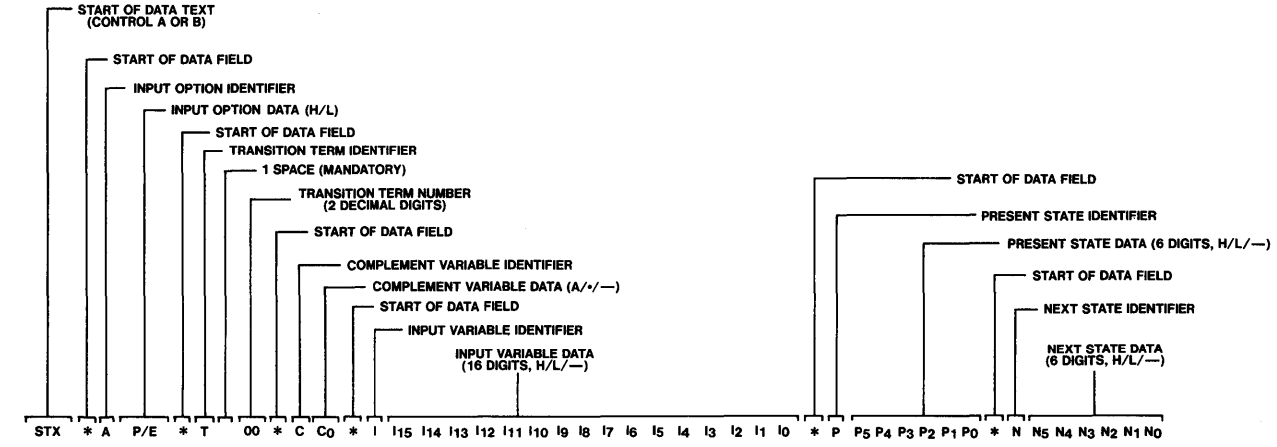
A. The MAIN HEADING at the beginning of tape includes the following information, with each entry preceded by a ($) character, whether used or not:

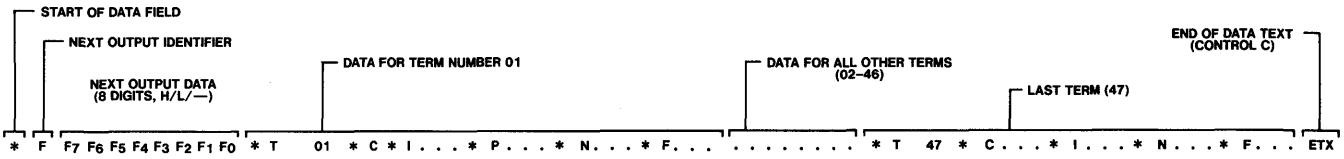
1. Customer Name _____
2. Customer TWX No. _____
3. Date _____
4. Purchase Order No. _____
5. Number of Program Tables _____
6. Total Number of Parts _____

B. Each SUB HEADING should contain specific information pertinent to each Program Table as follows, with each entry preceded by a ($) character, whether used or not:

1. Signetics Device No. _____
2. Program Table No. _____
3. Revision _____
4. Date _____
5. Customer Symbolized Part No. _____
6. Number of Parts _____

C. Program Table data blocks are initiated with an STX character, and terminated with an ETX character. The body of the data consists of preset/output enable option, transition term, and output term information separated by appropriate identifiers in accordance with the following format:





Entries for the Data Fields are determined in accordance with the following Table:

COMPLEMENT VARIABLE (C)			PRESENT STATE (P_S)/INPUT (I_m)			NEXT STATE (N_S)/OUTPUT (F_n)			OPTION (P/E)	
GENERATE	PROPA-GATE	TRANS-PARENT	I_m, P_S	$\overline{I_m}$, $\overline{P_S}$	DON'T CARE	N_S, F_n	$\overline{N_S}$, $\overline{F_n}$	NO CHANGE	PRESET	OUTPUT ENABLE
A	•	—	H	L	—	H	L	—	H	L

Although the Transition Term data are shown entered in sequence, this is not necessary. It is possible to input only one Transition Term, if desired. Unused Transition Terms require no entry. ETX signalling end of Program Table may occur with less than the maximum number of Transition Terms entered.

NOTES

1. Corrections to any entry can be made by backspace and rubout. However, limit consecutive rubouts to less than 25.
2. T-terms can be re-entered any number of times. The last entry for a particular T-term will be interrupted as valid data.
3. To facilitate an orderly Teletype printout, carriage returns, line feeds, spaces, rubouts, etc., may be interspersed between data groups.
4. Comments are allowed between data fields provided that an asterisk (*) is not used in any Heading or Comment entry.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	−30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			°C
T_A	Operating			
	N82S104/105	0	+75	
	S82S104/105	−55	+125	
T_{STG}	Storage	−65	+150	

THERMAL RATINGS

TEMPERATURE	MILITARY	COMMERCIAL
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHARACTERISTICS

N82S104/105: $0° \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S104/105: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

PARAMETER		TEST CONDITIONS	N82S104/105 Min	N82S104/105 Typ[2]	N82S104/105 Max	S82S104/105 Min	S82S104/105 Typ[2]	S82S104/105 Max	UNIT
	Input voltage[3]								V
V_{IH}	High	V_{CC} = Max	2			2			
V_{IL}	Low	V_{CC} = Min			0.85			0.8	
V_{IC}	Clamp[3,4]	V_{CC} = Min, I_{IN} = −18mA		−0.8	−1.2		−0.8	−1.2	
	Output voltage	V_{CC} = Min							V
V_{OH}	High (82S105)[3,5]	I_{OH} = −2mA	2.4			2.4			
V_{OL}	Low[3,6]	I_{OL} = 9.6mA		0.35	0.45		0.35	0.50	
	Input current								μA
I_{IH}	High	V_{IN} = 5.5V		<1	25		<1	50	
I_{IL}	Low	V_{IN} = 0.45V		−10	−100		−10	−150	
I_{IL}	Low (CK input)	V_{IN} = 0.45V		−50	−250		−50	−350	
	Output current	V_{CC} = Max							
I_{OLK}	Leakage[7]	V_{OUT} = 5.5V		1	40		1	60	μA
$I_{O(OFF)}$	Hi-Z state (82S105)[7]	V_{OUT} = 5.5V		1	40		1	60	μA
		V_{OUT} = 0.45V		−1	−40		−1	−60	
I_{OS}	Short circuit (82S105)[4,8]	V_{OUT} = 0V	−20		−70	−15		−85	mA
I_{CC}	V_{CC} supply current[9]	V_{CC} = Max		120	180		120	185	mA
	Capacitance[7]	V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_{OUT}	Output	V_{OUT} = 2.0V		10			10		

NOTES
1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device of these or any other condition above those indicated in the operation of the device specifications is not implied.
2. All typical values are at V_{CC} = 5V, T_A = 25°C.
3. All voltage values are with respect to network ground terminal.
4. Test one at a time.
5. Measured with V_{IL} applied to $\overline{O.E.}$ and a logic high stored, or with V_{IH} applied to PR.
6. Measured with a programmed logic condition for which the output is at a low logic level, and V_{IL} applied to PR/$\overline{O.E.}$ Output sink current is supplied thru a resistor to V_{CC}.
7. Measured with V_{IH} applied to PR/$\overline{O.E.}$.
8. Duration of short circuit should not exceed 1 second.
9. I_{CC} is measured with the PR/$\overline{O.E.}$ input grounded, all other inputs at 4.5V and the outputs open.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

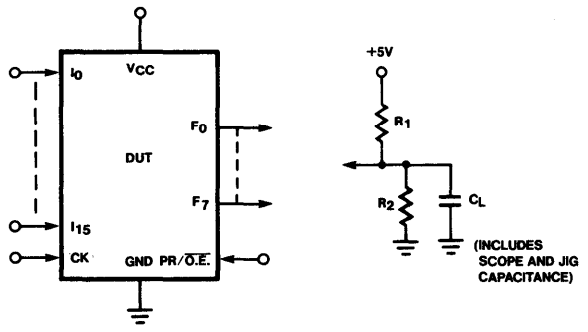
AC ELECTRICAL CHARACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1k\Omega$, $C_L = 30pF$
N82S104/105: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S104/105: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

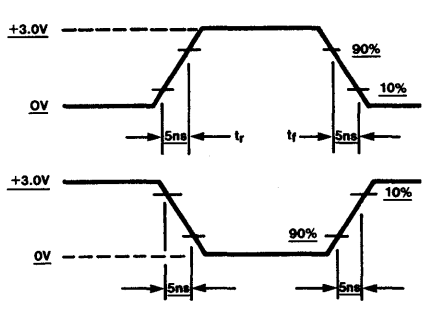
	PARAMETER	TO	FROM	N82S104/105			S82S104/105			UNIT
				Min	Typ[2]	Max	Min	Typ[2]	Max	
	Pulse width									ns
T_{CKH}	Clock[3] high	CK−	CK+	30	15		40	15		
T_{CKL}	Clock low	CK+	CK−	30	15		40	15		
T_{CKP}	Period (w/o c-array)	CK+	CK+	90	65		120	65		
T_{PRH}	Preset pulse	PR+	PR−	25	15		40	15		
	Set up time									ns
T_{IS1}	Input	CK+	Input ±	60	40		80	40		
T_{IS2}	Input (through Complement array)[4]	CK+	Input ±	90	70		120 5	70		
T_{VS}	Power-on preset	CK−	V_{CC}+	0	−10		5	10		
T_{PRS}	Preset	CK−	PR−	0	−10		5	−10		
	Hold time									ns
T_{IH}	Input	Input ±	CK+		−10	5		−10	10	
	Propagation delay									ns
T_{CKO}	Clock	Output ±	CK+		25	30		25	40	
T_{OE}	Output enable	Output−	$\overline{O.E.}$−		20	30		20	40	
T_{OD}	Output disable	Output+	$\overline{O.E.}$+		20	30		20	40	
T_{PR}	Preset	Output+	PR+		25	35		25	45	
T_{PPR}	Power-on preset	Output+	V_{CC}+		0	10		0	20	

NOTE
1. All typical values are at $V_{CC} = 5V$, $T_A = 25°C$.
2. Diagnostic mode only.
3. To prevent spurious clocking, clock rise time (10%-90%) ≤ 10ns.
4. When using the Complement Array, T_{CKP} = 120ns (min).

TEST LOAD CIRCUIT



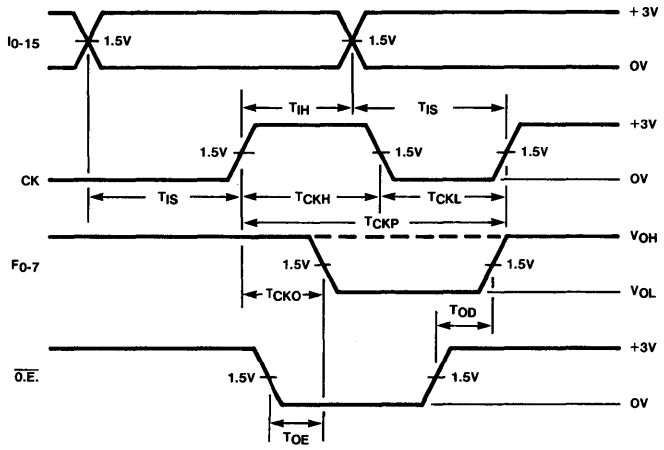
VOLTAGE WAVEFORM



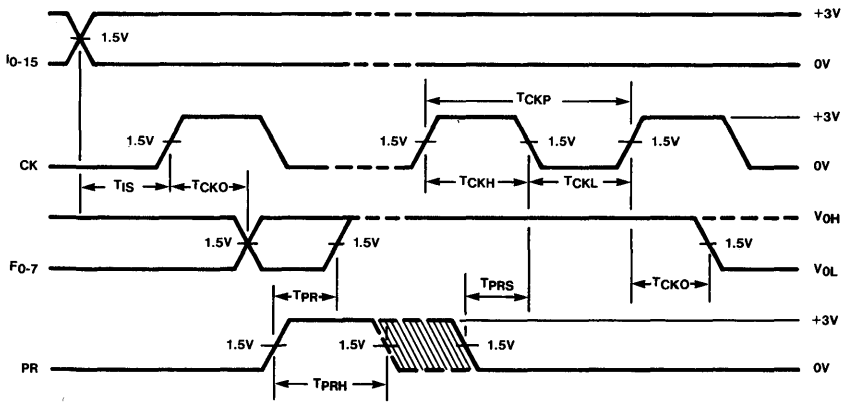
Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs, unless otherwise specified.

TIMING DIAGRAMS

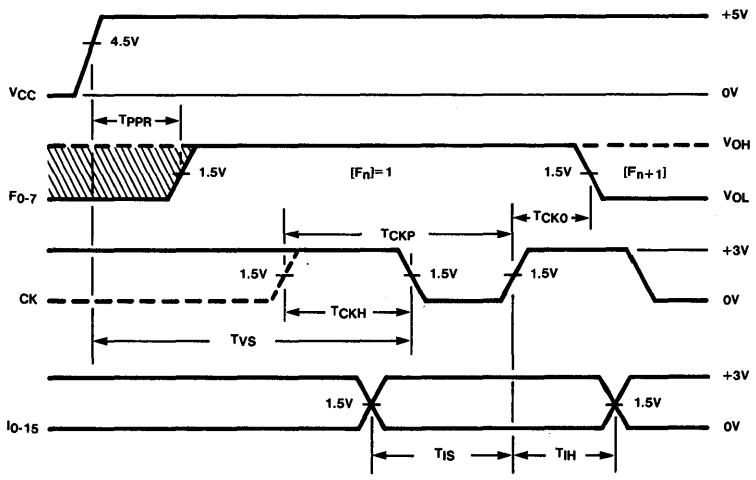
SEQUENTIAL MODE



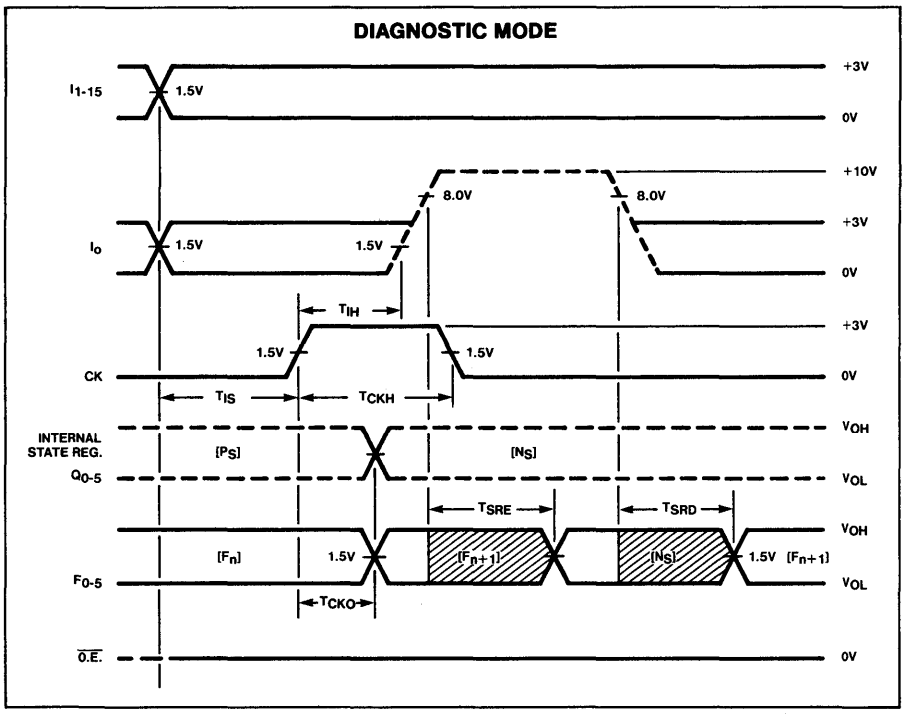
ASYNCHRONOUS PRESET



POWER-ON PRESET



TIMING DIAGRAMS (Cont'd)



TIMING DEFINITIONS

T_{CKH} Width of input clock pulse.

T_{CKL} Interval between clock pulses.

T_{CKP} Clock period.

T_{IS1} Required delay between beginning of valid Input and positive transition of clock.

T_{IS2} Required delay between beginning of valid Input and positive transition of clock, when using optional Complement Array (two passes necessary through the AND array).

T_{VS} Required delay between V_{CC} (after power-on) and negative transition of clock preceding first reliable clock pulse.

T_{PRS} Required delay between negative transition of asynchronous Preset and negative transition of clock preceding first reliable clock pulse.

T_{IH} Required delay between positive transition of clock and end of valid Input data.

T_{CKO} Delay between positive transition of clock and when Outputs become valid (with PR/$\overline{O.E.}$ low).

T_{OE} Delay between beginning of Output Enable Low and when Outputs become valid.

T_{OD} Delay between beginning of Output Enable High and when Outputs are in the off state.

T_{SRE} Delay between input I_0 transition to Diagnostic mode and when the Outputs reflect the contents of the State Register.

T_{SRD} Delay between input I_0 transition to Logic mode and when the Outputs reflect the contents of the Output Register.

T_{PR} Delay between positive transition of Preset and when Outputs become valid at "1".

T_{PPR} Delay between V_{CC} (after power-on) and when Outputs become preset at "1".

T_{PRH} Width of preset input pulse.

VIRGIN STATE[1,2,3]

A factory shipped virgin device contains all fusible links intact, such that:

1. PR/$\overline{OE}$ option is set to PR.
2. All T_n terms are disabled.
4. All S/R flip-flop inputs are disabled.
5. Test array is programmed with standard test pattern.

RECOMMENDED PROGRAMMING PROCEDURE

To program the AND, OR, and Complement arrays in addition to the PR/$\overline{OE}$ option the following procedure should be followed. To maximize recovery from programming errors, leave all links in unused device areas intact.

SET-UP

Terminate all device outputs with a 10kΩ resistor to V_{CC}. Set GND (pin 14) to 0V.

PROGRAM PR/$\overline{OE}$ OPTION

1. With PR/$\overline{OE}$ (pin 19) at GND, raise V_{CC} to V_{CCP}.
2. After t_D delay pulse PR/$\overline{OE}$ to Vix for a duration of tp.
3. t_D delay after PR/$\overline{OE}$ has returned to GND, lower V_{CC} to V_{CCV} or GND.

VERIFY PR/$\overline{OE}$ OPTION

1. With PR/$\overline{OE}$ at GND, set V_{CC} to V_{CCV}.
2. After a delay of t_D raise PR/$\overline{OE}$ to Vix for a minimum duration of T_{RS}.
3. Return PR/$\overline{OE}$ to GND with a fall time less than T_f.
4. After t_D delay, pulse PR/$\overline{OE}$ to V_{IH} for a minimum duration of T_{ps}.
5. After t_D delay, F_O (pin 18) indicates V_{OH} if the PR option is selected and V_{OL} if the $\overline{OE}$ option is programmed.

NOTES
1. All outputs will be at "1", as preset by initial power-up procedure.
2. Device can be clocked via test array function.
3. Test array function MUST be deleted before incorporating user program.

PROGRAM-VERIFY "AND" ARRAY

1. **SET-UP:**
With V_{CC} at GND and CK at V_{CKV}, select the fuse to be programmed by applying TTL voltage levels to input sets I_{0-5} and I_{7-13} in accordance with the binary address map on page 21. Also set I_{15} = V_{IH}, and I_{14} = V_{IL}. After t_D delay raise V_{CC} to V_{CCP}.
2. **PROGRAM CYCLE:**
After a delay of t_D raise I_{14} to V_{IH}. Proceed after another t_D delay to raise CK to V_{CKP}. Following still another t_D delay, pulse I_{15} to V_{IL} for a duration of t_p. t_D later return CK to V_{CKV}, and then t_D after that return I_{14} to V_{IL}.
3. **VERIFY CYCLE:**
After a t_D delay lower I_{15} to V_{IL} for a duration of t_V. At the end of t_V F_O should indicate a level of V_{OH}; a level of V_{OL} indicates an unsuccessful fusing attempt.
4. **NEXT VARIABLE SELECT (C,Im,Ps) (SAME TERM Tn):**
After t_D delay, apply the next variable select (C,Im,Ps) address to I_{7-13}, then continue with step 2.
5. **NEXT VARIABLE SELECT (C,Im,Ps) (DIFFERENT TERM Tn):**
After t_D, delay apply the next variable select (C,Im,Ps) and term (Tn) addresses to respectively I_{7-13} and I_{0-5}, then continue to step 2.

PROGRAM-VERIFY "OR" ARRAY

1. **SET-UP:**
With V_{CC} at GND and CK at V_{CKV}, select the fuse to be programmed by applying TTL voltage levels to input sets I_{0-5} and I_{7-13} in accordance with the binary address map on page 21. Also set I_{14} = V_{IL}, and I_{15} = V_{IH}. After t_D delay raise V_{CC} to V_{CCP}.
2. **PROGRAM CYCLE:**
After a delay of t_D raise I_{14} to V_{IH}. Proceed after another t_D delay to raise CK to V_{CKP}. Following still another t_D delay pulse I_{15} to V_{IL} for a duration of tp. t_D later return CK to V_{CKV} and then t_D after that return I_{14} to V_{IL}.
3. **VERIFY CYCLE:**
After a t_D delay lower I_{15} to V_{IL} for a duration of t_V. At the end of t_V F_O should indicate a level of V_{OH}; a level of V_{OL} indicates an unsuccessful fusing attempt.
4. **NEXT VARIABLE SELECT (C,Ns,Fn) (SAME TERM (Tn)):**
After t_D delay apply the next variable select (C,Ns,Fn) address to I_{7-13}, then continue with step 2.
5. **NEXT VARIABLE SELECT (C,Ns,Fn) (DIFFERENT TERM (Tn)):**
After t_D delay apply the next variable select (C,Ns,Fn) and term (Tn) addresses to respectively I_{7-13} and I_{0-5}, then continue to step 2.

PROGRAM CYCLE POWER DOWN

When programming of the device is complete, after t_D delay set I_{15} to V_{IH}, I_{14} to V_{IL} and CK to V_{CKV}. After another t_D delay reduce V_{CC} to 0V.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

PROGRAM CYCLE ROW/COLUMN FUSE ADDRESSING

VARIABLE SELECT Table[1]

ROW HEX ADDRESS I_{13} I_{12} I_{11}	ROW HEX ADDRESS I_{10} I_9 I_8 I_7	SELECTED VARIABLE			ROW HEX ADDRESS I_{13} I_{12} I_{11}	ROW HEX ADDRESS I_{10} I_9 I_8 I_7	SELECTED VARIABLE	
0	0	OR Array	N_0	SET	4	0	AND Array	I_0
0	1			RESET	4	1		$\overline{I_0}$
0	2		N_1	SET	4	2		I_1
0	3			RESET	4	3		$\overline{I_1}$
0	4		N_2	SET	4	4		I_2
0	5			RESET	4	5		$\overline{I_2}$
0	6		N_3	SET	4	6		I_3
0	7			RESET	4	7		$\overline{I_3}$
0	8		N_4	SET	4	8		I_4
0	9			RESET	4	9		$\overline{I_4}$
0	A		N_5	SET	4	A		I_5
0	B			RESET	4	B		$\overline{I_5}$
0	C		F_0	SET	4	C		I_6
0	D			RESET	4	D		$\overline{I_6}$
0	E		F_1	SET	4	E		I_7
0	F			RESET	4	F		$\overline{I_7}$
1	0		F_2	SET	5	0		I_8
1	1			RESET	5	1		$\overline{I_8}$
1	2		F_3	SET	5	2		I_9
1	3			RESET	5	3		$\overline{I_9}$
1	4		F_4	SET	5	4		I_{10}
1	5			RESET	5	5		$\overline{I_{10}}$
1	6		F_5	SET	5	6		I_{11}
1	7			RESET	5	7		$\overline{I_{11}}$
1	8		F_6	SET	5	8		I_{12}
1	9			RESET	5	9		$\overline{I_{12}}$
1	A		F_7	SET	5	A		I_{13}
1	B			RESET	5	B		$\overline{I_{13}}$
1	C	Complement Array		C	5	C		I_{14}
1	D	Empty Address Space			5	D		$\overline{I_{14}}$
↓	↓				5	E		I_{15}
↓	↓				5	F		$\overline{I_{15}}$
↓	↓				6	0		P_0
↓	↓				6	1		$\overline{P_0}$
↓	↓				6	2		P_1
↓	↓				6	3		$\overline{P_1}$
↓	↓				6	4		P_2
↓	↓				6	5		$\overline{P_2}$
↓	↓				6	6		P_3
↓	↓				6	7		$\overline{P_3}$
↓	↓				6	8		P_4
↓	↓				6	9		$\overline{P_4}$
↓	↓				6	A		P_5
↓	↓				6	B		$\overline{P_5}$
3	F				6	C	Complement Array	$\overline{C}$

TRANSITION TERM SELECT Table[2]

COLUMN HEX ADDRESS I_5 I_4	COLUMN HEX ADDRESS I_3 I_2 I_1 I_0	SELECTED TRANSITION TERM
0	0	0
0	1	1
0	2	2
0	3	3
0	4	4
0	5	5
0	6	6
0	7	7
0	8	8
0	9	9
0	A	10
0	B	11
0	C	12
0	D	13
0	E	14
0	F	15
1	0	16
1	1	17
1	2	18
1	3	19
1	4	20
1	5	21
1	6	22
1	7	23
1	8	24
1	9	25
1	A	26
1	B	27
1	C	28
1	D	29
1	E	30
1	F	31
2	0	32
2	1	33
2	2	34
2	3	35
2	4	36
2	5	37
2	6	38
2	7	39
2	8	40
2	9	41
2	A	42
2	B	43
2	C	44
2	D	45
2	E	46
2	F	47
3	0	48
3	1	49

NOTES
1. A row address identifies a particular variable coupled to all transition terms.
2. With a variable selected by the row address the column address further selects a coupling fuse for each term.

FIELD PROGRAMMABLE LOGIC SEQUENCER 82S104 (O.C.)/82S105 (T.S.)

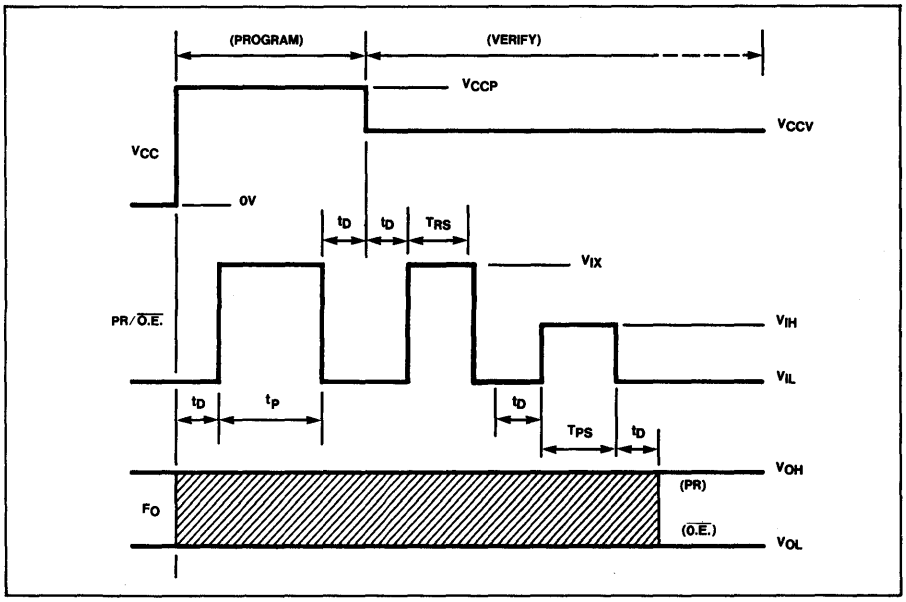
INTEGRATED FUSE LOGIC
SERIES 28

PROGRAMMING SYSTEM SPECIFICATIONS[1] (T_A = +25°C)

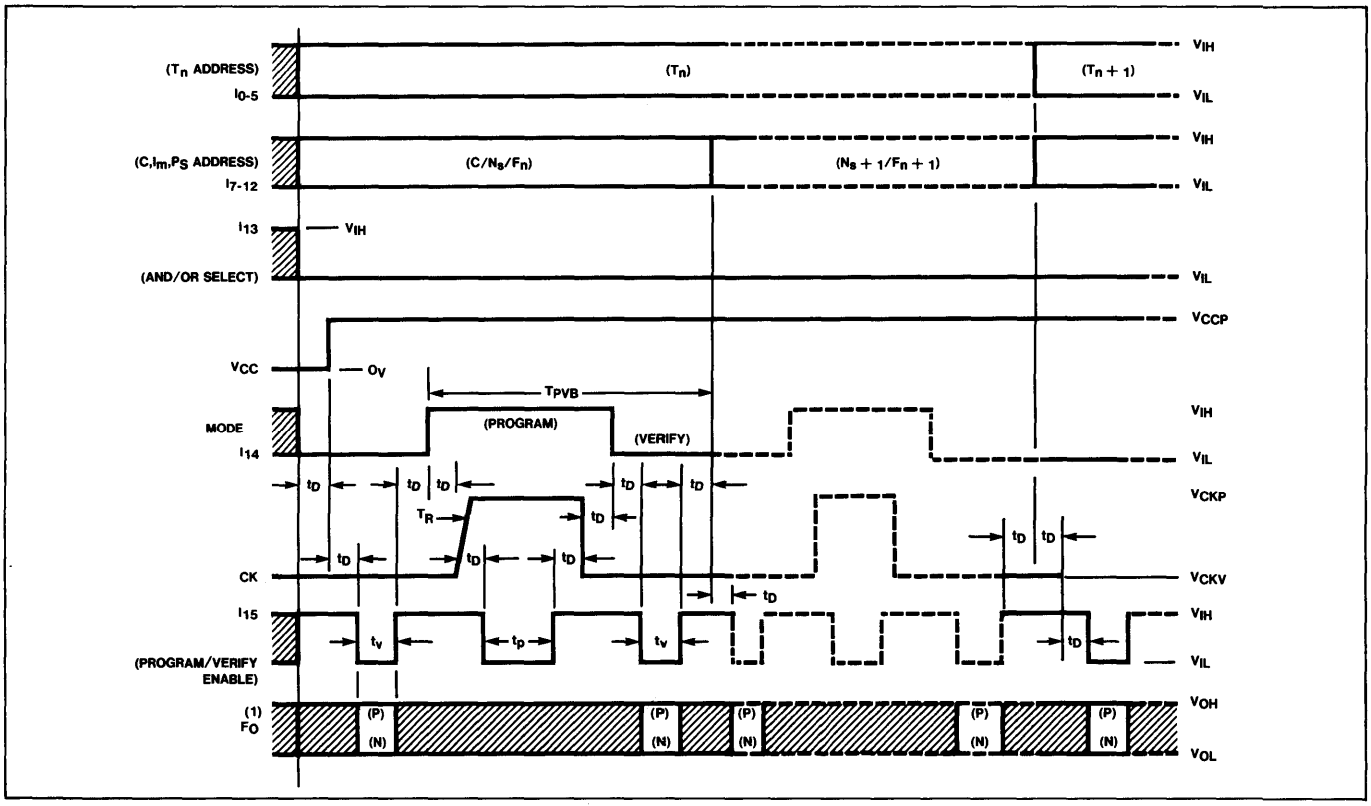
PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{CCP}	V_{CC} supply (program)	I_{CCP} = 550mA min	8.25	8.5	8.75	V
V_{CCV}	V_{CC} supply (verify)	Transient or steady state	4.75	5.0	5.25	V
I_{CCP}	I_{CC} limit program	V_{CCP} = +8.50 ± .25V	550		1,000	mA
	Input voltage					V
V_{IH}	High		2.4		5.5	
V_{IL}	Low		0	0.4	0.8	
	Input Current					μA
I_{IH}	High	V_{IH} = +5.5V			50	
I_{IL}	Low	V_{IL} = 0V			−500	
V_{IX}	$\overline{O.E.}$ program enable level		9.5	10	10.5	V
I_{IX}	$\overline{O.E.}$ program input current	V_{IX} = +10V			10	mA
V_{CKP}	CK supply (program)[3]	I_{CKP} = 300 ± 25mA	16.0	17.0	18.0	V
V		Transient or steady state				
V_{CKV}	CK supply (idle)	I_{CKV} = 1mA max	1.25	1.5	1.75	V
I_{CKP}	CK supply current limit	V_{CKP} = +17 ± 1V	275	300	325	mA
t_V	Verify time		1			μs
T_{RS}	Reset pulse width		1			μs
T_{PS}	Preset pulse width		1			μs
t_P	Programming pulse width		0.3	0.4	0.5	ms
t_D	Pulse sequence delay		10			μs
T_R	CK Pulse rise time	10% to 90%	10		50	μs
T_{PVB}	Program-Verify time per link			0.6		ms
P_{DC}	Programming duty cycle				100	%
F_L	Fusing attempts per link				2	cycle
V_S	Verify threshold[4]		1.4	1.5	1.6	V

NOTES
1. These are specifications which a Programming System must satisfy in order to be qualified by Signetics.
2. Bypass V_{CC} to GND with a 0.01μf capacitor to reduce voltage spikes.
3. Care should be taken to ensure that the voltage is maintained during the entire fusing cycle. The recommended supply is a constant current source clamped at the specified voltage limit.
4. V_S is the sensing threshold of the FPLS output voltage for a programmed link. It normally constitutes the reference voltage applied to a comparator circuit to verify a successful fusing attempt.

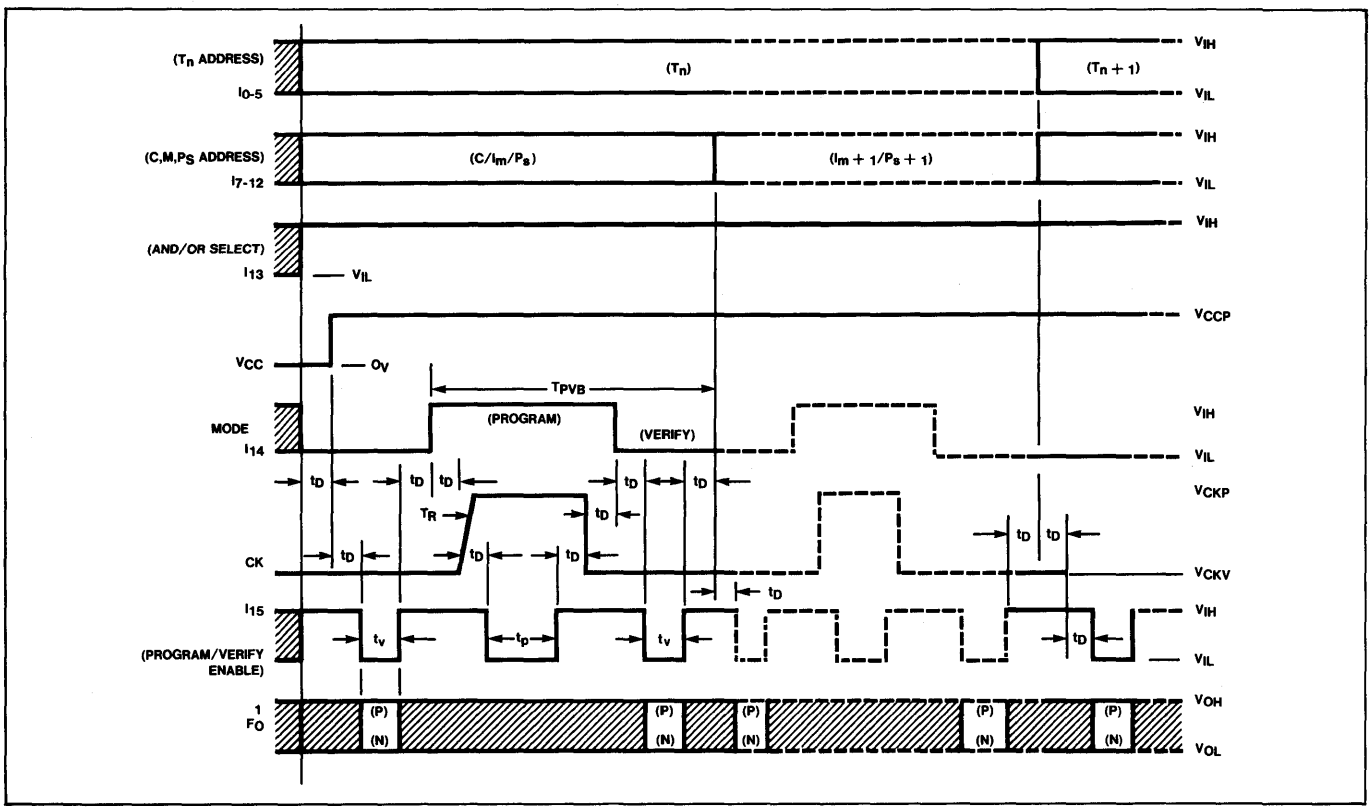
PR/$\overline{O.E.}$ OPTION PROGRAM-VERIFY SEQUENCE



"OR" ARRAY PROGRAM-VERIFY SEQUENCE (TYPICAL)



"AND" ARRAY PROGRAM-VERIFY SEQUENCE (TYPICAL)



NOTE

1. (P) and (N) represent respectively a programmed and non-programmed fuse, corresponding to logic "1" or "0" output voltage levels.

