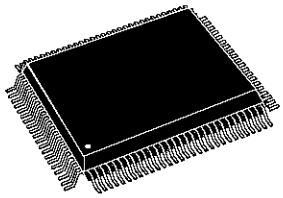
SGS-THOMSON MICROELECTRONICS

ST6280

8-BIT HCMOS MCU WITH DOT MATRIX LCD DRIVER EEPROM AND A/D CONVERTER

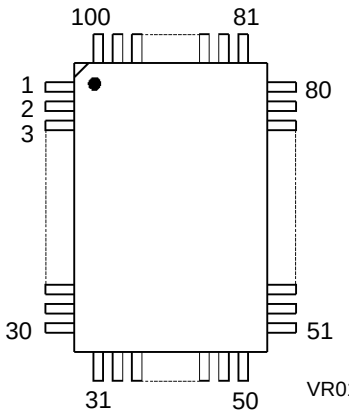
- 4.5 to 6V supply operating range
- 8.4 MHz Maximum Clock Frequency
- -40 to +85°C Operating Temperature Range
- Run, Wait & Stop Modes
- 5 different interrupt vectors
- Look-up table capability in ROM
- User ROM: 7948 bytes
  Data RAM: 192 bytes
  LCD RAM: 128 bytes
  EEPROM: 128 bytes
- PQFP100 package
- 12 fully software programmable I/O as:
  - Input with/without pull-up resistor
  - Input with interrupt generation
  - Open-Drain or Push-pull outputs
  - Analog Inputs
- 10 I/O lines can sink up to 20mA for direct LED or TRIAC driving and have SPI alternate functions
- One 8-bit counter with 7-bit programmable prescaler (Timer 1)
- One 8-bit auto-reload timer with 7-bit programmable prescaler (ARTimer)
- Software activated digital watchdog
- 8-bit A/D converter with up to 12 analog inputs
- 8-bit synchronous serial peripheral interface (SPI)
- LCD driver controller with 48 segments outputs, 8 backplane and 8 software selectable segment/backplane outputs able to drive up to 48x16 (768) or 56x8 (448) segments.
- 32kHz oscillator for stand-by LCD operation
- One external not maskable interrupt
- 9 powerful addressing modes
- The accumulator, the X, Y, V & W registers, the port and peripherals data & control registers are addressed in the data space as RAM locations.
- The ST62E80 is the EPROM version, ST62T80 is the OTP version
- Development tool: ST628x-EMU connected via a standard RS232 to an MS-DOS Personal Computer



PQFP100

(Ordering information at the end of the datasheet)

Figure 1. 100 Pin Quad Flat Pack (QFP) Package Pinout



ST6280 Pin Description

Pin number	Pin name	Pin number	Pin name	Pin number	Pin name	Pin number	Pin name
1	S39	31	PC7/Ain	51	PA3 (1)	81	S19
2	S40	32	PC6/Ain	52	PA2 (1)	82	S20
3	S41	33	PC5/Ain	53	OSC32out	83	S21
4	S42	34	PC4/Ain	54	OSC32in	84	S22
5	S43	35	PC3 (1)	55	COM1	85	S23
6	S44	36	PC2 (1)	56	COM2	86	S24
7	S45	37	PC1 (1)	57	COM3	87	S25
8	S46	38	PC0 (1)	58	COM4	88	S26
9	S47	39	NMI	59	COM5	89	S27
10	S48	40	V_{DD}	60	COM6	90	S28
11	S49	41	V_{SS}	61	COM7	91	S29
12	S50	42	VLCD	62	COM8	92	S30
13	S51	43	VLCD4/5	63	COM9/S1	93	S31
14	S52	44	VLCD3/5	64	COM10/S2	94	S32
15	S53	45	VLCD2/5	65	COM11/S3	95	S33
16	S54	46	VLCD1/5	66	COM12/S4	96	S34
17	S55	47	PA7/Sout (1)	67	COM13/S5	97	S35
18	S56	48	PA6/Sin (1)	68	COM14/S6	98	S36
19	PB7/ARTIMout /Ain	49	PA5/SCL (1)	69	COM15/S7	99	S37
		50	PA4/TIM1 (1)	70	COM16/S8	100	S38
20	PB6/ARTIMin /Ain			71	S9		
				72	S10		
21	PB5/Ain			73	S11		
22	PB4/Ain			74	S12		
23	PB3/Ain			75	S13		
24	PB2/Ain			76	S14		
25	PB1/Ain			77	S15		
26	PB0/Ain			78	S16		
27	TEST			79	S17		
28	OSCout			80	S18		
29	OSCin						
30	$\overline{\text{RESET}}$						

Note 1: 20mA SINK

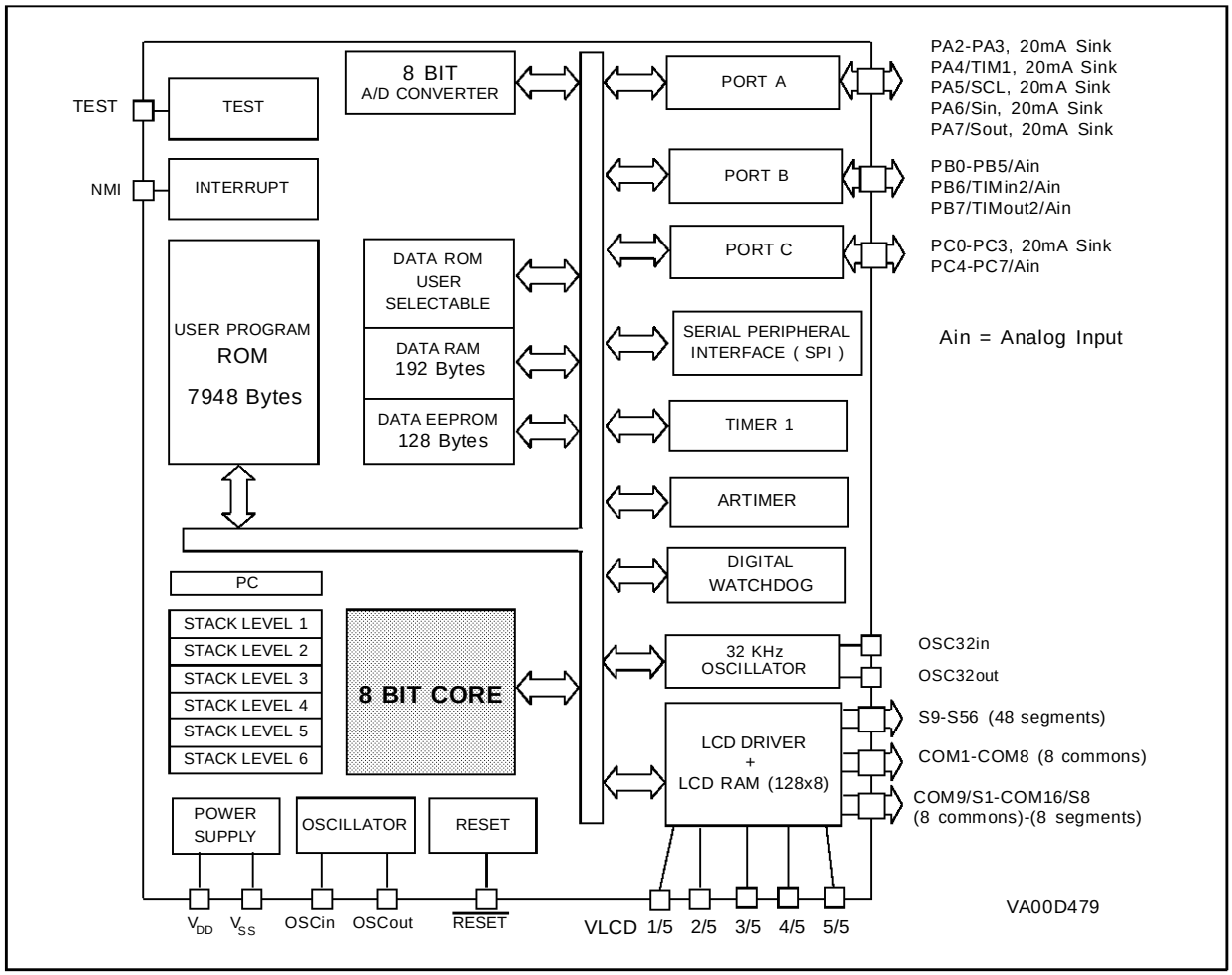
GENERAL DESCRIPTION

The ST6280 microcontrollers is a member of the 8-bit HCMOS ST62xx family, a series of devices oriented to low-medium complexity applications. All ST62xx members are based on a building block approach:

a common core is associated with a combination of on-chip peripherals (macrocells). The macrocells of the ST6280 are an advanced LCD driver/controller with 48 segments, 8 backplanes and 8 software selectable segment/backplane outputs able to drive up to 48 x16 (768) or 56 x 8 (448) segments, one 8 bit Autoreload timer with 7 bit programmable prescaler (ARTimer), one 8 bit standard timer/counter with a 7-bit software programmable prescaler (Timer 1), the digital watchdog (DWD), an 8-bit A/D Converter with up to 12 analog inputs, a 32kHz Oscillator, and an 8-bit synchronous serial peripheral interface (SPI). In addition this device offers 128 bytes of EEPROM for storage of non-volatile data.

Thanks to these peripherals the ST6280 is well suited for general purpose, automotive, security, appliance and industrial applications. The ST62E80 EPROM version is available for prototyping and low-volume production, an OTP version is also available (see separate datasheet).

Figure 2. ST6280 Block Diagram



PIN DESCRIPTION

V_{DD} and V_{SS}. Power is supplied to the MCU using these two pins. V_{DD} is power and V_{SS} is the ground connection.

OSCin and OSCout. These pins are internally connected with the on-chip oscillator circuit. A quartz crystal or a ceramic resonator can be connected between these two pins in order to allow the correct operation of the MCU with various stability/cost trade-offs. OSCin is the input pin, OSCout is the output pin. An external clock signal can be applied to OSCin.

$\overline{\text{RESET}}$. The active low $\overline{\text{RESET}}$ pin is used to restart the microcontroller at the beginning of its program. The $\overline{\text{RESET}}$ pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

TEST. TEST must be held at V_{SS} for normal operation (an internal pull-down resistor selects normal operating mode if TEST pin is not connected).

NMI. The NMI pin provides the capability for asynchronous applying an external top priority interrupt to the MCU. This pin is falling edge sensitive. The NMI pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

PA4/TIM1. This pin can be used as Timer 1 I/O pin. In input mode it is connected to the timer prescaler input and acts as the external timer clock or as the control gate for the internal timer clock. In the output mode the timer pin outputs the timer data bit when a time out occurs.
To use this pin as Timer output the I/O pin has to be programmed as open-drain output. To use this pin as Timer input the I/O pin has to be programmed as input.

PB6/TIMIN2, PB7/TIMOUT2. These pins are the Input and Output pins of the Autoreload Timer. The timer input pin ARTIMin is connected to port line PB6. To use the line as timer input function, PB6 has to be programmed as input with or without pull-up. The timer output pin is connected to the port line PB7. A dedicated bit in the ARTIMER mode control register sets the line as timer output function ARTIMout.

PA2-PA7. These 6 lines are organized as one I/O port (A). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. In output mode these lines can also sink 20mA for direct LED or triac driving. PA5-PA7 can also be used as respectively Clock, Data in and Data out pins for the on-chip SPI to carry the synchronous serial I/O signals. PA4 can also be used as the TIMER 1 I/O pin.

PB0-PB7. These 8 lines are organized as one I/O port (B). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, open-drain or push-pull output or as an analog input for the A/D converter. PB6 is also connected to the ARTIMER input function while PB7 can act as the ARTIMER output.

PC0-PC3, PC4-PC7. These 8 lines are organized as one I/O port (C). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. PC0-PC3 can also sink 20mA for direct LED or triac driving while PC4-PC7 can be programmed as analog inputs for the A/D converter.

COM1-COM8. These eight pins are the LCD peripheral common outputs. They are the outputs of the on-chip backplane voltage generator which is used for multiplexing the LCD segment lines.

S9-S56. These pins are the 48 LCD peripheral driver outputs of the ST6280. Segments S1-S8 are multiplexed with COM9-COM16 and their function is software selectable.

COM9/S1-COM16/S8. These pins are the 8 multiplexed common/segment lines. Under software selected control, they can act as LCD common outputs allowing a 48 x 16 dot matrix operation, or they can act as segment outputs allowing 56 x 8 dot matrix operation.

VLCD1/5-VLCD5/5. Resistor network nodes for determining the intermediate LCD voltage levels on COM1-COM8/COM16 and S1/S8-S56 pins during multiplex operation.

OSC32in and OSC32out. These pins are internally connected with the on-chip 32kHz oscillator circuit. A 32.768kHz quartz crystal can be connected between these two pins if it is necessary to provide the LCD stand-by clock and real time interrupt. OSC32in is the input pin, OSC32out is the output pin.

ST62xx CORE

The Core of the ST62xx Family is implemented independently from the I/O or memory configuration. Consequently, it can be treated as an independent central processor communicating with I/O and memory via internal addresses, data, and control busses. The in-core communication is arranged as shown in Figure 3; the controller being externally linked to both the reset and the oscillator, while the core is linked to the dedicated on-chip macrocells peripherals via the serial data bus and indirectly for interrupt purposes through the control registers.

Registers

The ST62xx Family Core has six registers and three pairs of flags available to the programmer. They are shown in Figure 4 and are explained in the following paragraphs.

Accumulator (A). The accumulator is an 8-bit general purpose register used in all arithmetic calculations, logical operations, and data manipulations. The accumulator is addressed in the data space as RAM location at address FFh. Accordingly, the ST62xx instruction set can use the accumulator as any other register of the data space.

Figure 4. ST62xx Core Programming Model

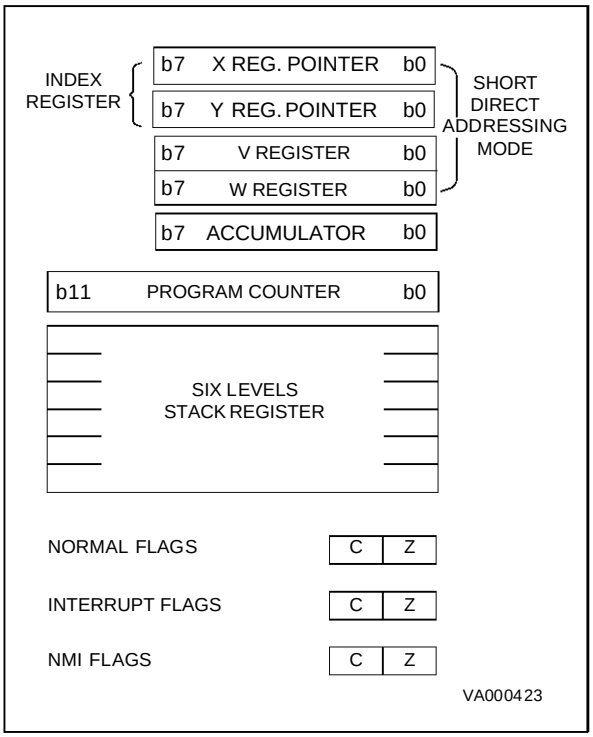
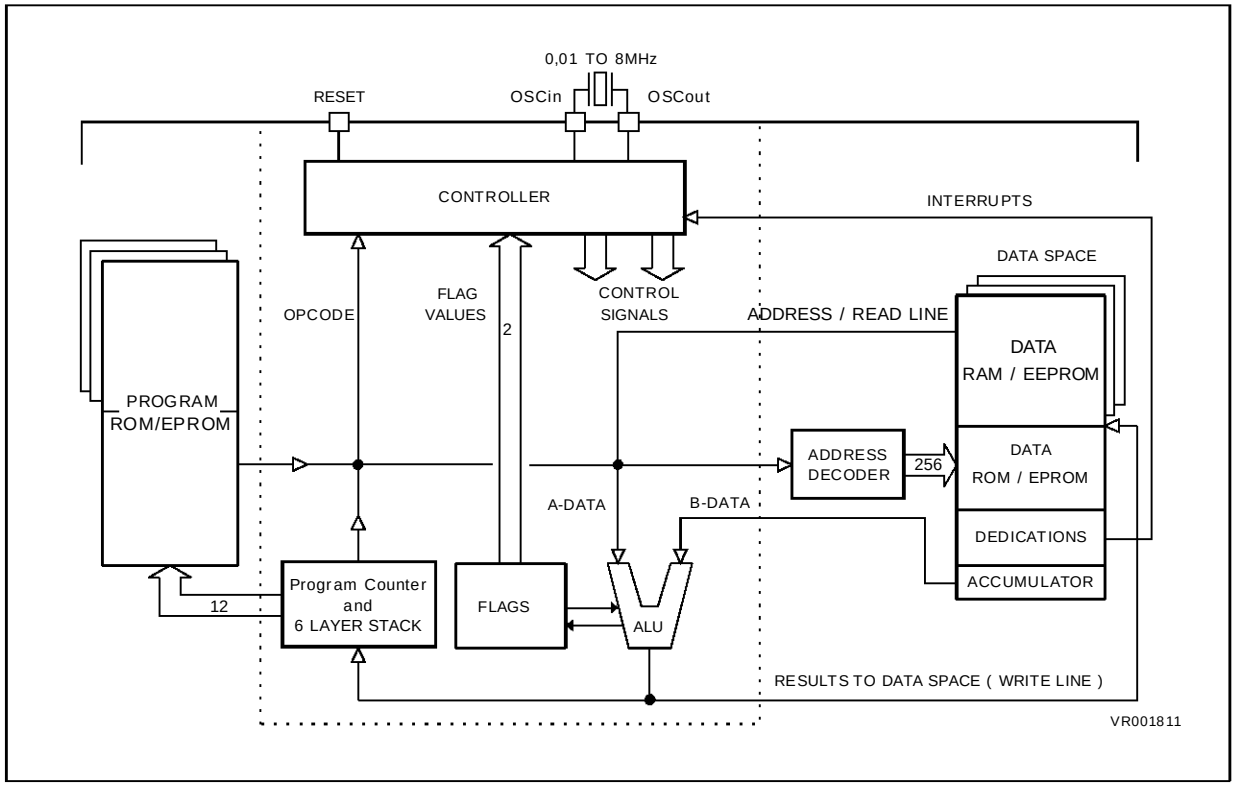


Figure 3. ST62xx Core Block Diagram



ST62xx CORE (Continued)

Indirect Registers (X, Y). These two indirect registers are used as pointers to memory locations in the data space. They are used in the register-indirect addressing mode. These registers can be addressed in the data space as RAM locations at 80h (X) and 81h (Y) addresses. They can also be accessed with the direct, short direct, or bit direct addressing modes. Accordingly, the ST62xx instruction set can use the indirect registers as any other register of the data space.

Short Direct Registers (V, W). These two registers are used to save one byte in short direct addressing mode. These registers can be addressed in the data space as RAM locations at 82h (V) and 83h (W) addresses. They can also be accessed with the direct and bit direct addressing modes. Accordingly, the ST62xx instruction set can use the short direct registers as any other register of the data space.