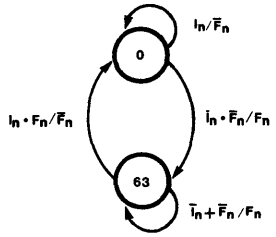
TEST ARRAY

The FPLS may be subjected to AC and DC parametric tests prior to programming via an on chip test array.

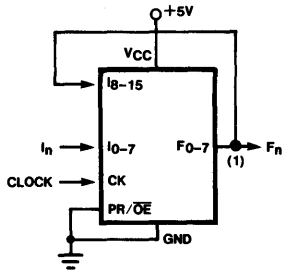
The array consists of test transition terms 48 and 49, factory programmed as shown below.

Testing is accomplished by clocking the FPLS and applying the proper input sequence to I_{0-7} as shown in the test circuit timing diagram.

STATE DIAGRAM



FPLS UNDER TEST



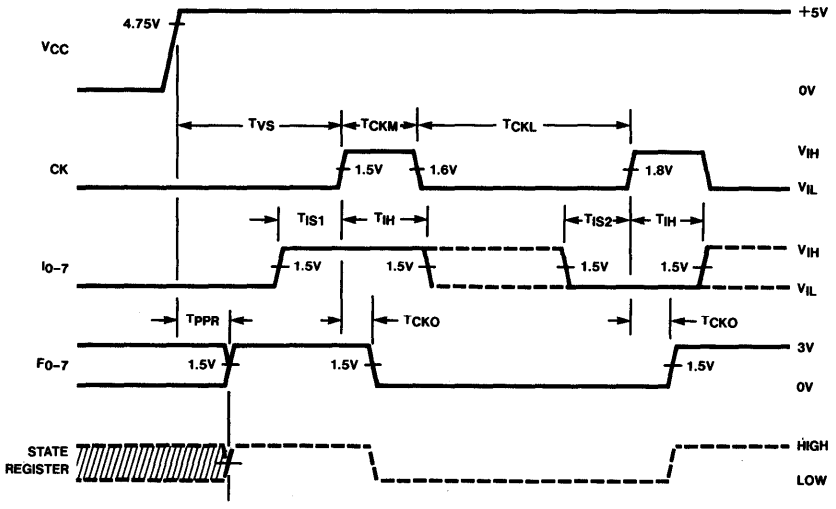
TEST ARRAY PROGRAM (LOGIC)

NO.	C	TRANSITION TERM: INPUT VARIABLE (Im) 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	PRESENT STATE (Ps) 5	4	3	2	1	0
48	A	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
49	●	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L

OPTION (P/E)	H												
OUTPUT TERM													
NEXT STATE (Ns)							OUTPUT FUNCTION (Fn)						
5	4	3	2	1	0	7	6	5	4	3	2	1	0
L	L	L	L	L	L	L	L	L	L	L	L	L	L
H	H	H	H	H	H	H	H	H	H	H	H	H	H

Both terms 48 and 49 must be deleted during user programming to avoid interfering with the desired logic function. This is accomplished automatically by any Signetics' qualified programming equipment.

TEST CIRCUIT TIMING DIAGRAM



TEST ARRAY DELETED (LOGIC)

NO.	C	TRANSITION TERM: INPUT VARIABLE (Im) 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	PRESENT STATE (Ps) 5	4	3	2	1	0
48	—	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
49	●	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L	L

OPTION (P/E)	H												
OUTPUT TERM													
NEXT STATE (Ns)							OUTPUT FUNCTION (Fn)						
5	4	3	2	1	0	7	6	5	4	3	2	1	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—
—	—	—	—	—	—	—	—	—	—	—	—	—	—

FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

DESCRIPTION

The 82S106 (Open collector outputs) and 82S107 (3-state outputs) are bipolar Programmable ROM Patches organized as 48 words by 8 bits, addressed via a 16 bit programmable address comparator. Each word can be assigned a unique address code, P_n, within a 64K (2^{16}) address range by programming the comparator inputs High, Low, or Don't care via True/Complement input buffers.

The contents of each word are also programmable, and are enabled to the active-High Patch outputs only when a programmed address is detected, which causes the $\overline{\text{Flag}}$ output to go Low. For all unprogrammed addresses, the device outputs remain High (82S106) or Hi-Z (82S107) while the $\overline{\text{Flag}}$ output remains High. The $\overline{\text{Flag}}$ is open collector to allow wire-ANDing for expansion to more than 48 patch words.

The 82S106 and 82S107 are fully TTL compatible and can be programmed in the field by following the fusing procedure outlined in this data sheet, or by means of commercially available equipment.

Both devices are available in commercial and military temperature ranges. For the commercial range (0°C to +75°C) specify N82S106/107, F or N, and for the military temperature range (−55°C to +125°C) specify S82S106/107, F, G, or R.

FEATURES

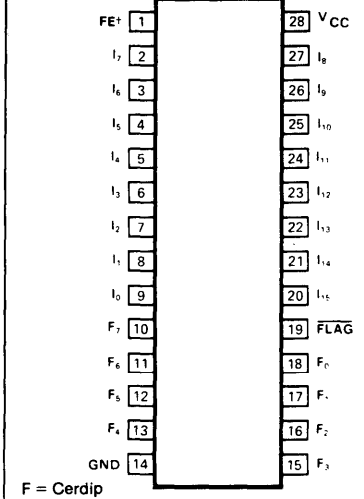
- **Field programmable (Ni-Cr link)**
- **Address Inputs: 16**
- **Data Outputs: 8**
- **Patch Words: 48**
- **Address access time:**
  S82S106/107—100ns Max
  N82S106/107—70ns Max
- **Power dissipation: 600mW typ**
- **Input loading:**
  S82S106/107: −150μA Max
  N82S106/107: −100μA Max
- **Open collector $\overline{\text{Flag}}$**
- **Output option:**
  82S106: Open collector
  82S107: 3-state
- **Output disabled state**
  3-state—Hi-Z
  Open collector—Hi

APPLICATIONS

- **ROM data modifications**
- **Memory address trap**
- **Digital filter**
- **Interrupt request/vector generator**
- **Data security encoder**

PIN CONFIGURATION

F,G,I,N,R PACKAGES

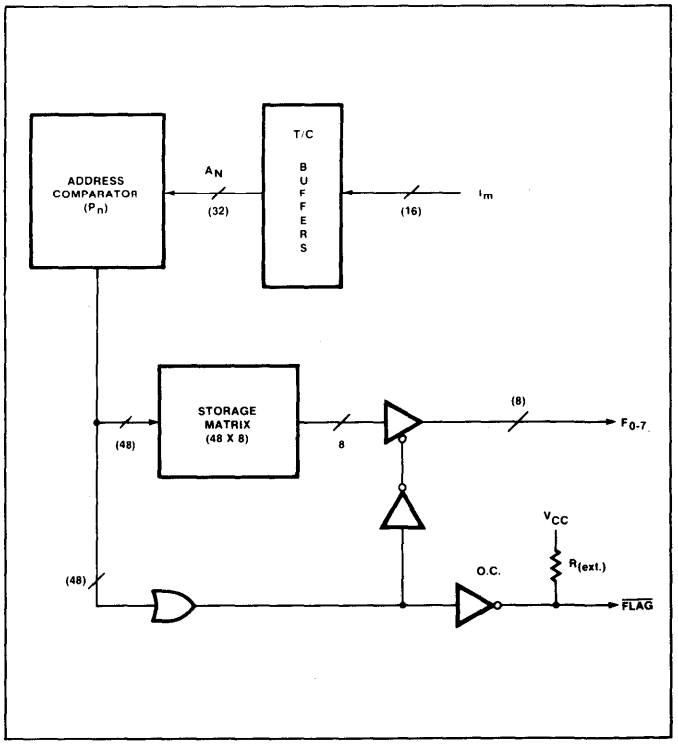


F = Cerdip
G = Leadless
I = Ceramic
N = Plastic
R = Flat pack
†Open or grounded during normal operation

TRUTH TABLE

$A_N = P_N$?	$\overline{\text{Flag}}$	F_{0-7}	
		82S106	82S107
NO	1	1	Hi-Z
YES	0	Stored Data	

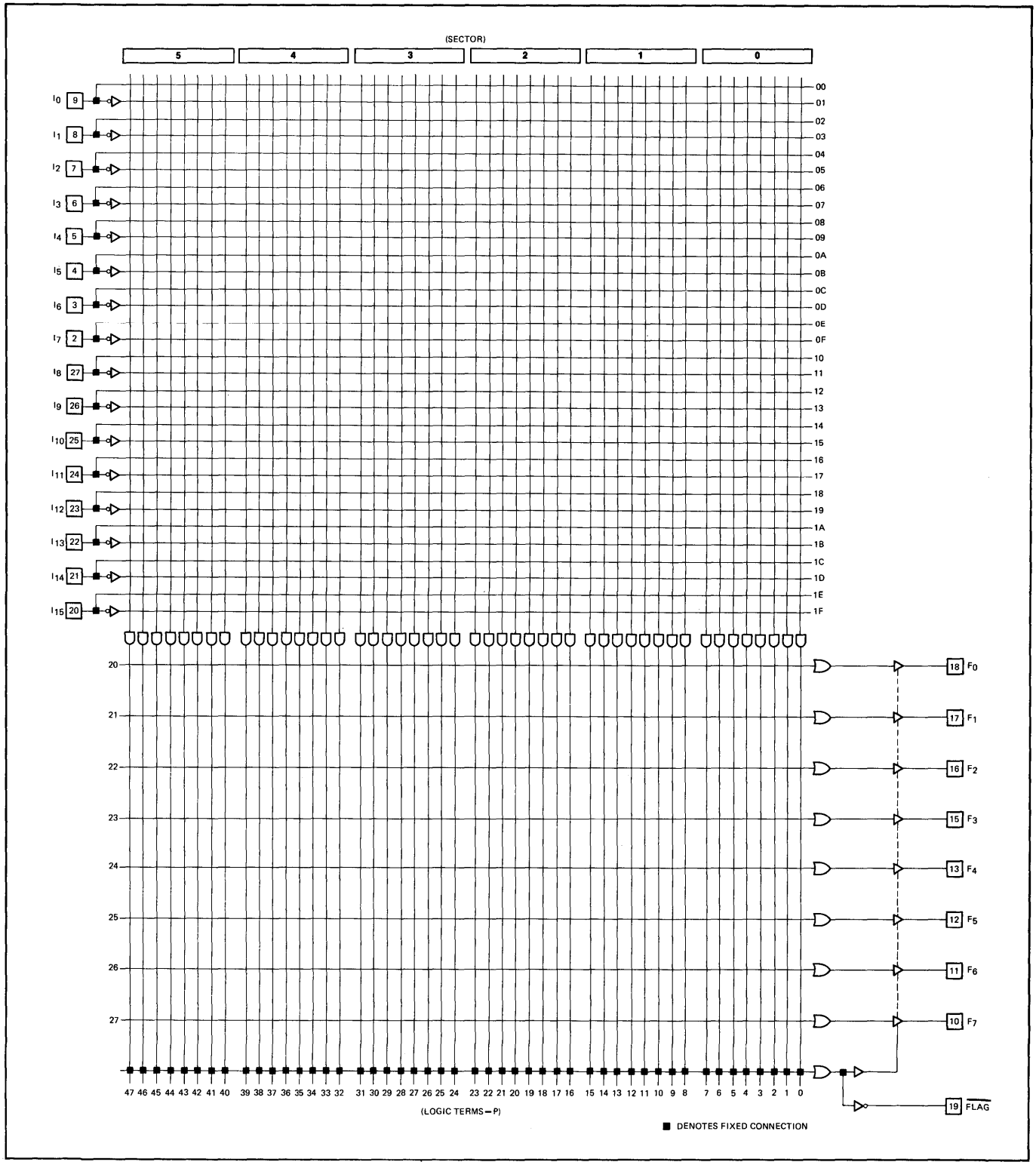
LOGIC DIAGRAM



FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPRP LOGIC DIAGRAM



FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	-30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			°C
T_A	Operating			
	N82S106/107	0	+75	
	S82S106/107	-55	+125	
T_{STG}	Storage	-65	+150	

THERMAL RATINGS

TEMPERATURE	MILITARY	COMMERCIAL
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHARACTERISTICS

N82S106/107: $0° \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S106/107: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TEST CONDITIONS	N82S106/107 Min	N82S106/107 Typ[2]	N82S106/107 Max	S82S106/107 Min	S82S106/107 Typ[2]	S82S106/107 Max	UNIT
	Input voltage[2]								V
V_{IH}	High	V_{CC} = Max	2			2			
V_{IL}	Low	V_{CC} = Min			0.85			0.8	
V_{IC}	Clamp[2,3]	V_{CC} = Min, I_{IN} = -18mA		-0.8	-1.2		-0.8	-1.2	
	Output voltage	V_{CC} = Min							V
V_{OH}	High (82S107)[2,4]	I_{OH} = -2mA	2.4			2.4			
V_{OL}	Low[2,5] (F_{0-7})	I_{OL} = 9.6mA		0.35	0.45		0.35	0.50	
V_{OL}	Low[2,5] ($\overline{Flag}$)	I_{OL} = 4.8mA		0.35	0.45		0.35	0.50	
	Input current								μA
I_{IH}	High	V_{IN} = 5.5V		<1	25		<1	50	
I_{IL}	Low	V_{IN} = 0.45V		-10	-100		-10	-150	
	Output current	V_{CC} = Max							
I_{OLK}	Leakage[7]	V_{OUT} = 5.5V		1	40		1	60	μA
$I_{O(OFF)}$	Hi-Z state (82S107)[6]	V_{OUT} = 5.5V		1	40		1	60	μA
		V_{OUT} = 0.45V		-1	-40		-1	-60	
I_{OS}	Short circuit (82S107)[3,7]	V_{OUT} = 0V	-20		-70	-15		-85	mA
I_{CC}	V_{CC} supply current[8]	V_{CC} = Max		120	170		120	180	mA
	Capacitance[6]	V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_{OUT}	Output	V_{OUT} = 2.0V		17			17		

AC ELECTRICAL CHARACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1k\Omega$, $C_L = 30pF$
N82S106/107: $0°C \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S106/107: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TO	FROM	N82S106/107 Min	N82S106/107 Typ	N82S106/107 Max	S82S106/107 Min	S82S106/107 Typ	S82S106/107 Max	UNIT
	Access time									ns
T_{IA}	Address	Output	Input		45	70		100	100	
T_{FL}	Enable	$\overline{Flag}$	Input		40	55		40	80	

NOTES on following page.

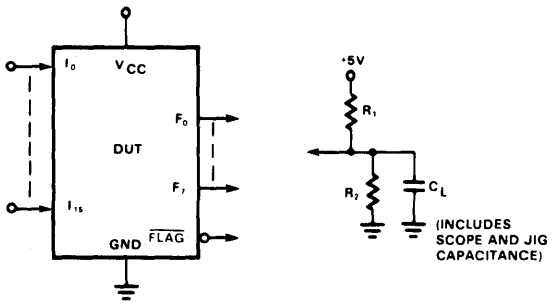
FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

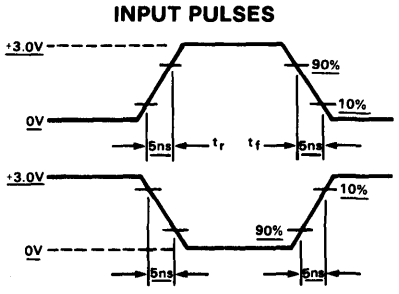
NOTES

1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device of these or any other condition above those indicated in the operation of the device specification is not implied.
2. **All typical values are at V_{CC} = 5V, T_A = 25°C.**
3. All voltage values are with respect to network ground terminal.
4. Test one at the time.
5. Measured with the Patch enabled ($A_N = P_N$) and a logic high stored.
6. Measured with a programmed logic condition for which the output under test is at a low logic level when the Patch is enabled ($A_N = P_N$). Output sink current is applied thru a resistor to V_{CC}.
7. Measured with the Patch disabled ($A_N \neq P_N$).
8. Duration of short circuit should not exceed 1 second.
9. I_{CC} is measured with all inputs at 4.5V and the outputs open.

TEST LOAD CIRCUIT

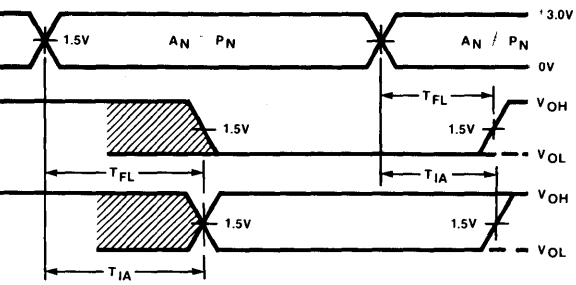


VOLTAGE WAVEFORM



Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs.

TIMING DIAGRAM



TIMING DEFINITIONS

T_{IA} Delay between latest Address variable change and when Data Output becomes stable.

T_{FL} Delay between latest Address variable change and when $\overline{\text{Flag}}$ output becomes stable.

VIRGIN DEVICE

The 82S106/107 are shipped in an unprogrammed state, characterized by:

1. All internal Ni-Cr links are intact.
2. Each comparator address (P-term) contains both true and complement values of every input variable I_m (P-terms always logically "false").
3. The storage Matrix contains all "1".
4. The polarity of each output is set to active high.
5. All outputs are at a low logic level.

RECOMMENDED PROGRAMMING PROCEDURE

To program up to 48 address-data pair locations, follow the program/verify procedures outlined below. To maximize recovery from programming errors, leave all links corresponding to unused address-data pairs intact.

SET-UP

Terminate all device outputs with a 10K resistor to +5V. Set GND (pin 14) to 0V.

ADDRESS COMPARATOR

Program Pn Address

Program one input at the time and one P-term at the time. Unused comparator inputs must be programmed as Don't Care for all programmed P-terms.

1. Set FE (pin 1) to V_{FEL}, and V_{CC} (pin 28) to V_{CCP}.
2. Disable all device outputs by setting $\overline{Flag}$ (pin 19) to V_{IH}.
3. Disable all comparator inputs by applying V_{IX} to inputs I_0 through I_{15}.
4. Address the P-term to be programmed (No. 0 through 47) by forcing the corresponding binary code on outputs F_0 through F_5 with F_0 as LSB. Use standard TTL logic levels V_{OHF} and V_{OLF}.

5a. If the P-term contains neither I_0 nor $\overline{I_0}$ (input is a Don't Care), fuse both I_0 and $\overline{I_0}$ links by executing both steps 5b and 5c, before continuing with step 7.

5b. If the P-term contains I_0, set to fuse the $\overline{I_0}$ link by lowering the input voltage at I_0 from V_{IX} to V_{IH}. Execute step 6.

5c. If the P-term contains $\overline{I_0}$, set to fuse the I_0 link by lowering the input voltage at I_0 from V_{IX} to V_{IL}. Execute step 6.

6a. After t_D delay, raise FE from V_{FEL} to V_{FEH}.

6b. After t_D delay, pulse the $\overline{Flag}$ input from V_{IH} to V_{IX} for a period t_p.

6c. After t_D delay, return FE input to V_{FEL}.

7. Disable programmed input by returning I_0 to V_{IX}.
8. Repeat steps 5 through 7 for all other input variables.
9. Repeat steps 4 through 8 for all other P-terms.
10. Remove V_{IX} from all input variables.

VERIFY P_n ADDRESS

1. Set FE to V_{FEL}, and V_{CC} to V_{CCP}.
2. Enable F_7 output by setting $\overline{Flag}$ to V_{IX}.
3. Disable all comparator inputs by applying V_{IX} to inputs I_0 through I_{15}.
4. Address the P-term to be verified (No. 0 through 47) by forcing the corresponding binary code on outputs F_0 through F_5.
5. Interrogate Input I_0 as follows:
   A. Lower the input voltage at I_0 from V_{IX} to V_{IH}, and sense the logic state of output F_7.
   B. Lower the input voltage at I_0 from V_{IH} to V_{IL}, and sense the logic state output F_7.

The state of I_0 contained in the P-term is determined in accordance with the following truth table:

I_0	F_7	COMPARATOR INPUT STATE CONTAINED IN P-TERM
0 1	1 0	$\overline{I_0}$
0 1	0 1	I_0
0 1	1 1	Don't Care
0 1	0 0	(I_0), ($\overline{I_0}$)

Note that 2 tests are required to uniquely determine the state of the input contained in the P-term.

6. Disable verified input by returning I_0 to V_{IX}.
7. Repeat steps 5 and 6 for all other input variables.
8. Repeat steps 4 through 7 for all other P-terms.
9. Remove V_{IX} from all comparator inputs.

STORAGE MATRIX

Program Output Data

Program one output at the time for one P-term at the time.

1. Set FE to V_{FEL}.
2. Disable the chip by setting $\overline{Flag}$ to V_{IH}.
3. After t_D delay, set V_{CC} to V_{CCS}, and inputs I_6 through I_{15} to V_{IH}, V_{IL}, or V_{IX}.
4. Address the P-term to be programmed (No. 0 through 47) by applying the corresponding binary code to comparator inputs I_0 through I_5, with I_0 as LSB.
5. To program a logic "0" at output F_0, force F_0 to V_{OPF}.

6a. After t_D delay, raise FE (pin 1) from V_{FEL} to V_{FEH}.

6b. After t_D delay, pulse the $\overline{Flag}$ input from V_{IH} to V_{IX} for a period t_p.

6c. After t_D delay, return FE input to V_{FEL}.

6d. After t_D delay, remove V_{OPF} from output F_0.

7. Repeat steps 5 and 6 for all other output functions.
8. Repeat steps 4 through 7 for all other P-terms.
9. Remove V_{CCS} from V_{CC}.

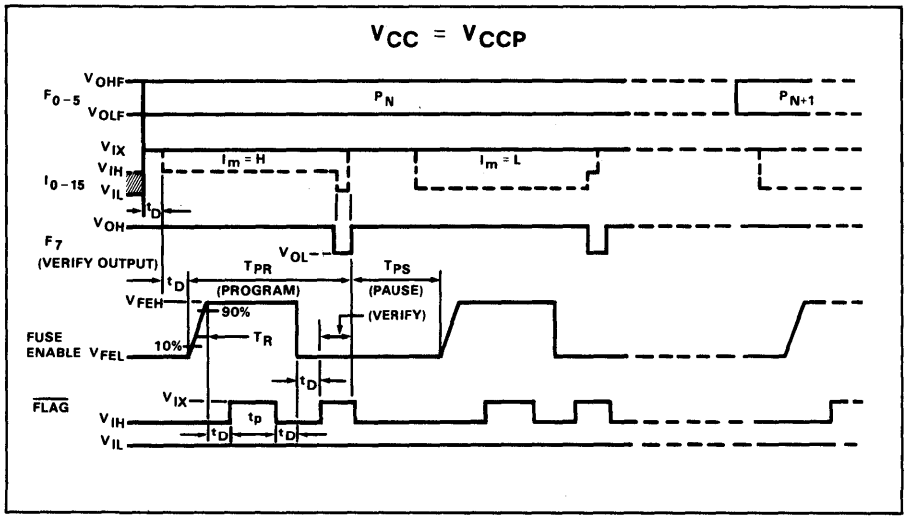
Verify Output Data

1. Set FE to V_{FEL}.
2. Disable the chip by setting $\overline{Flag}$ to V_{IH}.
3. After t_D delay, set V_{CC} to V_{CCS}, and inputs I_0 through I_{15} to V_{IH}, V_{IL}, or V_{IX}.
4. Address the P-term to be verified (No. 0 through 47) by applying the corresponding binary code to comparator inputs I_0 through I_5.
5. After t_D delay, enable the chip by setting $\overline{Flag}$ to V_{IL}.
6. To determine the status of each output link in the Storage Matrix, sense the state of outputs F_0 through F_7. The status of the link is given by the following truth table:

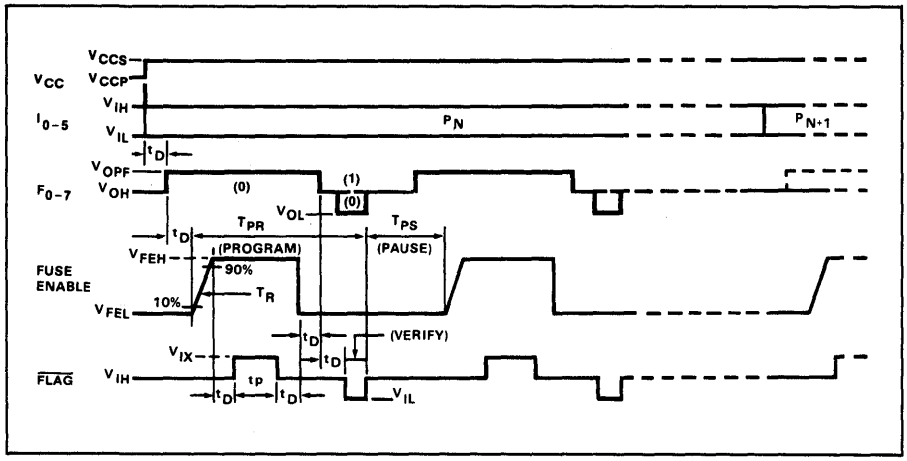
F_P	LINK
0	Fused
1	Present

7. Repeat steps 4 through 6 for all other P-terms.
8. Remove V_{CCS} from V_{CC}.

ADDRESS COMPARATOR PROGRAM-VERIFY SEQUENCE (TYPICAL)



STORAGE MATRIX PROGRAM-VERIFY SEQUENCE (TYPICAL)



FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

PROGRAMMING SYSTEM SPECIFICATIONS[1] (T_A = +25°C)

PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{CCS}	V_{CC} supply (program/verify Storage Matrix)[2]	I_{CCS} = 550mA, min. Transient or steady state	8.25	8.5	8.75	V
I_{CCS}	I_{CC} limit	V_{CCS} = +8.50 ± .25V	550		1,000	mA
	Input voltage					V
V_{IH}	High		2.4		5.5	
V_{IL}	Low		0	0.4	0.8	
	Input current					μA
I_{IH}	High	V_{IH} = +5.5V			50	
I_{IL}	Low	V_{IL} = 0V			-500	
	Forced output voltage					V
V_{OHF}	High		2.4		5.5	
V_{OLF}	Low		0	0.4	0.8	
	Output current					
I_{OHF}	High	V_{OHF} = +5.5V			100	μA
I_{OLF}	Low	V_{OLF} = 0V			-1	mA
V_{IX}	$\overline{Flag}$ program enable level		9.5	10	10.5	V
I_{IX1}	Input current	V_{IX} = +10V			10	mA
I_{IX2}	$\overline{Flag}$ input current	V_{IX} = +10V			10	mA
V_{FEH}	FE supply (program)[3]	I_{FEH} = 300 ± 25mA, Transient or steady state	16.0	17.0	18.0	V
V_{FEL}	FE supply (idle)	I_{FEL} = -1mA, max	1.25	1.5	1.75	V
I_{FEH}	FE supply current limit	V_{FEH} = +17 ± 1V	275	300	325	mA
V_{CCP}	V_{CC} supply (program/verify Address Comparator)	I_{CCP} = 550mA, min. Transient or steady state	4.75	5.0	5.25	V
I_{CCP}	I_{CC} limit	V_{CCP} = +5.0 ± .25V	550		1,000	mA
V_{OPF}	Forced output (program)		9.5	10	10.5	V
I_{OPF}	Output current (program)				10	mA
T_R	FE pulse rise time	10% to 90%	10		50	μs
t_P	$\overline{Flag}$ programming pulse width		0.3	0.4	0.5	ms[5]
t_D	Pulse sequence delay		10			μs
T_{PR}	Programming time			0.6		ms
$\frac{T_{PR}}{T_{PR}+T_{PS}}$	Programming duty cycle				50	%
F_L	Fusing attempts per link				2	cycle
V_S	Verify threshold[4]		1.4	1.5	1.6	V

NOTES
1. These are specifications which a Programming System must satisfy in order to be qualified by Signetics.
2. Bypass V_{CC} to GND with a 0.01μf capacitor to reduce voltage spikes.
3. Care should be taken to ensure that the voltage is maintained during the entire fusing cycle. The recommended supply is a constant current source clamped at the specified voltage limit.
4. V_S is the sensing threshold of the FPRP output voltage for a programmed link. It normally constitutes the reference voltage applied to a comparator circuit to verify a successful fusing attempt.
5. These are new limits resulting from device improvements, and which supersede, but do not obsolete the performance requirements of previously manufactured programming equipment.

FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

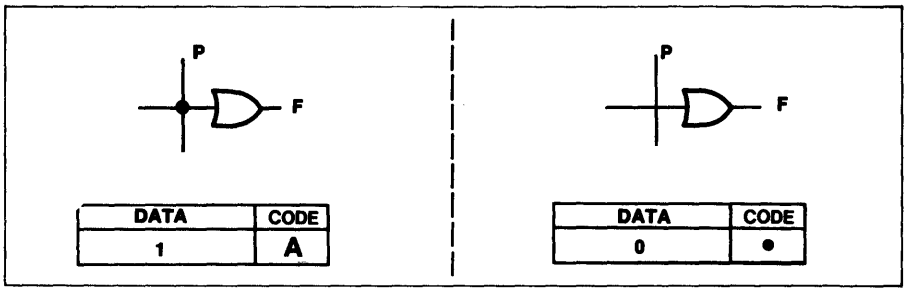
LOGIC PROGRAMMING

The FPRP can be programmed by means of Logic programming equipment.

With Logic programming, the AND/OR gate input connections necessary to implement the desired logic function are coded directly from memory patch specifications using the Program Table on the following page.

In this Table, the logic state of variables I and F associated with each Patch address P_n is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

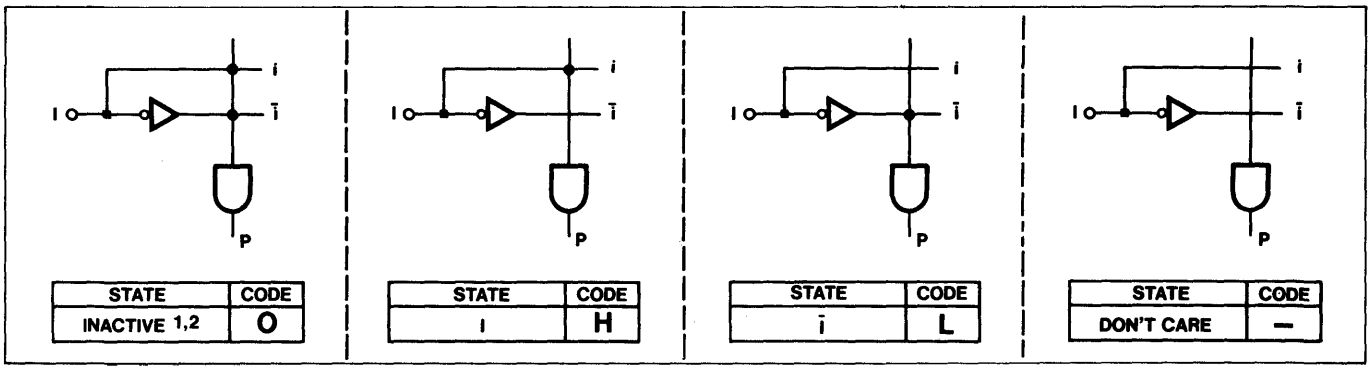
"OR" ARRAY - (F)



DATA	CODE
1	A

DATA	CODE
0	•

"AND" ARRAY - (I),



STATE	CODE
INACTIVE [1,2]	O

STATE	CODE
I	H

STATE	CODE
$\bar{I}$	L

STATE	CODE
DON'T CARE	–

NOTES
1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates P_n.
2. Any gate P_n will be unconditionally inhibited if any one of its (I) link pairs is left intact.

FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

FPRP PROGRAM TABLE (Logic)

THIS PORTION TO BE COMPLETED BY SIGNETICS

CF (XXXX) ______

CUSTOMER SYMBOLIZED PART # ______

DATE RECEIVED ______

COMMENTS ______

CUSTOMER NAME ______

PURCHASE ORDER # ______

SIGNETICS DEVICE # ______

TOTAL NUMBER OF PARTS ______

PROGRAM TABLE # ______

REV ______ DATE ______

PROGRAM TABLE ENTRIES				
COMPARATOR INPUT			OUTPUT FUNCTION	
I_m	$\overline{I_m}$	Don't care	"1"	"0"
H	L	— (dash)	A	• (period)
NOTE Enter (-) for unused inputs of all active P-terms.			NOTE Enter (A) for unused outputs of all active P-terms.	

NOTES
1. Unused inputs and outputs are FPRP terminals left floating.
2. Input and output fields of inactive P-terms can be left blank.

PRODUCT TERM	INPUT VARIABLE																ACTIVE LEVEL							
																	H	H	H	H	H	H	H	H
																	OUTPUT FUNCTION							
NO	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0																								
1																								
2																								
3																								
4																								
5																								
6																								
7																								
8																								
9																								
10																								
11																								
12																								
13																								
14																								
15																								
16																								
17																								
18																								
19																								
20																								
21																								
22																								
23																								
24																								
25																								
26																								
27																								
28																								
29																								
30																								
31																								
32																								
33																								
34																								
35																								
36																								
37																								
38																								
39																								
40																								
41																								
42																								
43																								
44																								
45																								
46																								
47																								

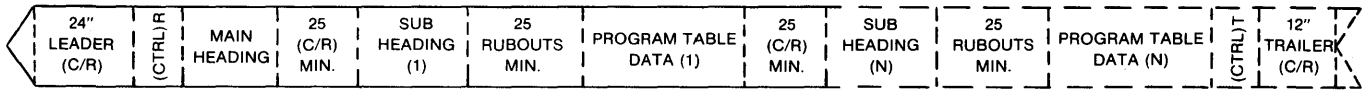
FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

TWX TAPE CODING FORMAT

The FPRP Program Table can be sent to Signetics in ASCII code format via airmail using any type of 8-level tape (paper, mylar; fanfold, etc.), or via TWX: just dial (910) 339-9283, tell the operator to turn the paper puncher on, and acknowledge. At the end of transmission instruct the operator to send tape to Signetics Order Entry.