Program Counter (PC)

The program counter is a 12-bit register that contains the address of the next ROM location to be processed by the Core. This ROM location may be an opcode, an operand, or an address of operand. The 12-bit length allows the direct addressing of 4096 bytes in the program space. Nevertheless, if the program space contains more than 4096 locations, as for the ST6280, the further program space can be addressed by using the Program ROM Page register.
The PC value is incremented after it is read from the address of the current instruction. To execute relative jumps the PC and the offset are shifted through the ALU, where they will be added, and the result is shifted back into the PC. The program counter can be changed in the following ways:

- JP (Jump) instruction..... PC=Jump address
- CALL instruction ..... PC=Call address
- Relative Branch instructions ..... PC=PC ± offset
- Interrupt..... PC=Interrupt vector
- Reset..... PC=Reset vector
- RET & RETI instructions PC=Pop (stack)
- Normal instruction ..... PC=PC+1

Flags (C, Z)

The ST62xx Core includes three pairs of flags that correspond to 3 different modes: normal mode, interrupt mode and Non-Maskable Interrupt-Mode. Each pair consists of a CARRY flag and a ZERO flag. One pair (CN, ZN) is used during normal operation, one pair is used during the interrupt mode (CI, ZI) and one is used during the not-maskable interrupt mode (CNMI, ZNMI).

The ST62xx Core uses the pair of flags that correspond to the actual mode: as soon as an interrupt (resp. a Non-Maskable-Interrupt) is generated, the ST62xx Core uses the interrupt flags (resp. the NMI flags) instead of the normal flags. When the RETI instruction is executed, the normal flags (resp. the interrupt flags) are restored if the MCU was in the normal mode (resp. in the interrupt mode) before the interrupt. It should be observed that each flag set can only be addressed in its own mode (Not-maskable interrupt, normal interrupt or main mode). The flags are not cleared during the context switching and so remain in the state they were at the exit of the last mode switch.

The Carry flag is set when a carry or a borrow occurs during arithmetic operations, otherwise it is cleared. The Carry flag is also set to the value of the bit tested in a bit test instruction, and participates in the rotate left instruction.

The Zero flag is set if the result of the last arithmetic or logical operation was equal to zero, otherwise it is cleared.

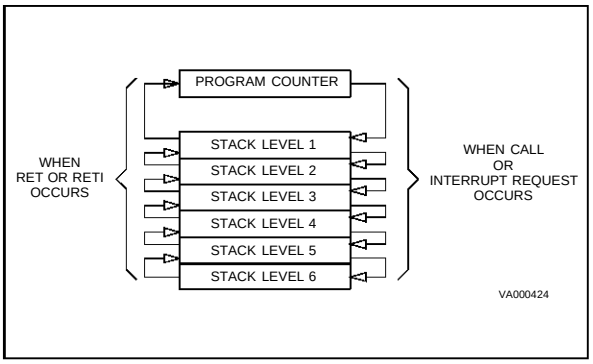
The switching between the three sets of Flags is automatically performed when an NMI, an interrupt or a RETI instruction occurs. As the NMI mode is automatically selected after the reset of the MCU, the ST62xx Core uses at first the NMI flags.

ST62xx CORE (Continued)

Stack

The ST62xx Core includes a true LIFO hardware stack that eliminates the need for a stack pointer. The stack consists of six separate 12-bit RAM locations that do not belong to the data space RAM area. When a subroutine call (or interrupt request) occurs, the contents of each level is shifted into the next level while the content of the PC is shifted into the first level (the value of the sixth level will be lost). When subroutine or interrupt return occurs (RET or RETI instructions), the first level register is shifted back into the PC and the value of each level is popped back into the previous level. These two operating modes are described in Figure 5. Since the accumulator, as all other data space registers, is not stored in the stack the handling of these registers should be performed inside the subroutine. The stack pointer will remain in its deepest position if more than 6 calls or interrupts are executed, so that the last return address will be lost. It will also remain in its highest position if the stack is empty and a RET or RETI is executed. In this case the next instruction will be executed.

Figure 5. Stack Operation



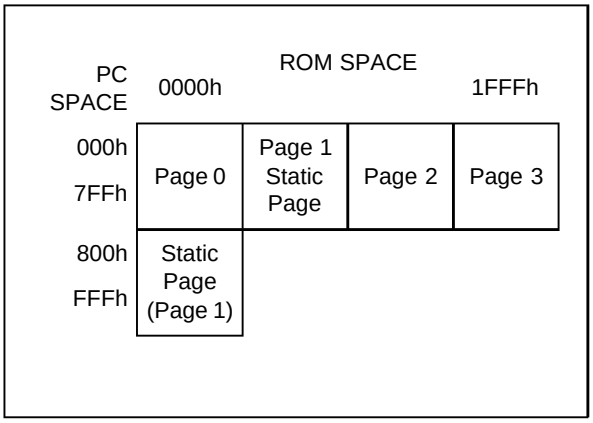
MEMORY SPACES

The MCU operates in three different memory spaces: program space, data space, and stack space. A description of these spaces is shown in the following figures.

Program Space

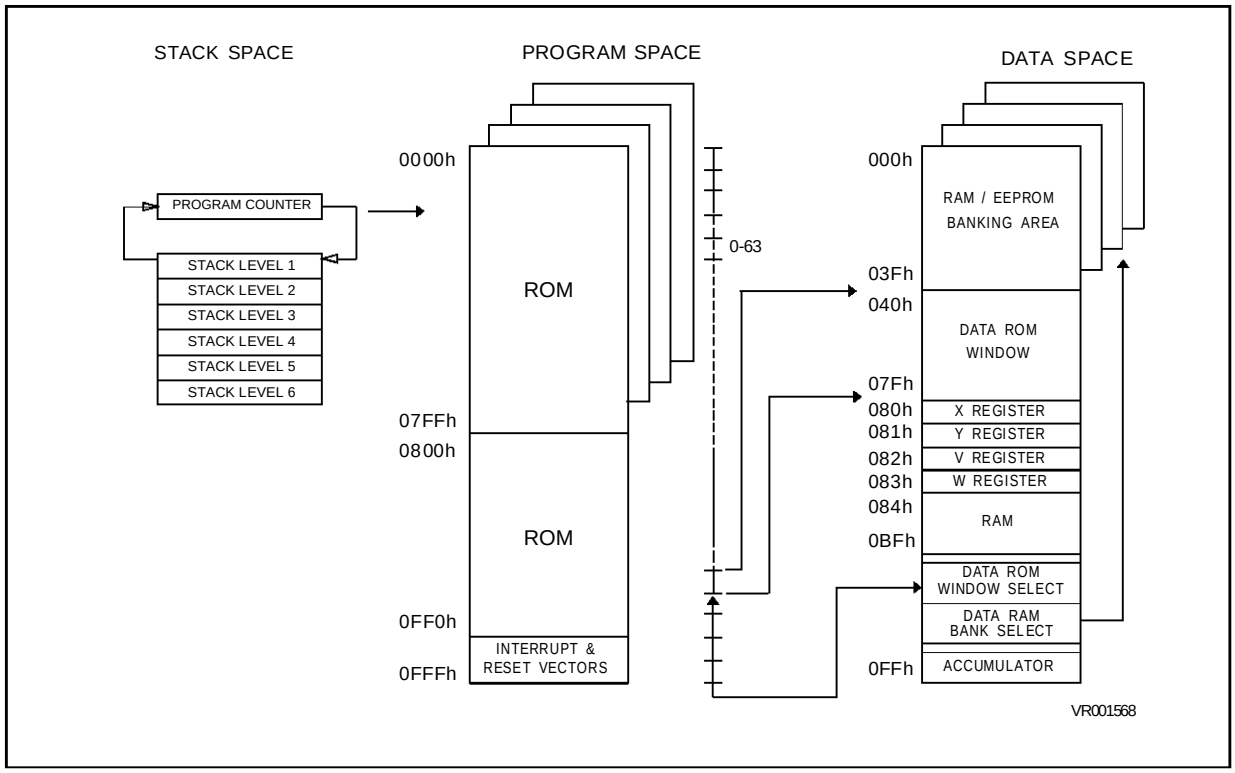
The program space is physically implemented in the ROM memory and includes all the instructions that are to be executed, as well as the data required for the immediate addressing mode instructions, the reserved test area and the user vectors. It is addressed by the 12-bit Program Counter register (PC register) and so the ST62xx Core can directly address up to 4Kbytes of Program Space. Nevertheless, the Program Space can be extended by the addition of 2Kbyte ROM banks as it is shown in the following figure in which the ST6280 8Kbyte memory is described.

Figure 6. Program Space Addressing Description



MEMORY SPACES (Continued)

Figure 7. ST62xx Memory Addressing Description Diagram



These banks are addressed in locations 00h to 7FFh of the Program Space by the Program Counter and by writing the appropriate code in the Program ROM Page Register (PRPR register) located at address CAh of the Data Space. Because interrupts and common subroutines should be available all the time, only the lower 2K bytes of the 4K program space are bank switched while the upper 2K can be seen as a static page. Table 2 gives the different codes that allow the selection of the corresponding banks. Note that, from the memory point of view, Page 1 and the Static Page represent the same physical memory: it is only two different ways of addressing the same locations. On the ST6280, a total of 8192 bytes of ROM have been implemented; 7948 are available as user ROM while 244 are reserved for SGS-THOMSON test purposes.

Table 1. ST6280 Program ROM Memory Map

ROM Page	Device Address	Description
Page 0	0000h-007Fh 0080h-07FFh	Reserved User ROM
Page 1 “STATIC”	0800h-0F9Fh 0FA0h-0FEFh 0FF0h-0FF7h 0FF8h-0FFBh 0FFCh-0FFDh 0FFEh-0FFFh	User ROM Reserved Interrupt Vectors Reserved NMI Vector Reset Vector
Page 2	0000h-000Fh 0010h-07FFh	Reserved User ROM
Page 3	0000h-000Fh 0010h-07FFh	Reserved User ROM

MEMORY SPACES (Continued)

Data Space

The instruction set of the ST62xx core operates on a specific space, named Data Space, that contains all the data necessary for the processing of the program. The Data Space allows the addressing of RAM memory, ST62xx core/peripheral registers, and read-only data such as constants and look-up tables.

Data ROM. All the read-only data is physically implemented in the ROM memory in which the Program Space is also implemented. The ROM memory contains consequently the program to be executed, the constants and the look-up tables needed for the program.

The locations of Data Space in which the different constants and look-up tables are addressed by the ST62xx core can be considered as being a 64-byte window through which it is possible to access to the read-only data stored in the ROM memory **.**

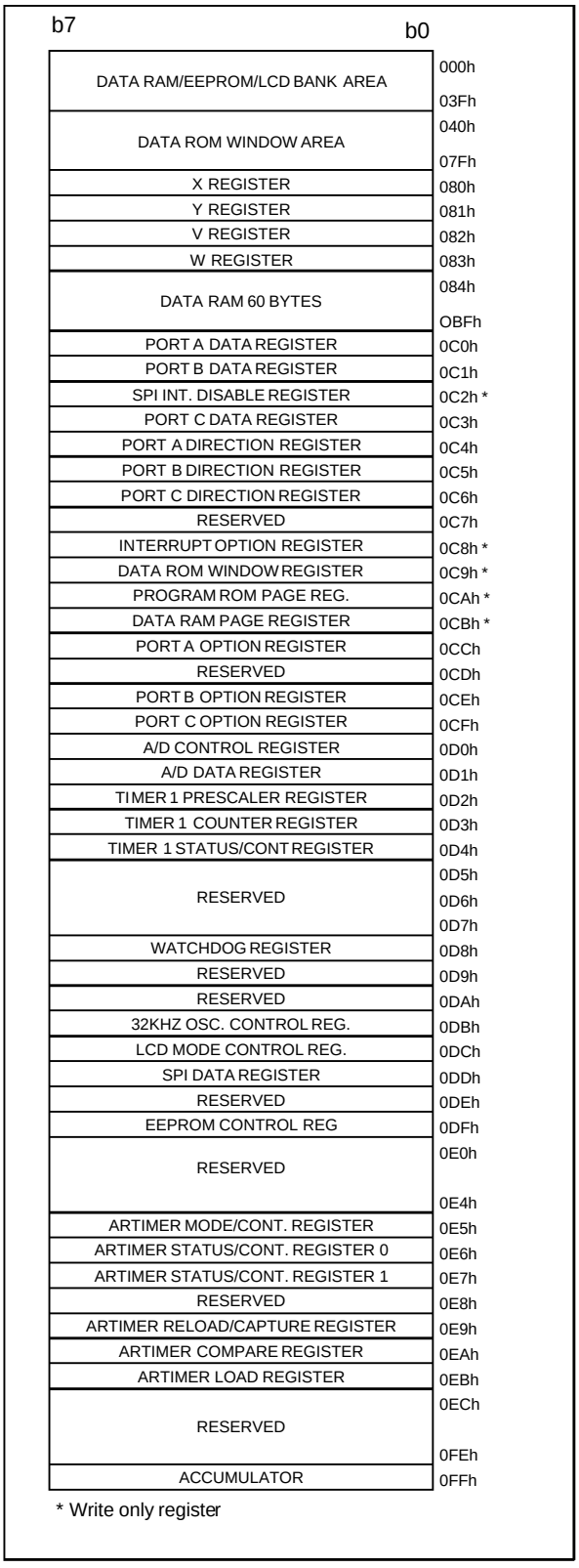
Data RAM/EEPROM. The ST6280 offers 192 bytes of data RAM memory and 128 bytes of EEPROM. 64 bytes of RAM are directly addressed in data space in the range 080h-0BFh (static space). The additional RAM and the EEPROM are addressed using the banks of 64 bytes located between addresses 00h and 3Fh.

Additionally banked RAM is available in the LCD data map.

Stack Space

The stack space consists of six 12 bit registers that are used for stacking subroutine and interrupt return addresses plus the current program counter register.

Figure 8. ST6280 Data Memory Space



MEMORY SPACES (Continued)

Program ROM Page Register (PRPR)

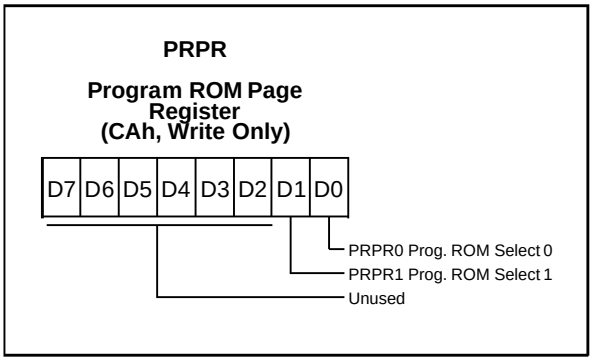
The PRPR register can be addressed like a RAM location in the Data Space at the address CAh. Nevertheless it is a write only register that cannot be accessed with single-bit operations. This register is used to select the 2-Kbyte ROM bank of the Program Space that will be addressed. The number of the page has to be loaded in the PRPR register. Refer to the Program Space description for additional information concerning the use of this register. The PRPR register is not modified when an interrupt or a subroutine occurs.

Care is required when handling the PRPR register as it is write only. For this reason, it is not allowed to change the PRPR contents while executing interrupt service routine, as the service routine cannot save and then restore its previous content. This operation may be necessary if common routines and interrupt service routines take more than 2K bytes; in this case it could be necessary to divide the interrupt service routine into a (minor) part in the static page (start and end) and to a second (major) part in one of the dynamic pages. If it is impossible to avoid the writing of this register in interrupt service routines, an image of this register must be saved in a RAM location, and each time the program writes to the PRPR it must write also to the image register. The image register must be written before PRPR, so if an interrupt occurs between the two instructions the PRPR is not affected.

Note:

Only the lower part of address space is bank switched because interrupt vectors and common subroutines should be available at all times. This structure is due to the fact that it is not possible to jump from one dynamic page to another except by jumping back to the static page, changing contents of PRPR, and then jumping to a different dynamic page.

Figure 9. Program ROM Page Register



D7-D2. These bits are not used.

PRPR1-PRPR0. These are the program ROM banking bits and the value loaded selects the corresponding page to be addressed in the lower part of the 4K program address space as specified in Table 2.

Table 2. Program ROM Page Register Coding

PRPR1	PRPR0	PC bit 11	Memory Page
X	X	1	Static Page (Page 1)
0	0	0	Page 0
0	1	0	Page 1 (Static Page)
1	0	0	Page 2
1	1	0	Page 3

This register is undefined on reset. Neither read nor single bit instructions may be used to address this register.

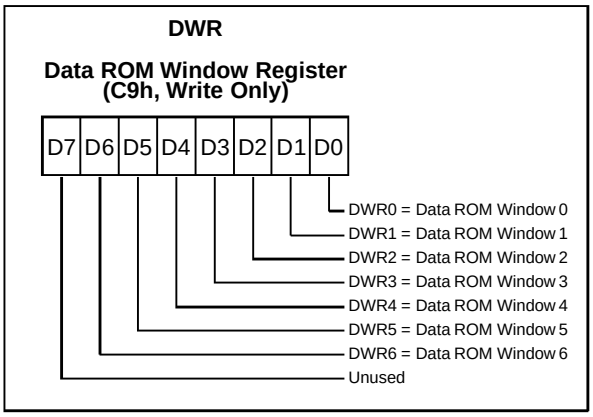
MEMORY SPACES (Continued)

Data Window register (DWR)

The Data ROM window is located from address 040h to address 7Fh in the Data space. It allows the direct reading of 64 consecutive bytes located anywhere in the ROM memory between the addresses 0000h and 1FFFh. All the bytes of the ROM memory can be used to store either instructions or read-only data. Indeed, the window can be moved by step of 64 bytes along the ROM memory in writing the appropriate code in the Write-only Data Window register (DWR register, location C9h).

The DWR register can be addressed like a RAM location in the Data Space at the address C9h, nevertheless it is a write only register that cannot be accessed with single-bit operations. This register is used to move the 64-byte read-only data window (from the 40h address to 7Fh address of the Data Space) up and down the ROM memory of the MCU in steps of 64 bytes. The effective address of the byte to be read as a data in the ROM memory is obtained by the concatenation of the 6 least significant bits of the register address given in the instruction (as least significant bits) and the content of the DWR register (as most significant bits, see Figure 10). So when addressing location 40h of dataspace, and 0 is loaded in the DWR register, the phisycal addressed location in ROM is 00h. The DWR register is not cleared at reset, therefore it must be written to before the first access to the Data ROM window area.

Figure 11. Data ROM Window Register



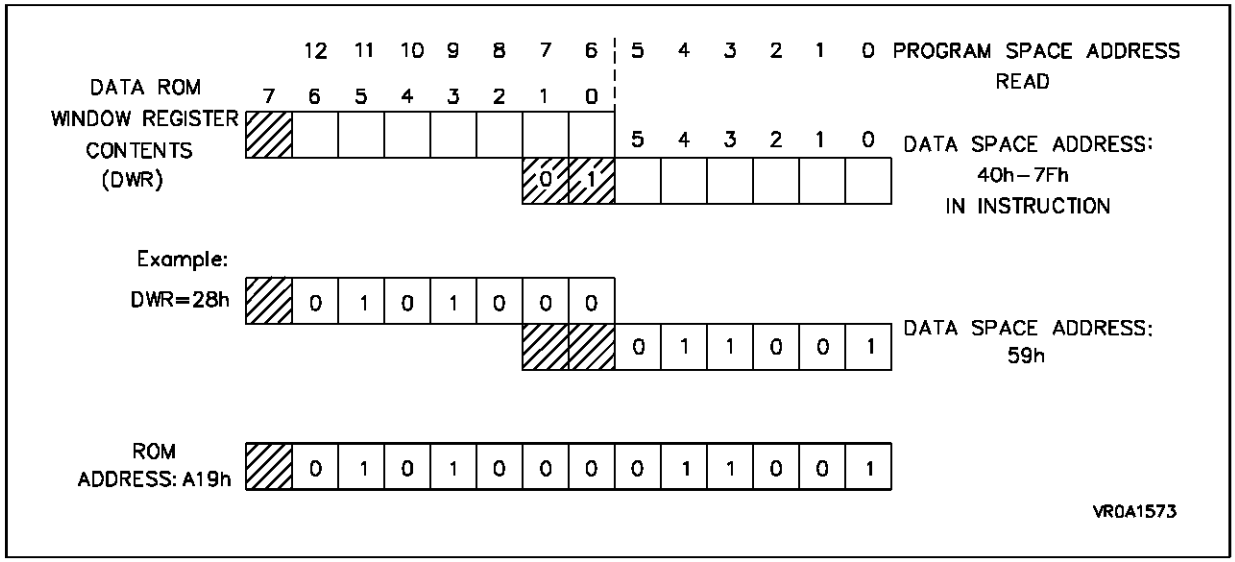
D7. This bit is not used.

DWR6-DWR0. These are the Data ROM Window bits that correspond to the upper bits of the data ROM space.

This register is undefined on reset. Neither read nor single bit instructions may be used to address this register.

Note: Care is required when handling the DWR register as it is write only. For this reason, it is not allowed to change the DWR contents while executing interrupt service routine, as the service routine cannot save and then restore its previous content. If it is impossible to avoid the writing of this register in the interrupt service routine, an image of this register must be saved in a RAM location, and each time the program writes to the DWR it must write also to the image register. The image register must be written first, so if an interrupt occurs between the two instructions the DWR is not affected.

Figure 10. Data ROM Window Memory Addressing



MEMORY SPACES (Continued)

Data RAM/EEPROM/LCD RAM Bank Register (DRBR)

The selection of the bank is made by programming the Data RAM Bank Switch register (DRBR register) located at address CBh of the Data Space. The number of the selected bank is equal to the bit content of the DRBR register. In this way each bank of RAM/LCD RAM or EEPROM can be selected 64 bytes at a time. No more than one bank should be set at a time.

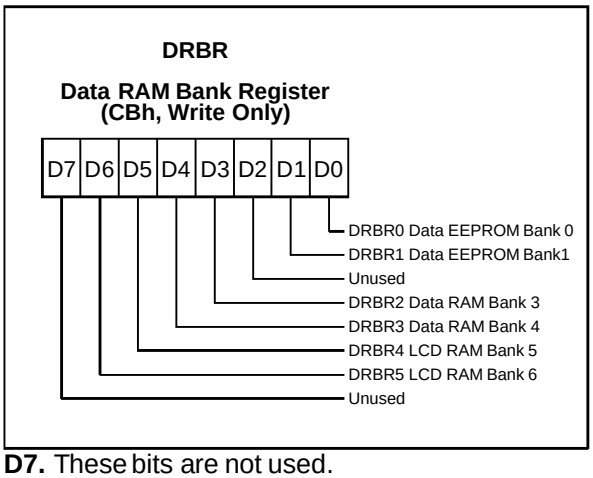
The DRBR register can be addressed like a RAM location in the Data Space at the address CBh; nevertheless it is a write only register that cannot be accessed with single-bit operations. This register is used to select the desired 64-byte RAM/EEPROM/LCD RAM bank of the Data Space. The number of the bank has to be loaded in the DRBR register and the instruction has to point to the selected location as if it was in bank 0 (from 00h address to 3Fh address). This register is not cleared during the MCU initialization, therefore it must be written before the first access to the Data Space bank region. Refer to the Data Space description for additional information. The DRBR register is not modified when a interrupt or a subroutine occurs.

The following table summarizes how to set the data RAM bank register in order to select the various banks or pages.

Table 3. Data RAM Bank Register Set-up

DRBR Value	Selection
01h	EEPROM Page 0
02h	EEPROM Page 1
08h	RAM Page 1
10h	RAM Page 2
20h	LCD RAM Page 1
40h	LCD RAM Page 2

Figure 12. Data RAM Bank Register



D7. These bits are not used.

DRBR6-DRBR5. Each of these bits, when set, will select one LCD RAM page.

DRBR4-DRBR3. Each of these bits, when set, will select one RAM page.

D2. This bit is not used.

DRBR1-DRBR0. Each of these bits, when set, will select one EEPROM page.

This register is undefined on reset. Neither read nor single bit instructions may be used to address this register.

Notes:

Care is required when handling the DRBR register as it is write only. For this reason, it is not allowed to change the DRBR contents while executing interrupt service routine, as the service routine cannot save and then restore its previous content. If it is impossible to avoid the writing of this register in interrupt service routine, an image of this register must be saved in a RAM location, and each time the program writes to DRBR it must write also to the image register. The image register must be written first, so if an interrupt occurs between the two instructions the DRBR is not affected.

In DRBR Register, *only 1 bit must be set*. Otherwise two or more pages are enabled in parallel, producing errors.

MEMORY SPACES (Continued)

EEPROM Description

The data space of ST62xx family from 00h to 3Fh is paged as described in Table 3. The ST6280 has 128 bytes of EEPROM located in two pages of 64 bytes (pages 0 and 1).

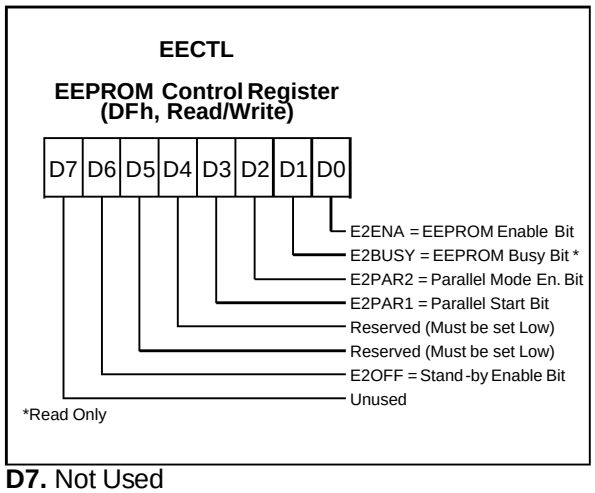
The EEPROM pages are physically organized in 32 byte modules (2 modules per page) and do not require dedicated instructions to be accessed in reading or writing. The EEPROM is controlled by the EEPROM Control Register (EECTL = DFh). In order to enable access to the EEPROM, bit 6 of this register must be cleared otherwise any access to the EEPROM will be meaningless.

Any EEPROM location can be read just like any other data location.

When writing to an EEPROM, the EEPROM is not accessible by the ST62xx. A busy flag can be read to identify the EEPROM status before attempting any access. Writing the EEPROM can work in two modes: Byte Mode (BMODE) and Parallel Mode (PMODE). BMODE is the normal way to use the EEPROM and consists in accessing one byte at a time. PMODE consists in accessing 8 bytes per time.

Reading the EEPROM is made at the same speed as RAM access.

Figure 13. EEPROM Control Register



D7. Not Used

E2OFF. *WRITE ONLY.* If this bit is set the EEPROM is disabled (any access will be meaningless) and the power consumption of the EEPROM is reduced to the lowest values.

D5, D4. Reserved, must be set to zero.

E2PAR1. *WRITE ONLY.* Once in Parallel Mode, as soon as the user software sets the E2PAR1 bit the parallel writing of the 8 adjacent registers will start. It is internally reset at the end of the programming procedure. Note that less than 8 bytes can be written; after parallel programming the undefined bytes will be unaffected

E2PAR2. *WRITE ONLY.* This bit must be set by the user program in order to perform parallel programming (more than one byte at a time). If E2PAR2 is set and the parallel start bit (E2PAR1) is low, up to 8 adjacent bytes can be written at maximum speed, the contents being stored in volatile registers. These 8 adjacent bytes are considered as a row, whose address lines A7, A6, A5, A4, A3 are fixed while A2, A1 and A0 are the changing bits. E2PAR2 is automatically reset at the end of any parallel programming procedure. It can be reset by the user software before starting the programming procedure, leaving the EEPROM registers unchanged.

E2BUSY. *READ ONLY*. This bit will be automatically set by the EEPROM control logic when the user program modifies an EEPROM register. The user program must test it before any read or write EEPROM operation; any attempt to access the EEPROM while the busy bit is set will be aborted and the writing procedure in progress completed.

E2ENA. *WRITE ONLY.* This bit MUST be set to one in order to write to any EEPROM register. If the user program attempts to write to the EEPROM when E2ENA = “0”, the involved registers will be unaffected and the BS will not be set.

After RESET the content of EECTL register will be 00h. No single bit instruction is allowed.

Notes:

a - The data to write has to be written directly at the address that it will have inside the EEPROM space. There is no buffer memory between the data RAM and the EEPROM spaces.

b - When the EEPROM is busy (E2BUSY = “1”) EECTL can not be accessed in write mode, it is only possible to read the status of E2BUSY. This implies that as long as the EEPROM is busy, it is not possible to change the status of the EEPROM control register. EECTL bits 4 and 5 are reserved for test purposes, and must be set to “0”.

c - E2ENA should be kept at zero when the EEPROM is not written. That is especially important to avoid corruption of the EEPROM content after reset and before turning-off the supply or the device.

d - The parts are delivered with the EEPROM space written at 0FFh.

e - Some bit of the EECTL register are write only. So take care not to use single bit instructions on it.

MEMORY SPACES (Continued)

Table 4. EEPROM Parallel Write Row Structure

Byte	0	1	2	3	4	5	6	7	Dataspace addresses. Banks 0 and 1.
ROW7									38h-3Fh
ROW6									30h-37h
ROW5									28h-2Fh
ROW4									20h-27h
ROW3									18h-1Fh
ROW2									10h-17h
ROW1									08h-0Fh
ROW0									00h-07h

Up to 8 bytes in each row may be programmed at the same time in Parallel Write mode

Additional Notes on Parallel Mode. If the user wishes to perform parallel programming, the first action should be to set the E2PAR2 bit to one. From this time the EEPROM will be addressed in writing, the ROW address will be latched and it will be possible to change it only at the end of the programming procedure or by resetting E2PAR2 without programming the EEPROM. After the ROW address latching the ST62xx can “see” only one EEPROM row (the selected one) and any attempt to write or read other rows will produce errors. Do not read the EEPROM while E2PAR2 is set.

As soon as E2PAR2 bit is set, the 8 volatile ROW latches are cleared. From this moment the user can load data in the whole ROW or in a subset. Setting E2PAR1 will modify the EEPROM registers corresponding to the ROW latches accessed after E2PAR2. For example, if the software sets E2PAR2 and accesses the EEPROM by writing to addresses 18h, 1Ah, 1Bh and then sets E2PAR1, these three registers will be modified at the same time; the remaining bytes will be unaffected. Note that E2PAR2 is internally reset at the end of the programming procedure. This implies that the user must set E2PAR2 bit between two parallel programming procedures. Note that if the user tries to set E2PAR1 while E2PAR2 is not set there will not be any programming procedure and the E2PAR1 bit will be unaffected. Consequently E2PAR1 bit cannot be set if E2ENA is low. E2PAR1 can be affected by the user to set it, only if E2ENA and E2PAR2 bits are also set to one.

TEST MODE

For normal operation the TEST pin must be held low when RESET is active. An on-chip 100kΩ pull-down resistor is internally connected to the TEST pin.

INTERRUPTS

The ST62xx core can manage 4 different maskable interrupt sources, plus one non-maskable interrupt source (top priority level interrupt). Each source is associated with a particular interrupt vector that contains a Jump instruction to the related interrupt service routine. Each vector is located in the Program Space at a particular address.
When a source provides an interrupt request, and the request processing is also enabled by the ST62xx core, then the PC register is loaded with the address of the interrupt vector (i.e. of the Jump instruction). Finally, the PC is loaded with the address of the Jump instruction and the interrupt routine is processed.

The ST6280 microcontroller has nine different interrupt sources associated to different interrupt vectors as described in Table 5.

Table 5. Interrupt Vectors - Sources Relationship

Interrupt Source	Associated Vector	Vector Address
NMI Pin	Interrupt Vector #0 (NMI)	(FFCh-FFDh)
SPI Peripheral	Interrupt Vector #1	(FF6h-FF7h)
Port A,B,C Pins	Interrupt Vector #2	(FF4h-FF5h)
TIMER 1, ARTIMER and 32kHz Oscillator	Interrupt Vector #3	(FF2h-FF3h)
ADC Peripheral	Interrupt Vector #4	(FF0h-FF1h)

Interrupt Vectors Description

The ST62xx core includes 5 different interrupt vectors in order to branch to 5 different interrupt routines in the static page of the Program Space:

- The interrupt vector associated with the non-maskable interrupt source is named interrupt vector #0. It is located at addresses FFCh, FFDh in the Program Space. On ST6280 this vector is associated with the external falling edge sensitive interrupt pin (NMI). An on-chip 100kΩ pull-up resistor is internally connected to the NMI pin.
- The interrupt vector located at the addresses FF6h, FF7h is named interrupt vector #1. It is associated with SPI peripheral and can be programmed by software to generate an interrupt request after the falling edge or low level of the eighth external clock pulse according to the code loaded in the Interrupt Option Register (IOR).
- The interrupt vector located at the addresses FF4h, FF5h is named interrupt vector #2. It is associated with Port A, B and C pins and can be programmed by software either in the falling edge detection mode or in the rising edge detection mode according to the code loaded in the Interrupt Option Register (IOR).
- The interrupt vector located at the addresses FF2h, FF3h is named interrupt vector #3. It is associated with Timer 1, ARTimer and the 32kHz Oscillator peripherals. All these interrupts are “ORed” together and are connected to interrupt line #3 of the core. Discrimination among the three interrupts must be made by polling the Status/Control registers of Timer 1 (D4h), ARTimer (E6h) and 32kHz oscillator (DBh).
- The interrupt vector located at the addresses FF0h, FF1h is named interrupt vector #4. It is associated with the A/D converter peripheral.

All the on-chip peripherals (refer to their descriptions for further details) have an interrupt request flag bit (TMZ for timer, EOC for A/D, etc.), this bit is set to one when the device wants to generate an interrupt request and a mask bit (ETI for timer, EAI for A/D, etc.) that must be set to one to allow the transfer of the flag bit to the Core.

Interrupt Priority

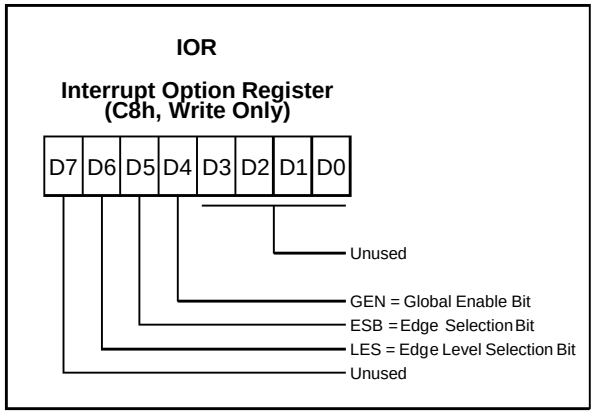
The non-maskable interrupt request has the highest priority and can interrupt any other interrupt routines at any time, nevertheless the four other interrupts cannot interrupt each other. If more than one interrupt request is pending, they are processed by the ST62xx core according to their priority level: vector #1 has the higher priority while vector #4 the lower. The priority of each interrupt source is fixed.

INTERRUPT (Continued)

Interrupt Option Register

The Interrupt Option Register (IOR register, location C8h) is used to enable/disable the individual interrupt sources and to select the operating mode of the external interrupt inputs. This register can be addressed in the Data Space as RAM location at the address C8h, nevertheless it is a write-only register that cannot be accessed with single-bit operations. The operating modes of the external interrupt inputs associated to interrupt vectors #1 and #2 are selected through bits 5 and 6 of the IOR register.

Figure 14. Interrupt Option Register



D7. This bit is not used.

LES. Level/Edge Selection Bit. When this bit is set to one, the interrupt #1 (SPI) is low level sensitive, when cleared to zero the negative edge sensitive interrupt is selected.

ESB. Edge Selection Bit. When this bit is set to one, the interrupt #2 (Port A & B lines) is positive edge sensitive, when cleared to zero the negative edge sensitive interrupt is selected.

GEN. Global Enable Interrupt. When this bit is set to one, all the interrupts are enabled. When this bit is cleared to zero all the interrupts (but NMI) are disabled.

This register is cleared on reset. No read or single bit instruction is allowed.

Table 6. Interrupt Option Register Description

GEN	SET	Enable all interrupts
	CLEARED	Disable all interrupts
ESB	SET	Rising edge mode on interrupt input #2
	CLEARED	Falling edge mode on interrupt input #2
LES	SET	Level-sensitive mode on interrupt input #1
	CLEARED	Falling edge mode on interrupt input #1
OTHERS	NOT USED	

External Interrupts Operating Modes

The NMI interrupt is associated to the NMI external pin of the ST6280. The NMI interrupt request will be generated by a falling edge applied to the NMI pin. The NMI interrupt pin signal is latched and is automatically reset by the core at the beginning of the non-maskable interrupt service routine. A 100kΩ on-chip pull-up resistor and a schmitt trigger is available with the NMI pin.

If the external interrupt input NMI is not used, it has to be connected to V_{DD} in order to avoid unintended interrupt.

The interrupt source associated with the falling/rising edge mode of the external interrupt pins (SPI vector #1, Ports A, B and C: vector #2,) are connected to two internal latches. Each latch is set when a falling/rising edge occurs and is cleared when the associated interrupt routine is started. So, the occurrence of an external interrupt request is stored: a second interrupt, that occurs during the processing of the first one, will be processed as soon as the first one has been finished (if there is not an higher priority interrupt request). If more than one interrupt occurs during the processing of the first one, these other interrupt requests will be lost.

The storage of the interrupt requests is not available in the level sensitive detection mode. To be taken into account, the low level must be present on the interrupt pin when the core samples the line after the execution of the instructions.

During the end of each instruction the core tests the interrupt lines and if there is an interrupt request the next instruction is not executed and the related interrupt routine is executed.

I**NTERRUPT** (Continued)

Interrupt Procedure. The interrupt procedure is very similar to a call procedure, indeed the user can consider the interrupt as an asynchronous call procedure. As this is an asynchronous event the user does not know about the context and the time at which it occurred. As a result the user should save all the data space registers which will be used inside the interrupt routines. There are separate sets of processor flags for normal, interrupt and non-maskable interrupt modes which are automatically switched and so these do not need to be saved.

The following list summarizes the interrupt procedure:

ST62xx actions

- Interrupt detection
- The flags C and Z of the main routine are exchanged with the flags C and Z of the interrupt routine (or the NMI flags)
- The value of the PC is stored in the first level of the stack
- The normal interrupt lines are inhibited (NMI still active)
- First internal latch is cleared
- The related interrupt vector is loaded in the PC.