A number of Program Tables can be sequentially assembled on a continuous tape as follows, however limit tape length to a roll of 1.75 inch inside diameter, and 4.25 inch outside diameter:



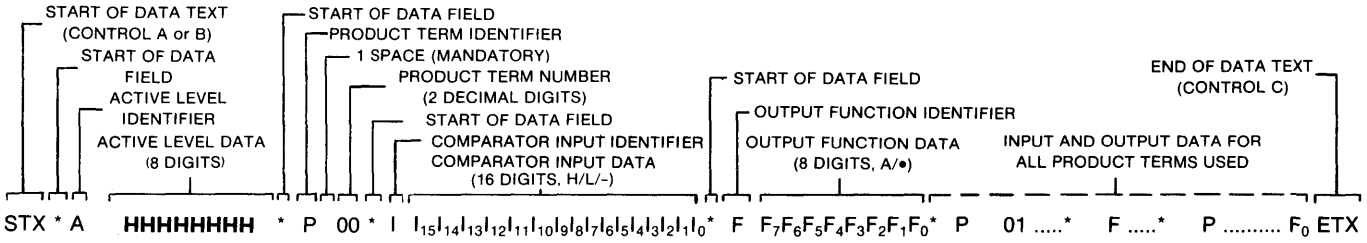
A. The MAIN HEADING at the beginning of tape includes the following information, with each entry preceded by a ($) character, whether used or not:

1. Customer Name _____
2. Customer TWX No. _____
3. Date _____
4. Purchase Order No. _____
5. Number of Program Tables _____
6. Total Number of Parts _____

B. Each SUB HEADING should contain specific information pertinent to each Program Table as follows, with each entry preceded by a ($) character, whether used or not:

1. Signetics Device No. _____
2. Program Table No. _____
3. Revision _____
4. Date _____
5. Customer Symbolized Part No. _____
6. Number of Parts _____

C. Program Table data blocks are initiated with an STX character, and terminated with an ETX character. The body of the data consists of Output Active Level, Product Term, and Output Function information separated by appropriate identifiers in accordance with the following format:



COMPARATOR INPUT		
I_m	$\overline{I_m}$	Don't care
H	L	—(dash)

OUTPUT FUNCTION	
"1"	"0"
A	• (period)

NOTES
1. Enter (-) for unused inputs of all active P-terms.
2. Enter (A) for unused outputs of all active P-terms.
3. Unused inputs and outputs are FPRP terminals left floating.

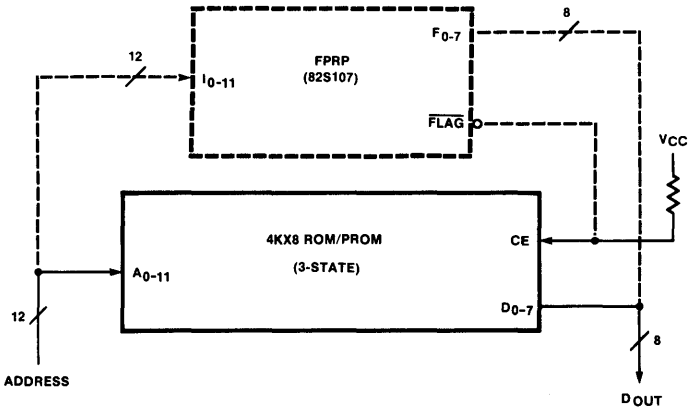
Although the Product Term data are shown entered in sequence, this is not necessary. It is possible to input only one Product Term, if desired. Unused Product Terms require no entry. ETX signalling end of Program Table may occur with less than the maximum number of Product Terms entered.

NOTES
1. Corrections to any entry can be made by backspace and rubout. However, limit consecutive rubouts to less than 25.
2. Terms can be re-entered any number of times. The last entry for a particular P-Term will be interpreted as valid data.
3. To facilitate an orderly Teletype print out, carriage returns, line feeds, spaces, rubouts etc. may be interspersed between data groups.
4. Comments are allowed between data fields, provided that an asterisk (*) is not used in any Heading or Comment entry.

FIELD PROGRAMMABLE ROM PATCH (16X48X8) 82S106 (O.C.)/82S107 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 28

TYPICAL APPLICATION



PATCH SPECIFICATIONS

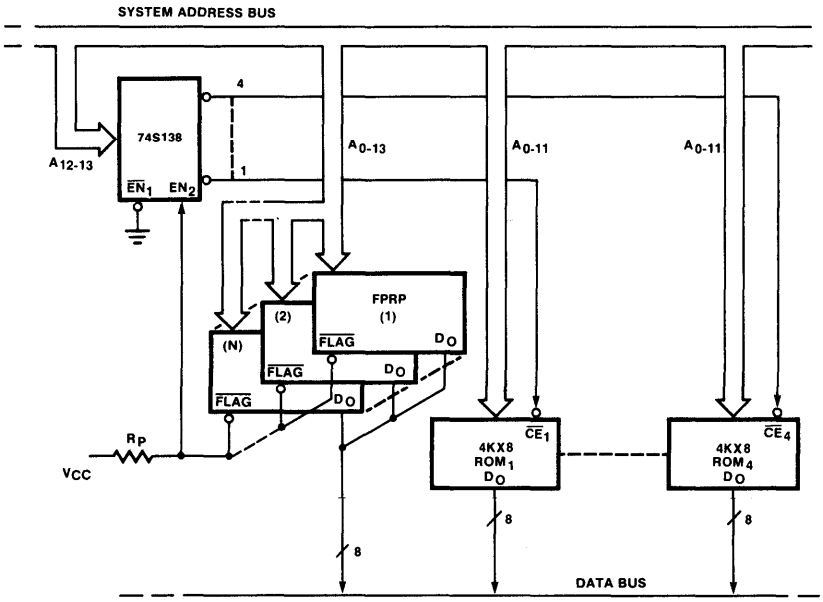
ADDRESS	NEW DATA
750	A9
5FE	7F
0A4	B4
0A5	
0A6	
0A7	

FPRP PROGRAM TABLE

INPUT ADDRESS	PRODUCT TERM NO.	COMPARATOR INPUT 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	ACTIVE LEVEL H H H H H H H H OUTPUT FUNCTION 7	6	5	4	3	2	1	0	OUTPUT DATA
750	0	–	–	–	–	L	H	H	H	L	H	L	H	L	L	L	L	A	•	A	•	A	•	•	A	A9
5FE	1	–	–	–	–	L	H	L	H	H	H	H	H	H	H	H	L	•	A	A	A	A	A	A	A	7F
0A4–0A7	2	–	–	–	–	L	L	L	L	H	L	H	L	L	H	–	–	A	•	A	A	•	A	•	•	B4
I/O ASSIGNMENT		UNUSED				A_{11}											A_0	D_7							D_0	

Since Patch inputs I_{12-15} are unused, they are programmed Don't Care (–), and left open in the circuit. Note however that I_0 and I_1 are programmed as (–) in P-term 2 because they define an address block of 4 memory locations.

EXPANSION



When more than 48 memory locations need to be patched, several devices can be paralleled by wire-ANDing the $\overline{Flag}$ outputs.

FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

DESCRIPTION

The 82S150 and the 82S151 are single level logic elements, consisting of 12 AND gates with fusible link connections for programming I/O polarity, I/O direction and output enable control.

All gates are linked to 6 inputs (I) and 12 bidirectional I/O lines (B). These yield variable I/O gate configurations via 3 direction control gates (D), ranging from 18 inputs to 12 outputs.

On chip T/C buffers couple either True (I, B) or Complement ($\overline{I}$, $\overline{B}$) input polarities to each AND gate. The polarity of all gate outputs is individually programmable through a set of EX-OR gates for implementing AND/NAND logic functions. Alternately, if desired, OR/NOR logic functions can also be realized by programming for each gate the complement of its inputs and output (DeMorgan's Theorem).

The 82S150 and the 82S151 are field programmable, enabling the user to quickly generate custom patterns using standard programming equipment.

Both devices are available in a 20-pin slim line package. For the commercial temperature range (0°C to +75°C) specify N82S150/151 N or F. For the military temperature range (−55°C to +125°C) specify S82S150/151 F only.

FEATURES

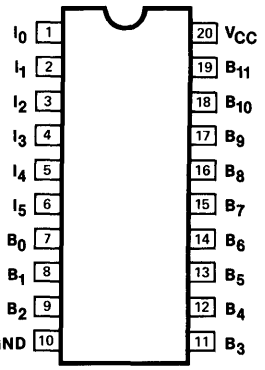
- **Field Programmable (Ni-Cr link)**
- **6 inputs**
- **12 AND gates**
- **12 bidirectional I/O lines**
- **Active high or low outputs**
- **Programmable output enable**
- ***Power dissipation: 650mW (typ)***
- **I/O propagation delay: 30ns (max)**
- **Input loading**
  N82S150/151: −100μA (max)
  S82S150/151: −150μA (max)
- **Output options**
  82S150: open collector
  82S151: tri-state
- **TTL compatible**

APPLICATIONS

- **Random gating functions**
- **Address decoding**
- **Code detectors**
- **Memory mapped I/O**
- **Fault monitors**
- **I/O port decoders**

PIN CONFIGURATION

F,N PACKAGES



F = Cerdip
N = Plastic

LOGIC FUNCTIONS

Typical Gate Functions:

At L = Open

$X = A \bullet \overline{B} \bullet C \bullet \ldots\ldots$

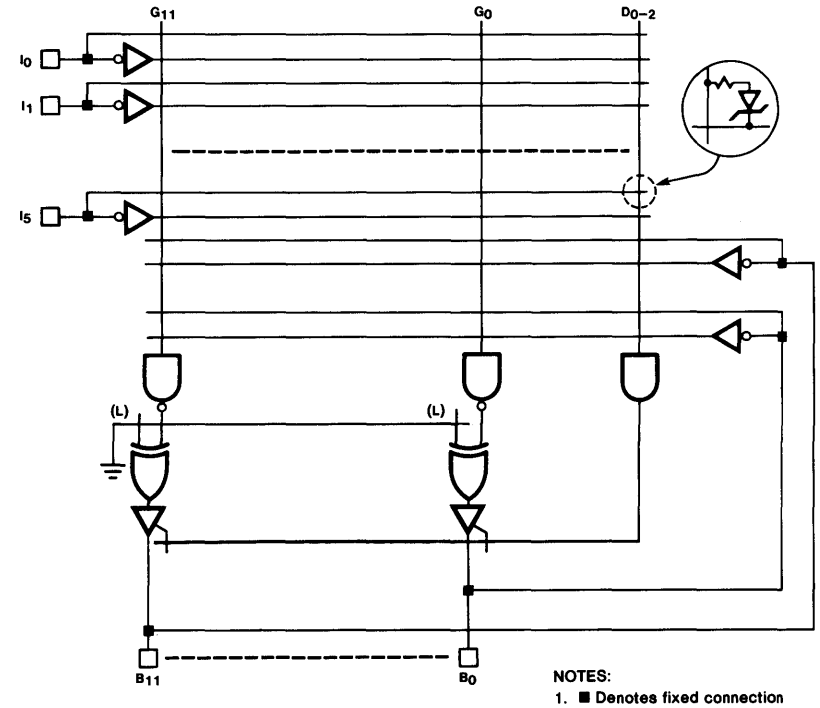
At L = Closed

$X = \overline{A \bullet \overline{B} \bullet C \bullet \ldots\ldots}$

$X = \overline{A} + B + \overline{C} + \ldots\ldots$

NOTES:
1. For each of the 12 outputs, either function X (active-high) or $\overline{X}$ (active-low) is available, but not both. The desired output polarity is programmed via link (L).
2. X, A, B, C, etc. are user defined connections to fixed inputs (I) and bidirectional pins (B).

FUNCTIONAL DIAGRAM



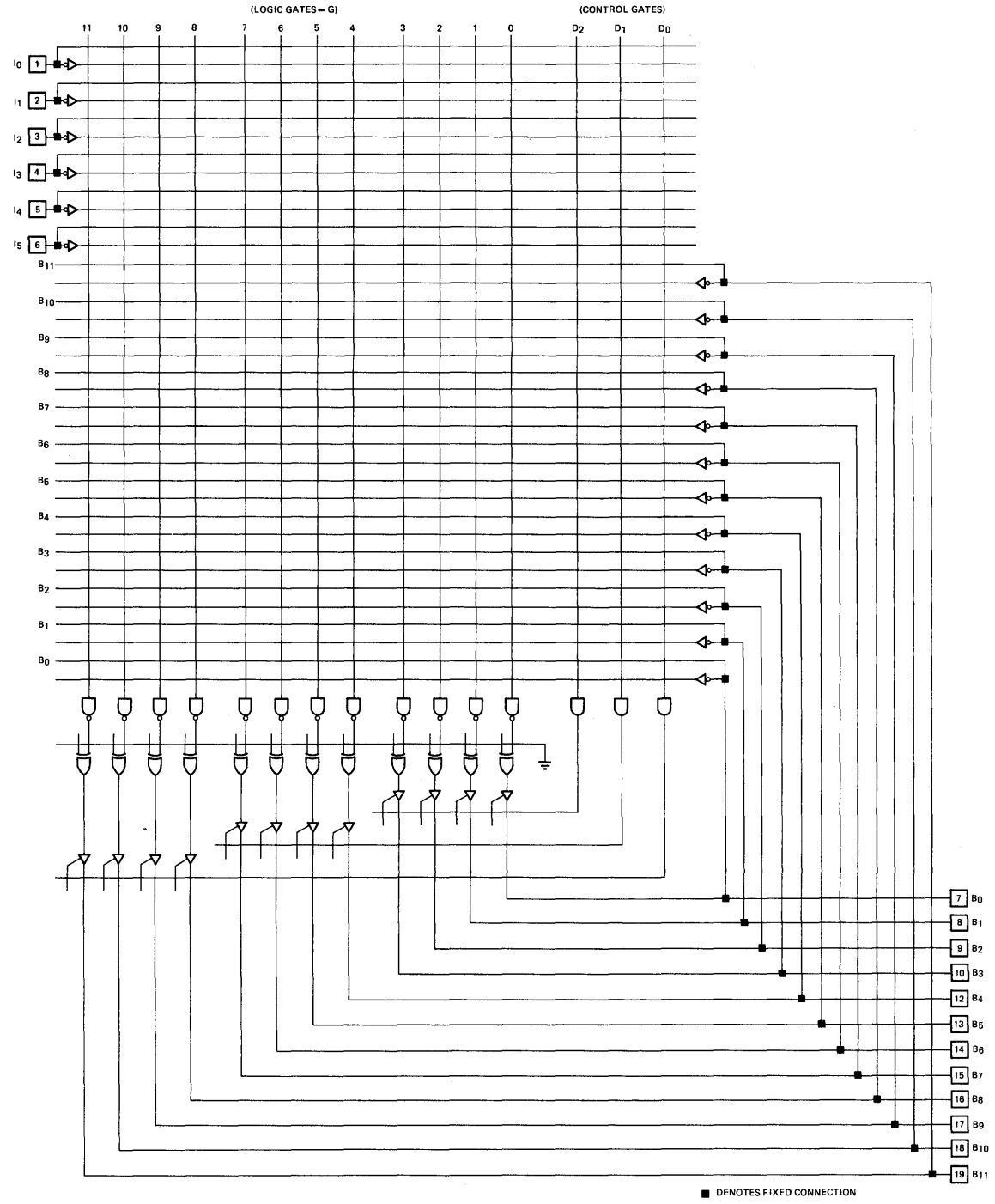
NOTES:
1. ■ Denotes fixed connection
2. All Ni-Cr links are initially intact.

FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

FPGA LOGIC DIAGRAM



FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

ABSOLUTE MAXIMUM RATINGS

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	−30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			C°
T_A	Operating			
	N82S150/151	0	+75	
	S82S150/151	−55	+125	
T_{STG}	Storage	−65	+150	

THERMAL RATINGS

TEMPERATURE	Military	Commercial
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHACTERISTICS

N82S150/151: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S150/151: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

PARAMETER		TEST CONDITION	N82S150/151 Min	N82S150/151 Typ[2]	N82S150/151 Max	S82150/151 Min	S82150/151 Typ[2]	S82150/151 Max	UNIT
	Input voltage[3]								V
V_{IL}	Low	V_{CC} = Min			.85			.80	
V_{IH}	High	V_{CC} = Max	2.0			2.0			
V_{IC}	Clamp[3,4]	V_{CC} = Min, Iin = −18mA		.8	−1.2		.8	−1.2	
	Output voltage	V_{CC} = Min							V
V_{OL}	Low[3,5]	I_{OL} = 10mA			.5				
V_{OL}	Low[3,5]	I_{OL} = 8mA						.5	
V_{OH}	High[3,6]	I_{OH} = −2mA	2.4			2.4			
	Input Current								μA
I_{IL}	Low	V_{IN} = 0.45V			−100			−150	
I_{IH}	High	V_{IN} = 5.5V			40			50	
	Output Current	V_{CC} = max							
I_{OLK}	Leakage (82S150)	V_{OUT} = 5.5V			40			60	μA
$I_{O(OFF)}$	Hi-Z state (82S151)	V_{OUT} = 5.5V			40			60	μA
		V_{OUT} = .45V			−40			−60	
I_{OS}	Short circuit (82S151)[4,6,7]	V_{OUT} = 0V	−20		−70	−15		−85	mA
I_{CC}	V_{CC} supply current[8]	V_{CC} = max		130	155		130	155	mA
	Capacitance	V_{CC} = 5V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_B	I/O	V_B = 2.0V		15			15		

AC ELECTRICAL CHACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1K\Omega$
N82S150/151: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S150/151: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

	PARAMETER	TO	FROM	TEST CONDITIONS	N82S150/151 Min	N82S150/151 Typ[2]	N82S150/151 Max	S82S150/151 Min	S82S150/151 Typ[2]	S82S150/151 Max	UNIT
T_{PD}	Progagation delay	Output ±	Input ±	C_L = 30pF		25	30		25		ns
T_{OE}	Output enable	Output−	Input ±			25	30		25		ns
T_{OD}	Output disable[9]	Output+	Input ±	C_L = 5pF		25	30		25		

NOTES

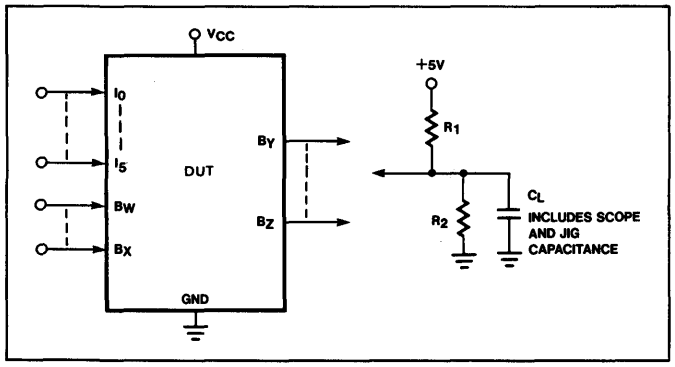
1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device of these or any other condition above those indicated in the operation of the device specifications is not implied.
2. All typical values are at V_{CC} = 5V, T_A = 25°C.
3. All voltage values are with respect to network ground terminal.
4. Test one at a time.
5.
6.
7. Duration of short circuit should not exceed 1 second.
8.
9. Measured at $V_T = V_{OL} + 0.5V$.

FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

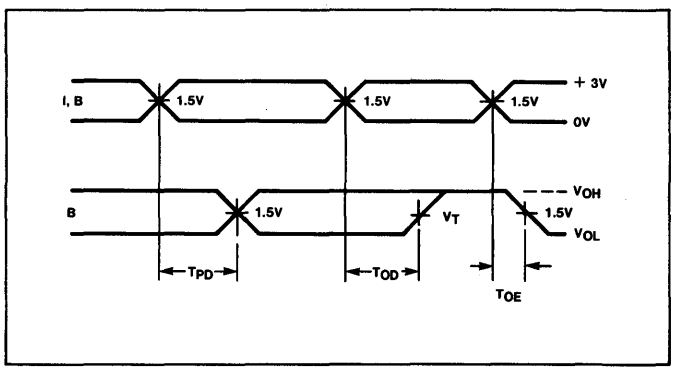
Preview

INTEGRATED FUSE LOGIC
SERIES 20

TEST LOAD CIRCUIT



TIMING DIAGRAM



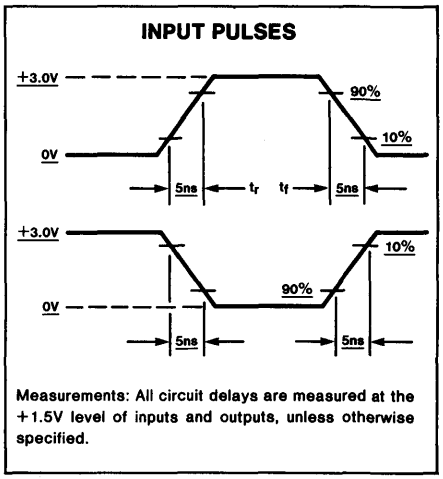
TIMING DEFINITIONS

TPD Propagation delay between input and output.

TOD Delay between input change and when output is off (Hi-Z or High).

TOE Delay between input change and when output reflects specified output level.

VOLTAGE WAVEFORM



Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs, unless otherwise specified.

FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

LOGIC PROGRAMMING

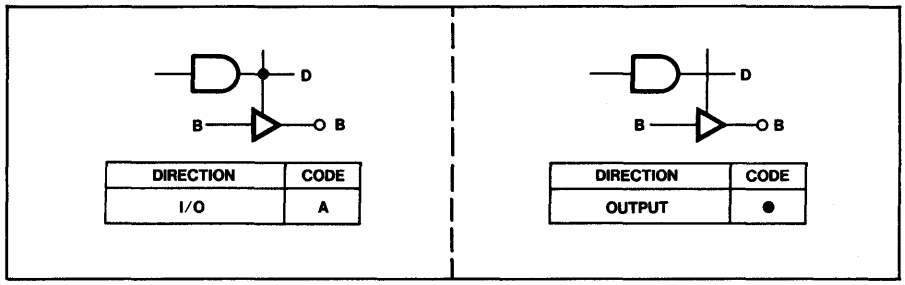
In a virgin device all Ni-Cr links are intact.

The FPGA can be programmed by means of Logic programming equipment.

With Logic programming, the AND/EX-OR gate input connections necessary to implement the desired logic function are coded directly from logic equations using the Program Table on the following page.

In this Table the logic state or action of variables I and B associated with each gate Gn, Dn is assigned a symbol which results in the proper fusing pattern of corresponding links defined as follows:

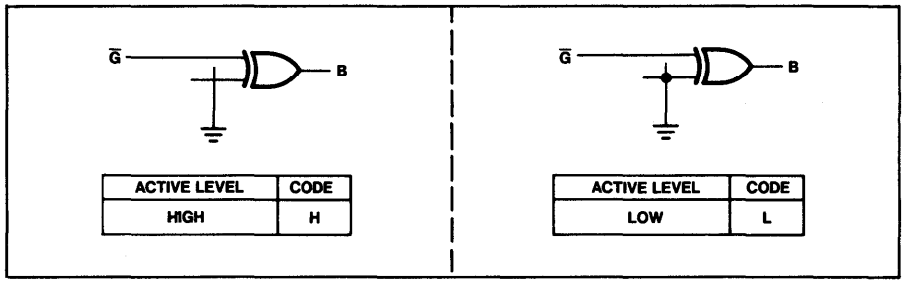
I/O DIRECTION—(B)



DIRECTION	CODE
I/O	A

DIRECTION	CODE
OUTPUT	●

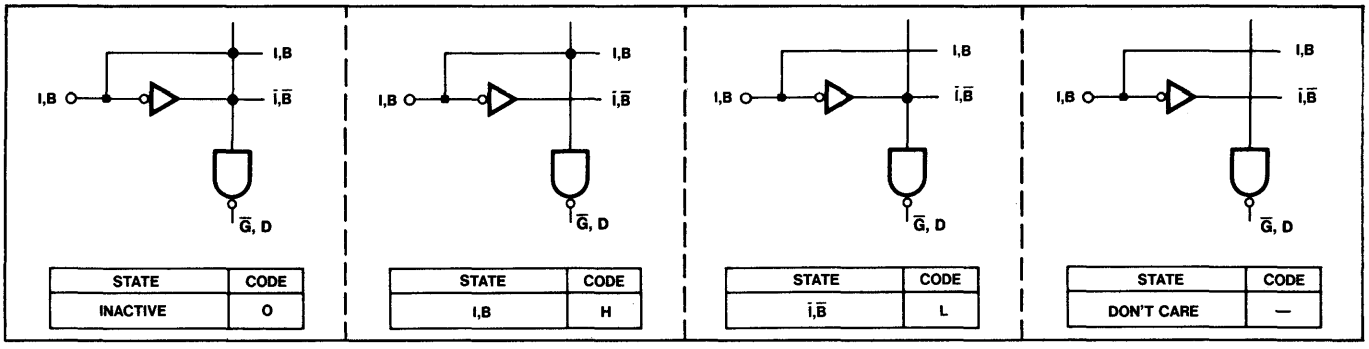
EX-OR ARRAY—(B)



ACTIVE LEVEL	CODE
HIGH	H

ACTIVE LEVEL	CODE
LOW	L

"AND" ARRAY—(I,B)



STATE	CODE
INACTIVE	0

STATE	CODE
I,B	H

STATE	CODE
$\bar{I}$,$\bar{B}$	L

STATE	CODE
DON'T CARE	—

NOTES

1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates Gn, Dn.
2. Any gate Gn, Dn will be unconditionally inhibited if any one of its (I, B) link pairs is left intact.

FIELD PROGRAMMABLE GATE ARRAY (18X12) 82S150 (O.C.)/82S151 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

FPGA PROGRAM TABLE (Logic)

NOTES
1. The FPGA is shipped with all links intact. Thus a background of entries corresponding to states of virgin links exists in the table, shown BLANK for clarity.
2. Unused I and B bits are normally programmed Don't Care (-).
3. Unused Gates can be left blank.
4. Disregard Polarity for Gates used only as inputs.

PROGRAM TABLE ENTRIES:

I, B (AND): INACTIVE = 0; I, B = H; $\bar{I}$, B = L; $\bar{I}$, $\bar{B}$ = -; Don't Care = -

G (or B) (POL.): HIGH = H; LOW = L

G (or B) (DIR.): I/O = A; OUTPUT = ●

D2 = _____
G0 = _____
G1 = _____
G2 = _____
G3 = _____
D1 = _____
G4 = _____
G5 = _____
G6 = _____
G7 = _____
D0 = _____
G8 = _____
G9 = _____
G10 = _____
G11 = _____

CUSTOMER NAME _____
PURCHASE ORDER # _____ CF (XXXX) _____
SIGNETICS DEVICE # _____
CUSTOMER SYMBOLIZED PART # _____
TOTAL NUMBER OF PARTS _____
PROGRAM TABLE # _____ REV _____ DATE _____

PIN	GATE	DIR.	POL.	AND I 5	I 4	I 3	I 2	I 1	I 0	B 11	B 10	B 9	B 8	B 7	B 6	B 5	B 4	B 3	B 2	B 1	B 0
	D2																				
6	00																				
7	01																				
8	02																				
9	03																				
	D1																				
11	04																				
12	05																				
13	06																				
14	07																				
	D0																				
15	08																				
16	09																				
17	10																				
18	11																				
		PIN		5	4	3	2	1	19	18	17	16	15	14	13	12	11	9	8	7	6

VARIABLE NAME

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

DESCRIPTION

The 82S152 and 82S153 are two-level logic elements, consisting of 32 AND gates and 10 OR gates with fusible link connections for programming I/O polarity and direction.

All AND gates are linked to 8 inputs (I) and 10 bidirectional I/O lines (B). These yield variable I/O gate configurations via 10 direction control gates (D), ranging from 18 inputs to 10 outputs.

On chip T/C buffers couple either True (I, B) or Complement ($\overline{I}$, $\overline{B}$) input polarities to all AND gates, whose outputs can be optionally linked to all OR gates. Their output polarity, in turn, is individually programmable through a set of EX-OR gates for inplementing AND-OR or AND-NOR logic functions.

The 82S152 and the 82S153 are field programmable, enabling the user to quickly generate custom patterns using standard programming equipment.

Both devices are available in a 20 pin slim line package. For the commercial temperature range (0°C to +75°C) specify N82S152/153 N or F. For the military temperature range (−55°C to +125°C) specify S82S152/153 F only.

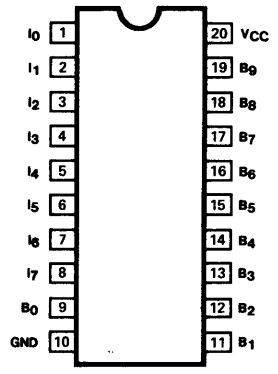
FEATURES

- **Field programmable (Ni-Cr links)**
- **8 inputs**
- **32 AND gates**
- **10 OR gates**
- **10 bidirectional I/O lines**
- **Active high or low outputs**
- **I/O propagation delay:**
  N82S152/153: 40ns (max)
  S82S152/153: 60ns (max)
- **Input loading**
  N82S152/153: −100μA (max)
  S82S152/153: −150μA (max)
- **Power dissipation:**
  650mW (typ)
- **Output options:**
  82S152: open collector
  82S153: tri-state
- **TTL compatible**

APPLICATIONS

- **Random logic**
- **Code converters**
- **Fault detectors**
- **Function generators**
- **Address mapping**
- **Multiplexing**

PIN CONFIGURATION



F = Cerdip
N = Plastic

LOGIC FUNCTION

Typical product term:

$Pn = A \bullet \overline{B} \bullet C \bullet D \bullet \ldots$

Typical logic function:

At L = Closed

$X = P0 + P1 + P2 \ldots$

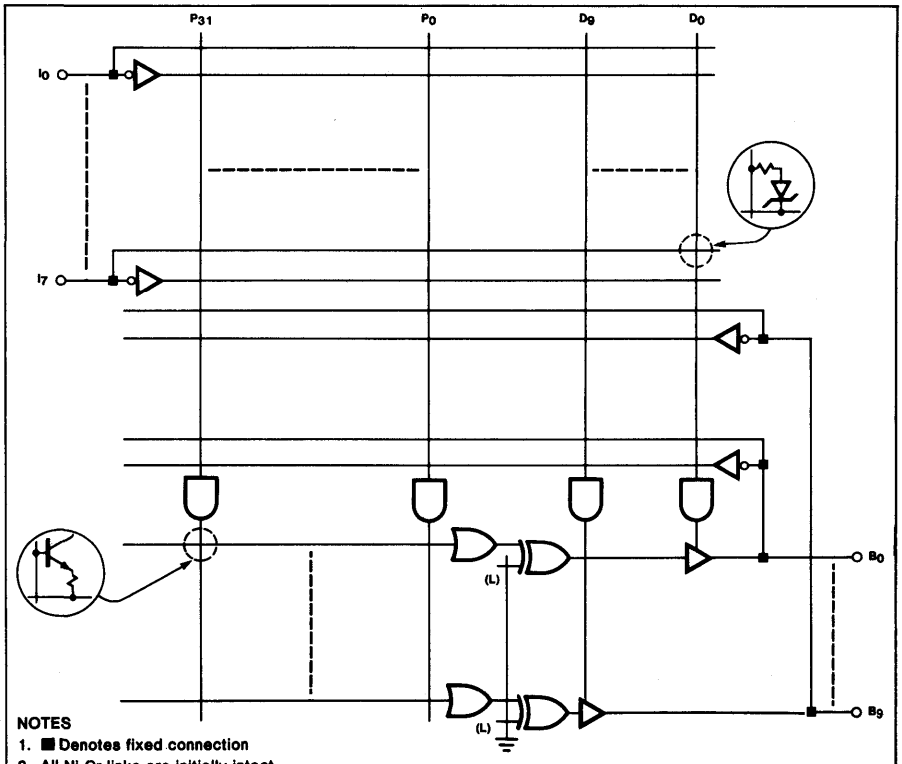
At L = Open

$X = \overline{P0 + P1 + P2 + \ldots}$

$X = \overline{P0} \bullet \overline{P1} \bullet \overline{P2} \ldots$

NOTES
1. For each of the 10 outputs, either function X (active high) or $\overline{X}$ (active low) is available, but not both. The desired output polarity is programmed via link (L).
2. X, A, B, C, etc. are user defined connections to fixed inputs (I) and bidirectional pins (B).

FUNCTIONAL DIAGRAM

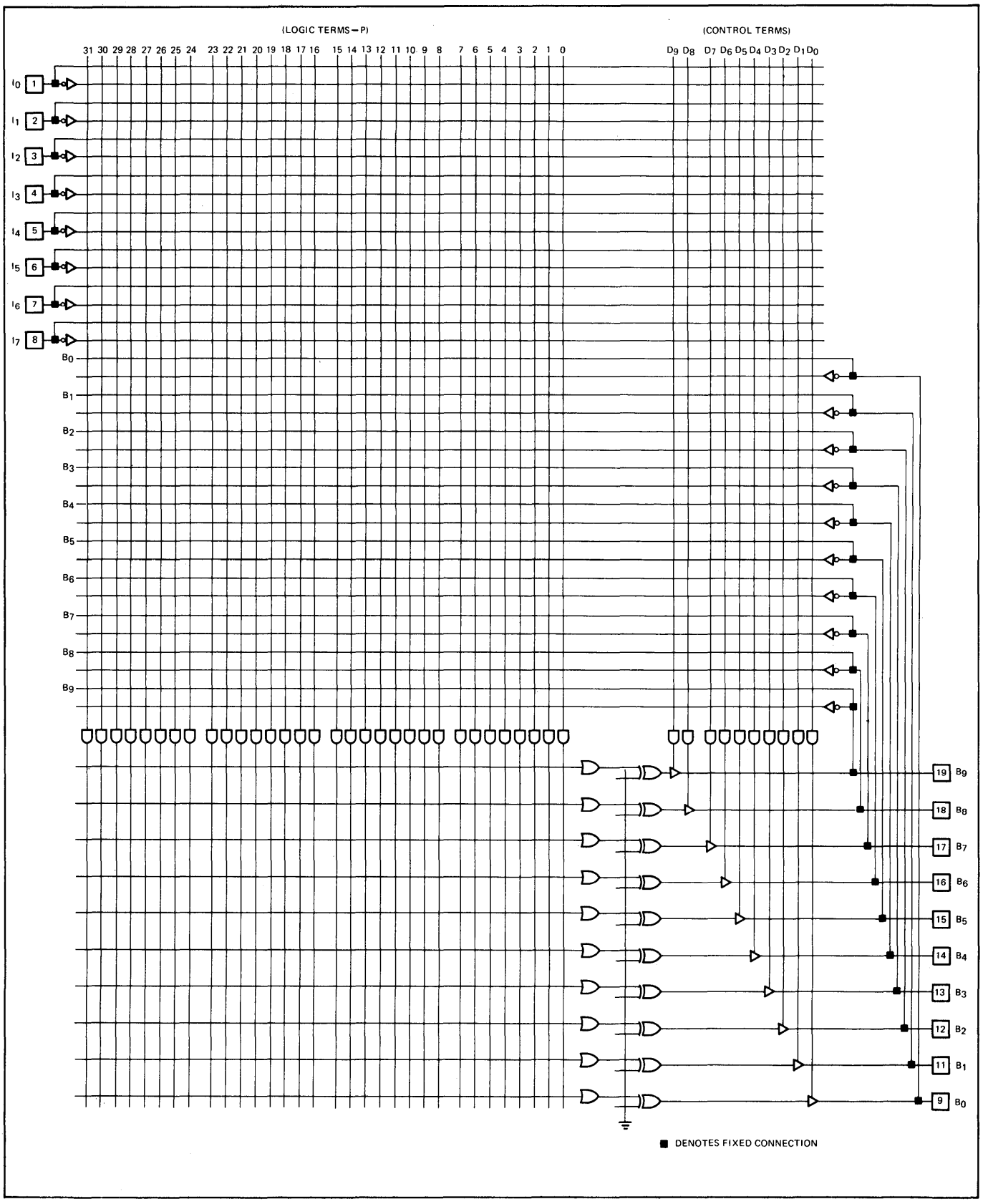


NOTES
1. ■ Denotes fixed connection
2. All Ni-Cr links are initially intact.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

FPLA LOGIC DIAGRAM



FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	−30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			C°
T_A	Operating			
	N82S152/153	0	+75	
	S82S152/153	−55	+125	
T_{STG}	Storage	−65	+150	

THERMAL RATINGS

TEMPERATURE	Military	Commercial
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHARACTERISTICS

N82S152/153: $0°C \leq T_A \leq +75°C$, $4.75 \leq V_{CC} \leq 5.25V$
S82S152/153: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

PARAMETER		TEST CONDITION	N82S152/153 Min	N82S152/153 Typ[2]	N82S152/153 Max	S82S152/153 Min	S82S152/153 Typ[2]	S82S152/153 Max	UNIT
	Input voltage[3]								V
V_{IL}	Low	V_{CC} = min			.85			.80	
V_{IH}	High	V_{CC} = max	2.0			2.0			
V_{IC}	Clamp[3,4]	V_{CC} = min, Iin = −18mA		−.8	−1.2		−.8	−1.2	
	Output voltage	V_{CC} = min							V
V_{OL}	Low[3,5]	I_{OL} = 15mA			.5				
V_{OL}	Low[3,5]	I_{OL} = 12mA						.5	
V_{OH}	High[3,6]	I_{OH} = −2mA	2.4			2.4			
	Input current								μA
I_{IL}	Low	V_{IN} = 0.45V			−100			−150	
I_{IH}	High	V_{IN} = 5.5V			40			50	
	Output current	V_{CC} = max							
I_{OLK}	Leakage (82S152)	V_{OUT} = 5.5V			40			60	μA
$I_{O(OFF)}$	Hi-Z state (82S153)	V_{OUT} = 5.5V			40			60	μA
		V_{OUT} = .45V			−40			−60	
I_{OS}	Short circuit (82S153)[4,6,7]	V_{OUT} = 0V	−20		−70	−15		−85	mA
I_{CC}	V_{CC} supply current[8]	V_{CC} = max		130	155		130	165	mA
	Capacitance	V_{CC} = 5V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_B	I/O	V_B = 2.0V		15			15		

AC ELECTRICAL CHACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1K\Omega$
N82S152/153: $0° C \leq Ta \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S152/153: $-55°C \leq Ta \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

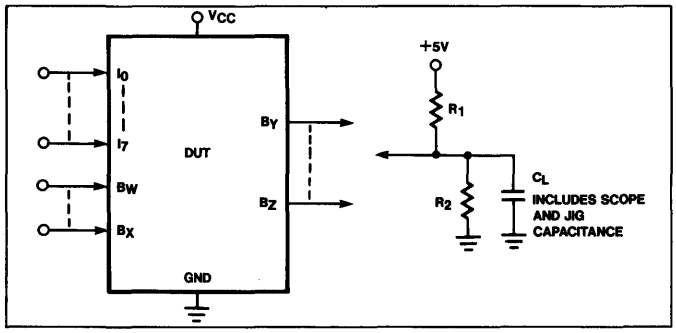
	PARAMETER	TO	FROM	TEST CONDITIONS	N82S152/153 Min	N82S152/153 Typ	N82S152/153 Max	S82S152/153 Min	S82S152/153 Typ	S82S152/153 Max	UNIT
T_{PD}	Progagation delay	Output ±	Input ±	C_L = 30pF		30	40		30	55	ns
T_{OE}	Output enable	Output−	Input ±			25	35		25	45	
T_{OD}	Output disable[9]	Output+	Input ±	C_L = 5pF		25	35		25	45	ns

NOTES
1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device of these or any other condition above those indicated in the operation of the device specifications is not implied.
2. All typical values are at V_{CC} = 5V, T_A = 25°C
3. All voltage values are with respect to network ground terminal.
4. Test one at a time.
5. Measured with +10V applied to I_7.
6. Measured with +10V applied to I_{0-7}. Output sink current is supplied thru a resistor to V_{CC}.
7. Duration of short circuit should not exceed 1 second.
8. I_{CC} is measured with I_{0-7} and B_{0-9} at 4.5V.
9. Measured at $V_T = V_{OL} + 0.5V$.

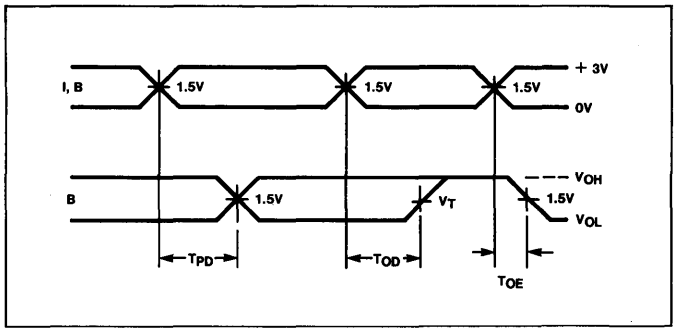
FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

TEST LOAD CIRCUIT



TIMING DIAGRAM



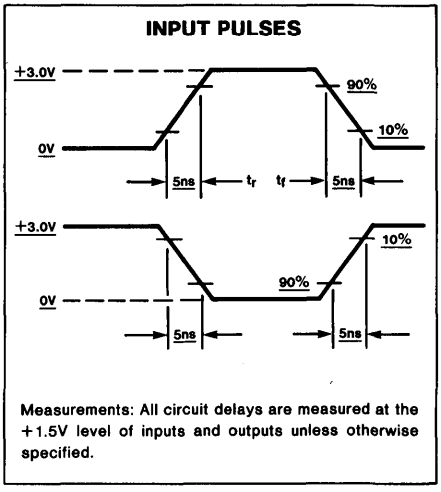
TIMING DEFINITIONS

T_{PD} Propagation delay between input and output.

T_{OD} Delay between input change and when output is off (Hi-Z or High).

T_{OE} Delay between input change and when output reflects specified output level.

VOLTAGE WAVEFORM



Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs unless otherwise specified.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

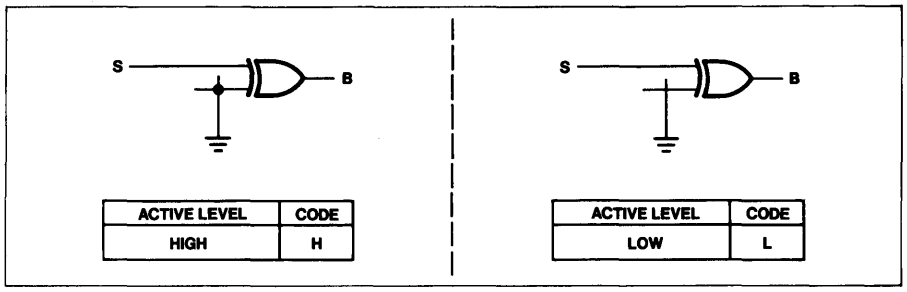
LOGIC PROGRAMMING

The FPLA can be programmed by means of Logic programming equipment.

With Logic programming, the AND/OR/EX-OR gate input connections necessary to implement the desired logic function are coded directly from logic equations using the Program Table on the following page.

In this Table the logic state or action of variables I, P, and B associated with each Sum Term S is assigned a symbol which results in the proper fusing pattern of corresponding link pairs, defined as follows:

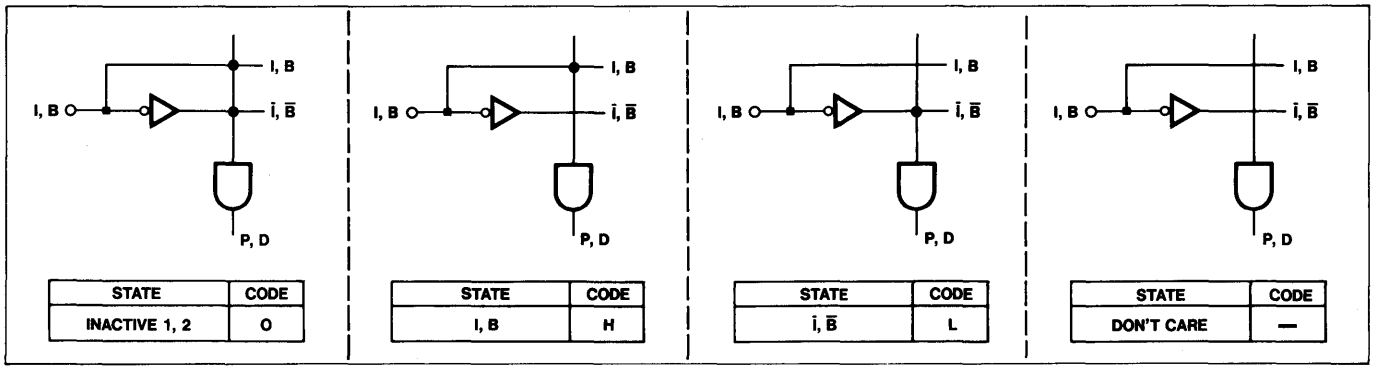
EX-OR ARRAY - (B)



ACTIVE LEVEL	CODE
HIGH	H

ACTIVE LEVEL	CODE
LOW	L

"AND" ARRAY - (I,B)



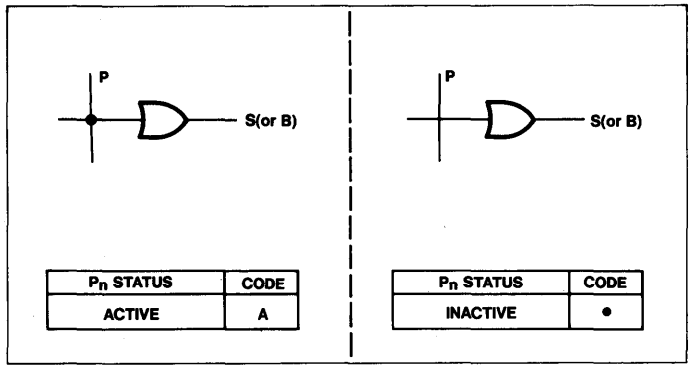
STATE	CODE
INACTIVE 1, 2	O

STATE	CODE
I, B	H

STATE	CODE
Ī, B̄	L

STATE	CODE
DON'T CARE	—

"OR" ARRAY - (B)



P_n STATUS	CODE
ACTIVE	A

P_n STATUS	CODE
INACTIVE	•

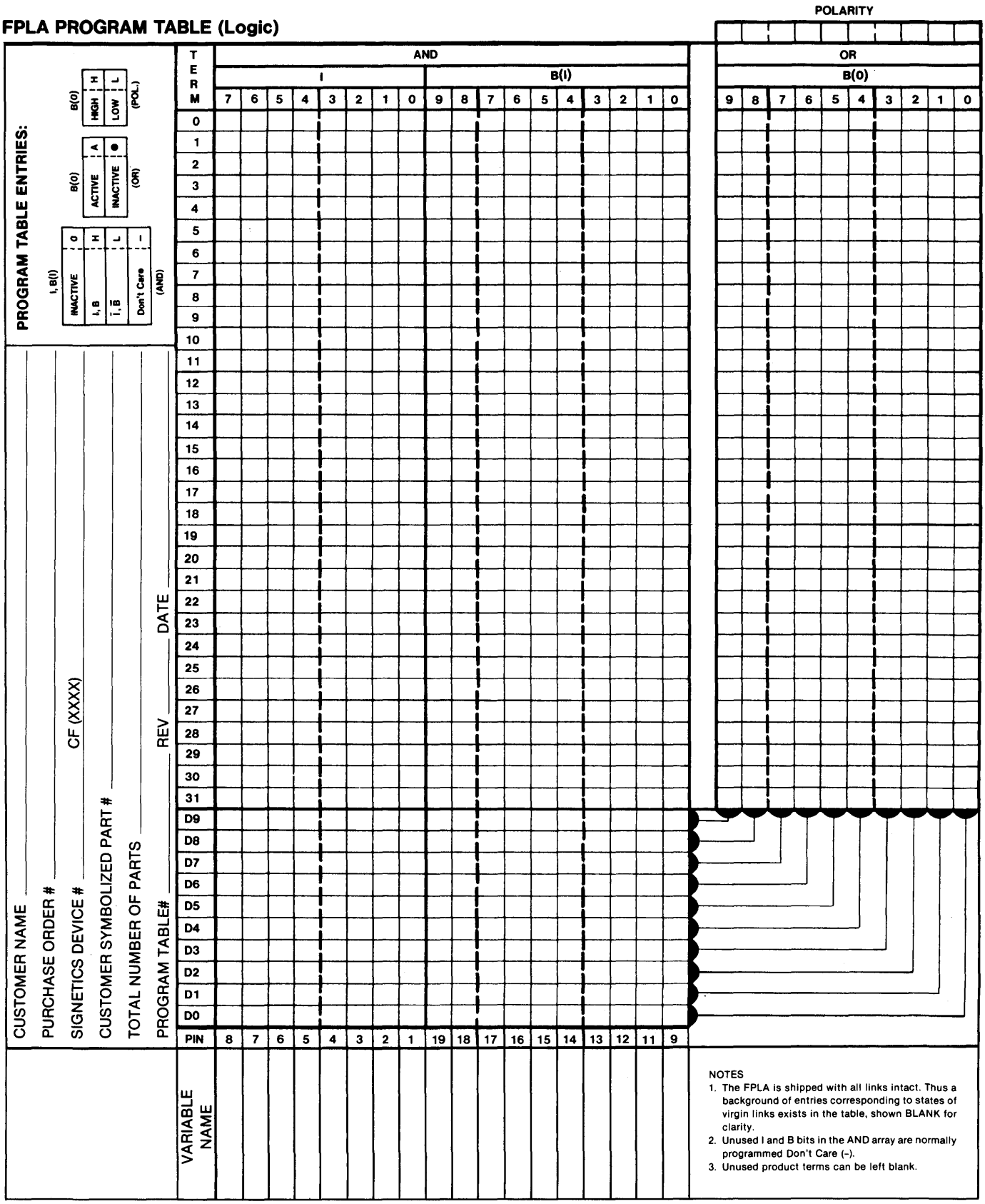
NOTES

1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates P_n, D_n.
2. Any gate P_n, D_n will be unconditionally inhibited if any one of its (I, B) link pairs is left intact.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

FPLA PROGRAM TABLE (Logic)



NOTES
1. The FPLA is shipped with all links intact. Thus a background of entries corresponding to states of virgin links exists in the table, shown BLANK for clarity.
2. Unused I and B bits in the AND array are normally programmed Don't Care (-).
3. Unused product terms can be left blank.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

VIRGIN STATE

A factory shipped virgin device contains all fusible links intact, such that:

1. All outputs at "H" polarity.
2. All P_n terms are disabled.
3. All P_n terms are active on all outputs.
4. Test array is programmed with standard test pattern.

RECOMMENDED PROGRAMMING PROCEDURE

To program the AND, OR, and polarity arrays, the following procedure should be followed. To maximize recovery from programming errors, leave all links in unused device areas intact.

SET-UP

Terminate all device outputs with a 10kΩ resistor to V_{CC}. Set GND (pin 10) to 0V.

PROGRAM-VERIFY

1. **SET-UP:**
   With V_{CC} at GND and B_8 to V_{FSV} select the fuse to be programmed by applying TTL voltage levels to the input sets in accordance with the binary address map on page 8. Also set $B_1 = V_{IH}$, and $B_7 = V_{IL}$. After t_D delay raise V_{CC} to V_{CCP}.
2. **PROGRAM CYCLE:**
   After a delay of t_D raise B_7 to V_{IH}. After another t_D, raise B_8 to V_{FSP}. Following t_D delay, pulse B_1 to V_{IL} for a duration of t_p. Wait t_D and return B_8 to V_{FSV}, and then after t_D return B_7 to V_{IL}.
3. **VERIFY CYCLE:**
   After a t_D delay lower B_1 to V_{IL} for a duration of t_V. At the end of t_V, B_6 should indicate a level of V_{OH}; a level of V_{OL} indicates an unsuccessful fusing attempt.
4. **NEXT VARIABLE SELECT (I_m, D_s, X_R) (SAME TERM P_n):**
   After t_D delay, apply the next variable select (I, D, X) address to the input set and continue with step 2.
5. **NEXT VARIABLE SELECT (I_m, D_s, X_R) (DIFFERENT TERM (P_n):**
   After t_D, delay apply the next variable select (I, D, X) and term (P_n) addresses to the input set then continue to step 2.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

PROGRAM CYCLE ROW/COLUMN FUSE ADDRESSING

VARIABLE SELECT Table[1]

ROW HEX ADDRESS		SELECTED VARIABLE	
$I_6I_7B_0$	$B_2B_3B_4B_5$		
0	0	AND Array	$\frac{I_0}{I_0}$
0	1		
0	2		$\frac{I_1}{I_1}$
0	3		
0	4		$\frac{I_2}{I_2}$
0	5		
0	6		$\frac{I_3}{I_3}$
0	7		
0	8		$\frac{I_4}{I_4}$
0	9		
0	A		$\frac{I_5}{I_5}$
0	B		
0	C		$\frac{I_6}{I_6}$
0	D		
0	E		$\frac{I_7}{I_7}$
0	F		
1	0		$\frac{B_9}{B_9}$
1	1		
1	2		$\frac{B_8}{B_8}$
1	3		
1	4		$\frac{B_7}{B_7}$
1	5		
1	6		$\frac{B_6}{B_6}$
1	7		
1	8		$\frac{B_5}{B_5}$
1	9		
1	A		$\frac{B_4}{B_4}$
1	B		
1	C		$\frac{B_3}{B_3}$
1	D		
1	E		$\frac{B_2}{B_2}$
1	F		
2	0		$\frac{B_1}{B_1}$
2	1		
2	2		$\frac{B_0}{B_0}$
2	3		
2	4	Empty Address Space	
↓	↓		
2	F		

ROW HEX ADDRESS		SELECTED VARIABLE	
$I_6I_7B_0$	$B_2B_3B_4B_5$		
3	0	OR Array	9
3	1		8
3	2		7
3	3		6
3	4		5
3	5		4
3	6		3
3	7		2
3	8		1
3	9		0
3	A	Polarity Enable	
3	B	Empty Address Space	
↓	↓		
3	F		

TERM SELECT Table[2]

COLUMN HEX ADDRESS		SELECTED PRODUCT TERM	
I_0I_1	$I_2I_3I_4I_5$		
0	0	Logic Terms	0
0	1		1
0	2		2
0	3		3
0	4		4
0	5		5
0	6		6
0	7		7
0	8		8
0	9		9
0	A		10
0	B		11
0	C		12
0	D		13
0	E		14
0	F		15
1	0		16
1	1		17
1	2		18
1	3		19
1	4		20
1	5		21
1	6		22
1	7		23
1	8		24
1	9		25
1	A		26
1	B		27
1	C		28
1	D		29
1	E		30
1	F		31
2	0	Control Terms	0
2	1		1
2	2		2
2	3		3
2	4		4
2	5		5
2	6		6
2	7		7
2	8		8
2	9		9
2	A	Polarity Terms	0
2	B		1
2	C		2
2	D		3
2	E		4
2	F		5
3	0		6
3	1		1
3	2		8
3	3		9

NOTES
1. A row address identifies a particular variable coupled to all product terms.
2. With a variable selected by the row address the column address further selects a coupling fuse for each term.

FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

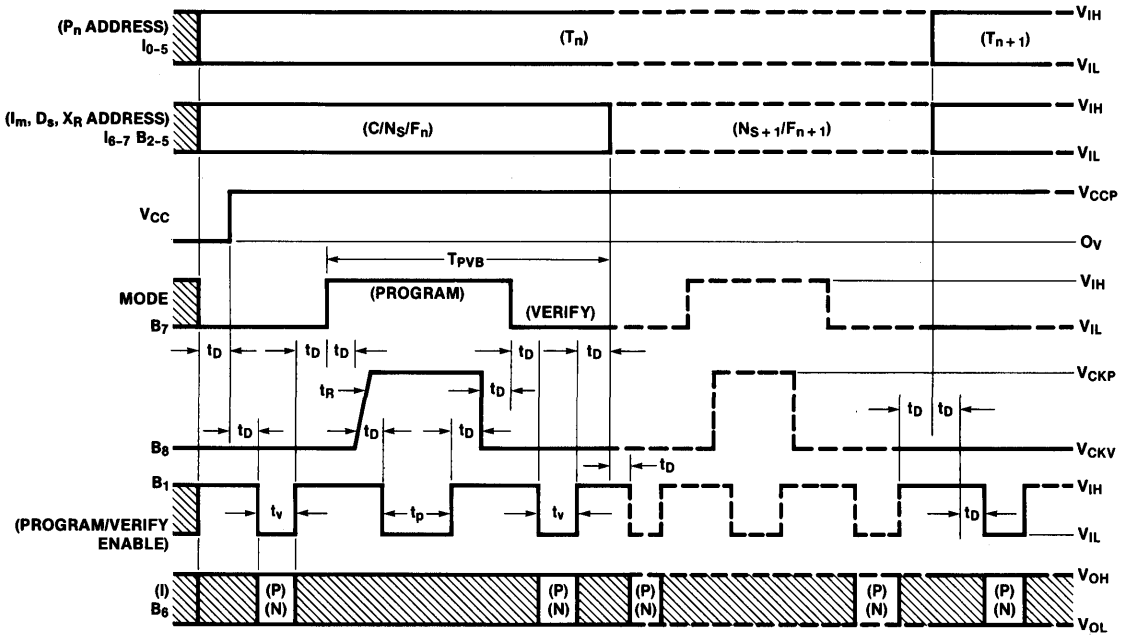
INTEGRATED FUSE LOGIC
SERIES 20

PROGRAMMING SYSTEM SPECIFICATIONS[1] (T_A = +25°C)

PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{CCP}	V_{CC} supply (program)	I_{CCP} = 550mA min	8.25	8.5	8.75	V
I_{CCP}	I_{CC} limit program	V_{CCP} = +8.50 ± .25V	550		1,000	mA
	Input voltage					V
V_{IH}	High		2.4		5.5	
V_{IL}	Low		0	0.4	0.8	
	Input Current					μA
I_{IH}	High	V_{IH} = +5.5V			50	
I_{IL}	Low	V_{IL} = 0V			−500	
V_{FSP}	Fuse supply (program)[3]	I_{FSP} = 300 ± 25mA Transient or steady state	16.0	17.0	18.0	V
V_{FSV}	Fuse supply (idle)	I_{FSV} = 1mA max	1.25	1.5	1.75	V
I_{FSP}	Fuse supply current limit	V_{FSP} = +17 ± 1V	275	300	325	mA
t_v	Verify time		1			μs
T_{RS}	Reset pulse width		1			μs
T_{PS}	Preset pulse width		1			μs
t_p	Programming pulse width		0.3	0.4	0.5	ms
t_D	Pulse sequence delay		10			μs
T_R	Fuse pulse rise time	10% to 90%	10		50	μs
T_{PVB}	Program-Verify time per link			0.6		ms
P_{DC}	Programming duty cycle				100	%
F_L	Fusing attempts per link				2	cycle
V_S	Verify threshold[4]		1.4	1.5	1.6	V

NOTES
1. These are specifications which a Programming System must satisfy in order to be qualified by Signetics.
2. Bypass V_{CC} to GND with a 0.01μf capacitor to reduce voltage spikes.
3. Care should be taken to ensure that the voltage is maintained during the entire fusing cycle. The recommended supply is a constant current source clamped at the specified voltage limit.
4. V_S is the sensing threshold of the FPLA output voltage for a programmed link. It normally constitutes the reference voltage applied to a comparator circuit to verify a successful fusing attempt.

ARRAY PROGRAM-VERIFY SEQUENCE (TYPICAL)



NOTE
1. (P) and (N) represent respectively a programmed and non-programmed fuse, corresponding to logic "1" or "0" output voltage levels.

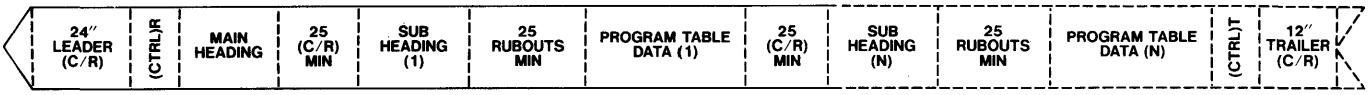
FIELD PROGRAMMABLE LOGIC ARRAY (18X32X10) 82S152 (O.C.)/82S153 (T.S.)

INTEGRATED FUSE LOGIC
SERIES 20

TWX TAPE CODING (LOGIC FORMAT)

The FPLA Program Table can be sent to Signetics in ASCII code format via airmail using any type of 8-level tape (paper, mylar, fanfold, etc.), or via TWX: just dial (910) 339-9283, tell the operator to turn the paper puncher on, and acknowledge. At the end of transmission instruct the operator to send tape to Signetics Order Entry.

A number of Program Tables can be sequentially assembled on a continuous tape as follows, however,limit tape length to a roll of 1.75 inch inside diameter and 4.25 inch outside diameter.



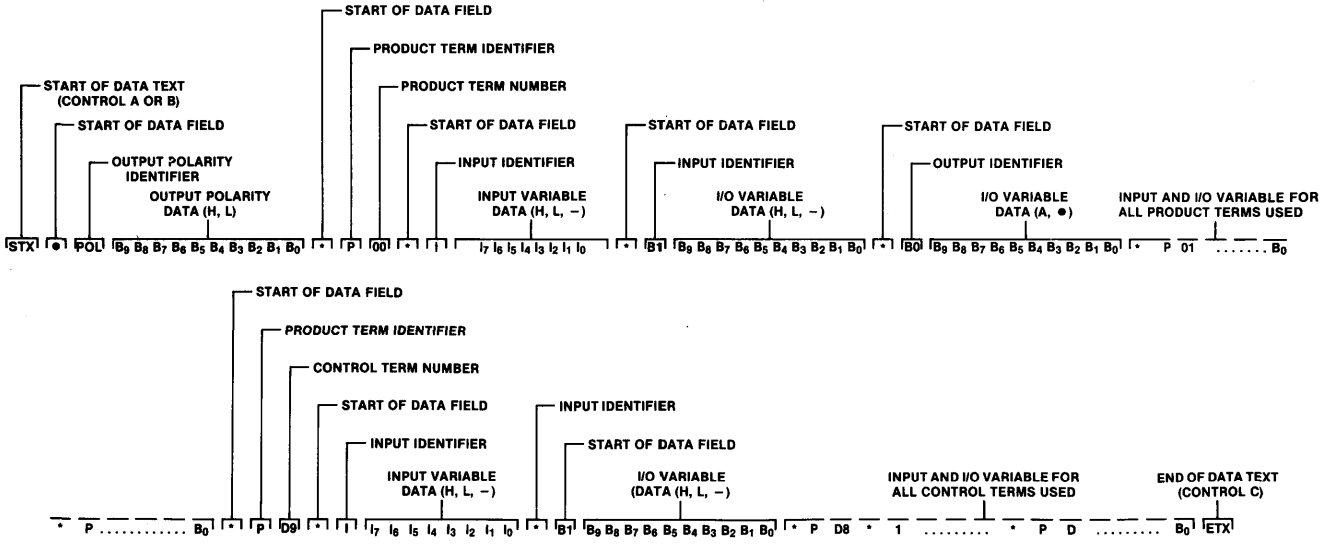
A. The MAIN HEADING at the beginning of tape includes the following information, with each entry preceded by a ($) character, whether used or not:

1. Customer Name _____
2. Customer TWX No. _____
3. Date _____
4. Purchase Order No. _____
5. Number of Program Tables _____
6. Total Number of Parts _____

B. Each SUB HEADING should contain specific information pertinent to each Program Table as follows, with each entry preceded by a ($) character, whether used or not:

1. Signetics Device No. _____
2. Program Table No. _____
3. Revision _____
4. Date _____
5. Customer Symbolized Part No. _____
6. Number of Parts _____

C. Program Table data blocks are initiated with an STX character, and terminated with an ETX character. The body of the data consists of output polarity, product term, and output information separated by appropriate identifiers in accordance with the following format:



Entries for the Data Field are determined in accordance with the following tables:

I, B (I)

Inactive	0
I, B	H
I, B	L
Don't Care	–

(AND)

B (0)

Active	A
Inactive	•

(OR)

B (0)

High	H
Low	L

(POL.)

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

DESCRIPTION

The 82S154/155/156/157/158/159 are Open Collector and Tri-state registered logic elements combining AND/OR gate arrays with clocked J/K flip-flops, optionally convertible to D-type via a "foldback" inverting buffer. They all have similar organization, featuring respectively 4, 6, or 8 registered I/O outputs (F), in conjunction with 8, 6, or 4 bidirectional I/O lines (B). These yield variable I/O gate and register configurations via control gates (D, L) ranging from 16 inputs to 12 outputs.

The AND/OR arrays consist of 32 AND gates and 21 OR gates with fusible link connections for programming I/O polarity and direction. All AND gates are linked to 4 inputs (I), bidirectional I/O lines (B), internal flip-flop outputs (Q), and Complement Array output ($\overline{C}$). The Complement Array consists of a NOR gate optionally linked to all AND gates for generating and propagating complementary AND terms.

On chip T/C buffers couple either True (I, B, Q) or Complement ($\overline{I}$, $\overline{B}$, $\overline{Q}$, $\overline{C}$) input polarities to all AND gates, whose outputs can be optionally linked to all OR gates. One group of OR gates drives bidirectional I/O lines (B), whose output polarity is individually programmable thru a set of EX-OR gates for implementing AND-OR or AND-NOR logic functions. Another group drives the J-K inputs of all flip-flops, as well as asynchronous Preset and Reset lines (P, R), (except the 82S158/159, where AND functions are provided).

All flip-flops are positive edge trigger and can be used as input, output, or I/O (for interfacing with a bidirectional data bus) in conjunction with load control gates (L), steering inputs (I), (B), (Q) and programmable output select lines (E).

The 82SXXX are field programmable, enabling the user to quickly generate custom patterns using standard programming equipment.

FLIP-FLOP TRUTH TABLE

V_{CC}	$\overline{O.E.}$	L	CK	P	R	J	K	Q	F
	H								H/HI-Z
↑	L	X	X	L	X	X	X	L	H
+5	L	X	X	H	L	X	X	H	L
	L	X	X	L	H	X	X	L	H
	L	L	↑	L	L	L	L	Q	$\overline{Q}$
	L	L	↑	L	L	L	H	L	H
	L	L	↑	L	L	H	L	H	L
	L	L	↑	L	L	H	H	$\overline{Q}$	Q
	H	H	↑	L	L	L	H	L	H*
	H	H	↑	L	L	H	L	H	L*
	+10V	X	↑	X	X	L	H	L	H**
		X	↑	X	X	H	L	H	L**

NOTES
1. Positive Logic:
   $J/K = T_0 + T_1 + T_2 \ldots\ldots\ldots T_{47}$
   $T_n = \overline{C} \bullet (I_0 \bullet I_1 \bullet I_2 \ldots) \bullet (Q_0 \bullet Q_1 \ldots) \bullet (B_0 \bullet B_1 \bullet \ldots)$
2. ↑ denotes transition from Low to High level.
3. X = Don't Care
4. * = Forced at F_n pin for loading J/K flip-flop in I/O mode. L must be enabled, and other active T_n disabled via steering input(s) I, B, or Q.
5. At P = R = H, Q = H. The final state of Q depends on which is released first.
6. * * = Forced at F_n pin to load J/K flip-flop independent of program code (Diagnostic mode).

All devices are available in a 20-pin, slim line package. For the commercial temperature range (0°C to +75°C) specify N82SXXX N or F. For the military temperature range (−55°C to +125°C) specify S82SXXX F only.

FEATURES

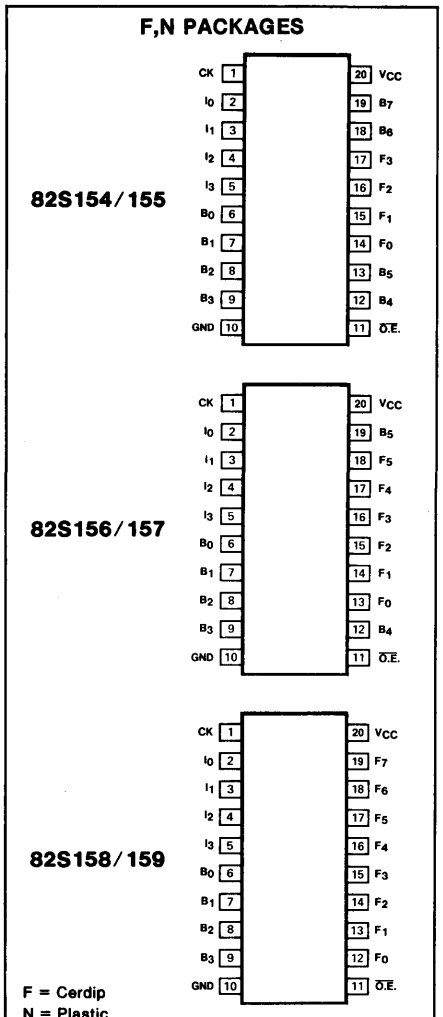
- **Field programmable (Ni-Cr link)**
- **4 Inputs**
- **32 AND gates**
- **21 OR gates**
- **Bidirectional I/O lines:** 82S154/155—8; 82S156/157—6; 82S158/159—4
- **Bidirectional Registers:** 82S154/155—4; 82S156/157—6; 82S158/159—8
- **J/K, T, or D-type flip-flops**
- **Asynchronous Preset/Reset**
- **Complement Array**
- **Active high or low outputs**
- **Programmable $\overline{O.E.}$ control**
- **Positive edge trigger clock**
- **Power-on reset of all flip-flops (F_n = "1")**
- **Clock frequency:** N82SXXX: 15 MHz (max); S82SXXX: MHz (max)
- **Input loading:** N82SXXX: −100μA (max); S82SXXX: −150μA (max)
- **Power dissipation: 650mW (typ)**
- **TTL Compatible**

APPLICATIONS

- **Random sequential logic**
- **Synchronous up/down counters**
- **Shift registers**
- **Bidirectional data buffers**
- **Timing function generators**
- **System controllers/synchronizers**
- **Priority encoder/registers**

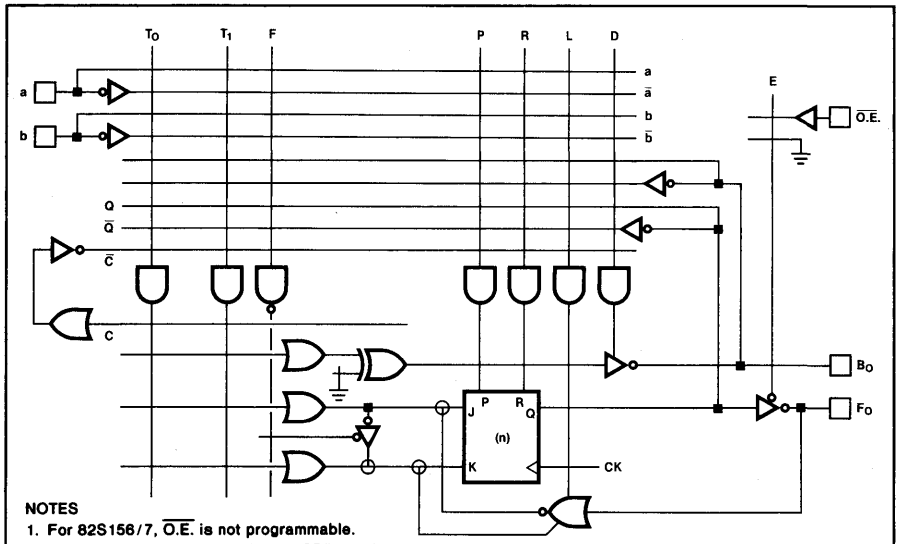
PIN CONFIGURATION

F,N PACKAGES



F = Cerdip
N = Plastic

FUNCTIONAL DIAGRAM



NOTES
1. For 82S156/7, $\overline{O.E.}$ is not programmable.
2. For 82S154/5/6/7, P and R are "OR" functions.