User actions

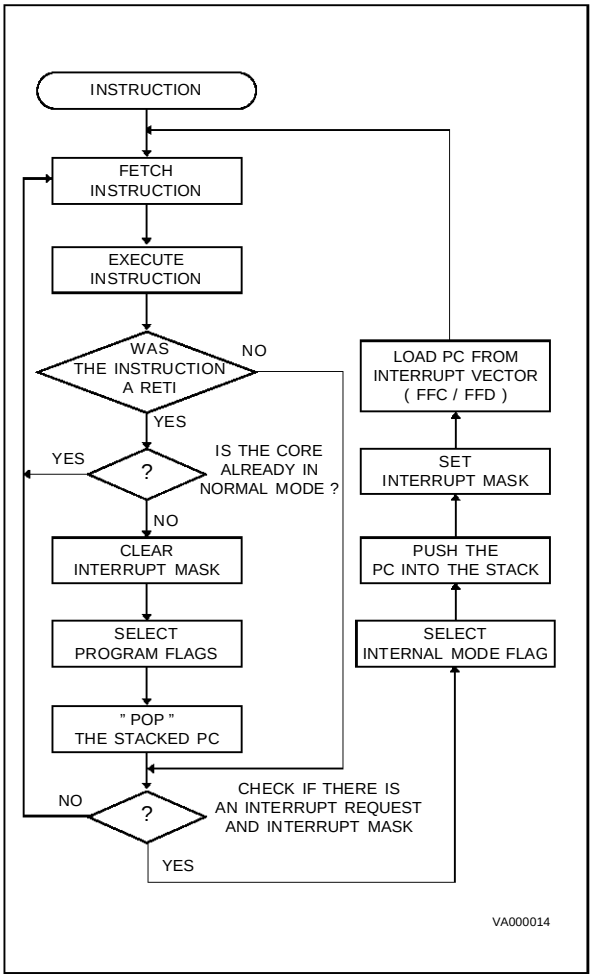
- User selected registers are saved inside the interrupt service routine (normally on a software stack)
- The source of the interrupt is found by polling (if more than one source is associated to the same vector) the interrupt flag of the source.
- Interrupt servicing
- Return from interrupt (RETI)

ST62xx actions

- Automatically the ST62xx core switches back to the normal flags (or the interrupt flags) and pops the previous PC value from the stack

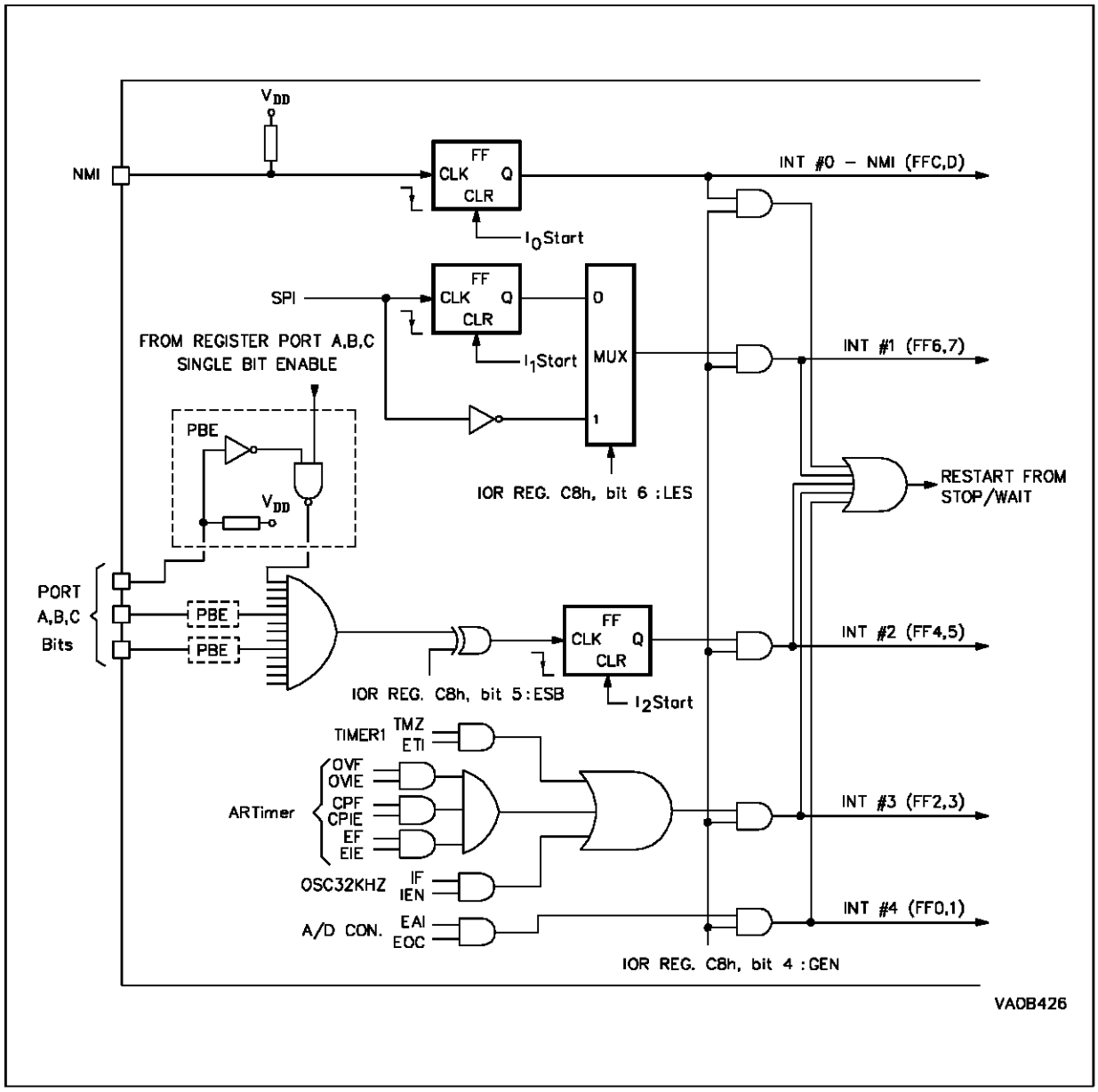
The interrupt routine begins usually by the identification of the device that has generated the interrupt request (by polling). The user should save the registers which are used inside the interrupt routine (that holds relevant data) into a software stack. After the RETI instruction execution, the core carries out the previous actions and the main routine can continue.

Figure 15. Interrupt Processing Flow-Chart



INTERRUPT (Continued)

Figure 16. ST6280 Interrupt Circuit Diagram



WARNING. *GEN is the global enable for all interrupts except NMI. If this bit is cleared, the NMI interrupt is accepted when the ST62xx core is in the normal RUN Mode.*

If the ST62xx core is in STOP or WAIT Mode, the *state can only be finished by a reset (from the Watchdog or an external Reset Signal).*

As a consequence the NMI can be masked in STO P and WAIT modes, but not in RUN mode.

INTERRUPT (Continued)

Table 7. Interrupt Request and Mask Bits

Peripheral	Register	Register Address	Mask bit	Masked Interrupt Source
All	IOR	C8h	GEN	All Interrupts, excluding NMI
I/O Ports	IOR	C8h	GEN, ESB, LES	All I/O Pins Interrupt Request
TIMER 1	TSCR1	D4h	GEN, ETI, TMZ	TMZ: TIMER 1 Overflow
A/D Converter	ADCR	D1h	GEN, EAI, EOC	EOC: End of Conversion
AR TIMER	ARMC	E5h	GEN OVIE CPIE EIE	OVF: AR TIMER Overflow CPF: Successful compare EF: Active edge on ARTIMin
SPI	SPI Interrupt	C2h	GEN, SPIID DIS	SPIF: End of Transmission
32 kHz Oscillator	320CR	DBh	GEN, EOSCI	OSC EOC: End of Counting

RESET

The ST6280 can be reset in three ways: by the external reset input ($\overline{\text{RESET}}$) tied low, by power-on reset and by the digital watchdog/timer peripheral.

RESET Input

The $\overline{\text{RESET}}$ pin can be connected to a device of the application board in order to restart the MCU during its operation. The activation of the $\overline{\text{RESET}}$ pin may occur in the RUN, WAIT or STOP mode. This input has to be used to reset the MCU internal state and provide a correct start-up procedure. The pin is active low. The internal reset signal is generated by adding a delay to the external signal. Therefore even short pulses at the $\overline{\text{RESET}}$ pin will be accepted. This feature is valid providing that V_{DD} has finished its rising phase and the oscillator is correctly running (normal RUN or WAIT modes).

If $\overline{\text{RESET}}$ activation occurs in the RUN or Wait mode, the MCU is configured in the Reset mode for as long as the signal of the $\overline{\text{RESET}}$ pin is low. The processing of the program is stopped (in RUN mode only) and the Input/Outputs are in the High-impedance state with pull-up resistors switched on. As soon as the level on the $\overline{\text{RESET}}$ pin becomes high, the initialization sequence is executed.

If a $\overline{\text{RESET}}$ pin activation occurs in the STOP mode, the oscillator starts and all the inputs/outputs are configured in the High-impedance state with pull-up resistors switched on for as long as the level on the $\overline{\text{RESET}}$ pin remains low. When the level of the $\overline{\text{RESET}}$ pin becomes high, a delay is generated by the ST62xx core to wait that the oscillator becomes completely stabilized. Then, the initialization sequence is started.

Power-On Reset (POR)

The function of the POR consists in waking up the MCU during the power-on sequence. At the beginning of this sequence, the MCU is configured in the Reset state: every Input/Output port is configured in the input mode (High-impedance state with pull-up) and no instruction is executed. When the power supply voltage becomes sufficient, the oscillator starts to operate, nevertheless the ST62xx core generates a delay to allow the oscillator to be completely stabilized before the execution of the first instruction. The initialization sequence is then executed.

Internal circuitry generates a Reset pulse when V_{DD} is switched on. In the case of fast rising V_{DD} (transition time $\leq 100\mu s$), this reset pulse starts the internal reset procedure without the need of external components at the $\overline{\text{RESET}}$ pin. In cases of slowly or non monotonously rising V_{DD}, an external reset signal must be provided for a proper reset of the MCU.

For as long as the reset pin is kept at the low level, the processor remains in the reset state. The reset will be released after the voltage at the reset pin reaches the high level.

Note:

To have a correct ST62xx start-up, the user should take care that the reset input does not change to the high level before the V_{DD} level is sufficient to allow MCU operation at the chosen frequency (see recommended operating conditions).

An on-chip counter circuit provides a delay of 2048 oscillator cycles between the detection of the reset high level and the release of the MCU reset.

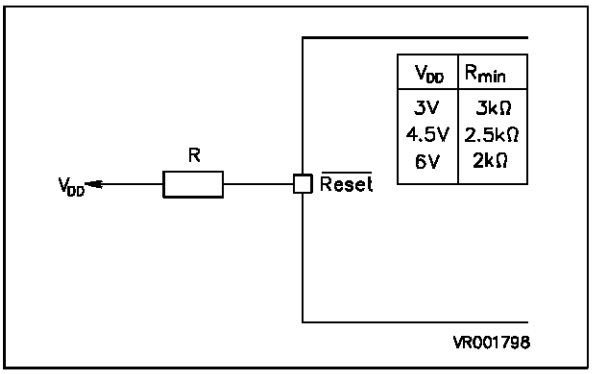
A proper reset signal for slow rising V_{DD}, i.e. the required delay between reaching sufficient operating voltage and the reset input changing to a high level, can be generally provided by an external capacitor connected between the $\overline{\text{RESET}}$ pin and V_{SS}.

RESET (Continued)

Watchdog Reset

The ST6280 provides an on-chip watchdog function in order to provide a graceful recovery from a software upset. If the watchdog register is not refreshed, preventing the end-of-count being reached, an internal circuit pulls down the RESET pin. The MCU will enter the reset state as soon as the voltage at RESET pin reaches the related low level. This also resets the watchdog which subsequently turns off the pull-down and activates the pull-up device at the RESET pin. This causes the positive transition at the RESET pin and terminates the reset state.

Figure 17. External Reset Resistance

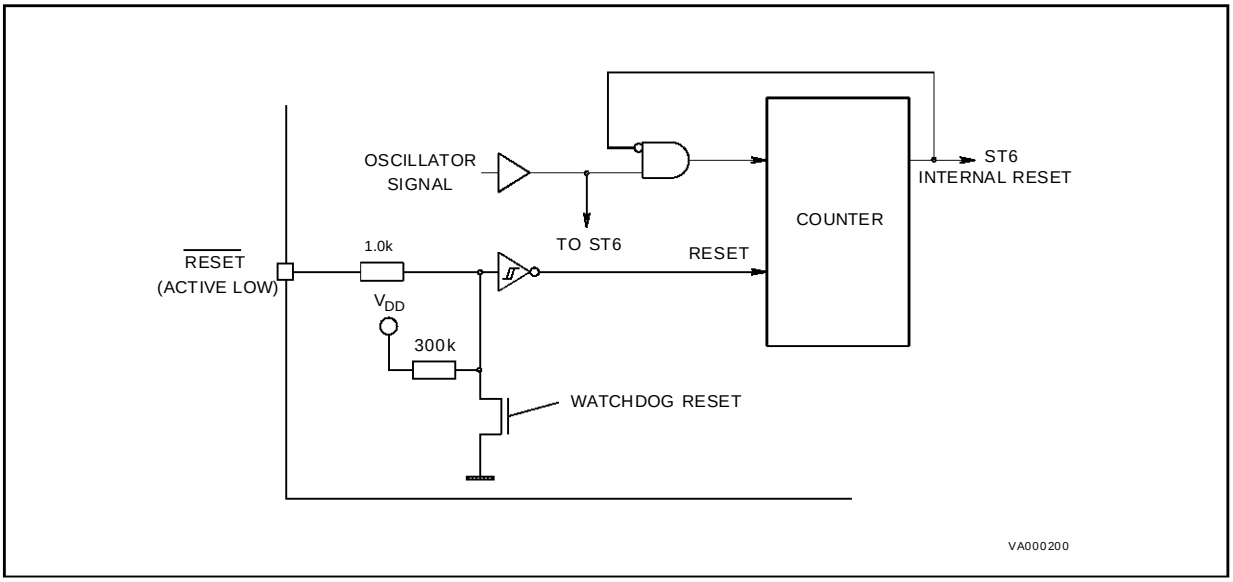


Application Notes

An external resistor between V_{DD} and reset pin is not required because an internal pull-up device is provided. If the user prefers, for any reason, to add an external pull-up resistor its value must comply with the Rmin value defined in Figure 17. If the value is lower than Rmin, the on-chip watchdog pull-down transistor might not be able to pull-down the reset pin resulting in an external deactivation of the watchdog function.

The POR function operates in a dynamic manner in the way that it brings about the initialization of the MCU when it detects a dynamic rising edge of the V_{DD} voltage. The typical detected threshold is about 2 volts, but the actual value of the detected threshold depends on the way in which the V_{DD} voltage rises up. The POR device DOES NOT allow the supervision of a static rising or falling edge of the V_{DD} voltage.

Figure 18. Reset Circuit



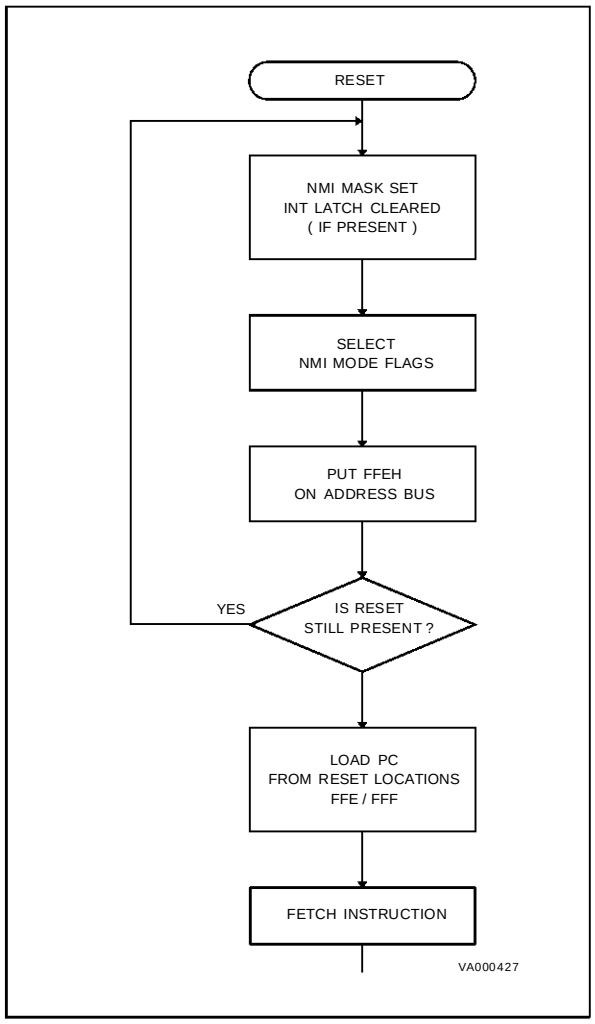
RESET (Continued)

Table 8. Reset Configuration

Feature	Register	Address	Value	Configuration
X,Y,V,W Accumulator Data RAM Data RAM/EEPROM/LCDRAM Data RAM Page EEPROM	 Control	80h to 83h FFh 84h to BFh 0h to 3Fh CBh DFh	Undefined Undefined Undefined Undefined Undefined 00h	 Enabled -no writing
Data ROM window Data ROM window Program ROM	Control Area Page	C9h 40h to 7Fh CAh	Undefined Undefined 00h	 Page 0 selected
Input/Output pins	Data Direction Option	C0h-C1h-C3h C4h-C5h-C6h CCh-CEh-CFh	00h 00h 00h	Input mode Pull-up No interrupt
A/D	Control Data	D0h D1h	00h Undefined	No conversion Idle mode
SPI	Data Interrupt	DDh C2h	00h 00h	Disabled Not connected to I/O
LCD	Mode Control	DCh	00h	LCD turned-off
Watchdog	Control	D8h	0FEh	Off-state
Timer 1	Status/Control Counter Prescaler	D4h D3h D2h	00h 0FFh 07Fh	Disabled
ARTimer	Mode Control Status Control 0 Status Control 1 Compare Load Reload/Capture	E5h E6h E7h EAh EBh E9h	00h 00h 00h 00h Undefined Undefined	Disabled
32kHz OSC	Control	DBh	00h	Disabled
Interrupt	Option	C8h	00h	Disabled

RESET (Continued)

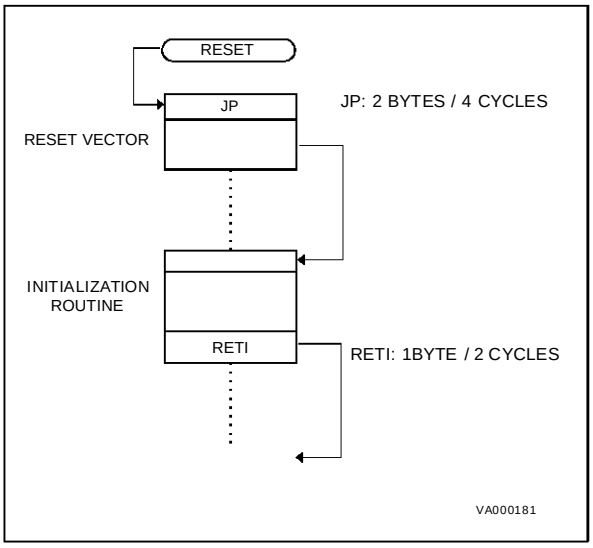
Figure 19. Reset & Interrupt Processing Flow-Chart



MCU Initialization Sequence

When a reset occurs the stack is reset to the program counter, the PC is loaded with the address of the reset vector (located in the program ROM at addresses FFEh & FFFh). A jump instruction to the beginning of the program has to be written into these locations. After a reset the interrupt mask is automatically activated so that the core is in non-maskable interrupt mode to prevent false or ghost interrupts during the restart phase. Therefore the restart routine should be terminated by a RETI instruction to switch to normal mode and enable interrupts. If no pending interrupt is present at the end of the reset routine the ST62xx will continue with the instruction after the RETI; otherwise the pending interrupt will be serviced.