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

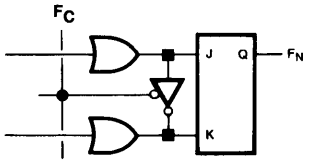
PIN DESIGNATION

PIN NO.	SYMBOL	NAME AND FUNCTION	POLARITY
1	CK	**CLOCK** The clock input to all flip-flops. A Low-to-High transition on this line is necessary to update the contents of flip-flops.	Active-High
2-5	I_{0-3}	**INPUTS** Fixed logic inputs to the AND array	Active-High/Low (user defined)
6-9	B_{0-3}	**STATIC I/O PINS** Bidirectional external inputs to the AND array, or outputs from the OR array, programmable via control gates D_{0-3}.	Active-High/Low (user defined)
11	$\overline{O.E.}$	**$\overline{\text{OUTPUT ENABLE}}$** Provides output enable functions E_A and E_B to flip-flop banks F_{0-3} and F_{4-7} respectively. May be programmed for external control, enable, or disable of flip-flop outputs.	Active-Low
12, 13	$(B_n/F_n)^*$	**STATIC OR REGISTERED I/O PINS** Bidirectional static or registered I/O pins, dependent on device configuration.	Active-High/Low (user defined)
14-17	$(F_n)^*$	**REGISTERED I/O PINS** Bidirectional flip-flop outputs or direct load inputs, selected via control gates L_{A-B} in conjunction with E_{A-B} output enables.	Active-Low
18, 19	$(B_n/F_n)^*$	**STATIC OR REGISTERED I/O PINS** Same as 12, 13.	Active-High/Low (user defined)

(*) Value of (n) is device dependent. Refer to circuit diagrams.

FLIP-FLOP FUNCTION

The output flip-flops (F_N) are programmable via the Flip-Flop Control Term (F_C). A tri-state inverter is positioned between the J-K inputs as shown. If (F_C) is programmed as active (A), then F_C controls the output state of the tri-state inverter.



When F_C = High, then flip-flop becomes a J-K type. When F_C = Low, the flip-flop becomes a D-type.

When the F_C is programmed as a • (i.e., disabled), the flip-flop is permanently defined as a J-K type.

LOGIC FUNCTION

COMBINATORIAL

Typical Product Term: $P_n = A \bullet \overline{B} \bullet \overline{C} \bullet D \bullet \overline{E} \bullet \ldots.$

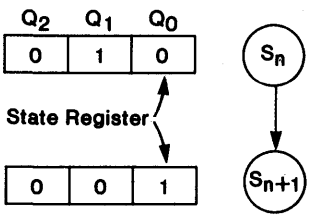
Typical Logic Function: At link (X) = Open at D_Y = "1": $Y = P_0 + P_1 + P_2 + \ldots.$

At link (X) = Closed: $Y = \overline{P_0 + P_1 + P_2 + \ldots.}$

$= \overline{P}_0 \bullet \overline{P}_1 \bullet \overline{P}_2 \bullet \ldots.$

SEQUENTIAL (J/K type)

Typical State Transition:



SET Q_0: $J_0 = (\overline{Q}_2 \bullet Q_1 \bullet \overline{Q}_0) \bullet \overline{A} \bullet \overline{B} \bullet C \ldots$
$K_0 = 0$

RESET Q_1: $J_1 = 0$
$K_1 = (\overline{Q}_2 \bullet Q_1 \bullet \overline{Q}_0) \bullet \overline{A} \bullet \overline{B} \bullet C \ldots$

HOLD Q_2: $J_2 = 0$
$K_2 = 0$

NOTES

1. For each of the combinatorial outputs, either function Y (active-high) or $\overline{Y}$ (active-low) is available, but not both. The desired output polarity is programmed via link (X).
2. Y, A, B, C, etc. are user defined connections to fixed inputs (I), bidirectional pins (B), "foldback" register outputs (Q), and Complement Array ($\overline{C}$).
3. Sequential state transitions occur on the positive edge of clock. External flip-flop outputs are given by $F_n = \overline{Q}_n$.
4. For D-type flip-flops, $K_n = \overline{J}_n$. For T-type flip-flops, $K_n = J_n$.

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

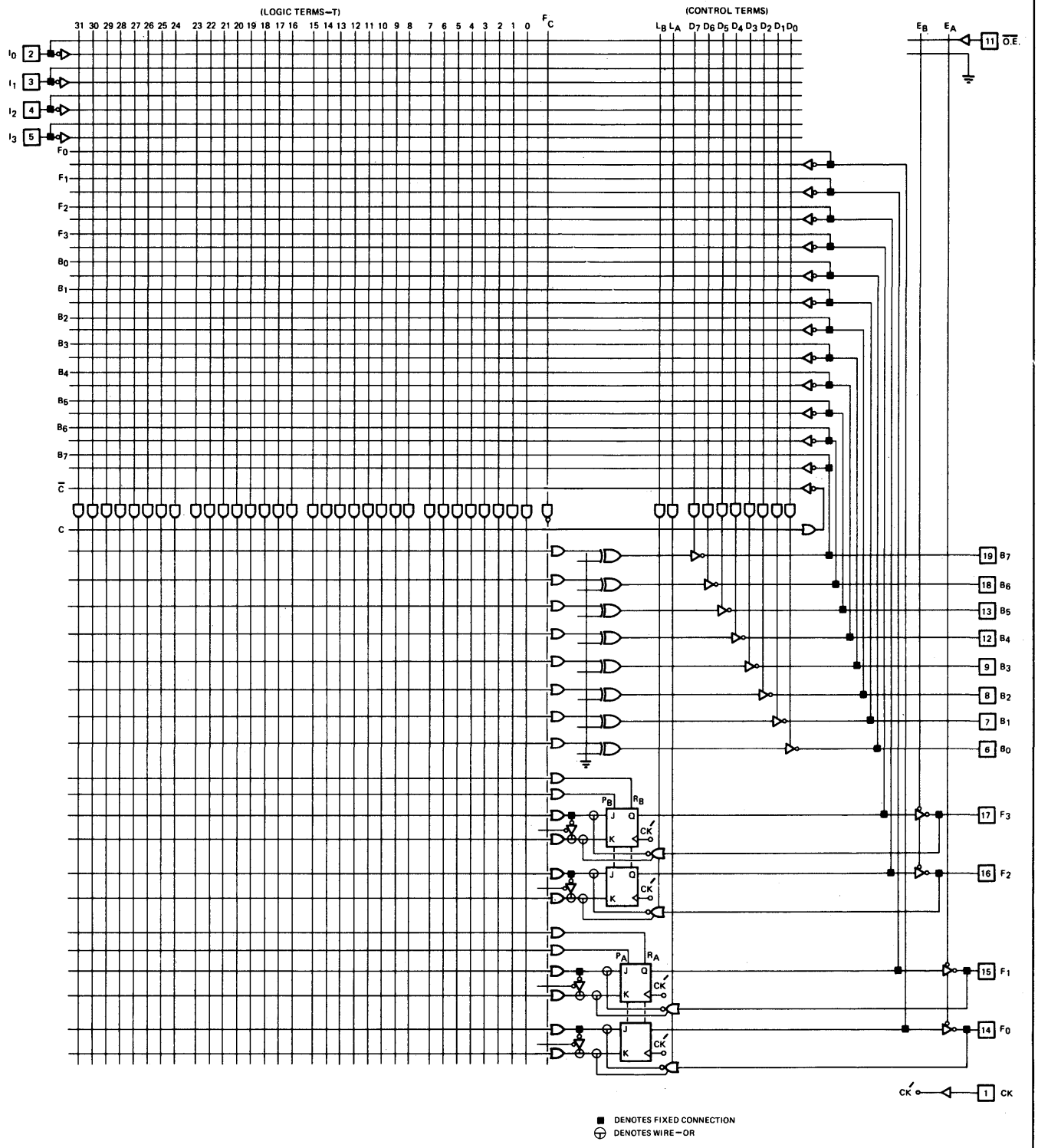
82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

FPLS LOGIC DIAGRAM

82S154/155



FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

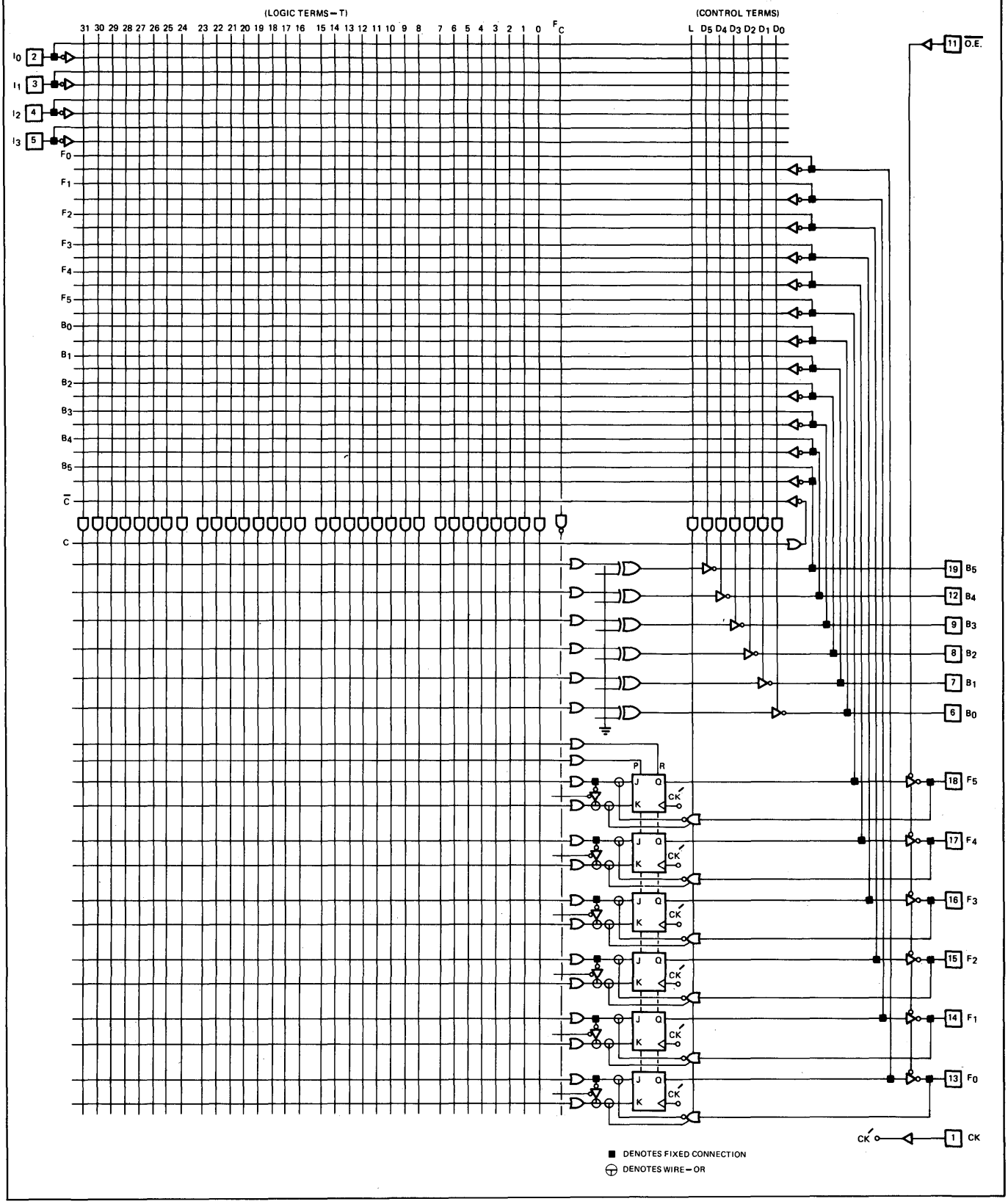
82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC SERIES 20

FPLS LOGIC DIAGRAM

82S156/157



FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

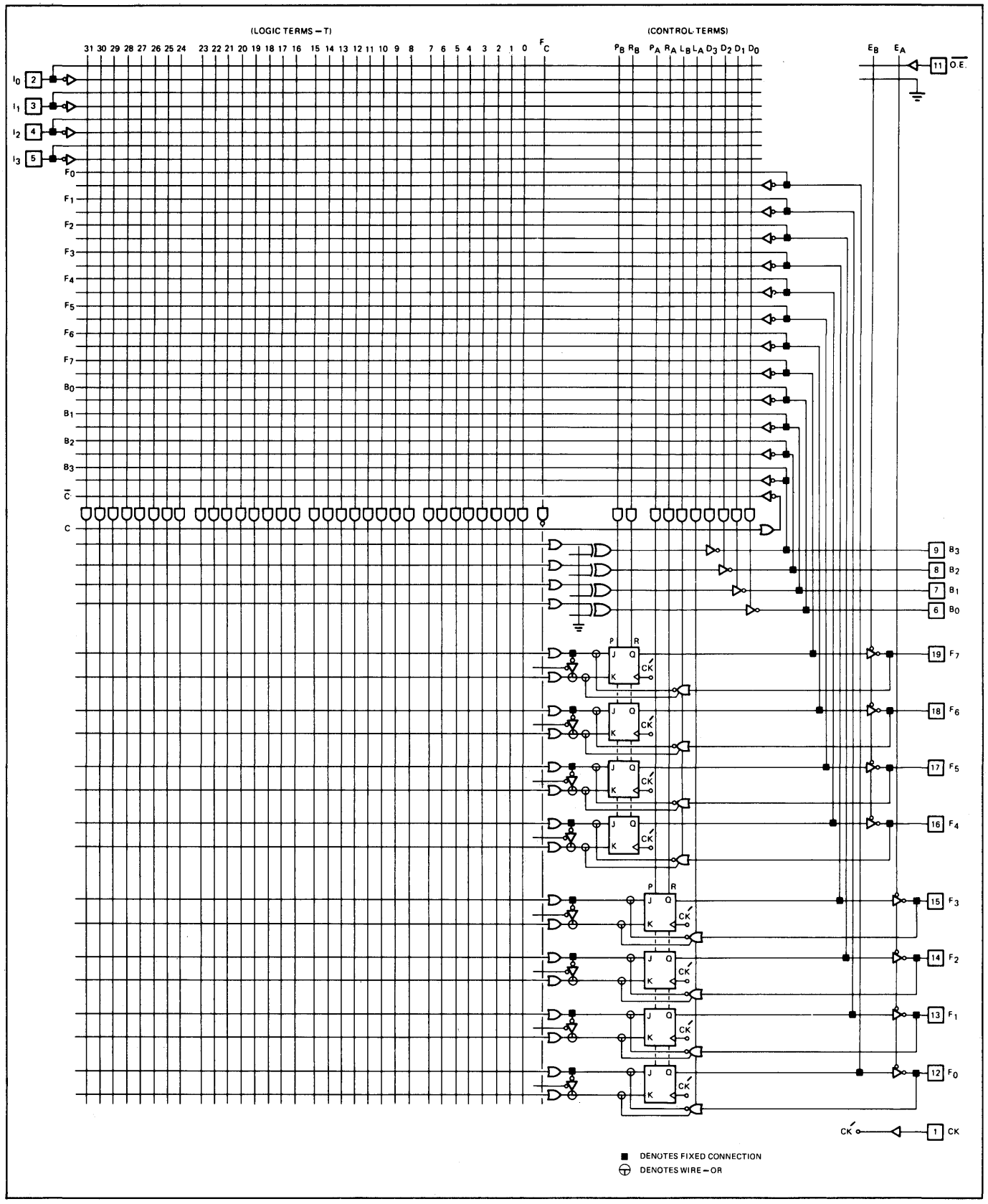
82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

FPLS LOGIC DIAGRAM

82S158/159



FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING Min	RATING Max	UNIT
V_{CC}	Supply voltage		+7	Vdc
V_{IN}	Input voltage		+5.5	Vdc
V_{OUT}	Output voltage		+5.5	Vdc
I_{IN}	Input currents	−30	+30	mA
I_{OUT}	Output currents		+100	mA
	Temperature range			°C
T_A	Operating			
	N82S154/5/6/7/8/9	0	+75	
	S82S154/5/6/7/8/9	−55	+125	
T_{STG}	Storage	−65	+150	

THERMAL RATINGS

TEMPERATURE	MILITARY	COMMERCIAL
Maximum junction	175°C	150°C
Maximum ambient	125°C	75°C
Allowable thermal rise ambient to junction	50°C	75°C

DC ELECTRICAL CHARACTERISTICS

N82S154/5/6/7/8/9: $0° \le T_A \le +75°C$, $4.75V \le V_{CC} \le 5.25V$
S82S154/5/6/7/8/9: $-55°C \le T_A \le +125°C$, $4.5V \le V_{CC} \le 5.5V$

PARAMETER		TEST CONDITIONS	N82S154/5/6/7/8/9 Min	N82S154/5/6/7/8/9 Typ[2]	N82S154/5/6/7/8/9 Max	S82S154/5/6/7/8/9 Min	S82S154/5/6/7/8/9 Typ[2]	S82S154/5/6/7/8/9 Max	UNIT
	Input voltage[3]								V
V_{IH}	High	V_{CC} = Max	2			2			
V_{IL}	Low	V_{CC} = Min			0.85			0.8	
V_{IC}	Clamp[3,4]	V_{CC} = Min, I_{IN} = −18mA		−0.8	−1.2		−0.8	−1.2	
	Output voltage	V_{CC} = Min							V
V_{OH}	High (82S155/7/9)[3,5]	I_{OH} = −2mA	2.4			2.4			
V_{OL}	Low[3,6]	I_{OL} = 10mA		0.35	0.5				
V_{OL}	Low[3,6]	I_{OL} = 10mA					0.35	0.5	
	Input current								μA
I_{IH}	High	V_{IN} = 5.5V		<1	40		<1	50	
I_{IL}	Low	V_{IN} = 0.45V		−10	−100		−10	−150	
I_{IL}	Low (CK input)	V_{IN} = 0.45V		−50	−250		−50	−350	
	Output current	V_{CC} = Max							
I_{OLK}	Leakage[7]	V_{OUT} = 5.5V		1	40		1	60	μA
$I_{O(OFF)}$	Hi-Z state (82S155/7/9)[7]	V_{OUT} = 5.5V		1	40		1	60	μA
		V_{OUT} = 0.45V		−1	−40		−1	−60	
I_{OS}	Short circuit (82S155/7/9)[4,6,7]	V_{OUT} = 0V	−20		−70	−15		−85	mA
I_{CC}	V_{CC} supply current[9]	V_{CC} = Max		130	155		130	155	mA
	Capacitance[7]	V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		8			8		
C_{OUT}	Output	V_{OUT} = 2.0V		15			15		

NOTES

1. Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device of these or any other condition above those indicated in the operation of the device specifications is not implied.
2. All typical values are at V_{CC} = 5V, T_A = 25°C.
3. All voltage values are with respect to network ground terminal.
4. Test one at a time.
5.
6.
7. Measured with V_{IH} applied to $\overline{O.E.}$
8. Duration of short circuit should not exceed 1 second.
9. I_{CC} is measured with the $\overline{O.E.}$ input grounded, all other inputs at 4.5V and the outputs open.

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

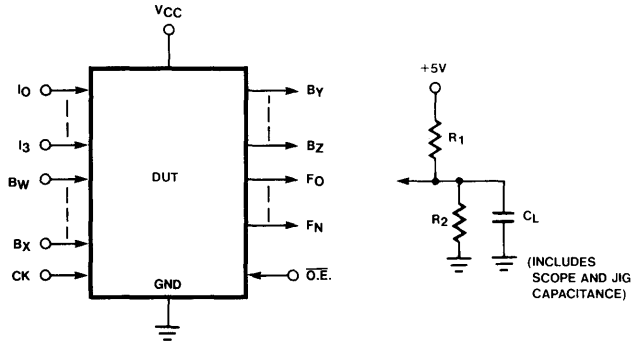
AC ELECTRICAL CHARACTERISTICS

$R_1 = 470\Omega$, $R_2 = 1k\Omega$
N82S154/5/6/7/8/9: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$
S82S154/5/6/7/8/9: $-55°C \leq T_A \leq +125°C$, $4.5V \leq V_{CC} \leq 5.5V$

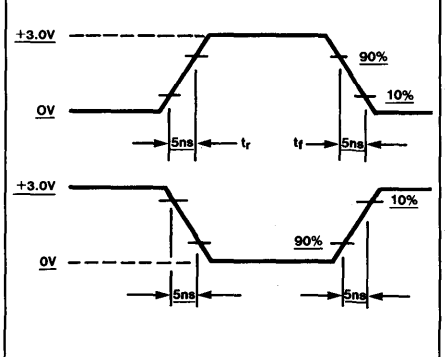
	PARAMETER	TO	FROM	TEST CONDITIONS	N82S154/5/6/7/8/9/ Min	Typ[1]	Max	S82S154/5/6/7/8/9 Min	Typ[1]	Max	UNIT
	Pulse width										ns
T_{CKH}	Clock[2] high	CK−	CK+		25	20			20		
T_{CKL}	Clock low	CK+	CK−		25	20			20		
T_{CKP}	Period	CK+	CK+		65	50			50		
T_{PRH}	Preset/Reset pulse	(I,B)+	(I,B)−		25	20			20		
	Set up time										ns
T_{IS1}	Input	CK+	(I,B)±		35	30			30		
T_{IS2}	Input (through F_n)	CK+	F±	C_L = 30pF	10	5			5		
T_{IS3}	Input (through Complement array)[4]	CK+	(I,B)±		45	40			40		
	Hold time										ns
T_{IH1}	Input	CK+	(I,B)±			−10	0		−10		
T_{IH2}		CK+	F±			−5	0		−5		
	Propagation delay										ns
T_{CKO}	Clock	F±	CK+			25	30		25		
T_{OE1}	Output enable	F−	$\overline{O.E.}$−			20	25		20		
T_{OD1}	Output disable[3]	F+	$\overline{O.E.}$+	C_L = 5pF		20	25		20		
T_{PD}	Output	B±	(I,B)±	C_L = 30pF		35	40		35		
T_{OE2}	Output enable	B±	(I,B)+			35	40		35		
T_{OD2}	Output disable[3]	B+	(I,B)−	C_L = 5pF		35	40		35		
T_{PRO}	Preset/Reset	F±	(I,B)+			50	65		50		
T_{PPR}	Power-on preset	F−	V_{CC}+	C_L = 30pF		0	10		0		

NOTE
1. All typical values are at V_{CC} = 5V, T_A = 25°C.
2. To prevent spurious clocking, clock rise time (10%-90%) ≤ 10ns.
3. Measured at $V_T = V_{OL}$ + 0.5 V.
4. When using the Complement Array T_{CKP} = 75ns (min).

TEST LOAD CIRCUIT



VOLTAGE WAVEFORM



Measurements: All circuit delays are measured at the +1.5V level of inputs and outputs, unless otherwise specified.

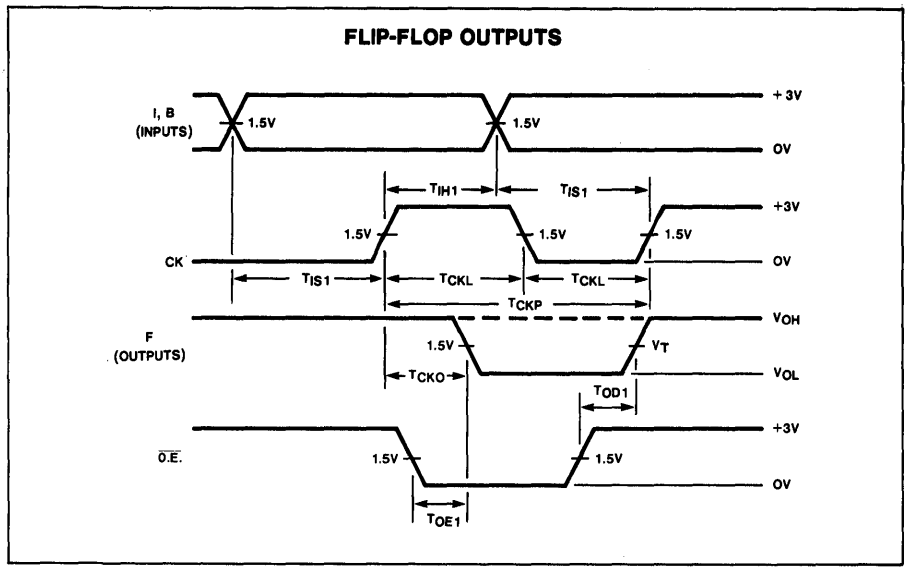
FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

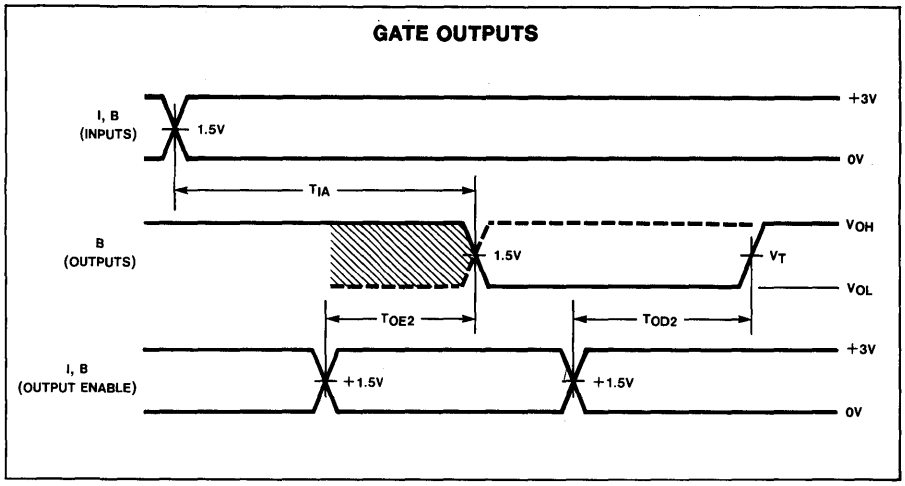
82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

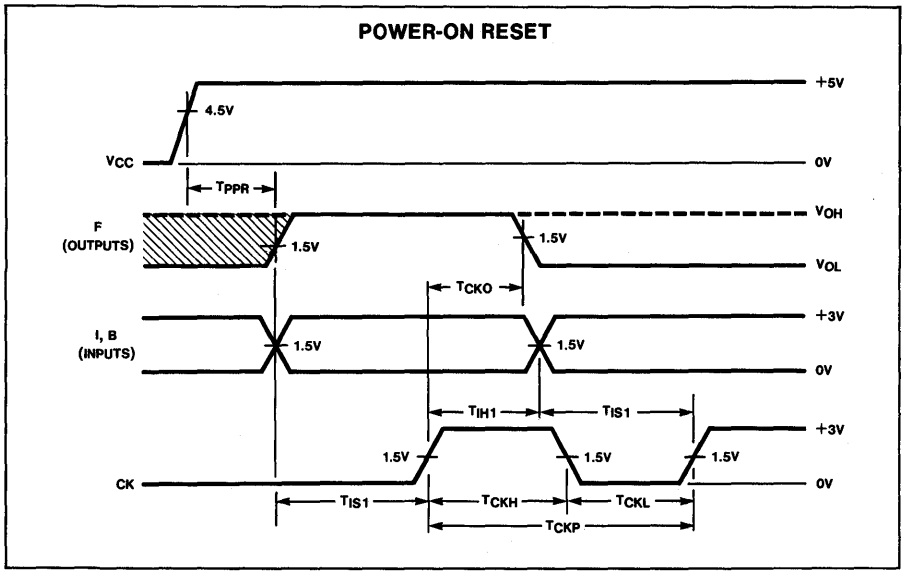
Preview

INTEGRATED FUSE LOGIC
SERIES 20

TIMING DIAGRAMS







MEMORY TIMING DEFINITIONS

T_{CKH} Width of input clock pulse.

T_{CKL} Interval between clock pulses.

T_{CKP} Clock period.

T_{PRH} Width of preset input pulse.

T_{IS1} Required delay between beginning of valid input and positive transition of clock.

T_{IS2} Required delay between beginning of valid input forced at flip-flop output pins, and positive transition of clock.

T_{IH1} Required delay between positive transition of clock and end of valid input data.

T_{IH2} Required delay between positive transition of clock and end of valid input data forced at flip-flop output pins.

T_{CKO} Delay between positive transition of clock and when Outputs become valid (with $\overline{O.E.}$ low).

T_{OE1} Delay between beginning of Output Enable Low and when Outputs become valid.

T_{OD1} Delay between beginning of Output Enable High and when Outputs are in the off state.

T_{PD} Propagation delay between combinational inputs and outputs.

T_{OE2} Delay between predefined Output Enable High, and when combinational Outputs become valid.

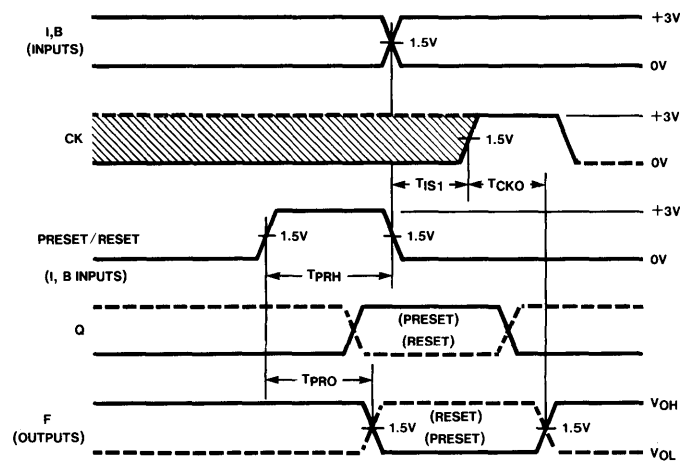
T_{OD2} Delay between predefined Output Enable Low and when combinational Outputs are in the off state.

T_{PRO} Delay between positive transition of predefined Preset/Reset input, and when flip-flop outputs become valid.

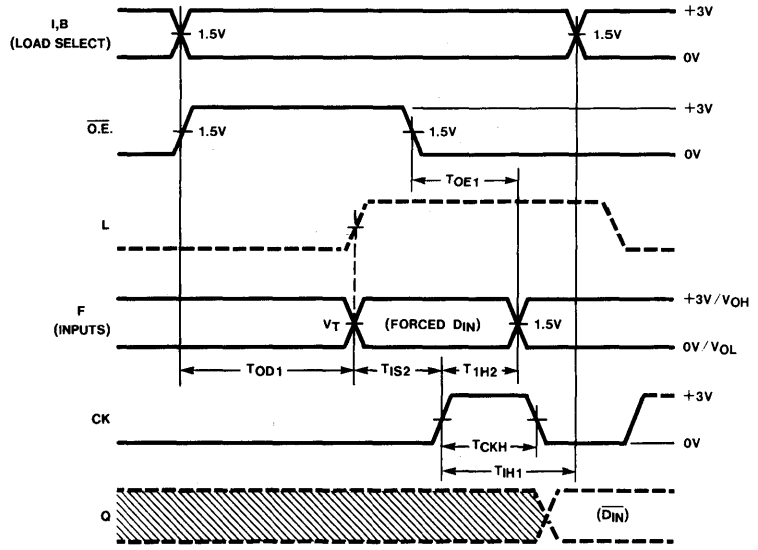
T_{PPR} Delay between V_{CC} (after power-on) and when flip-flop outputs become preset at "1" (internal Q outputs at "0").

TIMING DIAGRAMS (Cont'd)

ASYNCHRONOUS PRESET/RESET



FLIP-FLOP INPUT MODE



FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

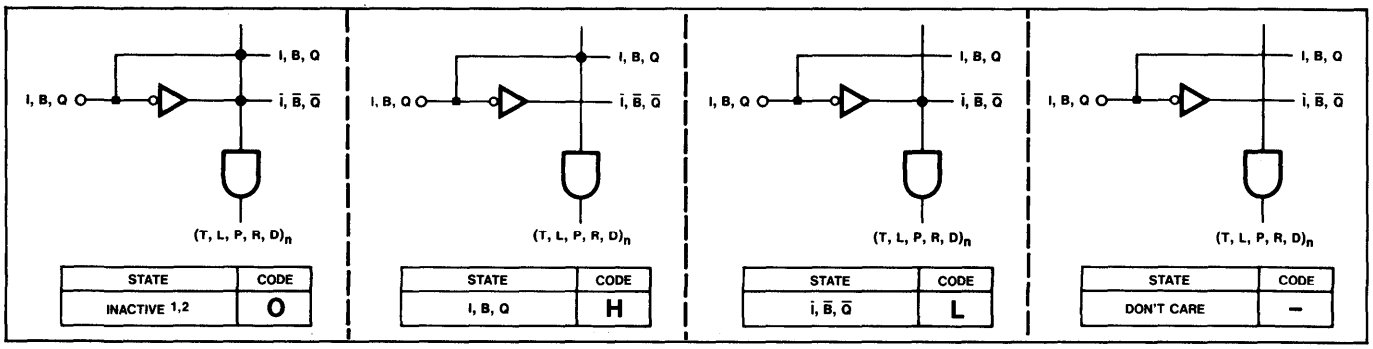
LOGIC PROGRAMMING

The FPLS can be programmed by means of Logic Programming equipment.

With Logic programming, the AND/OR/EX-OR input connections necessary to implement the desired logic function are coded directly from the State Diagram using the Program Tables on the following pages.

In this Table, the logic state or action of all I/O, control, and state variables is assigned a symbol which results in the proper fusing pattern of corresponding links defined as follows:

"AND" ARRAY = (I), (B), (Qp)



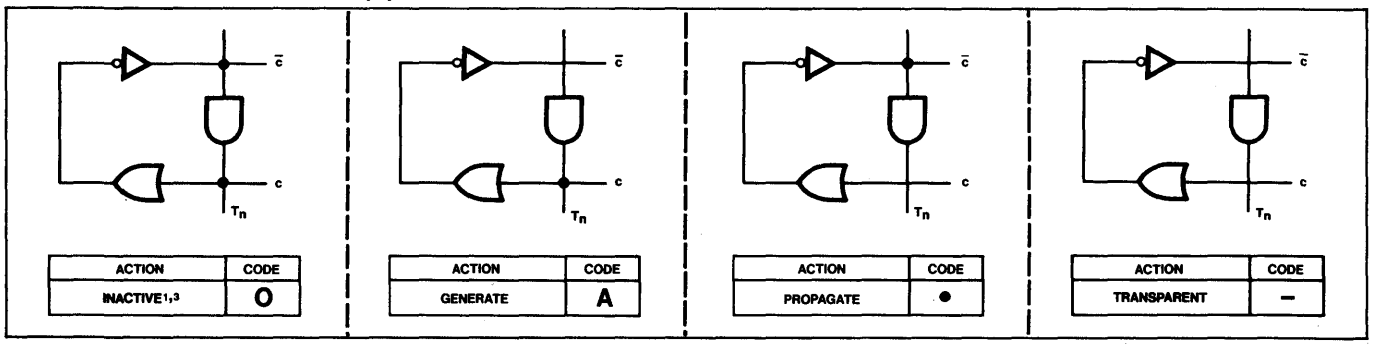
STATE	CODE
INACTIVE [1,2]	O

STATE	CODE
I, B, Q	H

STATE	CODE
$\bar{I}, \bar{B}, \bar{Q}$	L

STATE	CODE
DON'T CARE	–

"COMPLEMENT" ARRAY = (C)



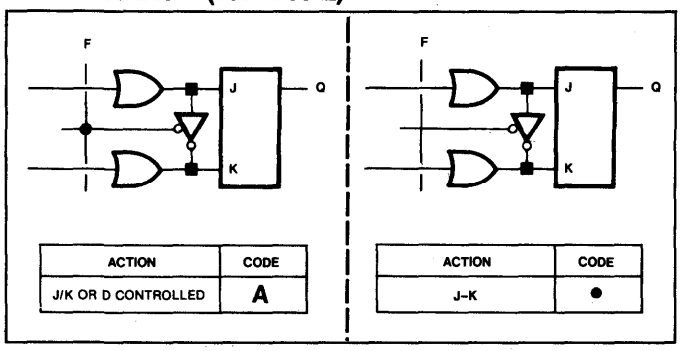
ACTION	CODE
INACTIVE[1,3]	O

ACTION	CODE
GENERATE	A

ACTION	CODE
PROPAGATE	•

ACTION	CODE
TRANSPARENT	–

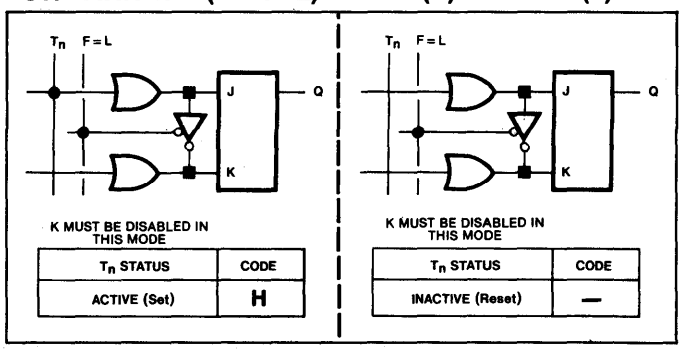
"OR" ARRAY = (F/F TYPE)



ACTION	CODE
J/K OR D CONTROLLED	A

ACTION	CODE
J–K	•

"OR" ARRAY = (D TYPE) F/F = (A) AND F = (L)



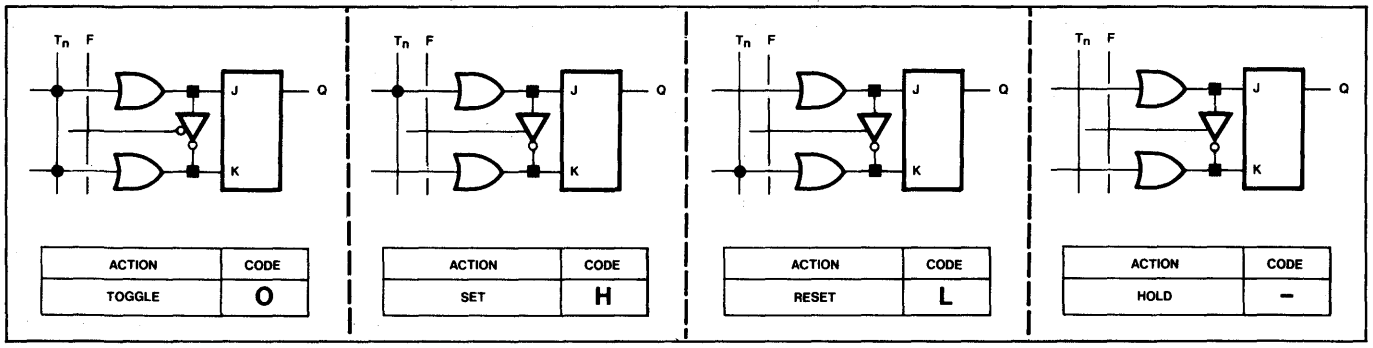
K MUST BE DISABLED IN THIS MODE

T_n STATUS	CODE
ACTIVE (Set)	H

K MUST BE DISABLED IN THIS MODE

T_n STATUS	CODE
INACTIVE (Reset)	—

"OR" ARRAY = (J/K TYPE) F/F = (A) AND F = (H) OR F = (•) AND F = (H) OR (L)



ACTION	CODE
TOGGLE	O

ACTION	CODE
SET	H

ACTION	CODE
RESET	L

ACTION	CODE
HOLD	–

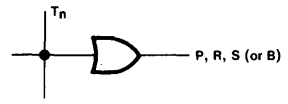
FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

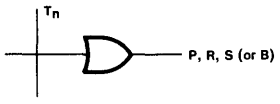
Preview

INTEGRATED FUSE LOGIC
SERIES 20

"OR" ARRAY - (S or B), (P), (R)

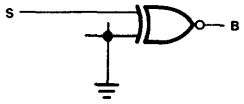


T_n STATUS	CODE
ACTIVE	A

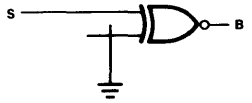


T_n STATUS	CODE
INACTIVE	•

"EX-OR" ARRAY - (B)

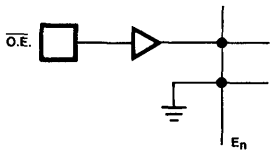


POLARITY	CODE
LOW	L

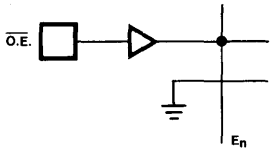


POLARITY	CODE
HIGH	H

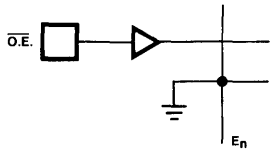
"$\overline{O.E.}$" ARRAY - (E)



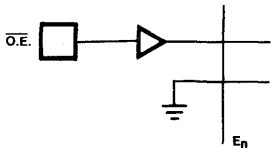
ACTION	CODE
IDLE[4]	O



ACTION	CODE
CONTROL	A



ACTION	CODE
ENABLE[4]	•



ACTION	CODE
DISABLE	–

NOTES

1. This is the initial unprogrammed state of all link pairs. It is normally associated with all unused (inactive) AND gates T_n.
2. Any gate $(T, L, P, R, D)_n$ will be unconditionally inhibited if any one of the I, B, or Q link pairs is left intact.
3. To prevent oscillations, this state is not allowed for C link pairs coupled to active gates T_n.
4. Although links in the $\overline{O.E.}$ array are isolated from each other, E_n = 0 and E_n = • are logically equivalent. But if register bank A is enabled with E_A = 0, then E_B can still be controlled (and vice versa).

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

82S154/155

FPLS PROGRAM TABLE (Logic)

PROGRAM TABLE ENTRIES:

I, B(I), Q(P) (AND)

INACTIVE	O
I, B, Q	H
I, $\overline{B}$, $\overline{Q}$	L
DON'T CARE	–

C (AND)

INACTIVE	O
GENERATE	A
PROPAGATE	●
TRANSPARENT	–

(Q = J/K) (OR)

TOGGLE	O
SET	H
RESET	L
HOLD	–

E_A, E_B

IDLE	O
CONTROL	A
ENABLE	●
DISABLE	–

P, R, B(O), (Q = D) (OR)

ACTIVE	A
INACTIVE	●

B(O) (POL.)

HIGH	H
LOW	L

F/F TYPE

J/a	●
J/K or C (controlled)	a

NOTES
1. The FPLS is shipped with all links intact. Thus a background of entries corresponding to states of virgin links exists in the table, shown BLANK for clarity.
2. Program unused C, I, B, and Q bits in the AND array as (–). Program unused Q, B, P, and R bits in the OR array as (–) or (A), as applicable.
3. Unused Terms can be left blank.
4. Q (P) and Q (N) are respectively the present and next states of flip-flops Q.

(F/F TYPE) | E_B = | E_A = | (POLARITY)

THIS PORTION TO BE COMPLETED BY SIGNETICS
CF (XXXX) _____
CUSTOMER SYMBOLIZED PART # _____
DATE RECEIVED _____
COMMENTS _____

TERM	AND C	I 3	I 2	I 1	I 0	B(I) 7	B(I) 6	B(I) 5	B(I) 4	B(I) 3	B(I) 2	B(I) 1	B(I) 0	Q(P) 3	Q(P) 2	Q(P) 1	Q(P) 0	OR Q(N) 3	Q(N) 2	Q(N) 1	Q(N) 0	P B	P A	R B	R A	B(O) 7	B(O) 6	B(O) 5	B(O) 4	B(O) 3	B(O) 2	B(O) 1	B(O) 0
0																																
1																																
2																																
3																																
4																																
5																																
6																																
7																																
8																																
9																																
10																																
11																																
12																																
13																																
14																																
15																																
16																																
17																																
18																																
19																																
20																																
21																																
22																																
23																																
24																																
25																																
26																																
27																																
28																																
29																																
30																																
31																																
F_C																																
L_B																																
L_A																																
D7																																
D6																																
D5																																
D4																																
D3																																
D2																																
D1																																
D0																																
PIN		5	4	3	2	19	18	13	12	9	8	7	6	17	16	15	14															

CUSTOMER NAME _____
PURCHASE ORDER # _____
SIGNETICS DEVICE # _____
TOTAL NUMBER OF PARTS _____
PROGRAM TABLE # _____ REV ____ DATE ____

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

FPLS PROGRAM TABLE (Logic)

82S156/157

PROGRAM TABLE ENTRIES:

C (AND)

INACTIVE	O
GENERATE	A
PROPAGATE	●
TRANSPARENT	–

(Q = J/K) (OR)

TOGGLE	O
SET	H
RESET	L
HOLD	–

I, B(I), Q(P) (AND)

INACTIVE	O
I, B, Q	H
$\overline{I}$, $\overline{B}$, $\overline{Q}$	L
DON'T CARE	–

P, R, B(O), (Q = D) (OR)

ACTIVE	A
INACTIVE	●

B(O) (POL.)

HIGH	H
LOW	L

F/F TYPE

J/a	•
J/K or C (controlled)	a

NOTES
1. The FPLS is shipped with all links intact. Thus a background of entries corresponding to states of virgin links exists in the table, shown BLANK for clarity.
2. Program unused C, I, B, and Q bits in the AND array as (–). Program unused Q, B, P, and R bits in the OR array as (–) or (A), as applicable.
3. Unused Terms can be left blank.
4. Q (P) and Q (N) are respectively the present and next states of flip-flops Q.

THIS PORTION TO BE COMPLETED BY SIGNETICS

CF (XXXX) _____

CUSTOMER SYMBOLIZED PART # _____

DATE RECEIVED _____

COMMENTS _____

CUSTOMER NAME _____

PURCHASE ORDER # _____

SIGNETICS DEVICE # _____

TOTAL NUMBER OF PARTS _____

PROGRAM TABLE # _____ REV ____ DATE ____

(F/F TYPE) (POLARITY)

TERM	AND C	AND I 3	AND I 2	AND I 1	AND I 0	AND B(I) 5	AND B(I) 4	AND B(I) 3	AND B(I) 2	AND B(I) 1	AND B(I) 0	AND Q (P) 5	AND Q (P) 4	AND Q (P) 3	AND Q (P) 2	AND Q (P) 1	AND Q (P) 0	OR Q (N) 5	OR Q (N) 4	OR Q (N) 3	OR Q (N) 2	OR Q (N) 1	OR Q (N) 0	OR P	OR R	OR B(O) 5	OR B(O) 4	OR B(O) 3	OR B(O) 2	OR B(O) 1	OR B(O) 0
0																														
1																														
2																														
3																														
4																														
5																														
6																														
7																														
8																														
9																														
10																														
11																														
12																														
13																														
14																														
15																														
16																														
17																														
18																														
19																														
20																														
21																														
22																														
23																														
24																														
25																														
26																														
27																														
28																														
29																														
30																														
31																														
F_C																														
L																														
D5																														
D4																														
D3																														
D2																														
D1																														
D0																														
PIN		5	4	3	2	19	12	9	8	7	6	18	17	16	15	14	13														

FIELD PROGRAMMABLE LOGIC SEQUENCER (16X32X12)

82S154/6/8 (O.C.)/82S155/7/9 (T.S.)

Preview

INTEGRATED FUSE LOGIC
SERIES 20

FPLS PROGRAM TABLE (Logic)

PROGRAM TABLE ENTRIES:

I, B(I), Q(P) (AND)	
INACTIVE	O
I, B, Q	H
$\overline{I}$, $\overline{B}$, $\overline{Q}$	L
DON'T CARE	–

C (AND)	
INACTIVE	O
GENERATE	A
PROPAGATE	●
TRANSPARENT	–

(Q = J/K) (OR)	
TOGGLE	O
SET	H
RESET	L
HOLD	–

$\overline{O.E.}$	
IDLE	O
CONTROL	A
ENABLE	●
DISABLE	–

F/F TYPE	
J/a	•
J/K or C (controlled)	a

P, R, B(O), (Q = D) (OR)	
ACTIVE	A
INACTIVE	●

B(O) (POL.)	
HIGH	H
LOW	L

NOTES
1. The FPLS is shipped with all links intact. Thus a background of entries corresponding to states of virgin links exists in the table, shown BLANK for clarity.
2. Program unused C, I, B, and Q bits in the AND array as (–). Program unused Q, B, P, and R bits in the OR array as (–) or (A), as applicable.
3. Unused Terms can be left blank.
4. Q (P) and Q (N) are respectively the present and next states of flip-flops Q.

THIS PORTION TO BE COMPLETED BY SIGNETICS

CF (XXXX) _____ CUSTOMER SYMBOLIZED PART # _____

DATE RECEIVED _____

COMMENTS _____

CUSTOMER NAME _____

PURCHASE ORDER # _____

SIGNETICS DEVICE # _____ REV ____ DATE ____

TOTAL NUMBER OF PARTS _____

PROGRAM TABLE # _____

TERM	(F/F TYPE)	E_B =	E_A =	(POLARITY)	AND																(OR)											
					C	I				B(I)				Q (P)							Q (N)								B(O)			
						3	2	1	0	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	3	2	1	0
0																																	
1																																	
2																																	
3																																	
4																																	
5																																	
6																																	
7																																	
8																																	
9																																	
0																																	
11																																	
12																																	
13																																	
14																																	
15																																	
16																																	
17																																	
18																																	
19																																	
20																																	
21																																	
22																																	
23																																	
24																																	
25																																	
26																																	
27																																	
28																																	
29																																	
30																																	
31																																	
F_C																																	
P_B																																	
R_B																																	
L_B																																	
P_A																																	
R_A																																	
L_A																																	
D3																																	
D2																																	
D1																																	
D0																																	
PIN		5	4	3	2	9	8	7	6	19	18	17	16	15	14	13	12																

Field-programmable arrays: powerful alternatives to random logic

Bridging design gap, TTL-compatible logic family is described in Part 1 of a two-part article

by Napoleone Cavlan and Stephen J. Durham, *Signetics Corp., Sunnyvale, Calif.*

☐ With the steady growth of integrated circuit technologies, hardly a day goes by without the news that yet another chip has made scores of discrete TTL packages obsolete. Yet, though large-scale integration is packing entire system architectures onto a few chips, it is still impossible to complete a design without some discrete logic to hold the framework together.

The increase of LSI has thus created the need for efficient ways to bridge the gaps between large functional islands. Because of complexity, performance, or uniqueness, these bridges have evolved into nontrivial random-logic configurations that still rely on clusters of small- and medium-scale integrated circuits, whose fixed functions never quite fit the problem. Now Signetics Corp. has attempted to meet the need with a field-programmable logic family.

The family spans three ranges of complexity: at the low end are the field-programmable gate arrays (FPGAs); covering the middle range are the more complex logic arrays (FPLAs); and finally there are the logic sequencers (FPLSs). These last, most complex elements have built-in registers and enable the designer to proceed from state diagram directly to hardware. The family, summed up in the table on p. 110, is compatible with TTL and operates from a +5-volt supply.

The devices provide a powerful and compact alternative to random logic, replacing discrete gates, wires, and connectors, with significant savings in board space,

FIELD-PROGRAMMABLE LOGIC FAMILY

Device	Organization	Device	Inputs	Outputs[1]	Chip enable ($\overline{CE}$)	I_{CC} (max)	Delay (max)	Availability	Package[2]
FPGA	• AND/NAND	82S102 82S103	16	9 OC 9 TS	yes	170 mA	35 ns	now	N
FPLA	• AND-OR/NOR	82S100 82S101		8 TS 8 OC	yes		50 ns	now	N, F
FPLA	• AND-OR • Self-enable output	82S106 82S107		8 OC 8 TS	no		70 ns	4Q79	N
FPLS	• AND-OR • Complement array • 6-bit state register • 8-bit output register	82S104 82S105		8 OC 8 TS	yes[3]		90 ns	4Q79	N, F

[1] OC = open collector TS = three-state
[2] N = plastic F = Cerdip
[3] $\overline{CE}$ input may be optionally programmed as preset.

power, and cost. Moreover, since all devices can be programmed and modified in the field (as programmable read-only memories can) using readily available programming equipment, the logic can be changed to meet new customer requirements or specifications, or to recover quickly from design errors—after delivery to the field—without expensive printed-circuit-board retooling.

Programming options

Depending on their complexity, members of the programmable logic family have internal AND gates, OR gates, S-R flip-flops, true or complement buffers, and exclusive-OR (EXOR) gates. Those elements can be combined to perform single-level, double-level, and sequential logic functions—all by blowing fuse links.

There are other fuse options in output structures for the entire logic family, too, since either active-high or active-low functions can be generated without additional hardware or signal delay. Finally, the family is well suited to bus-organized environments such as microprocessor systems, since all its members offer, in addition to open-collector outputs, three-state outputs whose signals are in the high-impedance state until activated by a chip-enable input.

All the logic elements perform standard logic functions that can be represented by augmenting conventional logic symbols with a few new definitions so that they can represent multiple-input gates (see "How the FPLF defines logic," p. 111).

The gate arrays

The simplest member in the family is the field-programmable gate array, which performs single-level logic functions. The equivalent logic diagram for the FPGA is shown in Fig. 1. The two gate arrays currently available are open-collector (82S102) and three-state output (82S103) versions of the same array, which comprise nine NAND gates fuse-selectively connected to 16 common inputs by true/complement buffers.

Fuses in the FPGAs allow individual outputs to be complemented to AND, so that by proper manipulation of the input polarities, and by using De Morgan's theorem, AND, OR, NAND, and NOR logic functions can be easily implemented. The parts thus serve as universal logic elements that can be tailored to applications requiring random logic, as in fault monitors, code detectors, and address decoders for microcomputer systems with memory-mapped I/O.

The logic arrays

Devices performing two-level combinational logic functions are grouped into the field-programmable logic array (FPLA) category. These elements are a step up in complexity from gate arrays, capable of generating AND-OR, AND-NOR, and their De Morgan equivalents. There are at present two array types in the FPLA family, each with either open-collector or three-state outputs.

The equivalent logic diagram of the first array type, the open-collector output 82S101 (or three-state output 82S100), is shown in Fig. 2. The first level of logic in the device is made up of 48 AND gates fuse-connectable to any of 16 common inputs by true/complement buffers. The second logic level consists of eight OR gates—one per device output—each capable of being selectively coupled to any of the 48 gates. Finally, fusing options are included for generating true or complementary outputs.

The second logic array type, the 82S106/107, has nearly the same organization as the first. The exception is that an additional OR gate with fixed inputs has been added to generate an internal enable command for the output structure. That self-enable is generated whenever any of the AND gates become logically true, which occurs when the external input code matches the internal AND-gate program. In the absence of such a match, all device outputs are unconditionally disabled. The self-enable signal is available externally—the chip-enable input ($\overline{CE}$) pin on the 82S100/101 becomes an open-collector output called $\overline{FLAG}$. Because of this feature, the 82S106/107 can be viewed as a content-addressable programmable read-only memory, ideally suited to modifying data in large ROMs, as will be shown in the second part of this article.

Shared gates

Both array types benefit from the second level of logic. The advantage here is that the AND gates can be shared—OR gates can couple with up to 48 AND gates. Also, a key advantage of this arrangement over single-level logic is that it allows editing—disconnecting invalid AND terms from the OR array and replacing them with spare AND gates (Fig. 3).

Open-collector versions of both gate-array and logic-array devices can form wired-AND outputs in order to expand the number of AND gates available on a single chip. This solves the problem that is posed by applications exceeding the resources of a single device. The only restriction is that the expanded outputs have to be programmed to be active-low.

By far the most powerful members of the family are the field-programmable logic sequencers (FPLS), which add on-chip registers to arrays of AND and OR gates. The

How the FPLF defines logic

For the most part, schematic representation of logic in the field-programmable logic family follows conventional notation—the devices include AND, OR, and exclusive-OR (EXOR) gates, as well as set-reset (S-R) flip-flops and true or complement buffers. To simplify the representation of fuse-link programmabiity, however, the FPLF schematics use a matrix arrangement with cross-point coupling to represent intact fuse links.

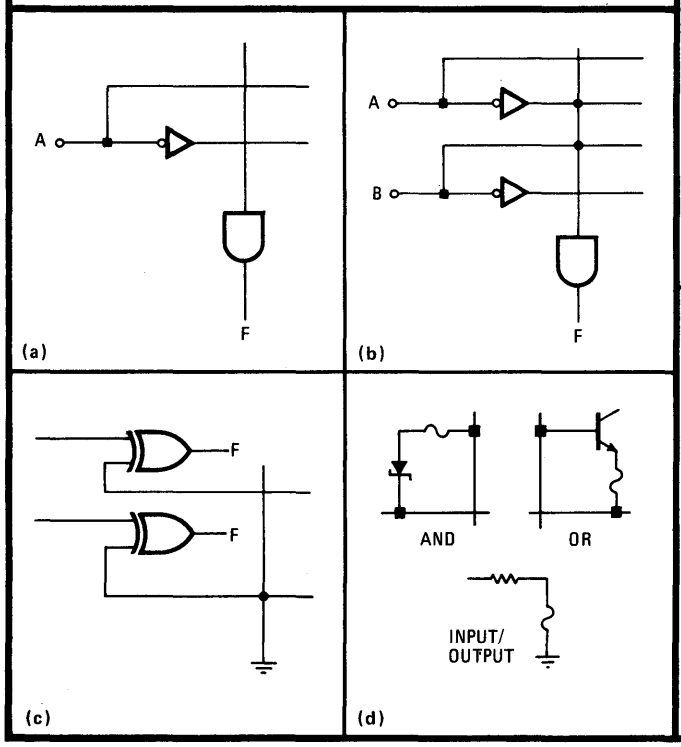
For example, (a) in the figure shows a typical input and AND gate of a gate array. The square "solder dot" represents a fixed internal connection. Both the line from input A and the line from the output of the inverter intersect the vertical input line of the AND gate; in actuality, fuse links make both connections. An intact fuse link is represented by a round solder dot. Blowing either of the fuse links will determine whether the input to the AND gate is A, or its complement, $\overline{A}$. (Leaving both fuses intact holds the output of the AND low, whereas blowing both fuses results in a "don't care" situation, an output that is independent of either input.)

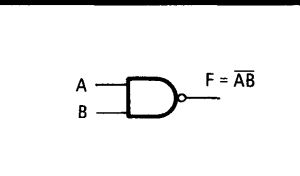
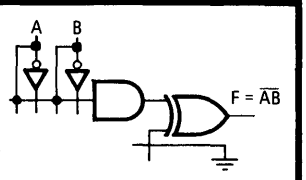
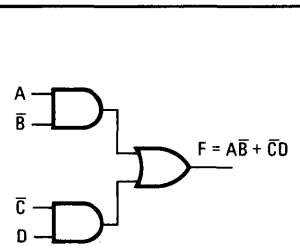
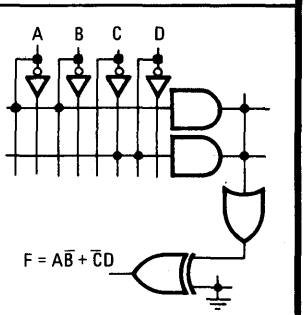
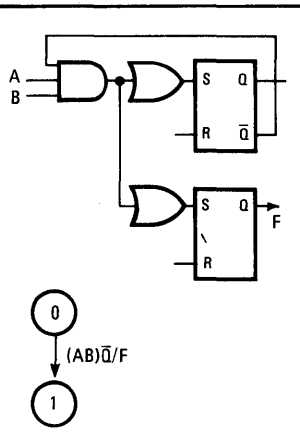
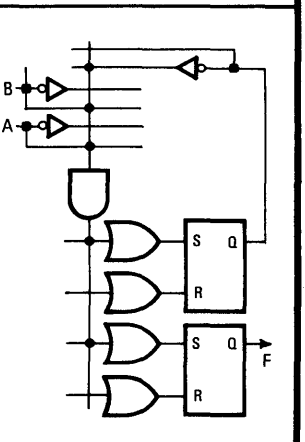
Extending the matrix and cross-point coupling approach a step further, (b) shows the configuration of a two-input AND gate. Since the input to the AND gate crosses the four lines of inputs A and B as well as their complements, the gate serves as a four-input AND while appearing to be a single-input gate. Since the members of the field-programmable logic family have 16 inputs intersecting each AND gate, the gates are actually 32-input devices; the number of inputs used is determined finally by the number of fuses left intact. Thus, in (b), solder dots (or intact fuse links) create the logical equation $F=\overline{A}B$.

The exclusive-OR gates on all outputs of the logic-family devices allow programming for either active-high or active-low output signals. As shown in (c), a fuse link grounding one of the two inputs of an EXOR gate results in an active-high output; blowing the fuse results in an output that is active-low.

The details of the fusing mechanisms are shown in (d). AND gates have a fuse in series with a Schottky diode, while OR gate fusing uses an npn transistor. The fuses for the true/complement input buffers and active-high/active-low outputs are in series with resistors.

The analogy between fixed and programmed logic is best shown by the examples in the table. The first example is typical of the single-level logic to which the gate array is applicable. The two-level logic of the second example is satisfied by the logic arrays. Finally, the registered state machine that executes the state transition of the third example is a candidate for the field-programmable logic sequencers.

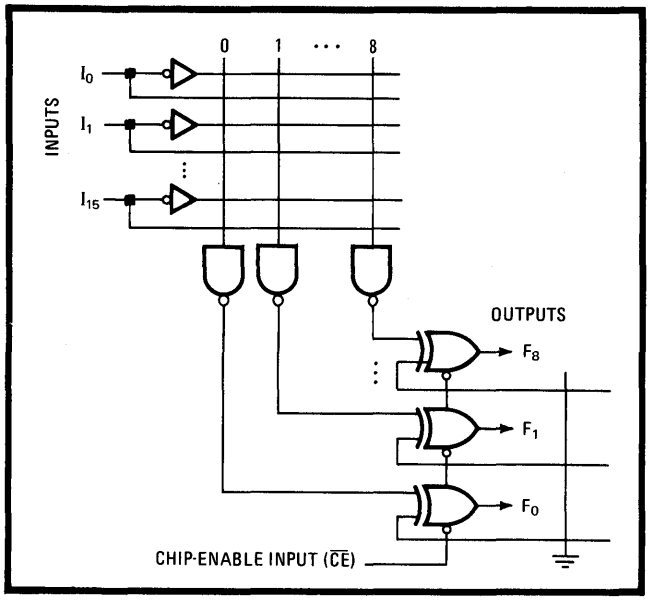


FIXED LOGIC	PROGRAMMED LOGIC
	
	
	

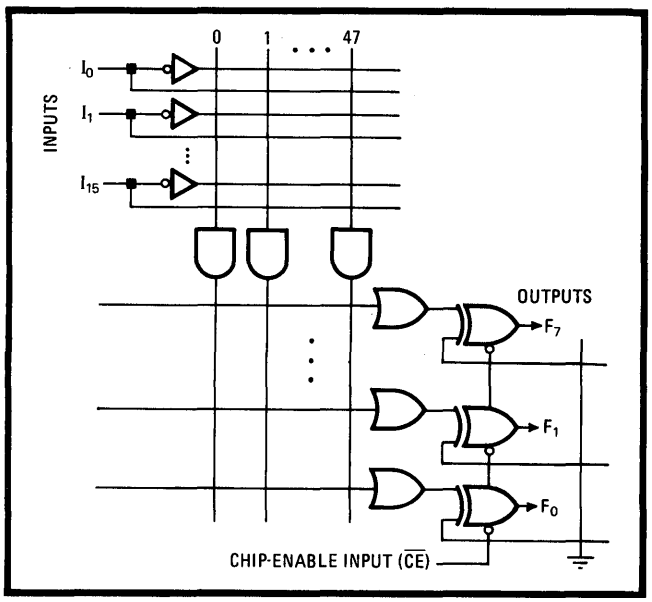
sequencers are actually self-contained state machines, since they can be programmed to perform any synchronously clocked logic sequence.

State machines, whose general structure is shown in Fig. 4a, usually take two forms: Moore machines, in which the output is a function of the present state only; and Mealy machines, whose output is a function of both the present state and the present input.

Figure 4b shows the basic architecture of the open-collector output 82S104 (or three-state output 82S105),



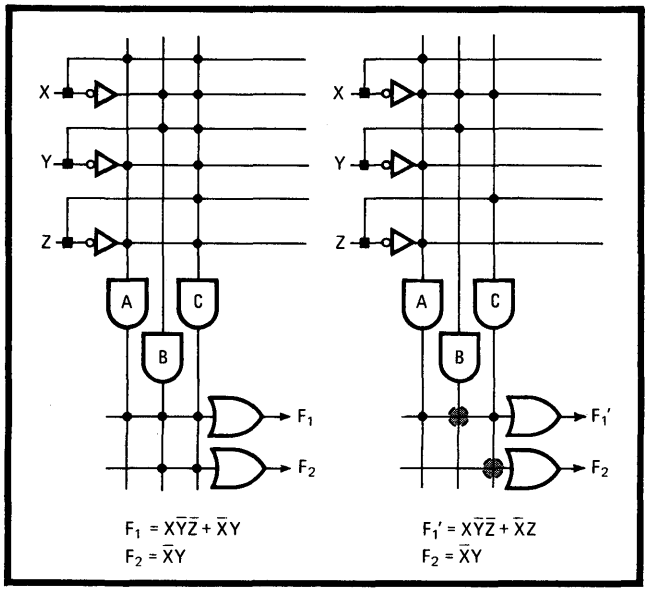
1. Gate array. The simplest device in Signetics' field-programmable logic is the gate array, capable of single-level logic. Any of 16 inputs can connect to nine NAND gates by true/complement buffers. Since outputs can be complemented to AND, manipulating De Morgan's theorem makes the device a universal logic element.



2. Double deep. The field-programmable logic array carries out two-level combinational logic. The 16 inputs couple to 48 AND gates, which in turn connect to any of nine OR gates. Either true or complement outputs are provided.

the first members of the FPLS family. With the FPLS, a user may program any logic sequence that can be expressed as a series of jumps between stable states triggered by a valid input condition I at clock time t. The number of states in the sequence depends on the length and complexity of the desired algorithm.

A typical state diagram is shown in Fig. 5. The state from which a jump originates is called the present state P, and that at which it terminates is the next state N. A jump always causes a change in state, but may or may not cause a change in the machine's output F.



3. Editing. The logic array's programmable OR gates allow sharing of AND gates, as with gate B at left. The OR array also allows easy editing of logic statements when design changes are made; note how spare gate C at right was used to modify output F_1 to F_1'.

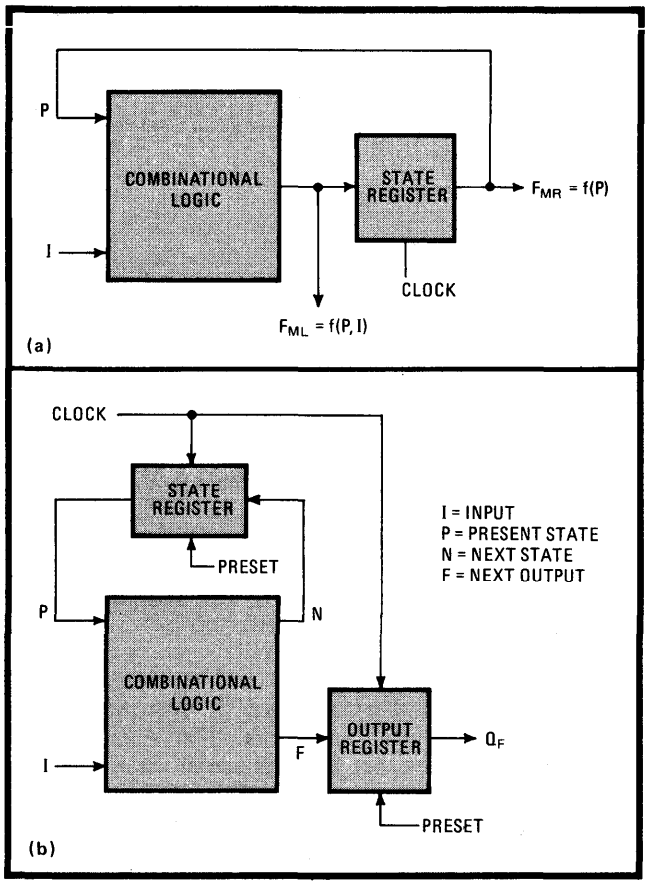
All states are arbitrarily assigned and stored in the state register, where the clock and next-state information from the combinational logic are the inputs. State jumps can occur only when transition terms are true. A transition term is, by definition, the logical AND function of the clock, present state, and valid inputs; hence, $T_n = t \cdot I \cdot P$. However, since the clock is actually applied to the state register, it may be removed from the equation. When T_n is true, a control signal is generated that, at clock time t, forces the contents of the state register from P to N and, if necessary, changes the contents of the output register.

FPLS organization

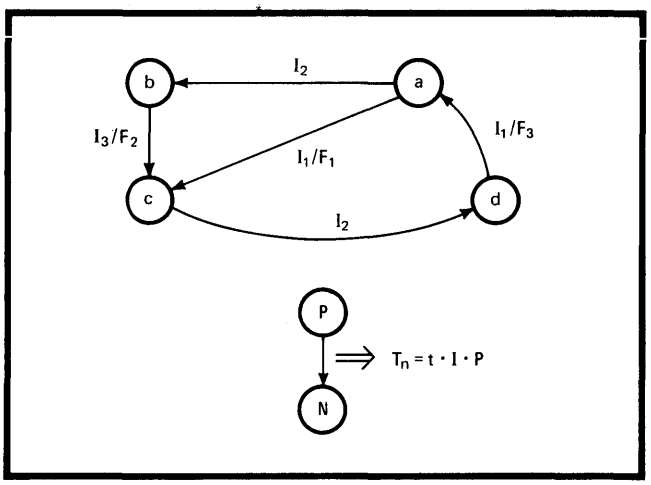
The architecture of the 82S104/105 is a natural extension of the static logic structure of the FPLA. It accepts 16 input variables and provides eight output functions. It has a 6-bit state register and an 8-bit output register; all the internal registers are automatically preset to logic 1 when power is applied. The FPLS provides for 48 transition terms, which can be selected to be either true or complementary.

A look at the equivalent logic diagram of the FPLS (Fig. 6) shows its extension of the static FPLA. The AND and OR gate arrays of the latter have been expanded to control the set and reset (S and R) inputs of six flip-flops (the state register) and to monitor the register's contents over an internal feedback path. Also, an independent 8-bit output register has been added to store output commands generated during state transitions and to hold the output constant during state sequences involving no output changes.

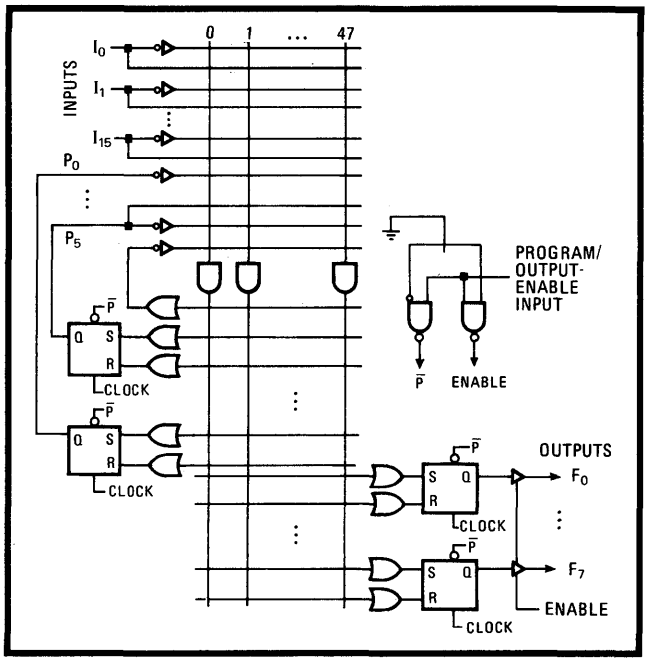
The AND array comprises 48 positive AND gates, each with 44 input connections from a set of true/complement buffers. The AND gates are used to form logic products of 16 external inputs (I_0 to I_{15}) with six present-state (P) inputs fed back from the state register. The gates are



4. State machine. A state machine (a) takes either a Mealy or Moore form. The architecture of the field-programmable logic sequencer (b) is that of a self-contained Mealy machine, where the output is a function of both the present state and the present input.