Figure 20. Restart Initialization Program Flow-Chart



WAIT & STOP MODES

The WAIT and STOP modes have been implemented in the ST62xx core in order to reduce the consumption of the product when the latter has no instruction to execute. These two modes are described in the following paragraphs

WAIT Mode

The configuration of the MCU in the WAIT mode occurs as soon as the WAIT instruction is executed. The microcontroller can also be considered as being in a "software frozen" state where the core stops processing the instructions of the routine, the contents of the RAM locations and peripheral registers are saved as long as the power supply voltage is higher than the RAM retention voltage but where the peripherals are still working. The WAIT mode is used when the user wants to reduce the consumption of the MCU when it is in idle, while not losing count of time or monitoring of external events. The oscillator is not stopped in order to provide a clock signal to the peripherals. The timer counting may be enabled (writing the PSI bit in TSCR register) and the timer interrupt may be also enabled before entering the WAIT mode; this allows the WAIT mode to be left when timer interrupt occurs. The above explanation related to the timers applies also to the A/D converter. If the exit from the WAIT mode is performed with a general RESET (either from the activation of the external pin or by watchdog reset) the MCU will enter a normal reset procedure as described in the RESET chapter.
If an interrupt is generated during WAIT mode the MCU behavior depends on the state of the ST62xx core before the initialization of the WAIT sequence, but also of the kind of the interrupt request that is generated. This case will be described in the following paragraphs. In any case, the ST62xx core does not generate any delay after the occurrence of the interrupt because the oscillator clock is still available.

STOP Mode

If the Watchdog is disabled the STOP mode is available. When in STOP mode the MCU is placed in the lowest power consumption mode. In this operating mode the microcontroller can be considered as being "frozen", no instruction is executed, the oscillator is stopped, the contents of the RAM locations and peripheral registers are saved as long as the power supply voltage is higher than the RAM retention voltage, and the ST62xx core waits for the occurrence of an external interrupt request or Reset activation to output from the STOP state.

If the exit from the STOP mode is performed with a general RESET (by the activation of the external pin) the MCU will enter a normal reset procedure as described in the RESET chapter. The case of an interrupt depends on the state of the ST62xx core before the initialization of the STOP sequence and also of the kind of the interrupt request that is generated.

This case will be described in the following paragraphs. In any case, the ST62xx core generates a delay after the occurrence of the interrupt request in order to wait the complete stabilization of the oscillator before the execution of the first instruction.

Exit from WAIT and STOP Modes

The following paragraphs describe the output procedure of the ST62xx core from WAIT and STOP modes when an interrupt occurs (not a RESET). It must be noted that the restart sequence depends on the original state of the MCU (normal, interrupt or non-maskable interrupt mode) before the start of the WAIT or STOP sequence, but also of the type of the interrupt request that is generated.

Normal Mode. If the ST62xx core was in the main routine when the WAIT or STOP instruction has been executed, the ST62xx core outputs from the stop or wait mode as soon as any interrupt occurs; the related interrupt routine is executed and at the end of the interrupt service routine the instruction that follows the STOP or the WAIT instruction is executed if no other interrupts are pending.

Non-maskable Interrupt Mode. If the STOP or WAIT instruction has been executed during the execution of the non-maskable interrupt routine, the ST62xx core outputs from the stop or wait mode as soon as any interrupt occurs: the instruction that follows the STOP or the WAIT instruction is executed and the ST62xx core is still in the non-maskable interrupt mode even if another interrupt has been generated.

Normal Interrupt Mode. If the ST62xx core was in the interrupt mode before the initialization of the STOP or WAIT sequence, it outputs from the stop or wait mode as soon as any interrupt occurs. Nevertheless, two cases have to be considered:

- If the interrupt is a normal interrupt, the interrupt routine in which the WAIT or STOP was entered will be completed with the execution of the instruction that follows the STOP or the WAIT and the ST62xx core is still in the interrupt mode. At the end of this routine pending interrupts will be serviced in accordance to their priority.
- If the interrupt is a non-maskable interrupt, the non-maskable routine is processed at first. Then the routine in which the WAIT or STOP was entered will be completed with the execution of the instruction that follows the STOP or the WAIT and the ST62xx core remains in the normal interrupt mode.

Notes:

WAIT & STOP MODES (Continued)

To reach the lowest power consumption the user software must take care of:

- placing the A/D converter in its power down mode by clearing the PDS bit in the A/D control register before entering the STOP instruction.
- switching off the 32kHz oscillator by clearing the oscillator start/stop bit in the 32kHz oscillator control register.
- putting the EEPROM on-chip memory in standby mode by setting the E2OFF bit in EEPROM Control Register to one.

The LCD Driver peripheral is automatically switched-off by the STOP instruction when the 32kHz oscillator operation is not selected.

When the hardware activated watchdog is selected or the software watchdog enabled, the STOP instruction is deactivated and any attempt to execute the STOP instruction will cause an execution of a WAIT instruction.

If all the interrupt sources are disabled (including NMI if GEN="0"), the restart of the MCU can only be done by a $\overline{\text{RESET}}$ activation. The WAIT and STOP instructions are not executed if an enabled interrupt request is pending.

If all the interrupt sources are disabled, the restart of the ST6280 MCU can only be made by a Reset. The Wait and Stop instructions are not executed but skipped if an enabled interrupt request is pending.

ON-CHIP CLOCK OSCILLATOR

The internal oscillator circuit is designed to require a minimum of external components. A crystal, a ceramic resonator or an external signal (provided to the OSCin pin) may be used to generate a system clock with various stability/cost tradeoffs.

The different oscillator connection methods are shown in Figure 21, crystal specifications are given in Figure 22. One machine cycle takes 13 oscillator pulses; 12 clock pulses are needed to increment the PC while an additional 13th pulse is needed to stabilize the internal latches during memory addressing. This means that with a clock frequency of 8MHz the machine cycle is 1.625µs.

The crystal oscillator start-up time is a function of many variables: crystal parameters (especially R_S) oscillator load capacitance (C_L), IC parameters, ambient temperature, and supply voltage.It must be observed that the crystal or ceramic leads and circuit connections must be as short as possible.

The oscillator output frequency is internally divided by 13 to produce the machine cycle, by 12 to produce the Timer 1, the Watchdog and the A/D peripheral clock and not divided at all or divided by 3 to produce ARTimer clock. A machine cycle is the smallest unit needed to execute any operation (i.e., increment the program counter). An instruction may need two, four, or five machine cycles to be executed. Typical values for CL1 and CL2 are in the range of 15pF to 22pF but these should be chosen based on the crystal manufacturers specification. Typical input capacitance for OSCin and OSCout pins is 5pF.

Figure 21. Internal Clock Circuits

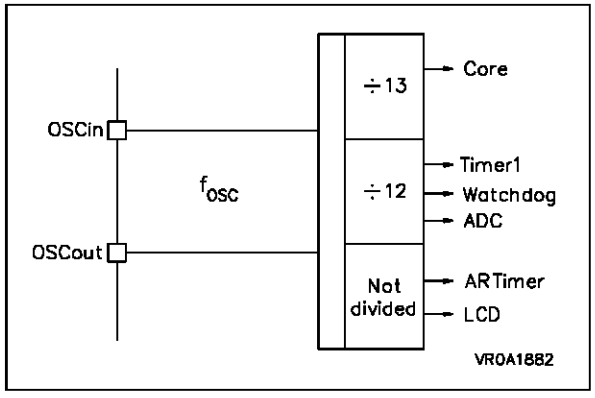
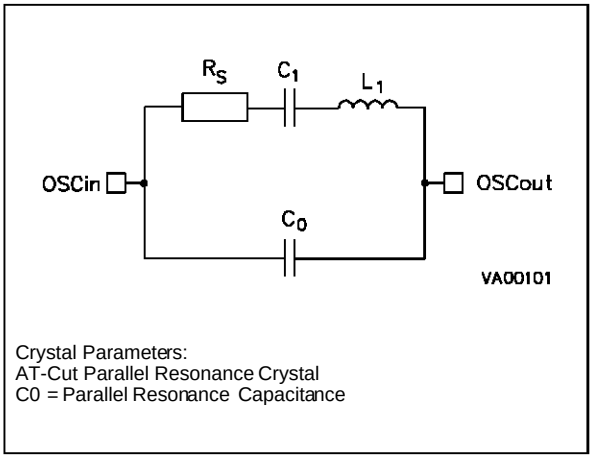
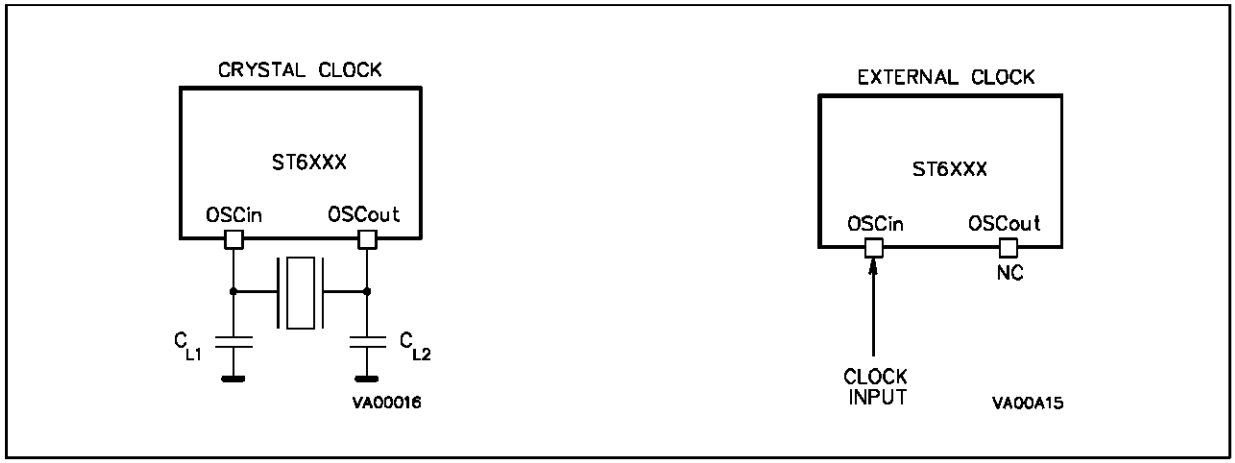


Figure 22. Crystal Parameters



ON-CHIP CLOCK OSCILLATOR (Continued)

Figure 23. Oscillator Connection



C_{L1} = C_{L2} = 12 to 22pF for a 4/8MHz Crystal (typical)

INPUT/OUTPUT PORTS

The ST6280 has 22 Input/Output lines respectively that can be individually programmed either in the input or output mode with the following, software selectable, options.

- Input without pull-up and without interrupt
- Input with pull-up and with interrupt
- Input with pull-up without interrupt
- Timer 1 I/O line (PA4)
- ARTimer I/O lines (PB6, PB7)
- Analog input (PB0-PB7, PC4-PC7 only)
- SPI control signals (PA5-PA7)
- Push-pull Output
- 20mA Open drain Output (PA2-PA7, PC0-PC3 only)
- Standard Open drain Output (PB0-PB7, PC4-PC7 only)

The lines are organized in 3 ports (Ports A,B and C). The ports occupy 9 registers in the data space. Each bit of these registers is associated with a particular line (for instance, bit 0 of the Port A Data, Direction and Option registers are associated with the PA0 line of Port A).

Three DATA registers (DRA, DRB and DRC) are used to read the voltage level values of the lines programmed in the input mode, or to write the logic value of the signal to be output on the lines configured in the output mode. The port data registers can be read to get the effective logic levels of the pins, however they can also be written by the user software, with the related option register, to select the different input mode options.

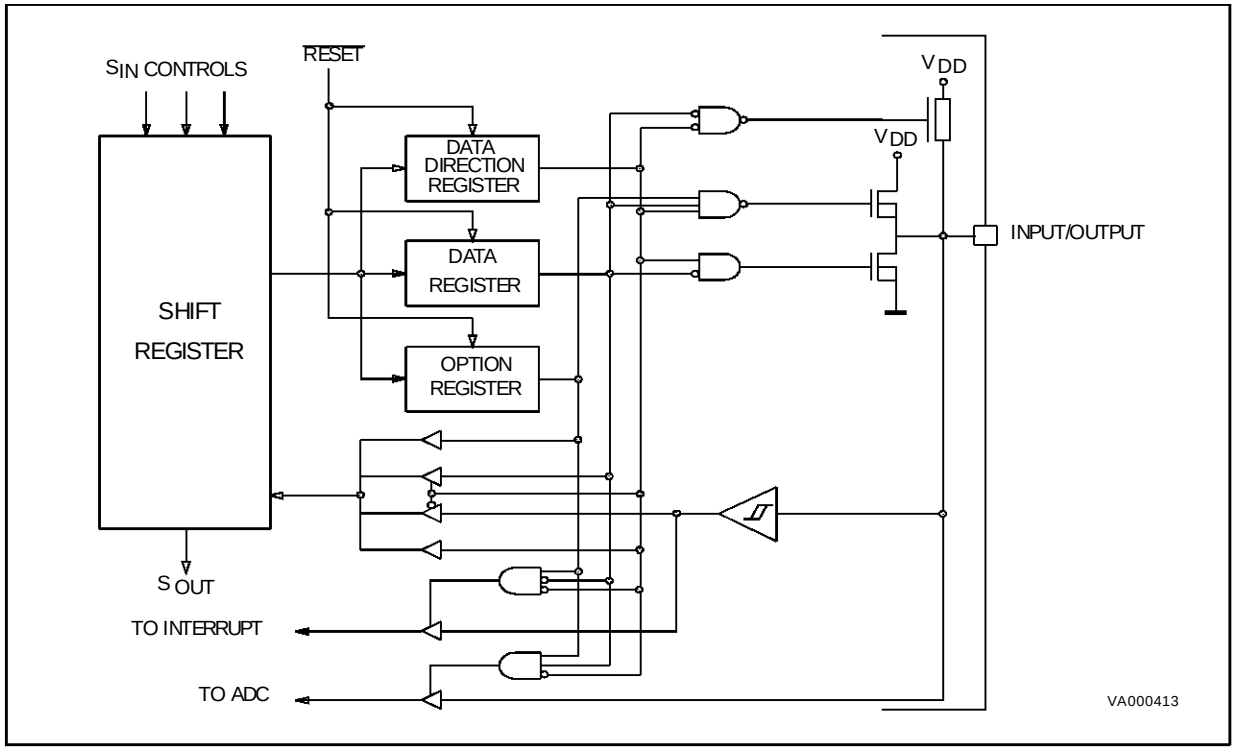
Single-bit operations on I/O registers are possible but care is necessary because reading in input mode is made from I/O pins while writing will directly affect the Port data register causing an undesired change of the input configuration.

The three DATA DIRECTION registers (DDRA, DDRB and DDRC) allow the selection of the direction of each pin (input or output).

The three OPTION registers (ORPA, ORPB and ORPC) are used to select the different port options that are available both in input and output mode.

All I/O registers can be read or written as any other RAM location of the data space. During the initialization of the MCU, all the I/O registers are cleared and the input mode with pull-up/no-interrupt is selected on all pins (except PB0-PB3 whose function is mask selectable) thus avoiding potential pin conflicts.

Figure 24. I/O Port Block Diagram



INPUT/OUTPUT PORTS (Continued)

Figure 25. I/O Port Data Option Registers

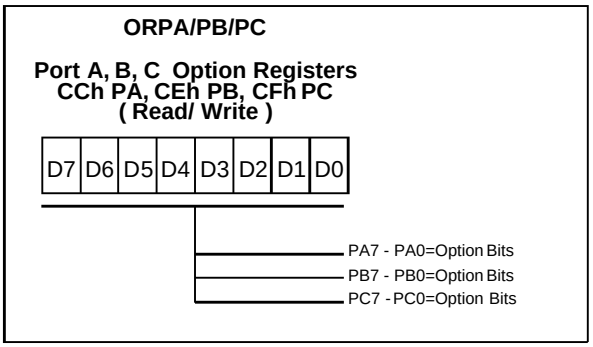


Figure 26. I/O Port Data Option Registers

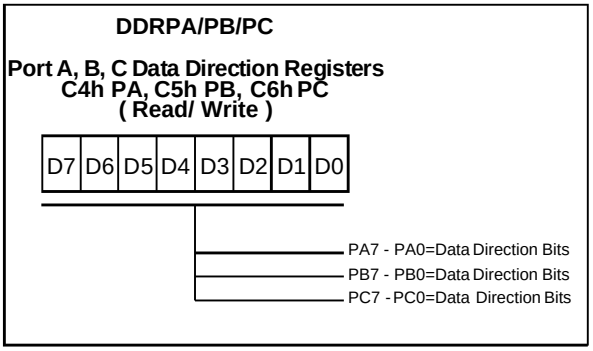
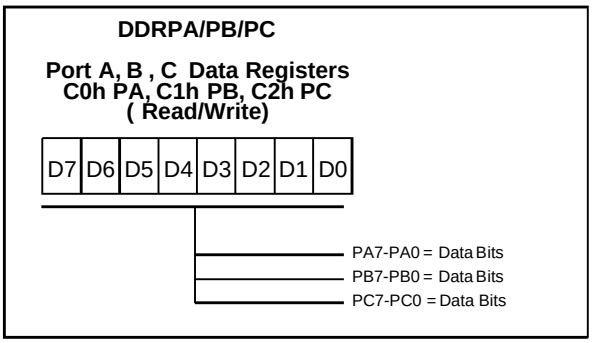


Figure 27. I/O Port Data Register



I/O Pin Programming

Each pin can be individually programmed as input or output with different input and output configuration. This is achieved by writing to the relevant bit in the data (DR), data direction (DDR) and option (OR) registers. The following table shows all the I/O port configuration that can be selected by the user software:

Option Description

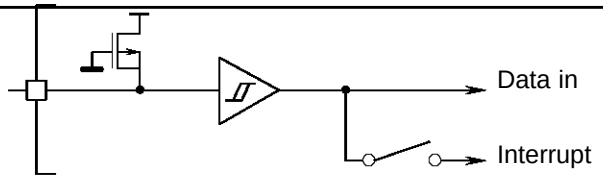
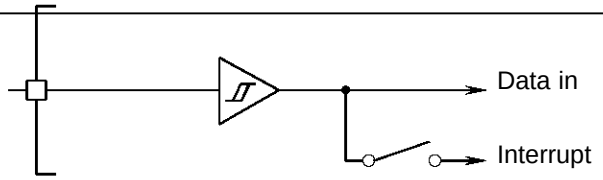
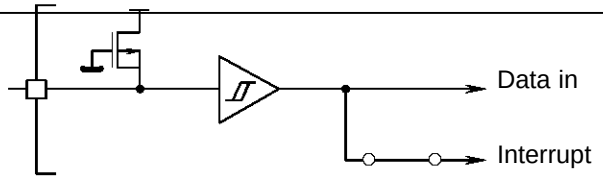
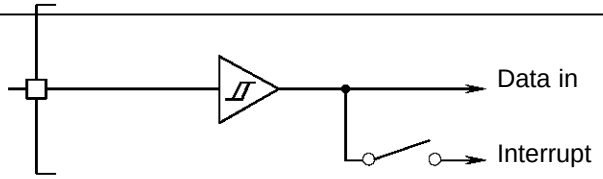
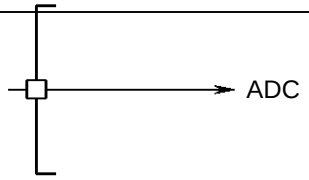
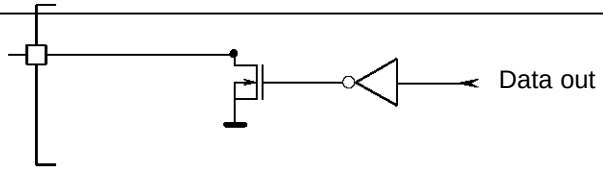
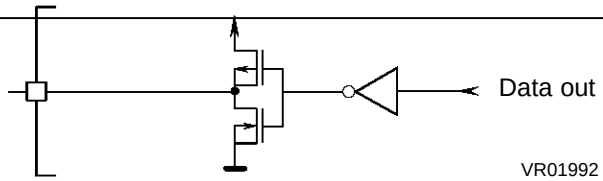
Pull-up, high impedance option. All the input lines can be individually programmed with or without an internal pull-up according to the codes programmed in the OR and DR registers (see table 9). If the pull-up option is not selected, the input pin is set to a high impedance state.

Interrupt Option. All input lines can be individually connected by software to the interrupt lines of the ST62xx Core according to the codes programmed in the OR and DR register (see table 9). The pins of Port A, B and C are “ORed” and are connected to interrupt vector #2. The different interrupt modes (falling/rising edge, level sensitive) can be selected by software by programming bit 5 and bit 6 of the IOR register.

Analog input Option. The twelve PB0-PB7, PC4-PC7 pins can be configured to be analog inputs according to the codes programmed in the OR and DR registers (see table 9). These analog inputs are connected to the on-chip 8-bit Analog to Digital Converter. ONLY ONE input line should be programmed to be analog input at the same time otherwise the selected inputs will be shorted.

INPUT/OUTPUT PORTS (Continued)

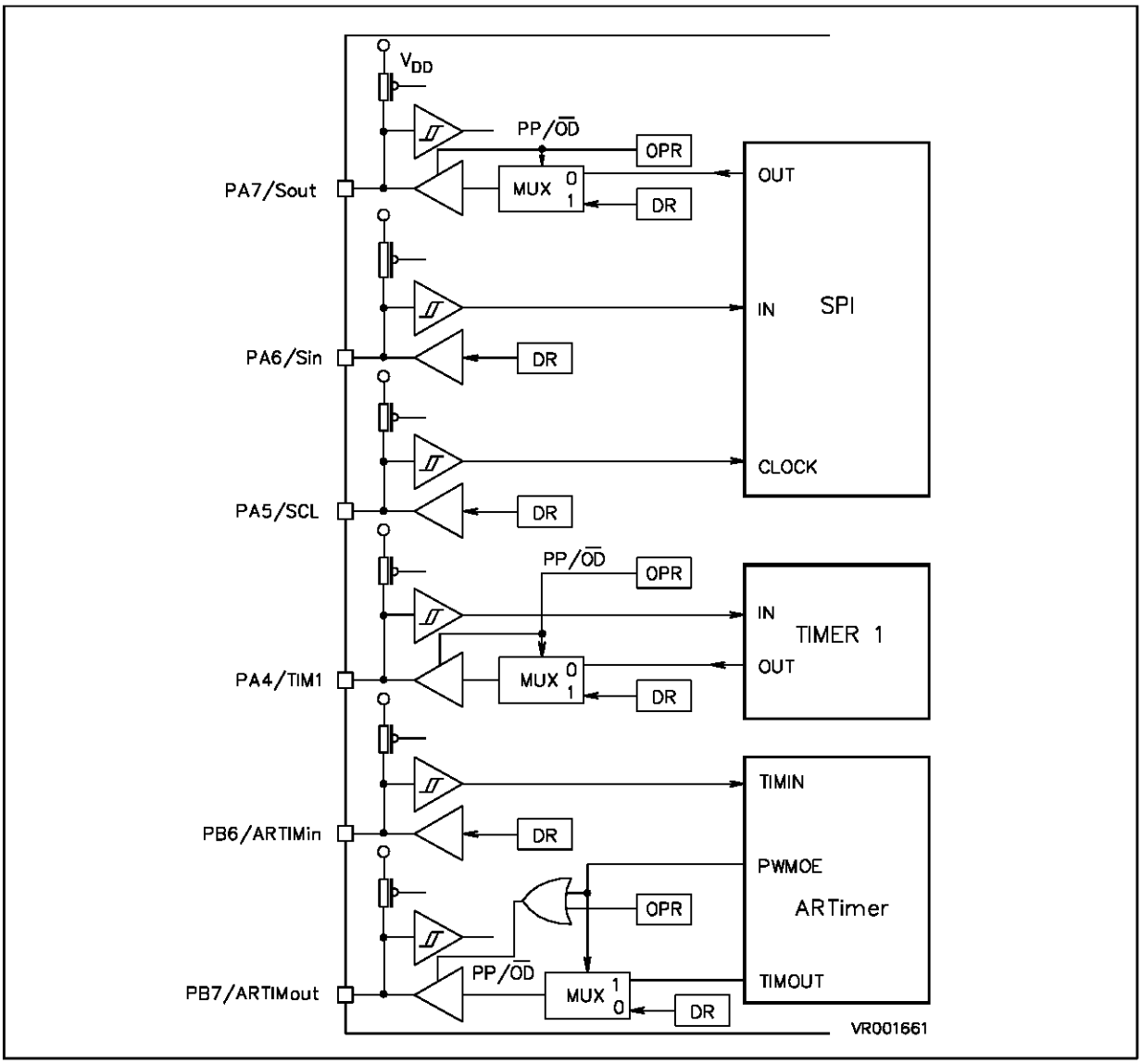
Table 9. I/O Port Options Selection

DDR, OR	DR	Mode	Option	Schematic	
0	0	0	Input	Pull-up No interrupt (RESET state)	
0	0	1	Input	No pull-up No interrupt	
0	1	0	Input	Pull-up Interrupt	
0	1	1	Input	No pull-up No interrupt PA2-PA7 PC0-PC3	
			Input	Analog input: PB0-PB7 PC4-PC7	
1	0	X	Output	Open drain: 20mA PA2-PA7, PC0-PC3 Open drain: 5mA PB0-PB7, PC4-PC7	
1	1	X	Output	Push-pull: 20mA PA2-PA7, PC0-PC3 Push-pull: 5mA PB0-PB7, PC4-PC7	

INPUT/OUTPUT PORTS (Continued)

SPI alternate function Option. The I/O pins PA5-PA7 are also used by serial peripheral interface SPI. PA5 is connected with the SPI clock input SCL, PA6 is connected with the SPI data input Sin and PA7 is connected with the SPI data output Sout. For serial input operation PA5 and PA6 have to be programmed as inputs. For serial output operation PA7 has to be programmed as open-drain output (DDR = “1”, OPR = ”0”). In this operating mode the output of the SPI shift register instead of the port data register is connected to the port buffer. When PA7 is programmed as push-pull output (DDR = “1”, OPR = ”1”), the port data register is connected to the port buffer. When the SPI peripheral is not used PA5-PA7 can be used as general purpose I/O lines (providing that PA7 is not selected to be open-drain in output mode).

Figure 28. Peripheral Interface Configuration of Serial I/O Timer 1, ARTimer



INPUT/OUTPUT PORTS (Continued)

Timer 1 Alternate function Option.

When bit TOUT of register TSCR1 is low, pin PA4/Timer 1 is configured through the port registers as any standard pin of Port B. It is in addition connected to the Timer 1 input for Gated and Event counter modes. When bit TOUT of register TSCR1 is high, pin PA4/Timer 1 is forced as Timer 1 output. The port registers configuration is meaningless.

AR Timer Alternate function Option

When bit PWMOE of register ARMC is low, pin ARTIMout/PB7 is configured as any standard pin of port B through the port registers. When PWMOE is high, ARTIMout/PB7 is the PWM output. The port registers configuration is meaningless.

ARTIMin/PB6 is connected to the AR Timer input. It is configured through the port registers as any standard pin of port B. To use ARTIMin/PB6 as AR Timer input, it must be configured as input through DDRB.

Figure 28 shows the Peripheral Interface Configuration of the SPI, Timer 1 and ARTimer

Notes:

The WAIT and STOP instructions allow the ST62xx to be used in situations where low power consumption is needed. The lowest power consumption is achieved by configuring I/O pins:

- Input high impedance if the I/O is connected to V_{DD} or V_{SS}
- Input pull-up if the I/O is floating
- Output at “1” or “0” if the I/O is floating
- Output open drain at 0 if the I/O is floating

The first configuration is prefered when possible. The key point is to define the potentials and avoid conflicts with the environment of the MCU.

The user has to take care not to switch outputs with heavy loads during the conversion of one of the analog inputs in order to avoid any disturbance in the measurement.

Notes:

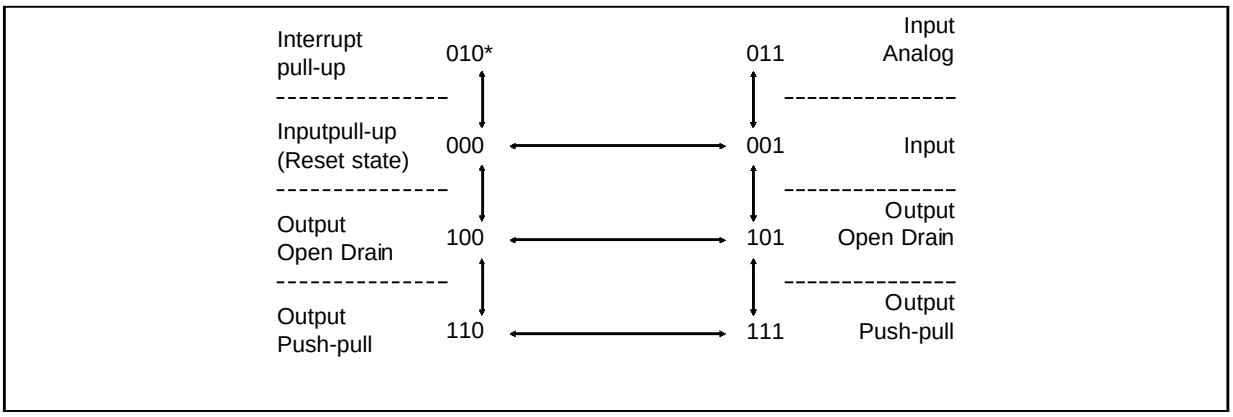
Switching the I/O ports from one state to another should be done in a way that no unwanted side effects can happen. The recommended safe transitions are shown below. All other transitions are risky and should be avoided during change of operation mode as it is most likely that there will be an unwanted side-effect such as interrupt generation or two pins shorted together by the analog input lines.

Single bit SET, RES, INC, DEC instructions should be used very carefully with Port A, B and C data registers because these instructions make an implicit read and write back of the whole addressed register byte. In port input mode however data register address reads from the input pins, not from the data register latches and data register information in input mode is used to set characteristics of the input pin (interrupt, pull-up, analog input), therefore these characteristics may be unintentionally reprogrammed depending on the state of the input pins. As a general rule is better to use SET, RES, INC and DEC instructions on data registers only when the whole port is in output mode. If input or mixed configuration is needed it is recommended to keep a copy of the data register in RAM. On this copy it is possible to use single bit instructions, then the copy register could be written into the port data register.

Example:

```
SET bit, data copy
LD a, datacopy
LD DRA, a
```

Figure 29. State Diagram for Safe Transitions



Note *. xxx = DDR, OR, DR Bits respectively

TIMERS

The ST6280 offers two on-chip Timer peripherals named Timer 1 and Auto-reload Timer. Timer 1 consists of an 8-bit counter with a 7-bit programmable prescaler, thus giving a maximum count of 2^{15}, and control logic that allows configuring the peripheral in three operating modes. The Auto-reload Timer is an 8-bit Timer with Auto-reload, Input Capture and Output Compare capabilities. 4 modes are available for PWM, PLL, time measurement and period measurement.

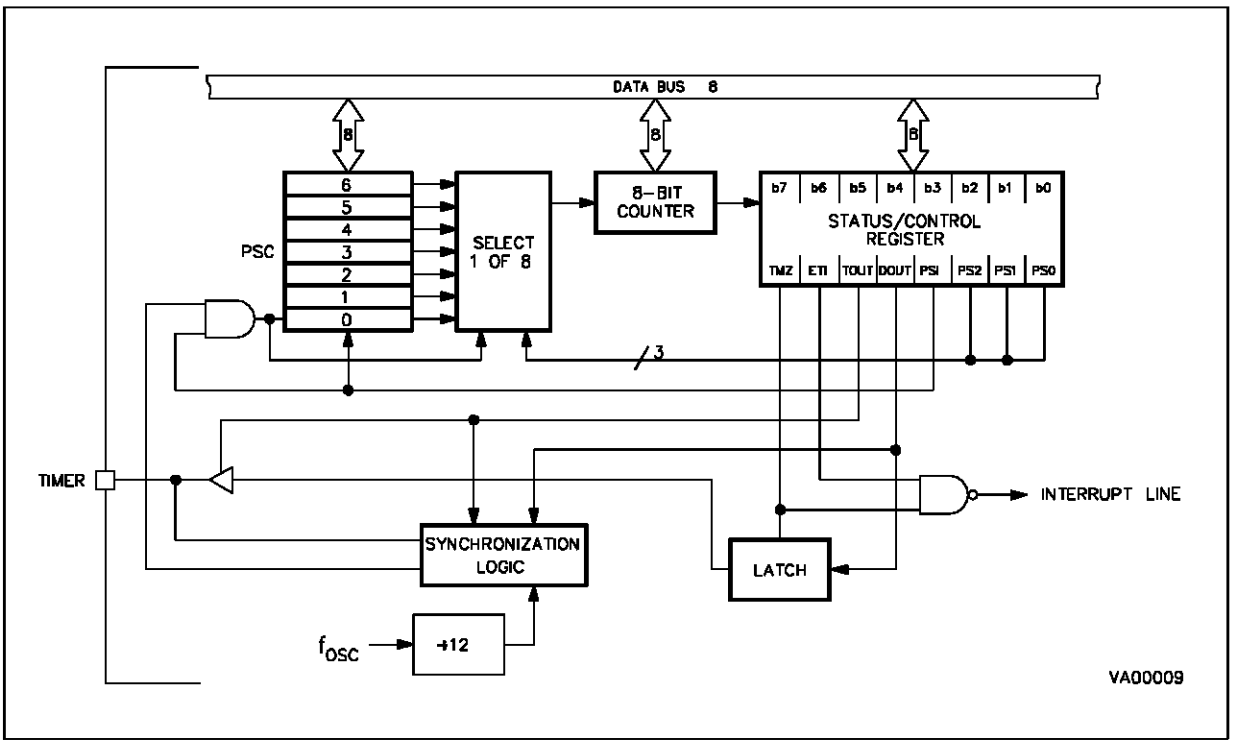
Timer 1

Figure 30 shows the Timer 1 block diagram. An external Timer pin PA4/TIM1 is available for the user. The content of the 8-bit counter can be read/written in the Timer/Counter register TCR which is addressed in the data space as a RAM location at addresses D3h. The state of the 7-bit prescaler is read in the PSC register at addresses D2h. The control logic device is managed in the TSCR1 register (addresses D4h) as described in the following paragraphs.

The 8-bit counter is decremented by the output (rising edge) coming from the 7-bit prescaler and can be loaded and read under program control. When it decrements to zero then the TMZ (Timer Zero) bit in the TSCR1 is set to one. If the ETI (Enable Timer Interrupt) bit in the TSCR1 is also set to one an interrupt request, associated to interrupt vector #3, is generated. The interrupt service routine then should poll bit TMZ in TSCR1 to determine if the interrupt has been generated by Timer 1, ARTimer or 32kHz oscillator. The Timer 1 interrupt can be used to exit the MCU from the WAIT mode.

The Timer 1 Prescaler input can be the internal clock f_{OSC} divided by 12 or an external clock at the Timer I/O pin. The prescaler decrements on the rising edge. Depending on the division factor programmed by PS2, PS1 and PS0 bits in TSCR1, the clock input of the timer/counter register is multiplexed to different sources. On division factor 1, the clock input of the prescaler is also that of timer/counter; on factor 2, bit 0 of prescaler register is connected to the clock input of TCR1. This bit changes its state with the half frequency of prescaler clock input. On factor 4, bit 1 of PSC1 is connected to clock input of TCR1, and so on. The prescaler initialize bit PSI in the TSCR1 register must be set to one to allow the prescaler (and hence the counter) to start. If it is cleared to zero then all of the prescaler bits are set to one and the counter is inhibited from counting. The prescaler can be given any value between 0 and 7Fh by writing to addresses D2h, if bit PSI in the TSCR1 register is set to one. The tap of the prescaler is selected using the PS2,PS1,PS0 bits in the control register. Figure 31 shows the Timer 1 working principle.

Figure 30. Timer 1 Block Diagram



TIMERS (Continued)

Timer 1 can be configured in 3 modes using the TOUT and DOUT bits of the TSCR1 register. These modes are Event counter, Gated or Output signal.

The internal Timer I/O can in addition be connected to either the PA4/TIM1 pin or the DRC1 bit depending on the configuration of bit DDRC1. Table 11 summarize the modes of Timer 1.

- **Event counter:** The Prescaler is decremented at each rising edge of the Timer I/O. The Timer I/O is the PA4/TIM1 pin.
- **Gated:** The Timer 1 is decremented by the Timer clock (f_{INT} divided by 12) when the internal Timer I/O is held high. The Timer I/O is pin PA4/TIM1.
- **Output signal:** The PA4/TIM1 pin is connected to the DOUT latch and is configured as output regardless of DOUT and DDRC1 bits. The low to high transition of bit TMZ (when counter reaches 00h) is used to latch the data previously stored in DOUT and pass it to the PA4/TIM1 through the Timer I/O. This operating mode allows signal generation.

Timer 1 Interrupt

When the counter register decrements to zero and the software controlled ETI (Enable Timer Interrupt) bit is set to one then an interrupt request associated to interrupt vector #3 is generated. When the counter decrements to zero also the TMZ bit in the TSCR register is set to one.

Since one interrupt vector is available for both Timer 1, ARTimer and 32kHz oscillator, the interrupt service routine should determine from which source the interrupt came. That is achieved by polling the TMZ bit, the OSCEOC of the 32kHz Control Register and OVIE bit of the ARTimer Control Register.

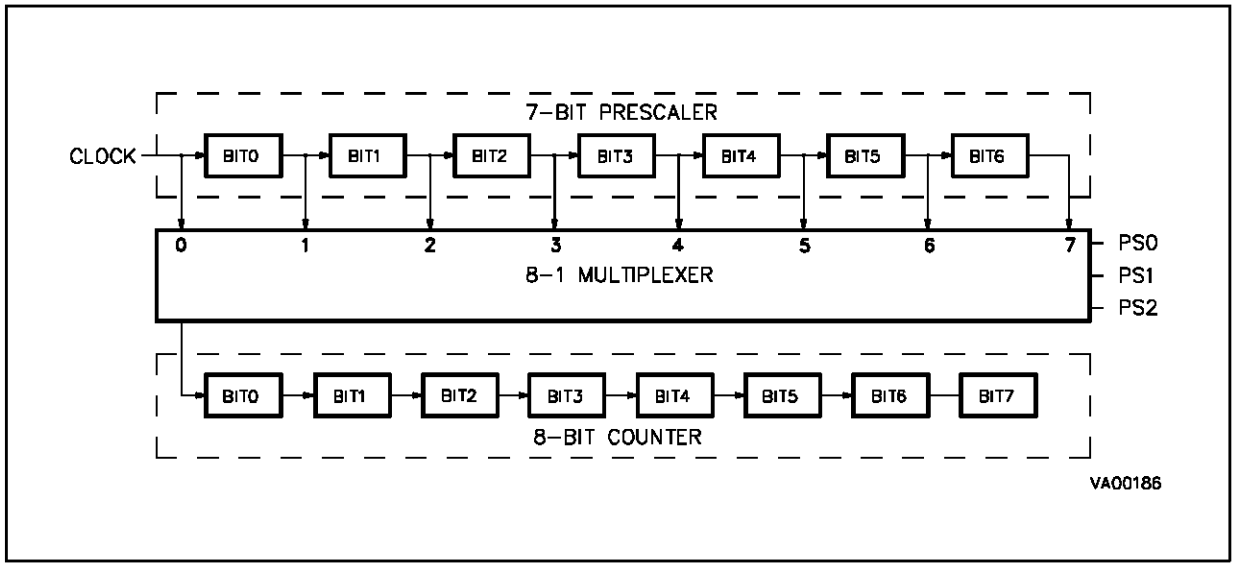
Notes:

TMZ is set when the counter reaches 00h; however, it may be set by writing 00h in the TCR1 register or setting bit 7 of the TSCR1 register. TMZ bit must be cleared by user software when servicing the Timer 1 interrupt to avoid undesired interrupts when leaving the interrupt service routine. After reset, the 8-bit counter register is loaded to FFh while the 7-bit prescaler is loaded to 7Fh , and the TSCR1 register is cleared which means that Timer 1 is stopped and the Timer 1 interrupt is disabled.