5. State diagram. Example of a state diagram (a) with four states—A, B, C, and D. I_1-I_3 are jump conditions, which trigger output changes F_1-F_3. A state change (b) gives rise to transition term T_n, which is logical AND of clock t, input I, and present state P.



6. Sequencer. The field-programmable logic sequencer has 16 outputs, 48 AND gates, and 28 OR gates, plus 14 flip-flops that serve as state and output registers. Either an asynchronous preset input or an output-enable input is available as a programming option.

therefore called transition terms because, like the transition terms in state diagrams, they issue next-state commands.

The OR array contains 28 positive OR gates, each with 48 input connections to all 48 AND gates. The outputs of the ORs drive the set and reset inputs of the 14 S-R flip-flops that are state and output registers.

The FPLS is made still more flexible by a complement array comprising a single 48-input OR gate that drives an inverter, which then feeds back into the AND array. The complement array forms a bridge between the AND and OR arrays for generating NAND functions of input-jump conditions; the user programs it in such a way as to suit each transition term.

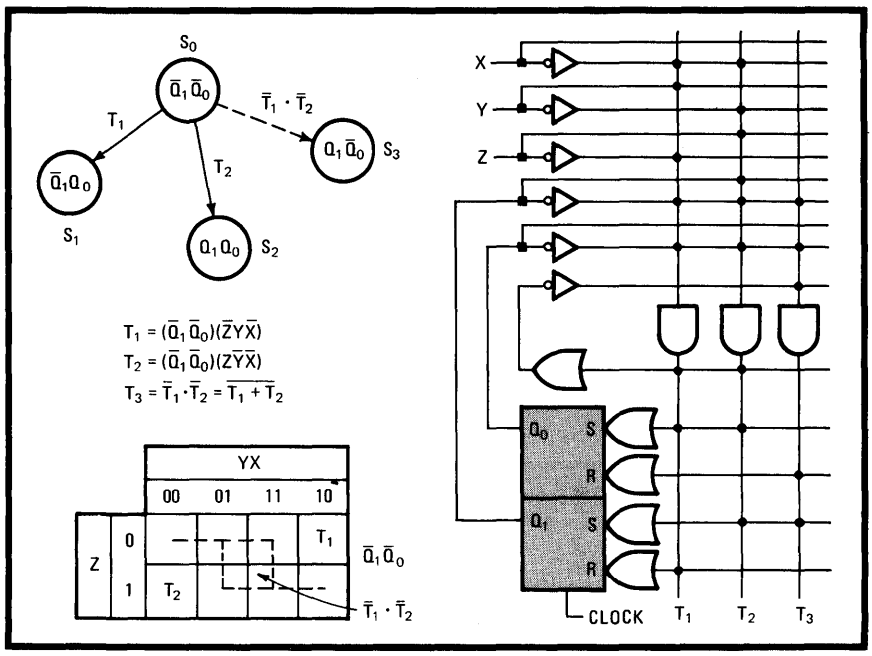
De Morgan's theorem

De Morgan's theorem to reduce logic terms can be easily implemented with the complementary array so that the most use is made of the AND gates. For example, if the transition term is $T = (Q)(\overline{X}+\overline{Y}+\overline{Z})$, where Q is the output of the state register and $\overline{X}$, $\overline{Y}$, and $\overline{Z}$ are inputs, three AND gates in the FPLS are required. However, De Morgan's theorem changes the transition term to $T = (Q)\,\overline{XYZ}$, which requires only two AND gates.

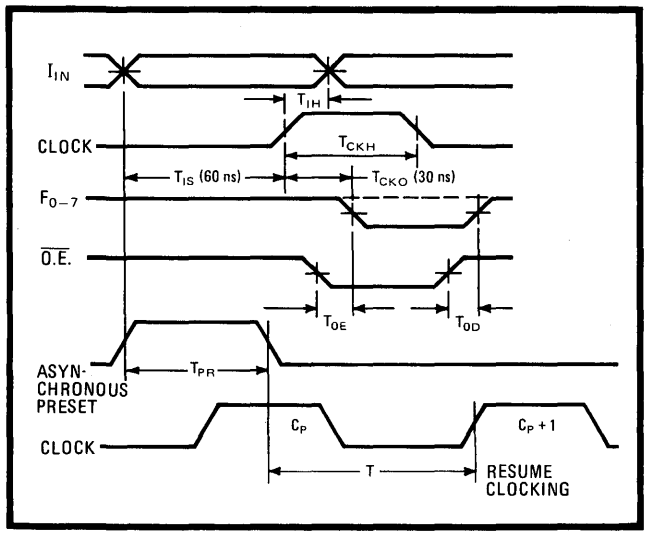
The complementary array is also an efficient means of aborting a clocked sequence in the absence of valid jump conditions. As Fig. 7 shows, considerable minimization of AND gates is possible when the detection of valid jumps involves many complements of jump functions, especially as the number of variables increases.

All clocked S-R flip-flops that make up the state and output registers offer the option of asynchronous presetting to all 1s. The 64-state total that can be represented by the state register is adequate in most cases to chart algorithms involving fewer than 48 nonredundant transitions. The register accepts next-state commands (N) from the OR array and supplies present-state information (P) to the AND array.

The output register is similar to the state register, except it has eight states for servicing eight output functions. It accepts the next-output commands F_0-F_7



7. Complementary. Use of AND gates in the FPLS is greatly reduced by the complement array, a single 48-input OR gate driving an inverter that feeds back into the AND array. In this example, the default jump from state S_0 to S_3 is reduced from three AND gates to a single gate T_3.



8. Timing. Minimum clock duration for the FPLS is 20 nanoseconds. Minimum width of preset input, which overrides clock, is 40 ns. Normal clocking resumes with the first full clock pulse following a negative clock transition after the trailing edge of the present signal.

from the OR array and then reflects its contents to the device outputs through the buffered Q outputs of each of the flip-flops. Also, as an added feature to enhance fault isolation, driving input I_0 to +10 volts will route the contents of the state register (P_0–P_5) directly to outputs F_0–F_5 without any alteration of the contents of the output register. However, the feature is not recommended for use in a normal mode of operation (as in a Moore machine). This is because it increases the device's maximum current by 5 to 10 milliamperes and thereby lowers the maximum ambient temperature rating of the package by approximately 5°C.

As a final programming option in the 82S104/105, a pin can function as either an active-high asynchronous preset (pr) or an active-low output enable ($\overline{OE}$). The output-enable function forces all outputs to logic 1 (or to high impedance in the 82S105) and is normally used when the device is sharing a bus. It does not inhibit clocking of the internal registers. The asynchronous preset option, on the other hand, is useful when the logic sequence requires an immediate state-independent return to initial conditions. The state register and output register can also be synchronously preset independently of one another by dedicating that function to one of the input variables in conjunction with a single transition term and a clock pulse.

Timing constraints

The maximum clock rate of the 82S104/105 can be inferred from its timing diagram (Fig. 8), which shows worst-case delays and setup requirements during a typical I/O cycle. Using stable external inputs as a reference, the device can be clocked after a minimum setup time of 60 nanoseconds. The next output (as well as the next internal state) will be valid 30 ns after the positive edge of the clock, giving a total I/O delay of 90 ns. Since both output enable and disable delays are also 30 ns, when the $\overline{OE}$ pin is used its signal's edge should occur prior to or coincidentally with the clock in order to avoid increasing I/O delays.

The asynchronous preset option includes a clock lockout feature that eliminates the potential hazard of spurious clocking. But, as the timing diagram shows, when using the lockout feature it is possible to miss one clock pulse, which may be prohibitive in some applications.

Applications

The second part of this article, to appear in the next issue of *Electronics*, will provide examples of applications for the gate- and logic-array devices. It will also describe in detail the development of a full-blown cartridge-tape drive controller built with a single logic-sequencer chip. The design example proceeds from flow chart to state-sequence diagram to hardware.

Sequencers and arrays transform truth tables into working systems

by Napoleone Cavlan and Stephen J. Durham
Signetics Corp., Sunnyvale, Calif.

□ Because of its power and flexibility, the Signetics field-programmable logic family is ideal for replacing the discrete logic normally used to interface large-scale integrated devices, as shown in Part 1 [July 5, 1979, p. 109]. The examples of applications that follow show how to exploit its special features.

In designing with these gate and logic arrays and logic sequencers, the user need concern himself only with generating truth tables associated with the state diagrams or sets of Boolean logic equations that define his function. The one restriction is that he must use logic symbols corresponding to the status of fuse links.

As indicated in Fig. 1, an extra set of symbols is needed to describe all the states of FPLF gates corresponding to all combinations of blown and unblown fuse links. Once ordered into truth tables, the user-defined functions are then directly mapped onto standard program tables furnished with FPLF elements, whose fuses are then blown by a logic-type programmer. As the user gains experience, he can manipulate logic variables intuitively and can eventually implement algorithms directly on the program tables with only the device schematics for reference. (The formal step of deriving state diagrams and logic equations will not be considered here.)

Because of their simple and uncommitted structure, FPLF elements are suited to a wide variety of applications, several of them already well documented. The following examples illustrate the typical use of each logic element and match devices with applications.

Bus translator

Signetics' Instructor 50 microcomputer system is built around the 2650 microprocessor; but for compatibility with other systems and peripheral devices in the hobbyist market, it interfaces to the S100 bus, which is based mainly on 8080 microprocessor signals. Yet to carry out the seemingly unwieldy task of bus translation, only a single FPGA is needed. The gate array translates the logical combinations of timing, enable, and control signals supplied by the 2650 and its I/O hardware into control signals entirely compatible with the S100 bus definitions, as shown in Fig. 2.

The programmable feature of the FPGA is strategically invaluable in this case since the S100 bus is not yet totally standardized. The FPGA permits easy adaptation of the interface to changes in specifications, which are subject to arbitrary manipulation by manufacturers in the hobby arena.

Two-level logic

The logic arrays add a second level of combinational logic to the gate arrays, and thus another level of versatility. AND/OR combinations of the FPLAs are well suited to carrying out polynomial equations and the like, as shown in the next example.

In systems that transfer large blocks of data, a cyclic redundancy check (CRC) scheme can significantly improve data integrity. The technique appends a check word to a transmitted sequence of data, and the receiving end uses that word to check for errors. A cyclical division of the transmitted data by an industry-standard polynomial generates the CRC word; the remainder from the division forms the check word.

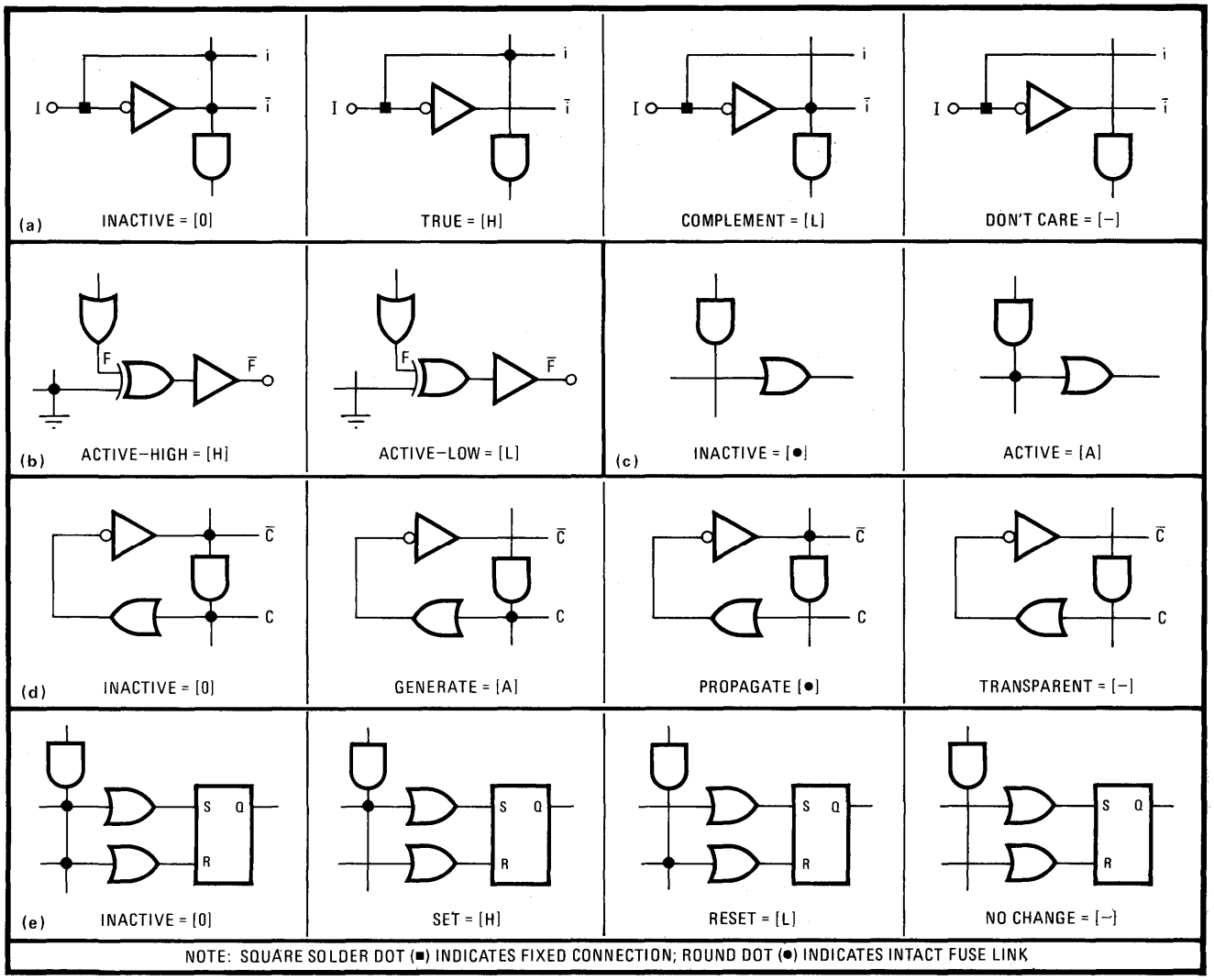
Polynomials lend themselves to serial manipulation, and serial CRC generation and checking are easy to implement. But in a multiple-line data system with parallel organization, a considerable amount of hardware may be needed for parallel-to-serial conversion. Moreover, the multiple-bit clocking for each word carries an inherent speed loss—a factor of 8 for a byte-oriented system. A parallel CRC generator-checker circuit is the answer, developed from the set of logic equations describing the function of the circuit in the form of a state machine.

The general design of the CRC circuit is shown in Fig. 3a, along with the logic equation set for the popular CRC polynomial $P(x) = x^{16}+x^{15}+x+1$. Figure 3b shows that the entire byte-wide parallel CRC generator-checker circuit can be implemented with only five chips: two 8-bit latches, two FPLAs, and an FPGA. The FPLAs contain the set of logic equations controlling the flip-flop inputs, which are expanded from EXOR form to sum-of-products form. In Fig. 3a, variables N_0–N_{15} represent the next CRC word after clocking, based on the current word B_0–B_{15} and the present input byte D_0–D_7.

CRC generation begins by driving the $\overline{\text{RESET}}$ line low to initialize the latches to zero. Pulsing the clock line then transfers the first byte of the data block in at D_0–D_7. Subsequent bytes are clocked in the same way. The cyclic nature of this design places no limit on the size of the data block that can be processed. During data transmission, the 16-bit CRC word is available at outputs B_0–B_{15} after the last data byte has been clocked in; it is appended as two check bytes to the data in the block.

Checking

The circuit is used in the check mode when receiving data containing CRC characters. The last 2 bytes in the data block received are CRC send characters. They too are clocked in and contribute to form a final receive pair of CRC characters, which, for error-free transmission, must both be zero. If an error has occurred, B_0–B_{15} will be nonzero. The FPGA will detect the nonzero condition and generate an error signal. This parallel CRC format can operate on data blocks at speeds in excess of 5.7



1. New notation. The many combinations of blown and unblown fuse links in the field-programmable logic family require new notation. The four possibilities for AND gates are shown in (a), while those for exclusive-OR outputs are in (b). The combinations for OR gates are in (c). The complement array in the logic sequencers is detailed in (d). Finally, OR gates controlling the flip-flops in sequencers are in (e).

megabytes per second.

An interesting use for the FPLA is in changing data at a few locations of a read-only memory (see "How to patch a read-only memory," p. 137).

The abilities of the field-programmable logic sequencer are well demonstrated by its use as a controller for a cartridge-tape transport. In this example, one chip replaces many—a distinct advantage if the controller is to be packed on a single-board microcomputer. Although the chip's function is complex, it can be programmed methodically and worked directly from a flow chart.

Controller routines

The controller executes fixed routines in response to status and input commands that may originate from an input/output bus or a monitoring station. Its outputs operate the velocity servo that drives the cartridge, form I/O status signals, and enable writing of data. The input and output signals of the one-chip controller are shown in detail in Fig. 4.

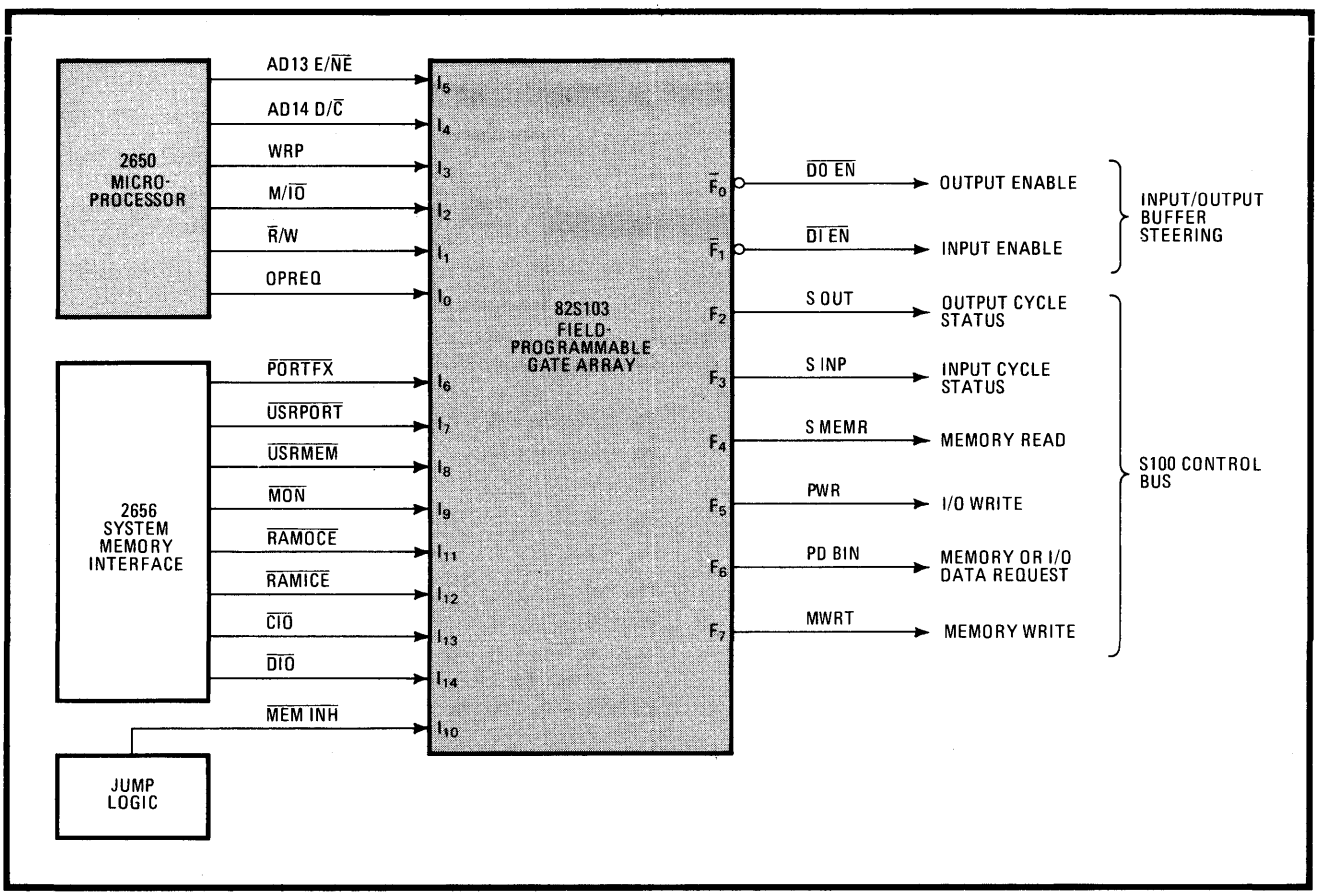
The controller carries out these eight routines:

- Move tape fast-forward.
- Move tape slow-forward.
- Move tape fast-reverse.
- Move tape slow-reverse.
- Bring tape to load point when cartridge is inserted.
- Rewind tape to load point.
- Rewind tape to beginning and eject cartridge in response to unload command.
- Rewind tape to beginning and eject cartridge in response to auto-unload true condition.

The routines could be represented concisely in a conventional Mealy state diagram, but that often obscures the actual machine function. Flow charts are more easily understood, where input variables, machine states, and output functions are given variable names. Such a chart is shown in Fig. 5.

Diagramming the flow

What would be transition terms in a Mealy state machine become true/false statements regarding the system inputs (taken one at a time) in the chart. The correlation is most obvious in the simple example in Fig. 6. The flow chart in (a) shows a conditional change from



2. Translator. Getting S100 bus signals, which are mostly 8080 microprocessor signals, out of a 2650 microprocessor calls for a field-programmable gate array. One 82S103 translates signals from the 2650 and its companion 2656 interfacing chip to the hobby bus.

state A to state B. The conditions in the flow chart's diamonds must be simultaneously satisfied for the state change to occur. The conditions take on variable names, and for this example, which arbitrarily assumes a 4-bit state register, three inputs, and two outputs, the corresponding state diagram is shown in Fig. 6b.

The transition from A to B denotes a jump from 10 (1010_2) to 13 (1101_2) and an output transition to 2 (10_2) at the next clock pulse if the combination $X_n = 4$ (100_2) is true. The transition is synthesized by forming a transition term $T = P_3\overline{P}_2P_1\overline{P}_0I_2\overline{I}_1\overline{I}_0$ and using term T at the next clock pulse to generate next-state and next-output commands for the state and output registers, respectively. For the state register, flip-flops N_0 and N_2 are set by connecting T to set lines S_o and S_2, and flip-flop N_1 is reset by coupling T to the R_1 reset line. Similarly, for the output register bit F_o is reset and bit F_1 is set by connecting T to corresponding flip-flop reset (R_0) and set (S_1) lines.

Controller conditions

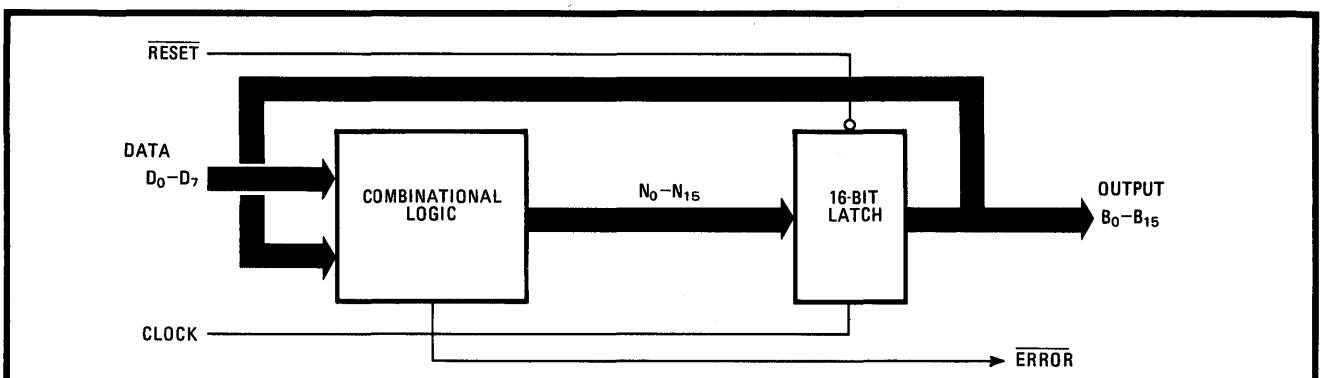
Referring again to the controller flow chart, it can be seen that whenever the tape-drive power is turned on, or when an interlock is opened, the transport must be stopped. That is achieved by an input signal to the controller called $\overline{INTRDY}$ that resets the state register with an unconditional jump to state 1 or STOP. When that occurs, all outputs on the FPLS chip become inactive, WRITE is inhibited, and speed and direction are arbitrarily set to SLOW and REVERSE. From the STOP state, operation into any mode ocurs by state and output jumps when all of the intervening conditions are simultaneously satisfied.

As an example, writing at normal speed will occur with a jump from state 1 to state 3, which requires that the following criteria be satisfied:

- The data cartridge is in place; therefore CIP is true.
- The drive has been addressed; SEL is true.
- The tape has been commanded to run; TR is true.
- The controller is not in state 6; state 6 is false.
- The tape should move slowly; therefore $\overline{FAST}$ is true (an active-low signal).
- The tape should move forward; $\overline{FWD}$ is true.

In tracing the jump between these states two things must be noted. First, the commands RWD, UNL, and TR are mutually exclusive, so that when either is true the others can be considered false or "don't care." Second, after TR = true, the condition (State = 6?) is inserted to indicate invalid jumps to states 2 and 3, which could originate from state 6 with an AUTO UNL false. Clearly, these should be avoided to inhibit honoring requests for read slow (or fast) forward while stopped at the end of the tape. So, the (State = 6?) condition is a reminder to avoid programming 6 → 2 and 6 → 3 state jumps in the FPLS. A similar argument holds for (State = 7?) and (State = 11?) conditions.

After data has been either written or read, the tape drive is commanded to stop by TR false, which causes a



$N_0 = D_0 \oplus D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus B_8 \oplus B_9 \oplus B_{10} \oplus B_{11} \oplus B_{12} \oplus B_{13} \oplus B_{14} \oplus B_{15}$

$N_1 = D_0 \oplus D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus B_9 \oplus B_{10} \oplus B_{11} \oplus B_{12} \oplus B_{13} \oplus B_{14} \oplus B_{15}$

$N_2 = D_6 \oplus D_7 \oplus B_8 \oplus B_9$

$N_3 = D_5 \oplus D_6 \oplus B_9 \oplus B_{10}$

$N_4 = D_4 \oplus D_5 \oplus B_{10} \oplus B_{11}$

$N_5 = D_3 \oplus D_4 \oplus B_{11} \oplus B_{12}$

$N_6 = D_2 \oplus D_3 \oplus B_{12} \oplus B_{13}$

$N_7 = D_1 \oplus D_2 \oplus B_{13} \oplus B_{14}$

$N_8 = D_0 \oplus D_1 \oplus B_0 \oplus B_{14} \oplus B_{15}$

$N_9 = D_0 \oplus B_1 \oplus B_{15}$

$N_{10} = B_2$

$N_{11} = B_3$

$N_{12} = B_4$

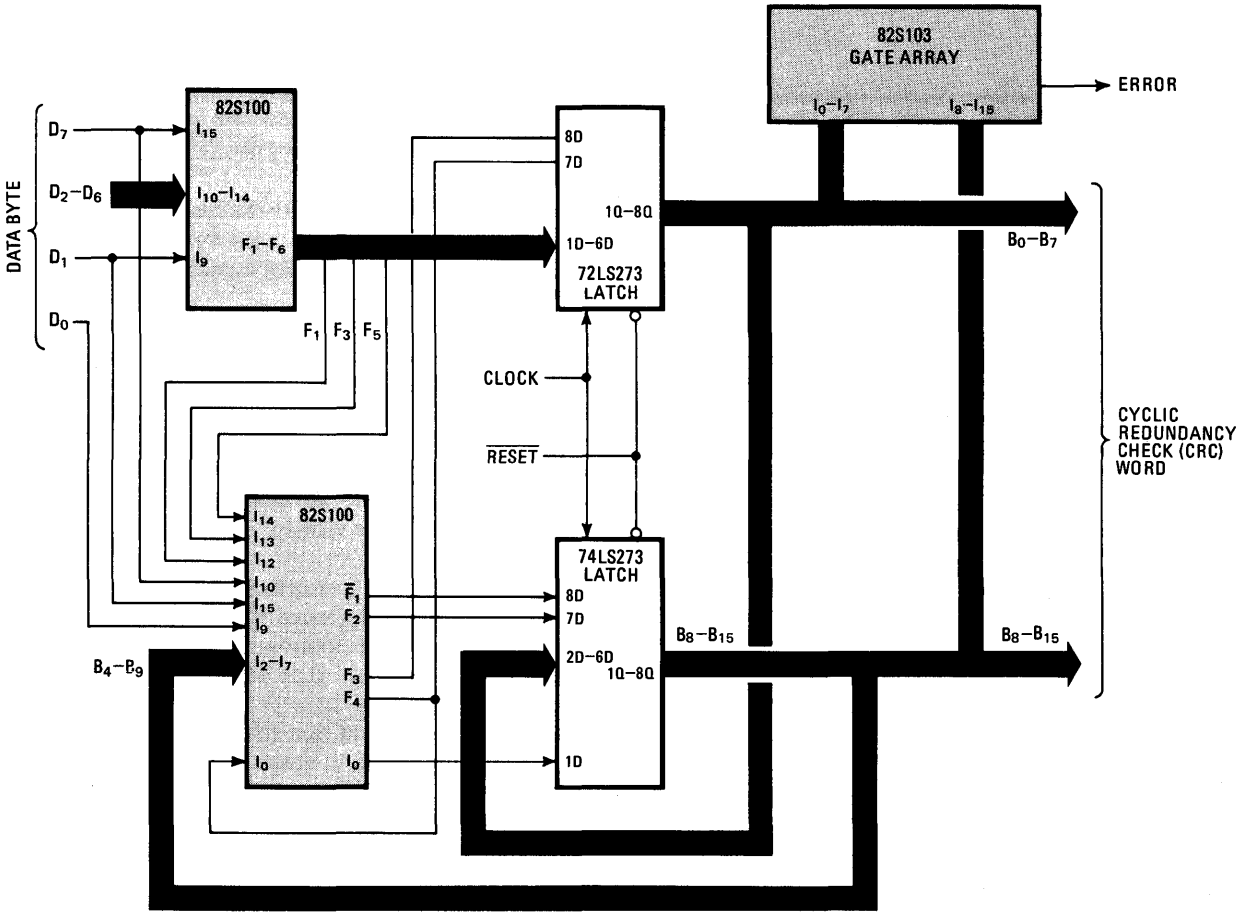
$N_{13} = B_5$

$N_{14} = B_6$

$N_{15} = D_0 \oplus D_1 \oplus D_2 \oplus D_3 \oplus D_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus B_7 \oplus B_8 \oplus B_9 \oplus B_{10} \oplus B_{11} \oplus B_{12} \oplus B_{13} \oplus B_{14} \oplus B_{15}$

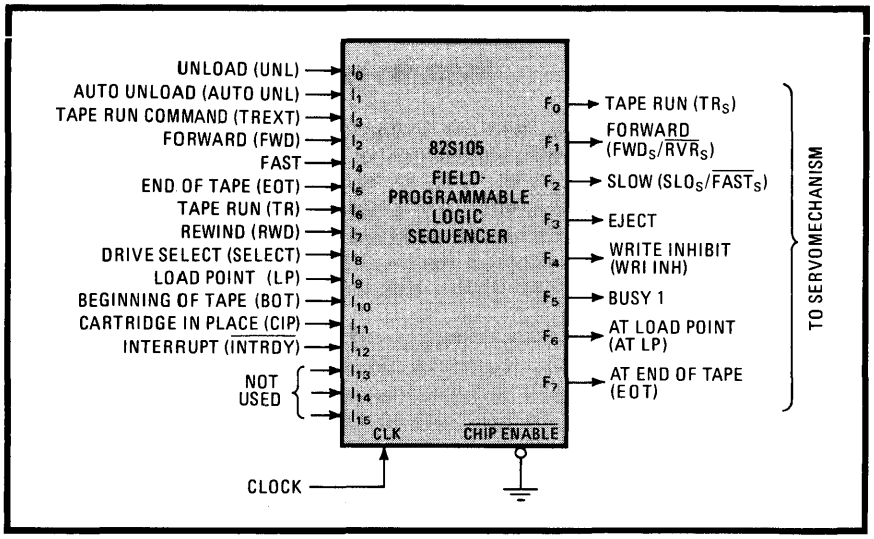
$\overline{ERROR} = \bar{B}_0 \cdot \bar{B}_1 \cdot \bar{B}_2 \cdot \bar{B}_3 \ldots\ldots\ldots\ldots\ldots\ldots \cdot \bar{B}_{15}$

(a)

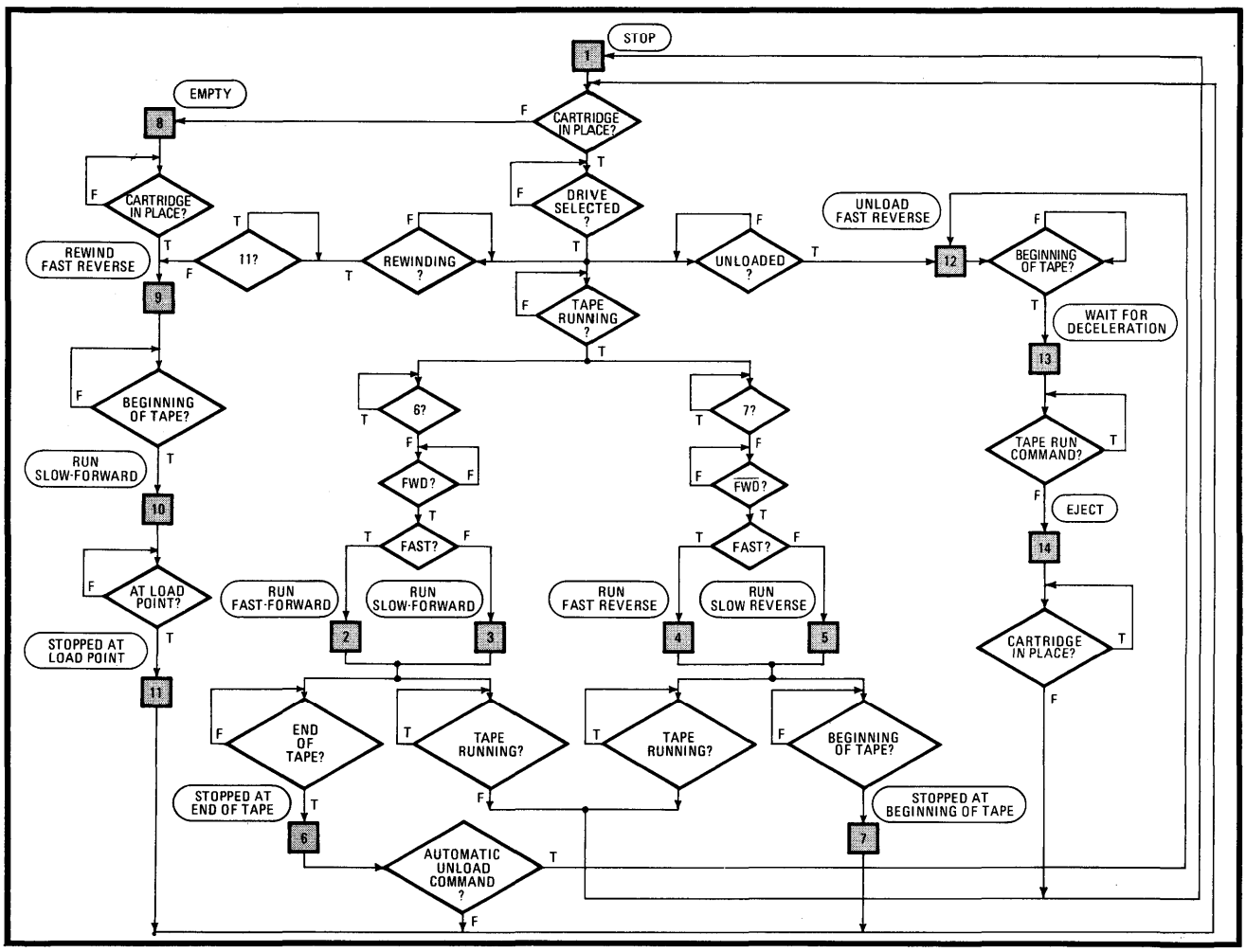


(b)

3. Error-free. The technique of using a cyclic redundancy check (CRC) word for error-free data transmission requires complex logic to generate the word (a). A pair of logic arrays, two latches, and a gate array (b) do the job, which usually requires a boardful of chips.