If the Timer 1 is programmed in output mode, DOUT bit is transferred to the TIM1 pin when TMZ is set to one (by software or due to counter decrement). When TMZ is high, the latch is transparent and DOUT is copied to the timer pin. When TMZ goes low, DOUT is latched.

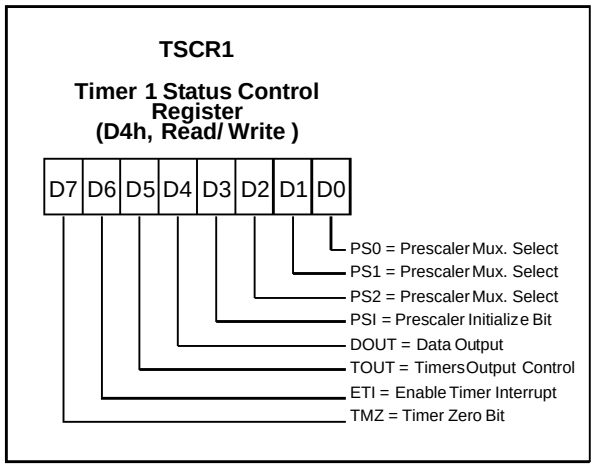
A write to the TCR1 register will predominate over the 8-bit counter decrement to 00h function, i.e. if a write and a TCR1 register decrement to 00h occur simultaneously, the write will take precedence, and the TMZ bit is not set until the 8-bit counter reaches 00h again. The values of the TCR1 and the PSC1 registers can be read accurately at any time.

Figure 31. Timer 1 Working Principle



TIMERS (Continued)

Figure 32. Timer Status Control Register



TMZ. Low-to-high transition indicates that the timer count register has decremented to zero. This bit must be cleared by user software before starting with a new count.

ETI. This bit, when set, enables the timer interrupt request (vector #3). If ETI="0" the timer interrupt is disabled. If ETI="1" and TMZ="1" an interrupt request is generated.

TOUT. When low, this bit selects an input mode for the Timer I/O pin. When high the output mode is selected.

DOUT. If Timer 1 is in Output mode, DOUT is the data sent to the PC1/TIM1 pin when TMZ goes high. DOUT enables discrimination between Event Counter and Gated modes if TOUT is low.

PSI. Used to initialize the prescaler and inhibit its counting. When PSI="0" the prescaler is set to 7Fh and the counter is inhibited. When PSI="1" the prescaler is enabled to count downwards. As long as PSI="0" both counter and prescaler are not running.

PS2, PS1, PS0. These bits select the division ratio of the prescaler register.

Table 10. Prescaler Division Factors

PS2	PS1	PS0	Divided by	PS2	PS1	PS0	Divided by
0	0	0	1	1	0	0	16
0	0	1	2	1	0	1	32
0	1	0	4	1	1	0	64
0	1	1	8	1	1	1	128

Figure 33. Timer Counter Register

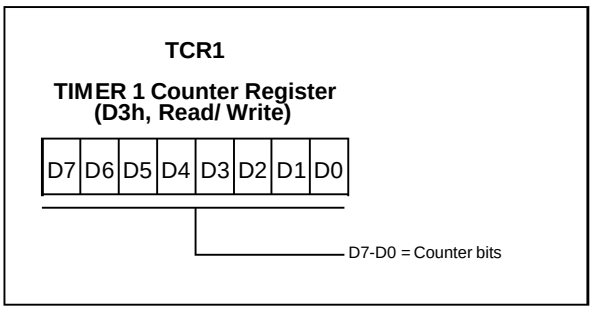


Figure 34. Prescaler Register

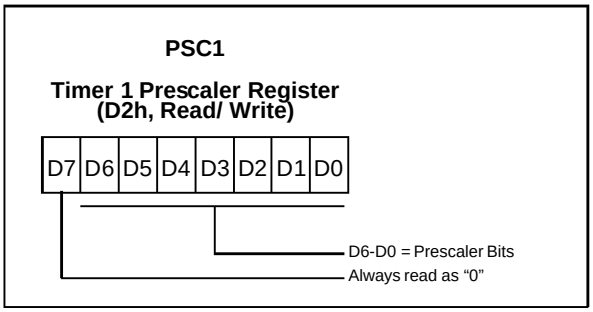


Table 11. Timer 1 Operating Modes

TOUT	DOUT	Timer Pin	Timer Function
0	0	Input	Event Counter
0	1	Input	Gated
1	0	Output	Output
1	1	Output	Output

TIMERS (Continued)

Auto-reload Timer

The Auto-reload Timer (AR Timer) on-chip peripheral consists of an 8-bit timer/counter (AR COUNTER) with compare and capture/reload capabilities and a 7-bit prescaler with a clock multiplexer enabling the clock input to be selected as f_{OSC}, $f_{OSC}/3$ or external clock. One Mode Control Register (AR MODE), two Status Registers (ARSC0, ARSC1), an output pin (ARTIMout/PB7) and an input pin (ARTIMin/PB6) allow the Auto-reload Timer to be used in 4 modes:

- Auto-reload (PWM generation),
- Output compare and reload on external event (PLL),
- Input capture and output compare for time measurement.
- Input capture and output compare for period measurement.

The AR Timer can be used to wake the MCU from WAIT mode with either an internal or an external clock. It also can be used to wake the MCU from STOP mode if used with an external clock provided at pin ARTIMin. A Load register allows the program to read and write the counter on the fly.

AR Timer Description

The AR COUNTER is an 8-bit up-counter incremented on the clock input rising edge. It is loaded from the ReLoad/Capture Register REL/CAP (address D9h) for auto-reload or capture operations as well as for initialization. Direct access to the AR COUNTER is not possible, however by reading/writing the Load Register AR LOAD (address DBh) it is possible to read/write the TC counter content.

The AR Timer input clock is either the internal clock (from Oscillator Divider), the internal clock divided by 3 or the ARTIMin pin. Selection between these clock sources is made through the AR Multiplexer by bits CC0-CC1 of Register ARSCR1. The output of the AR Multiplexer feeds the AR Prescaler, ARPSC. ARPSC is a software programmable 7-bit prescaler. Programming of ARPSC is performed by the AR Prescaler Multiplexer AR MUX which selects one of the 8 available taps of the prescaler outputs under the control of PSC0,1,2 in the AR Mode Control Register (address D5h). So the division factor of PSC prescaler can be set to 2^n (where n = 0, 1,..7).

The clock input to the TC counter is enabled by bit TEN (Timer Enable) in the Status Control Register 1. When TEN is cleared to “0” the TC counter is stopped and the prescaler and counter contents are frozen. When the TEN bit is set to “1” the TC counter runs at the rate of the selected clock source. TC is cleared after system reset.

The ARTC counter can also be initialized by writing into the load register ARLR, which causes also the immediate copy of the value into the ARTC counter regardless of whether ARTC is running or not. Initialization of ARTC, in both ways, will also clear the ARPSC in order to start counting from a known state.

Each interrupt generated by the AR Timer operating modes is associated to interrupt vector #3.

Timer Operating Modes

Four different operating modes are available for the AR Timer:

Auto-reload Mode with PWM Generation. This mode allows a Pulse Width Modulated signal to be generated on the ARTIMout output pin with minimum Core processing time used.

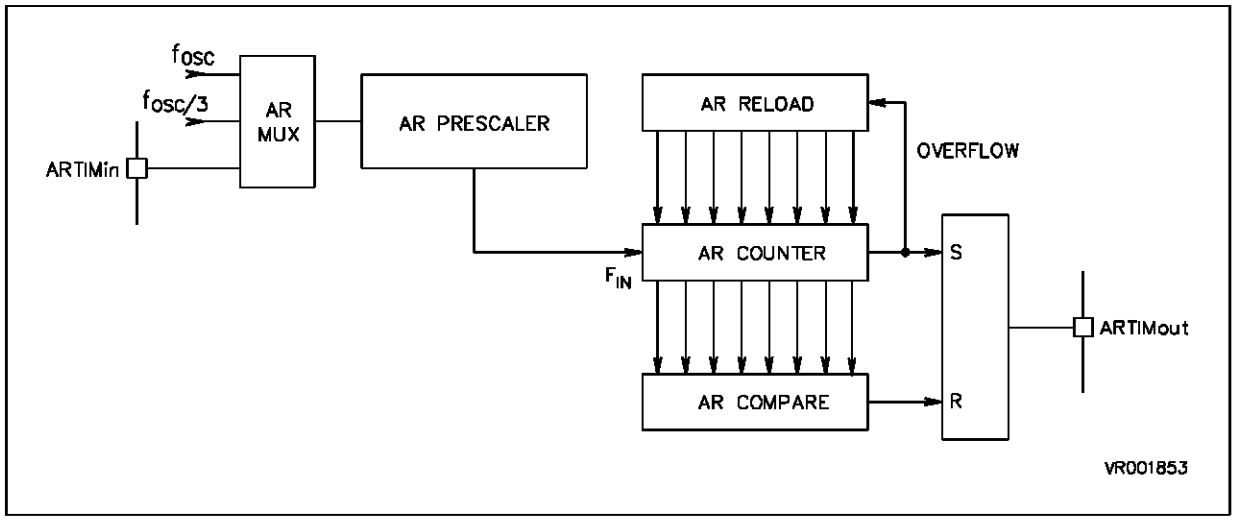
ARTC is a free running 8-bit counter fed by the ARPSC prescaler output and is incremented on every rising edge of the clock signal.
When a counter overflow occurs the ARTC counter is automatically reloaded with the contents of the Reload Register (ARRC, address D9h) while ARTIMout is set. When the counter reaches the value contained in the compare register ARCP, ARTIMout is reset.

At overflow, an overflow interrupt request is generated if the overflow interrupt enable bit, OVIE in the mode control register (ARMC, address D5h), is set to “1”.
When the counter reaches the compare value a compare interrupt request is generated if the Compare Interrupt enable bit, CPIE, in the Mode Control Register (ARMC, address D5h), is set to one. The interrupt service routine may then adjust the PWM period by loading a new value into ARCP.

The PWM signal is generated at ARTIMout (refer to block diagram) connected to the ARTIMout output pin. The frequency of this signal is controlled by the prescaler and by the auto-reload value present in the Reload/Capture register ARRC (address D9h). The duty cycle of the PWM signal is controlled by the Compare Register (ARCP, address DAh).

TIMERS (Continued)

Figure 35. Auto-reload Timer Block Diagram



Note that the reload values will also affect the value and the resolution of the duty cycle of PWM output signal. To achieve a ARTIMout signal the contents of the ARCP register must be greater than the contents of the ARRC register.

The maximum available resolution for the ARTIMout duty cycle is:

Resolution = 1/[255-(ARRC)]

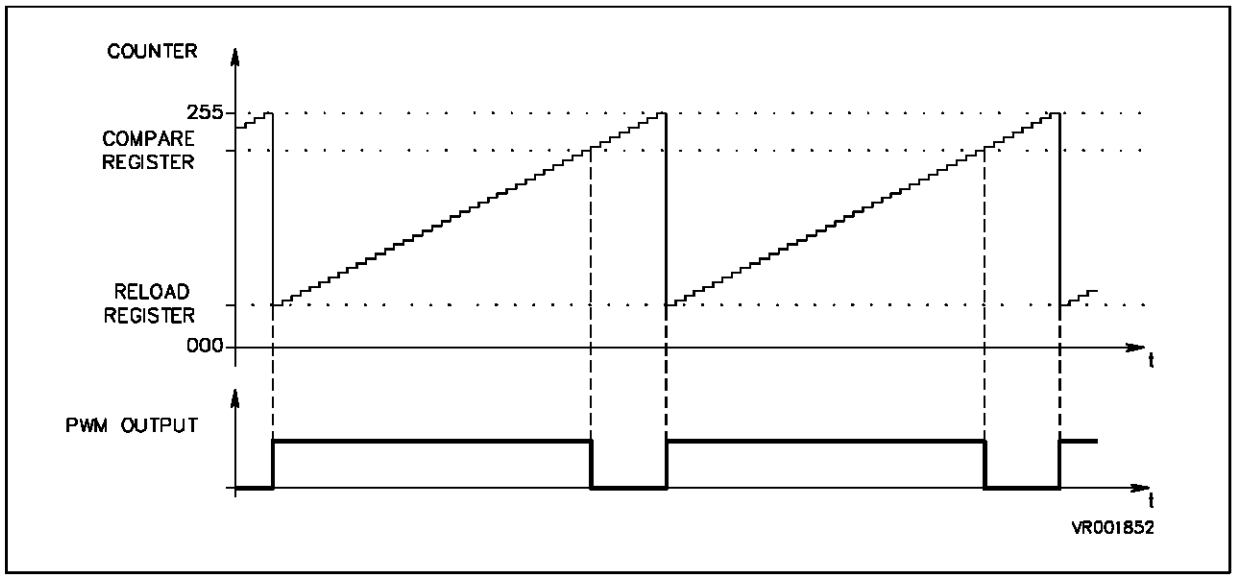
Where ARRC is the content of Reload/Capture register and the compare value loaded in the Compare Register, ARCP, must be in the range from (ARRC) to 255.

The initialization of the ARTC counter is made by writing into the ARRC register, then by setting the TCLD (Timer Load) and the TEN (Timer Clock Enable) bits in the Mode Control register ARMC.

The enable and the selection of clock sources are controlled by CC0, CC1, SL0 and SL1 bits in the Status Control Register ARSC1. The prescaler division ratio is selected by PS0, PS1 and PS2 bits in the ARSC1 Register.

In Auto-reload Mode three clock sources can be selected: Internal Clock (f_{OSC}), Internal Clock divided by 3 or the signal at the ARTIMin input pin.

Figure 36. Auto-reload Timer PWM Function



TIMERS (Continued)

Capture Mode with PWM Generation. In this case, ARTC is a free running 8-bit counter fed by the PSC prescaler output. ARTC is incremented on every clock rising edge.

An 8-bit capture operation from ARTC counter to ARRC register is performed on every active edge at ARTIMin/PC Input pin when enabled by Edge Control bits SL0, SL1 in the ARSC1 register. At the same time the External Flag EF, in the SC0 register, is set and an external interrupt request is generated if the External Interrupt Enable bit EIE, in the ARMC register, is set to one.

Each ARTC overflow sets ARTIMout, while a match between ARTC and ARCP (Compare Register) contents resets ARTIMout and sets the compare flag ARCPF and the compare interrupt request is generated if the related compare interrupt enable bit CPIE is set. A PWM signal is generated at ARTIMout.

The frequency of this signal is controlled by the prescaler. The duty cycle is controlled by register ARCP from 0-255/256.

Initialization and reading of ARTC counter are made in the same way as in the auto-reload mode (see previous paragraph).

The enable and selection of clock sources is controlled by CC0, CC1 bits in the AR Status Control Register ARSC1.

The prescaler division ratio is selected by PS0, PS1 and PS2 bits in the ARSC1 Register.

In Capture mode the possible clock sources are the internal clock and the internal clock divided by 3; the external ARTIMin input pin should not be used.

Capture Mode with Reset of ARTC, ARPSC and PWM Generation. This mode is identical to the previous one, with the difference that a capture condition also resets the ARTC counter and ARPSC prescaler allowing easy measurement of the time between two captures (for input period measurement on ARTIMin pin).

Load on External Input. ARTC is a free running 8-bit counter fed by the ARPSC prescaler. TC is incremented on every clock rising edge.

Each ARTC overflow sets the ARTIMout. A match between ARTC and ARCP (Compare Register) contents resets the ARTIMout and sets the compare flag CPF and the compare interrupt request is generated if the related compare interrupt enable bit CPIE is set. A PWM signal is generated at ARTIMout.

The initialization of ARTC can be done in the same way as described in the previous paragraph. In addition if the external ARTIMin input is enabled, an active edge on the input pin will copy the contents of the ARRC register into the ARTC counter, whether ARTC is running or not.

General Notes:

a - The allowed AR Timer clock sources are the following:

b - The timer clock frequency should not be modified while ARTC is counting as the ARTC counter may take an unpredictable value. For example the multiplexer setting should not be modified while ARTC is counting.