4. Tape controller. A field-programmable logic sequencer like this tape controller can perform extremely complex tasks. The 82S105 receives commands from an input/output bus or monitor, and provides all the necessary signals for driving the tape-transport servo-motor mechanism.



5. Goes with the flow. The first step in designing the controller is preparing a flow chart of the operation. The chart is much easier to understand than a state diagram or Mealy machine, yet provides all the information needed for programming the logic-sequencer chip.

jump from state 3 (RUN SLOW FORWARD) to state 1. By similar arguments, the tape drive can be run either fast or slow in either forward or reverse directions by jumping to states 2, 4, and 5.

When the end of tape is reached (EOT true), the tape drive is stopped. That is implemented by jumps 2 →6 or 3 →6. Once in state 6, the tape drive can no longer move in the forward direction because of the State 6 false condition preceding states 2 and 3. If AUTO UNL is true, the drive will automatically rewind (state 12), wait for tape to decelerate (state 13), eject the tape cartridge (state 14) and stop. If AUTO UNL is false, the drive must

How to patch a read-only memory

It is a shame to throw away read-only memories. But often firmware-based systems must commit control programs to large mask-programmed ROMs, only to have a design revision requiring a new program—and a new ROM. If no pin-compatible, user-programmable ROM is available, the customer could end up waiting out the 5-to-10-week turnaround time for the new mask parts—and throwing away his inventory of old ROMs.

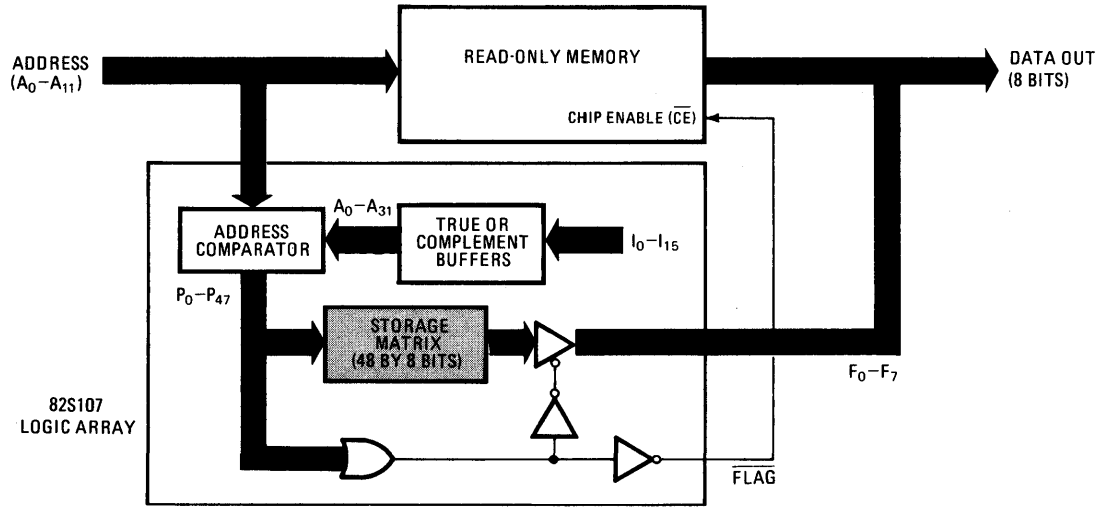
One way to save an obsolete ROM (or even PROMs—it hurts to throw them away, too) is by patching, which redirects certain addresses to an adjunct smaller memory. This can be done most efficiently with an 82S107 field-programmable logic array.

As a ROM patch (FPRP), the FPLA becomes a programmable, content-addressable PROM that continually monitors the address bus. As shown in the figure, when the FPRP encounters a match that signals a correction of data, its flag output ($\overline{FL}$) disables the ROM, and new data from the FPRP is put on the output bus. If, for example, address 750 were to be given new data A9, address 5FE were to be given 7F, and addresses OA4–OA7 were all to be reassigned B4, the FPRP would be programmed as in the table. For a 12-bit address, only inputs I_0–I_{11} are used, and the remaining four, I_{12}–I_{15} then become "don't care." (Incidentally, inputs I_0 and I_1 in the second product term are also "don't care" because they define an address block of four locations.)

The address comparator can patch up to 48 non-overlapping addresses anywhere within a memory field of 64 kilobytes. Block addressing is possible, too, using the FPRP's true or complement input buffers. Moreover, the number of addresses can be expanded by hooking several devices in parallel and wire-ANDing their flag outputs.

Since the outputs of the ROM patch primarily define a byte of memory data rather than a set of logic functions, output polarity is not controlled. Also, to maintain compatibility with the gate array, the FPRP generates its self-enable signal with a fixed multiple-input OR gate; the only disadvantage of that method is addresses (AND terms), once programmed, may no longer be deleted.

The ROM patch affords a recovery strategy effective in several design situations, including modifications of dedicated application programs, operating systems, assemblers, and monitor routines. It also permits on-site optimization of system parameters, in accordance with, say, environmental variables, and allows custom function options and product-line diversification. The customer need only allot board space next to the mask ROM for an FPRP; no parts are actually used until program changes are required after the product is in the field.



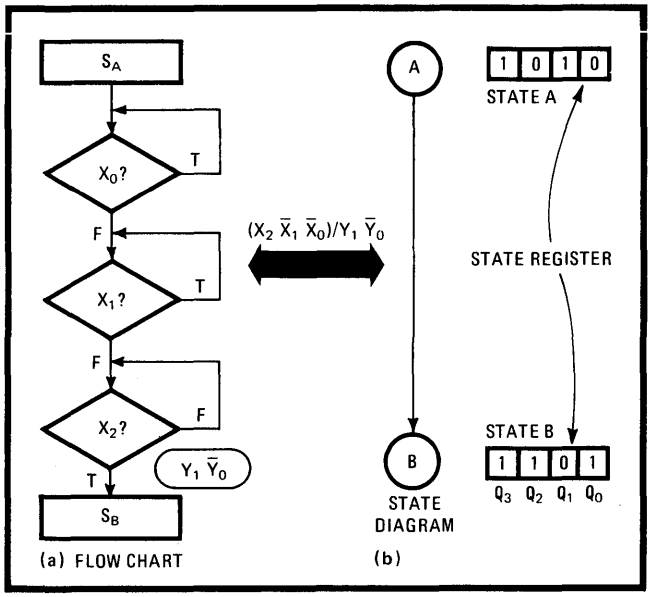
PROGRAM TABLE FOR ROM PATCH

Input address	Product term: No.	Comparator input: 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Active level: H	H	H	H	H	H	H	H	Output data
																		Output function: 7	6	5	4	3	2	1	0	
750	0	–	–	–	–	L	H	H	H	L	H	L	H	L	L	L	L	A	•	A	•	A	•	•	A	A9
5FE	1	–	–	–	–	L	H	L	H	H	H	H	H	H	H	H	L	•	A	A	A	A	A	A	A	7F
OA4–OA7	2	–	–	–	–	L	L	L	L	H	L	H	L	L	H	–	–	A	•	A	A	•	A	•	•	B4
		A_{11}															A_0	D_7							D_0	

wait for either a rewind command (RWD), an unload command (UNL), or reverse command (FWD).

If the tape is moved in the reverse direction until the beginning (BOT), the drive is stopped. This is implemented by a jump from states 4 or 5 to state 7. Once in state 7, the tape drive can no longer move in the reverse direction because of the state 7 false condition preceding states 4 and 5. The tape will remain stopped at the beginning until TWD, UNL, or FWD commands are given.

If no cartridge is in place (CIP false) when the tape drive is turned on, the controller will jump from state 1 to 8, and signal EMPTY. When a cartridge is installed,



6. Flow chart to state diagram. Simple transition from state A to state B is shown in flow chart (a). Three inputs (X_0,X_1,X_2) and two outputs (Y_0,Y_1) are assumed. The contents of a four-bit state register show the transition from state A (1010_2) to state B (1101_2).

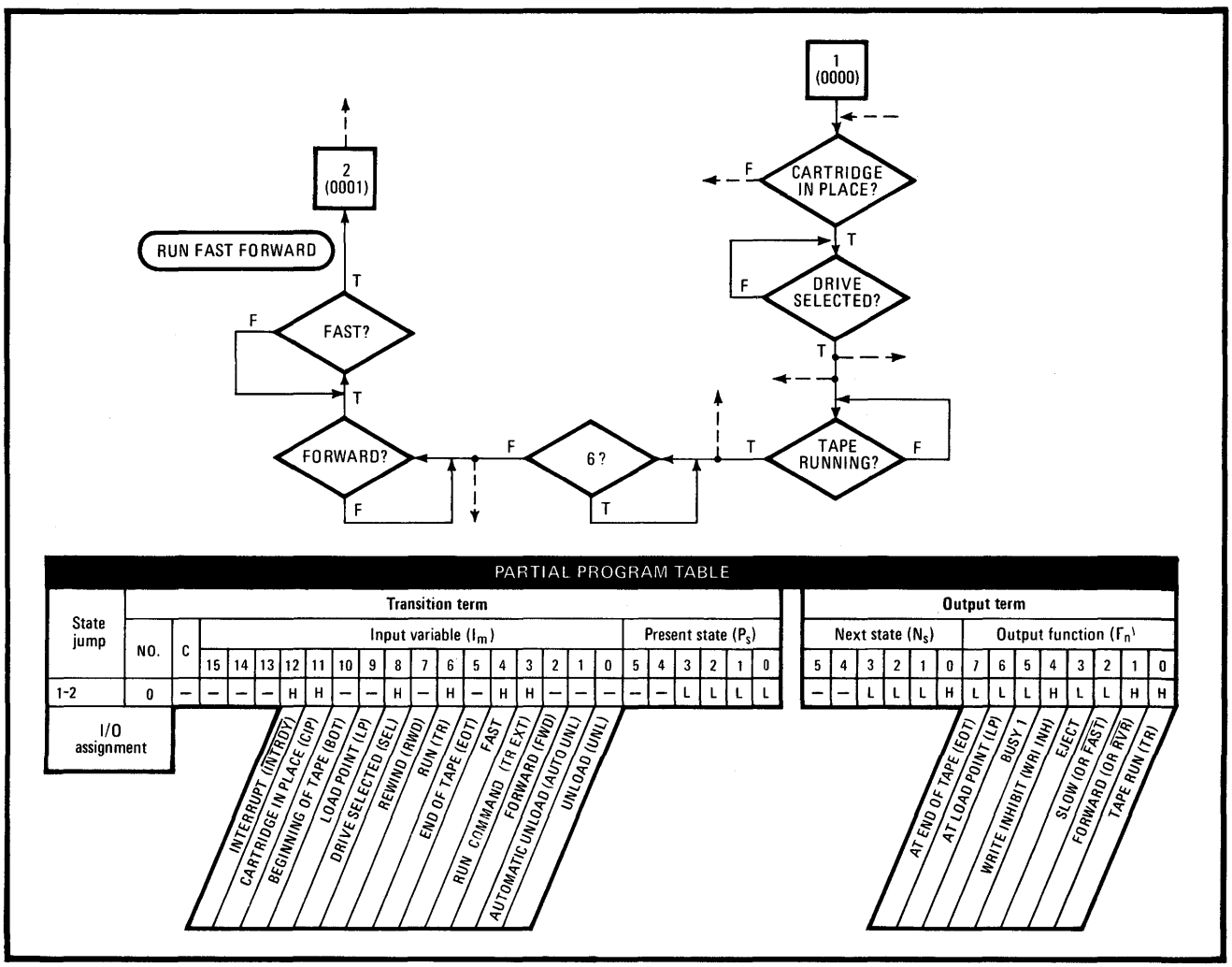
CIP = true implements a jump from state 8 to 9. In state 9 the tape will rewind in fast reverse until a BOT mark is reached. BOT true implements a jump from state 9 to 10. The tape now runs at slow speed in the forward direction until the load point (LP) is reached. LP true implements a jump from state 10 to 11 indicating STOPPED AT LP. From state 11, forward, reverse, or unload commands can be executed, but not rewind, because of the state 11 condition preceding state 9. That keeps RWD from being needlessly repeated.

State jump

A single state jump is shown in detail in Fig. 7. The transition is from state 1 to 2. In the latter, the controller is required to enter the READ FAST FORWARD routine from STOP when:

- CIP is true.
- SEL is true.
- TR is true.
- State 6 is false.
- FWD is true.
- FAST is true.

In response to this jump, the controller outputs that must change to issue the appropriate commands are run (TR),



PARTIAL PROGRAM TABLE

State jump	NO.	C	Transition term: Input variable (I_m) 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Transition term: Present state (P_s) 5 4 3 2 1 0	Output term: Next state (N_s) 5 4 3 2 1 0	Output term: Output function (F_n) 7 6 5 4 3 2 1 0
1-2	0	–	– – – – H H – – H – H – H H – – –	– – L L L L	– – L L L H	L L L H L L H H
I/O assignment			INTERRUPT (INTRDY), CARTRIDGE IN PLACE (CIP), BEGINNING OF TAPE (BOT), LOAD POINT (LP), DRIVE SELECTED (SEL), REWIND (RWD), RUN (TR), END OF TAPE (EOT), FAST, RUN COMMAND (TR EXT), FORWARD (FWD), AUTOMATIC UNLOAD (AUTO UNL), UNLOAD (UNL)		AT END OF TAPE (EOT), AT LOAD POINT (LP), BUSY 1, WRITE INHIBIT (WRI INH), EJECT, SLOW (OR FAST), FORWARD (OR RVR), TAPE RUN (TR)	

7. Detailed state jump. The transition from state 1 (STOP) in the tape cartridge controller to state 2 (READ FAST FORWARD) is shown in flow-chart form. That part of the logic sequencer coding of the state jump is shown below, including transition and output terms.

TABLE 1: COMPARISON OF DESIGN ALTERNATIVES FOR TAPE CONTROLLER

Parameter	Field-programmable logic sequencer	Discrete logic	Monolithic Memories Inc.'s Programmable Array Logic
Chip count	1 chip	6 chips	14 chips
Circuit-board area	0.84 in.^{2}	2.13 in.^{2}	3.78 in.^{2}
Power (typical)	0.60 W	1.36 W	4.8 W
Speed	90 ns/state	132 ns/state	105 ns/state
Voltage	+5 V	+5 V	+5 V
Cost (high-volume production)	$12	$14	$48

TABLE 2: PROGRAMMING EQUIPMENT FOR THE FIELD PROGRAMMABLE LOGIC FAMILY

Type	Manufacturer	Model	Field-programmable device				Availability
			Gate array	Logic array	ROM patch	Logic sequencer	
Logic	Signetics	FP-103	•				now
		FP-104		•	•	•	
	Curtis	PR-100		•			
		PR-100A		•	•		
	Data I/O	10		•	•		
Memory	Data I/O	17,19	•	•	•	•	3Q79
	Sunrise Electronics	SM100		•	•		now
			•			•	in development
Hybrid	Stag	PPX-Plus		•	•		now
			•			•	in development

forward (FWD), and fast (FAST).

The flow chart of the controller routines is complete with 14 states and 36 state jumps (including synchronous reset). As such, four state-register flip-flops sufficiently represent all states. All state jumps can be directly programmed into the chip from the flow chart. All state jumps occur on the leading edge of the clock.

The advantage of a controller built with the FPLS is best shown by a comparison to discrete logic, which would comprise PROMs, latches, and gates, using the same state diagram as for the FPLS. Table 1 compares the FPLS controller with a discrete implementation as well as with Monolithic Memories Inc.'s Programmable Array Logic chips, in several aspects.

Programming

The key to design flexibility with programmable logic is the availability of programming equipment. The need for PROMs in this equipment has led to a large number of memory programmers being offered by several manufacturers. Generally, they operate with personality card sets that meet the requirements of various PROM technologies. Suppliers have already begun developing sets compatible with memory programmers for logic devices. Hardware is expected to be available by the end of the third quarter of this year.

For the concept to work, the logic devices must be manipulated as memory chips are—by defining the desired fusing pattern in terms of an address-data relationship. Although this tends to obscure the logic function of the device, which is not visible on the program table, it is sure to provide low-cost programming equipment that can be manned by low-skilled labor.

Logic programming is another possibility, and low-cost equipment is already available from Signetics. Logic programmers allow direct entry of the logic function from the program table; no reference to the device logic diagram is necessary, and the user need not specify the status of each individual link in a device. Such programmers are more convenient for engineering use during the initial design phase, but with their high programming speed—about 10 seconds per device—can also be effective in production. Their only drawback is that they are dedicated machines and cannot program PROMs.

Some manufacturers offer a hybrid type of PROM programmer that can also be configured to do logic programming. Table 2 shows the various options available to prospective users now, or in the near future. □

SALES OFFICES

SIGNETICS HEADQUARTERS

811 East Argues Avenue
P.O. Box 409
Sunnyvale, California 94086
Phone: (408) 739-7700

ARIZONA
Phoenix
Phone: (602) 265-4444

CALIFORNIA
Canoga Park
Phone: (213) 340-1431

Cupertino
Phone: (408) 725-8100

Inglewood
Phone: (213) 670-1101

Irvine
Phone: (714) 833-8980
(213) 588-3281

San Diego
Phone: (714) 560-0242

COLORADO
Aurora
Phone: (303) 751-5011

CONNECTICUT
Danbury
Phone: (203) 744-6066

DELAWARE
Call: Micro-Comp, Baltimore
Phone: (301) 247-0400

FLORIDA
Lighthouse Point
Phone: (305) 782-8225

GEORGIA
Atlanta
Phone: (404) 953-0067

ILLINOIS
Schaumburg
Phone: (312) 843-7805

INDIANA
Kokomo
Phone: (317) 453-6462

KANSAS
Overland Park
Phone: (913) 341-8181

MARYLAND
Glen Burnie
Phone: (301) 787-0220

MASSACHUSETTS
Woburn
Phone: (617) 938-1000

MICHIGAN
Farmington Hills
Phone: (313) 476-1610

MINNESOTA
Edina
Phone: (612) 835-7455

NEW JERSEY
Cherry Hill
Phone: (609) 665-5071

Piscataway
Phone: (201) 981-0126

NEW YORK
Liverpool
Phone: (315) 451-5470

Melville
Phone: (516) 752-0130

Wappingers Falls
Phone: (914) 297-4074

NORTH CAROLINA
Raleigh
Phone: (919) 851-2013

OHIO
Worthington
Phone: (614) 888-7143

TENNESSEE
Greeneville
Phone: (615) 639-0251

TEXAS
Austin
Phone: (512) 458-2591

Dallas
Phone: (214) 661-1296

UTAH
Bountiful
Phone: (801) 298-2624

CANADA
SIGNETICS CANADA, LTD.
Etobicoke, Ontario
Phone: (416) 626-6675

SIGNETICS CANADA, LTD./LTEE.
Pointe-Claire, Quebec
Phone: (514) 697-3385

REPRESENTATIVES

ALABAMA
Huntsville
Electronic Sales, Inc.
Phone: (205) 533-1735

CALIFORNIA
Los Gatos
Sierra Technology
Phone: (408) 354-9986

San Diego
Mesa Engineering
Phone: (714) 278-8021

CONNECTICUT
Danbury
Kanan Associates
Phone: (203) 265-2404

FLORIDA
Altamonte Springs
Semtronic Associates
Phone: (305) 831-8233

Clearwater
Semtronic Associates
Phone: (813) 461-4675

ILLINOIS
Elk Grove Village
Micro-Tex, Inc.
Phone: (312) 640-9633

INDIANA
Fort Wayne
Insul-Reps, Inc.
Phone: (219) 482-1596

Indianapolis
Insul-Reps, Inc.
Phone: (317) 842-5203

IOWA
Cedar Rapids
Comstrand Inc.
Phone: (319) 377-1575

KANSAS
Shawnee Mission
B. C. Electronic Sales
Phone: (913) 888-6680

MARYLAND
Baltimore
Micro-Comp, Inc.
Phone: (301) 247-0400

MASSACHUSETTS
Reading
Kanan Associates
Phone: (617) 944-8484

MICHIGAN
Bloomfield Hills
Enco Marketing
Phone: (313) 642-0203

MINNESOTA
Edina
Mel Foster Technical Sales
Phone: (612) 941-9790

MISSOURI
St. Louis
B. C. Electronic Sales
Phone: (314) 731-1255

NEVADA
Sierra Technology
Phone: (408) 354-9986

NEW MEXICO
Albuquerque
Power Enterprises
Phone: (505) 298-1918

NEW YORK
Ithaca
Bob Dean, Inc.
Phone: (607) 272-2187

OHIO
Cleveland
Norm Case Associates
Phone: (216) 333-4120

Dayton
Norm Case Associates
Phone: (800) 362-6631

OKLAHOMA
Tulsa
Cunningham Co.
Phone: (918) 492-0390

OREGON
Hillsboro
Western Technical Sales
Phone: (503) 640-4621

TEXAS
Austin
Cunningham Co.
Phone: (512) 459-8947

Dallas
Cunningham Co.
Phone: (214) 233-4303

Houston
Cunningham Co.
Phone: (713) 461-4197

VIRGINIA
Lynchburg
Micro-Comp, Inc.
Phone: (804) 237-6221

WASHINGTON
Bellevue
Western Technical Sales
Phone: (206) 641-3900

Spokane
Western Technical Sales
Phone: (509) 922-7600

WASHINGTON, D.C.
Call: Micro-Comp, Baltimore
Phone: (301) 247-0400

WISCONSIN
Waukesha
Micro-Tex, Inc.
Phone: (414) 542-5352

DISTRIBUTORS

ALABAMA
Huntsville
Hall-Mark Electronics
Phone: (205) 837-8700
Hamilton/Avnet Electronics
Phone: (205) 837-7210
Pioneer Electronics
Phone: (205) 837-9300

ARIZONA
Phoenix
Hamilton/Avnet Electronics
Phone: (602) 894-9600
Wyle Distribution Group
Phone: (602) 249-2232

CALIFORNIA
Costa Mesa
Avnet Electronics
Phone: (714) 754-6111
Hamilton Electro Sales
Phone: (714) 641-4100

Culver City
Hamilton/Avnet Electronics
Phone: (213) 558-2121
Hamilton/Avnet Electronics
Military
Phone: (213) 558-2901

El Segundo
Wyle Distribution Group
Phone: (213) 322-8100

Irvine
Schweber Electronics
Phone: (714) 556-3880
Wyle Distribution Group
Phone: (714) 979-2125

San Diego
Anthem Electronics
Phone: (714) 279-5200
Hamilton/Avnet Electronics
Phone: (714) 571-7510
Wyle Distribution Group
Phone: (714) 565-9171

San Jose
Anthem Electronics Inc.
Phone: (408) 946-8000

Santa Clara
Schweber Electronics
Phone: (408) 496-0200
Wyle Distribution Group
Phone: (408) 727-2500

Sunnyvale
Arrow Electronics
Phone: (408) 745-6600
Hamilton/Avnet Electronics
Phone: (408) 743-3366

CANADA
Calgary, Alberta
Hamilton/Avnet Electronics
Phone: (403) 230-3586

Downsview, Ontario
Cesco Electronics
Phone: (416) 661-0220

Mississauga, Ontario
Hamilton/Avnet Electronics
Phone: (416) 677-7432
Zentronics
Phone: (416) 451-9600

Montreal, Quebec
Cesco Electronics
Phone: (514) 735-5511
Zentronics
Phone: (514) 735-5361

Ottawa, Ontario
Cesco Electronics
Phone: (613) 729-5118
Hamilton/Avnet Electronics
Phone: (613) 226-1700
Zentronics
Phone: (613) 238-6411

Quebec City
Cesco Electronics
Phone: (418) 524-4641

Ville St. Laurent, Quebec
Hamilton/Avnet Electronics
Phone: (514) 331-6443

COLORADO
Denver
Arrow Electronics
Phone: (303) 758-2100
Wyle Distribution Group
Phone: (303) 457-9953

Englewood
Hamilton/Avnet Electronics
Phone: (303) 740-1000

CONNECTICUT
Danbury
Hamilton/Avnet Electronics
Phone: (203) 797-2800
Schweber Electronics
Phone: (203) 792-3500

Wallingford
Arrow Electronics
Phone: (203) 265-7741

FLORIDA
Ft. Lauderdale
Arrow Electronics
Phone: (305) 776-7790
Hamilton/Avnet Electronics
Phone: (305) 971-2900

Hollywood
Schweber Electronics
Phone: (305) 927-0511

Palm Bay
Arrow Electronics
Phone: (305) 725-1480

St. Petersburg
Hamilton/Avnet Electronics
Phone: (813) 576-3930

GEORGIA
Atlanta
Schweber Electronics
Phone: (404) 449-9170

Norcross
Arrow Electronics
Phone: (404) 449-8252
Hamilton/Avnet Electronics
Phone: (404) 447-7500

ILLINOIS
Chicago
Bell Industries
Phone: (312) 982-9210

Elk Grove
Schweber Electronics
Phone: (312) 364-3750

Schaumburg
Arrow Electronics
Phone: (312) 893-9420

Bensenville
Hamilton/Avnet Electronics
Phone: (312) 860-7700

INDIANA
Indianapolis
Pioneer Electronics
Phone: (317) 849-7300
Arrow Electronics
Phone: (317) 243-9353
Hamilton/Avnet Electronics
Phone: (317) 844-9333

KANSAS
Overland Park
Hamilton/Avnet Electronics
Phone: (913) 888-8900

MARYLAND
Baltimore
Arrow Electronics
Phone: (301) 247-5200

Columbia
Hamilton/Avnet Electronics
Phone: (301) 995-3500

Gaithersburg
Pioneer Washington Electronics
Phone: (301) 948-0710
Schweber Electronics
Phone: (301) 840-5900

MASSACHUSETTS
Bedford
Schweber Electronics
Phone: (617) 275-5100

Burlington
Lionex Corp.
Phone: (617) 272-9400

Woburn
Arrow Electronics
Phone: (617) 933-8130
Hamilton/Avnet Electronics
Phone: (617) 273-7500

MICHIGAN
Ann Arbor
Arrow Electronics
Phone: (313) 971-8220

Grand Rapids
Hamilton/Avnet Electronics
Phone: (616) 243-8805

Livonia
Hamilton/Avnet Electronics
Phone: (313) 522-4700
Pioneer Electronics
Phone: (313) 525-1800
Schweber Electronics
Phone: (313) 525-8100

MINNESOTA
Eden Prairie
Schweber Electronics
Phone: (612) 941-5280

Minneapolis
Arrow Electronics
Phone: (612) 830-1800
Hamilton/Avnet Electronics
Phone: (612) 932-0600

MISSOURI
Earth City
Hamilton/Avnet Electronics
Phone: (314) 344-1200

St. Louis
Arrow Electronics
Phone: (314) 567-6888

NEW HAMPSHIRE
Manchester
Arrow Electronics
Phone: (603) 668-6968

NEW JERSEY
Cherry Hill
Hamilton/Avnet Electronics
Phone: (609) 424-0100

Fairfield
Hamilton/Avnet Electronics
Phone: (201) 575-3390
Schweber Electronics
Phone: (201) 227-7880

Moorestown
Arrow Electronics
Phone: (609) 235-1900

Saddlebrook
Arrow Electronics
Phone: (201) 797-5800

NEW MEXICO
Albuquerque
Hamilton/Avnet Electronics
Phone: (505) 765-1500

NEW YORK
Buffalo
Summit Distributors
Phone: (716) 887-2800

East Syracuse
Arrow Electronics
Phone: (315) 652-1000
Hamilton/Avnet Electronics
Phone: (315) 437-2642

Farmingdale, L.I.
Arrow Electronics
Phone: (516) 694-6800

Liverpool
Arrow Electronics
Phone: (315) 652-1000

Melville
Hamilton/Avnet Electronics
Phone: (516) 454-6012

Rochester
Arrow Electronics
Phone: (716) 275-0300
Hamilton/Avnet Electronics
Phone: (716) 475-9130
Schweber Electronics
Phone: (716) 424-2222

Westbury, L.I.
Schweber Electronics
Phone: (516) 334-7474

NORTH CAROLINA
Greensboro
Pioneer Electronics
Phone: (919) 273-4441

Raleigh
Hamilton/Avnet Electronics
Phone: (919) 829-8030

Winston-Salem
Arrow Electronics
Phone: (919) 725-8711

OHIO
Beechwood
Schweber Electronics
Phone: (216) 464-2970

Cleveland
Hamilton/Avnet Electronics
Phone: (216) 831-3500
Pioneer Electronics
Phone: (216) 587-3600

Centerville
Arrow Electronics
Phone: (513) 435-5563

Dayton
Hamilton/Avnet Electronics
Phone: (513) 433-0610
Pioneer Standard Electronics
Phone: (513) 236-9900

Solon
Arrow Electronics
Phone: (216) 248-3990

OKLAHOMA
Tulsa
Quality Components
Phone: (918) 664-8812

OREGON
Lake Oswego
Hamilton/Avnet Electronics
Phone: (503) 635-8831

PENNSLYVANIA
Horsham
Schweber Electronics
Phone: (215) 441-0600

Pittsburgh
Arrow Electronics
Phone: (412) 856-7000
Pioneer/Pittsburgh
Phone: (412) 782-2300

TEXAS
Austin
Hamilton/Avnet Electronics
Phone: (512) 837-8911
Quality Components
Phone: (512) 835-0220

Dallas
Hall-Mark Electronics
Phone: (214) 341-1147
Hamilton/Avnet Electronics
Phone: (214) 659-4111
Quality Components
Phone: (214) 387-4949
Schweber Electronics
Phone: (214) 661-5010

Houston
Hamilton/Avnet Electronics
Phone: (713) 780-1771
Quality Components
Phone: (713) 772-7100
Schweber Electronics
Phone: (713) 784-3600

UTAH
Salt Lake City
Hamilton/Avnet Electronics
Phone: (801) 972-4300

WASHINGTON
Bellevue
Arrow Electronics
Phone: (206) 643-4800
Hamilton/Avnet Electronics
Phone: (206) 453-5844
Wyle Distribution Group
Phone: (206) 453-8300

WISCONSIN
Oak Creek
Arrow Electronics
Phone: (414) 764-6600

New Berlin
Hamilton/Avnet Electronics
Phone: (414) 784-4510

FOR SIGNETICS PRODUCTS WORLDWIDE:

ARGENTINA
Fapesa I.y.C
Buenos-Aires
Phone: 652-7438/7478

AUSTRALIA
Philips Industries-ELCOMA
Lane-Cove, N.S.W.
Phone: (02) 427-0888

Queensland
Brisbane
(07) 277-3332

South Australia
Adelaide
(08) 45-0211

Victoria
Melbourne
(03) 544-7833

Western Australia
Perth
(09) 277-4199

AUSTRIA
Osterrichische Philips
Wien
Phone: 93 26 22

BELGIUM
M.B.L.E.
Brussels
Phone: 523 00 00

BRAZIL
Ibrape Electronica Ltda.
Sao Paulo
Phone: (011) 211-2600

CHILE
Philips Chilena S.A.
Santiago
Phone: 39-4001

COLOMBIA
Sadape S.A.
Bogota
Phone: 600600

DENMARK
Miniwatt A/S
Kobenhavn
Phone: (01) 69 16 22

FINLAND
Oy Philips Ab
Helsinki
Phone: 1 72 71

FRANCE
R.T.C.
Paris
Phone: 355-44-99

GERMANY
Valvo
Hamburg
Phone: (040) 3296-1

GREECE
Philips S.A. Hellenique
Athens
Phone: 915 311

HONG KONG
Philips Hong Kong, Ltd.
Hong Kong
Phone: 12-245121

INDIA
Peico Electronics & Elect. Ltd.
Elcoma Div.
Bombay
Phone: 295-144

INDONESIA
P.T. Philips-Ralin Electronics
Jakarta
Phone: 581058

IRELAND
Philips Electrical Ltd.
Dublin
Phone: 693355

ISRAEL
Rapac Electronics, Ltd.
Tel Aviv
Phone: 477115-6-7

ITALY
Philips S.p.A.
Milano
Phone: 2-6994

JAPAN
Signetics Japan, Ltd.
Tokyo
Phone: (03) 230-1521

KOREA
Philips Elect Korea Ltd.
Seoul
Phone: 44-4202

MEXICO
Electronica S.A. de C.V.
Mexico D.F.
Phone: 533-1180

NETHERLANDS
Philips Nederland B.V.
Eindhoven
Phone: (040) 79 33 33

NEW ZEALAND
Philips Electrical Ind. ELCOMA
Auckland
Phone: 867119

NORWAY
Electronica A.S.
Oslo
Phone: (02) 15 05 90

PERU
Cadesa
Lima
Phone: 628599

PHILIPPINES
Philips Industrial Dev., Inc.
Makata-Rizal
Phone: 868951-9

PORTUGAL
Philips Portuguesa SARL
Lisbon
Phone: 68 31 21

SINGAPORE/MALAYSIA
Philips Singapore Pte., Ltd.
Singapore
Phone: 538811

SOUTH AFRICA
E.D.A.C. (PTY), Ltd.
Johannesburg
Phone: 24-6701-3

SPAIN
Copresa S.A.
Barcelona
Phone: 329 63 12

SWEDEN
Elcoma A.B.
Stockholm
Phone: 08/67 97 80

SWITZERLAND
Philips A.G.
Zurich
Phone: 01/44 22 11

TAIWAN
Philips Taiwan, Ltd.
Taipei
Phone: (02) 563-1717

THAILAND
Saeng Thong Radio, Ltd.
Bangkok
Phone: 34985/36980

TURKEY
Turk Philips
Ticaret A.S.
Istanbul
Phone: 453250

UNITED KINGDOM
Mullard, Ltd.
London
Phone: 01-580-6633

UNITED STATES
Signetics International Corp.
Sunnyvale, California
Phone: (408) 739-7700

URUGUAY
Luzilectron SA
Montevideo
Phone: 943 21

VENEZUELA
Industrias Venezolanas
Philips S.A.
Caracas
Phone: 239-8180

Printed in U.S.A. August 1981

signetics

a subsidiary of U.S. Philips Corporation

Signetics Corporation
811 East Arques Avenue
P.O. Box 409
Sunnyvale, California 94086
Telephone 408/739-7700

700124B

Printed in U.S.A.

20M1181