ARTimer Mode	Clock Sources
Auto-reload Mode	f_{OSC}, $f_{OSC}/3$, ARTIMin
Capture Mode	f_{OSC}, $f_{OSC}/3$
Capture/Reset Mode	f_{OSC}, $f_{OSC}/3$
External Load Mode	f_{OSC}, $f_{OSC}/3$

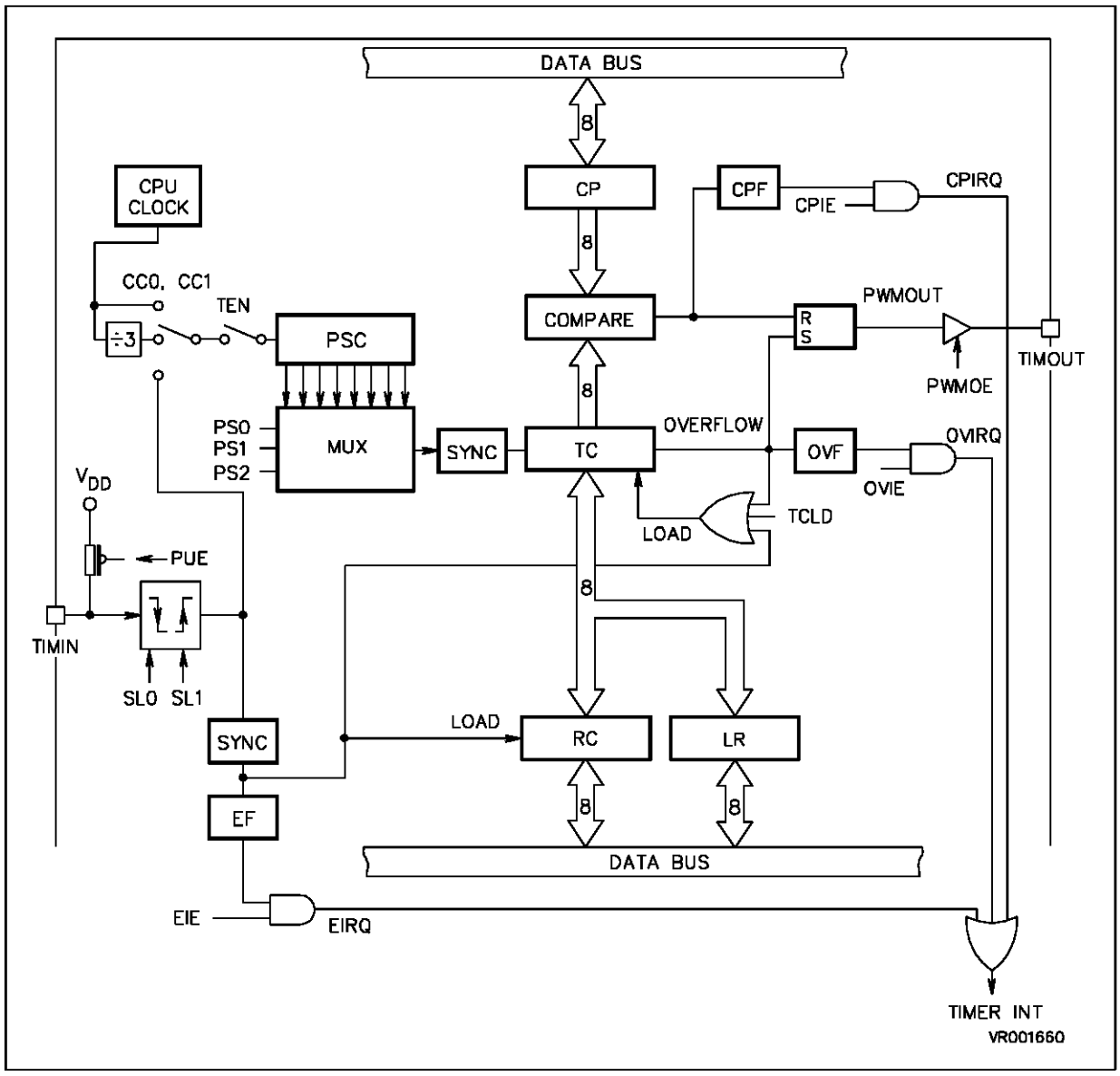
c - Any loading of ARTC (by auto-reload, through ARLR, ARRC or by the Core) resets ARPSC at the same time.

d - Care must be taken when using the auto-reload timer in Input capture mode with capture interrupt and overflow interrupt. Infact when the overflow interrupt routine is served, the user sw must clear the OVF flag in register ARSC0. However, since the capture event is asynchronous to the overflow event, it could happen when the sw is clearing the OVF flag.

In this case the capture event is lost, because the EF flag is set when the register ARSC0 is under manipulation by the CPU of the micro. To avoid this problem, the user software in the overflow interrupt routine must clear the OVF flag. Immediately after the pin PB6/ARTIMin has to be tested to verify if the capture event has happened. In conclusion the EF flag has to be set to generate the capture interrupt at the end of the overflow interrupt routine.

TIMERS (Continued)

Figure 37. ARTimer Detailed Block Diagram

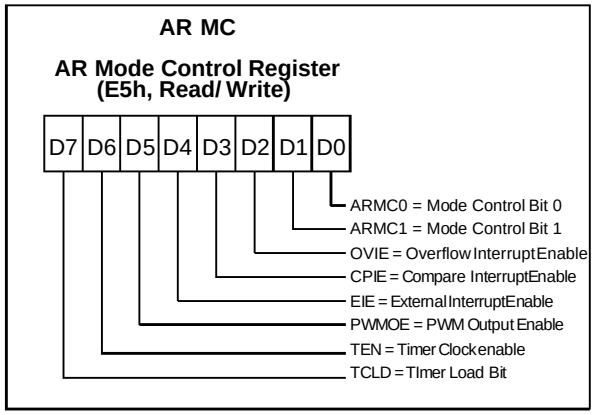


TIMERS (Continued)

AR Timer Registers

Mode Control Register ARMC. The AR Mode Control Register ARMC is used to program the different operation modes of AR Timer, to enable the clock of the Timer/Counter and to initialize it. It can be read and written by the Core and it is cleared to zero on system reset (AR Timer is disabled).

Figure 38. AR Mode Control Register



TCLD. This bit, when set to one, will cause the contents of ARRC register to be loaded into the ARTC counter and the contents of ARPSC register are cleared in order to initialize the timer before starting to count. This bit is write only and any attempt to read it will show a logical zero.

TEN. This bit, when set to one, will allow the timer to count. When cleared to zero it will stop the timer and freeze the ARPSC and ARTSC values.

PWMOE. This bit, when set, enables the PWM output to be carried on ARTIMout output pin. When cleared to zero the PWM output is disabled.

EIE. This bit, when set, enables the external interrupt request. If EIE = "0" the external interrupt request is masked. If EIE = "1" and the related flag EF in the ARSC0 register is also set an interrupt request is generated.

CPIE. This bit, when set, enables the compare interrupt request. If CPIE = "0" the compare interrupt request is masked. If CPIE = "1" and the related flag CPF into the ARSC0 register is also set an interrupt request is generated.

OVIE. This bit, when set, enables the overflow interrupt request. If OVIE = "0" the compare interrupt request is masked. If OVIE = "1" and the related flag OVF into the ARSC0 register is also set, an interrupt request is generated.

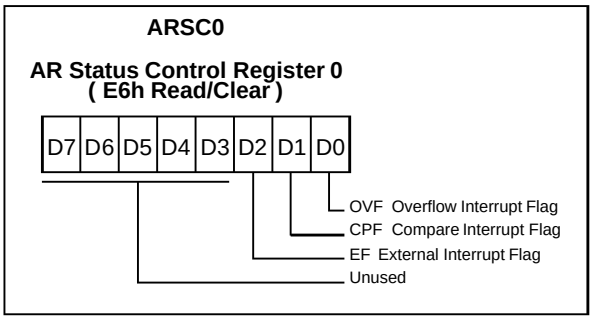
ARMC1, ARMC0. These are the operation mode control bits. The following bit combination will select the different operating modes:

ARMC1	ARMC0	Operating Mode
0	0	Auto-reload Mode
0	1	Capture Mode
1	0	Capture Mode with Reset of ARTC and ARPSC
1	1	Load on External Edge Mode

AR Timer Status/Control Registers ARSC0 & ARSC1. These registers provide the AR Timer status information bits and also allows the programming of clock sources, active edge and prescaler multiplexer programming.

ARSC0 register bits 0,1 and 2 contain the interrupt flags of the AR Timer. These bits can be read and cleared by the Core. A normal write operation is not possible, it is only possible to reset the bits by writing a zero on the selected position. Writing a one will not affect the bit flag bits.

Figure 39. AR Mode Control Register



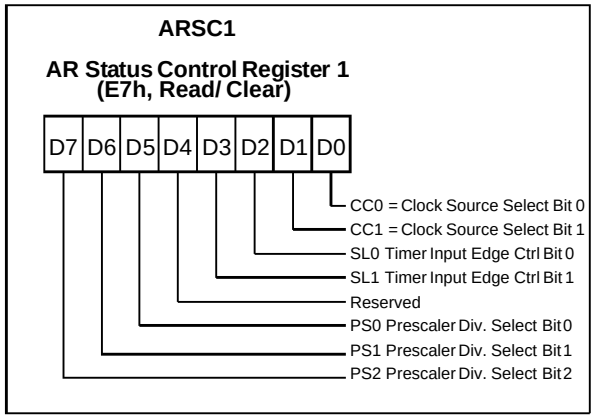
EF. This bit is set to one by any active edge at the external ARTIMin input pin. The flag is cleared by writing a zero in the EF bit.

CPF. This bit is set to one if the contents of ARTC counter and ARCP register are equal. The flag is cleared by writing a zero into ARCPF bit.

OVF. This bit is set to one by a transition of TC counter from FFh to 00h. The flag is cleared by writing a zero into OVF bit.

TIMERS (Continued)

Figure 40. AR Mode Control Register



PS2-PS0. These bits control the AR Prescaler division ratio. The prescaler itself is not affected by these bits. The AR PSC division is listed in the following Table 12:

Table 12. Prescaler Division Ratio Selection

PS2	PS1	PS0	PSC Division Ratio
0	0	0	1
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

D4. Reserved. Must be kept to 0.

SL1-SL0. These bits control the edge function on AR Timer input pin for external synchronization. If bit SL0 is cleared to zero the edge detection is disabled, if set to one the edge detection is enabled. If bit SL1 is cleared to zero the AR Timer input pin is rising edge sensitive, if set to one it is falling edge sensitive.

SL1	SL0	Edge Detection
X	0	Disabled
0	1	Rising Edge
1	1	Falling Edge

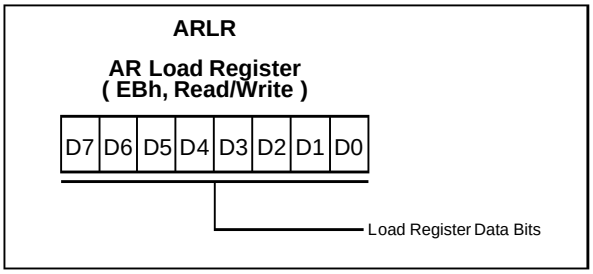
CC1-CC0. These bits select the clock source for the AR Timer through the AR Multiplexer. The programming of the clock sources is explained in the following Table 13:

Table 13. Clock Source Selection

CC1	CC0	Clock Source
0	0	Oscillator Clock
0	1	Oscillator Clock Divided by 3
1	0	ARTIMin Input Clock
1	1	Reserved

AR Load Register ARLR. The ARLR load register is used to read or write “on the fly” the ARTC counter register, while it is counting. ARLR register is not affected by system reset.

Figure 41. AR Load Register

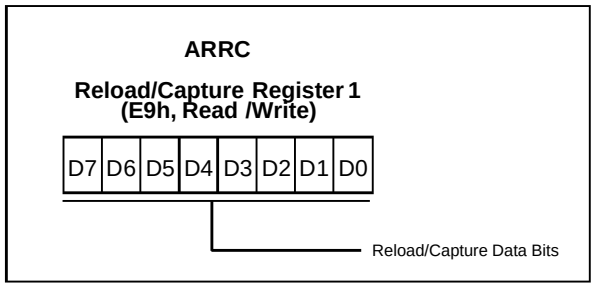


D7-D0. These are the load register data bits.

TIMERS (Continued)

AR Reload/Capture Register. The ARRC reload/capture register is used to hold the auto-reload value that is automatically loaded into ARTC counter from ARRC when overflow occurs.

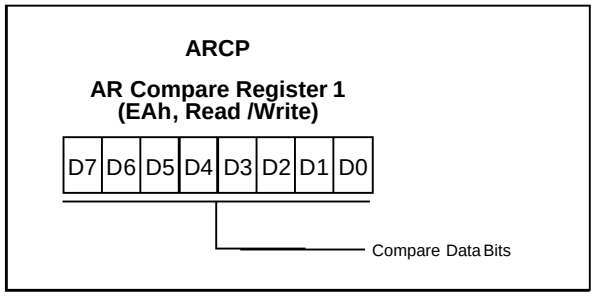
Figure 42. AR Reload/Capture Register



D7-D0. These are the Reload/Capture register data bits.

AR Compare Register. The CP compare register is used to hold the compare value to perform the compare function with TC counter.

Figure 43. AR Compare Register



D7-D0. These are the Compare register data bits.

Synchronization

The clock of the counter, the capture and reload operations and the reading of the reload/capture register are triggered by independent asynchronous sources. To prevent capture of a transient counter content or a reading of transient capture value the different sources are synchronized internally by the AR Timer synchronization logic. The Clock and Trigger sources, AR Timer Input and internal clock are synchronized with the processor read/write strobes. To avoid any loss of clock or trigger pulses the frequency of the external signal should be equal to or lower than 1/4 of the internal clock.

DIGITAL WATCHDOG

The ST6280 software activated digital watchdog consists of a down counter that can be used to provide a controlled recovery from a software upset. The watchdog can be enabled only by software. If enabled, any STOP instruction is interpreted as a WAIT instruction. The watchdog can not be disabled by software. The watchdog uses one data space register (DWDR location D8h). The watchdog register is set to 0FEh after reset and the watchdog function is disabled.

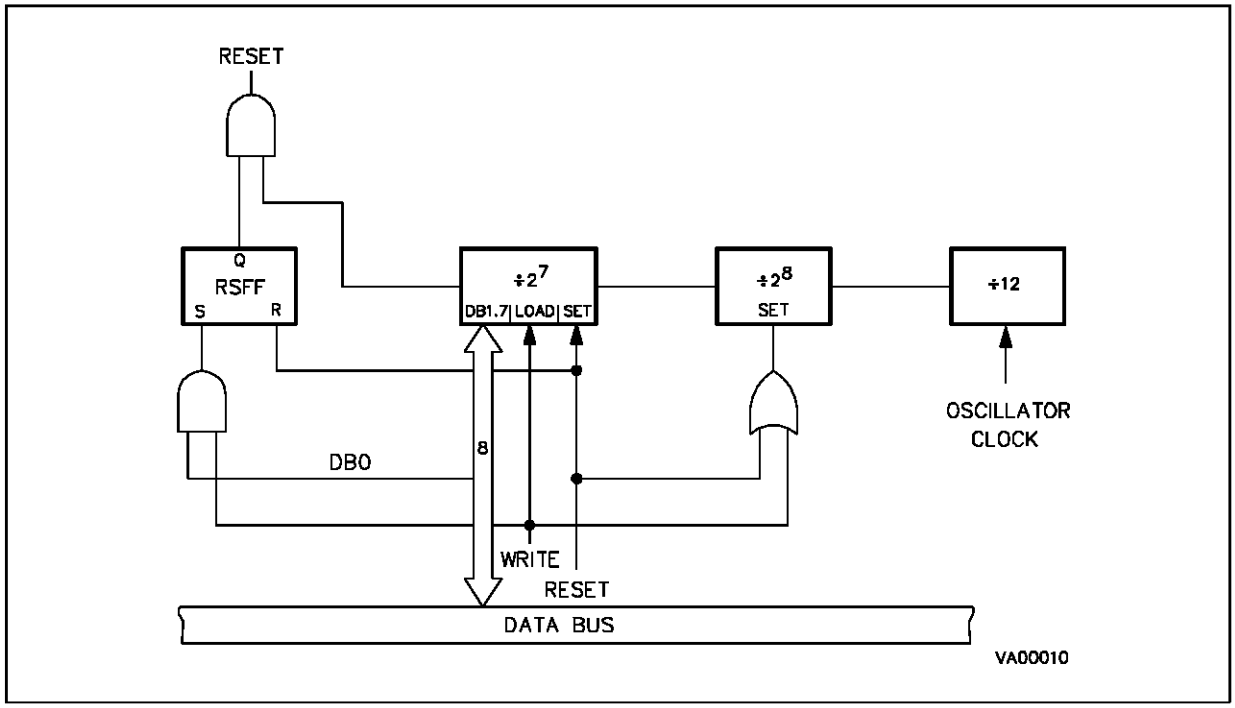
The watchdog time can be programmmed using the 6 Most Significant Bits in the Watchdog register. The check time can be set differently for different routines within the general program. Figure 44 shows the Watchdog block diagram, while Figure 46 shows its working principle.

After a reset the software Watchdog is in the offstate. The watchdog should be activated inside the Reset restart routine by writing a “1” in watchdog timer register bit 0. Bit one of this register must be set to one before programming bit zero as otherwise a reset will be immediately generated when bit 0 is set. This allows the user to generate a reset by software (bit 0 = “1”, bit 1 = “0”). Once bit 0 is set, it can not be cleared by software without generating a Reset. The delay time is defined by programming bits 2-7 of the Watchdog register. Bit 7 is the Least Significant Bit while bit 2 is the MSB. This gives the possibility to generate a reset in a time between 3072 to 196608 clock cycles in 64 possible steps: (With a clock frequency of 8MHz this means from 384µs to 24.576ms). The reset is prevented if the register is reloaded with the desired value before bits 2-7 decrement from all zeros to all ones. If the watchdog is active the STOP instruction is deactivated and a WAIT instruction is automatically executed instead of a STOP. If bit 0 of the watchdog register is never set to one then bits 1-7 of the register can be used as a simple 7-bit counter which is decremented every 3072 clock cycles.

Note:

In many applications the user may need to synchronize the Reset with the external circuitry. When the watchdog initiates the ST62xx reset the external RESET pin is held low until the on-chip reset circuit ensures the good start-up condition (see RESET description for additional information). This time is arround 50ns.

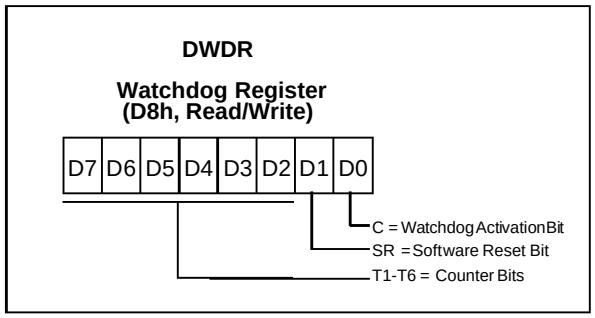
Figure 44. Digital Watchdog Block Diagram



DIGITAL WATCHDOG (Continued)

Watchdog Register

Figure 45. Watchdog Register



C. This is the watchdog activation bit, that, if set to one, will activate the watchdog function. When cleared to zero it allows the use of the counter as a 7-bit timer. This bit is cleared on reset.

SR. This bit is set to one during the reset and will generate a software reset if cleared to zero. When C = "0" (watchdog disabled) it is the MSB of the 7-bit timer.

T1-T6. These are the watchdog counter bits. It should be noted that D7 (T1) is the LSB of the counter and D2 (T6) is the MSB of the counter. These bits are in the opposite order to normal.

Application notes:

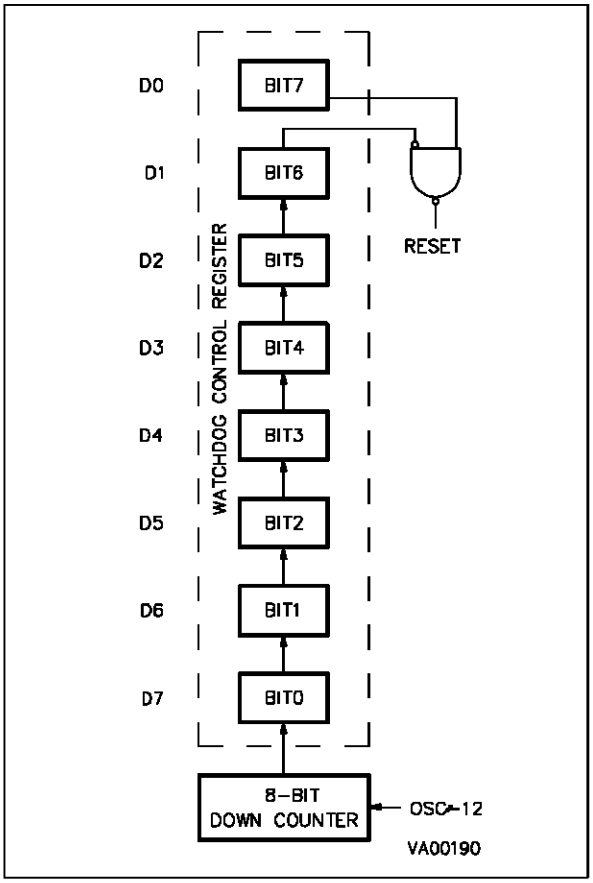
If the watchdog is not used during power-on reset, external noise may cause the undesired activation of the watchdog with a generation of an unexpected reset. To avoid this risk, two additional instructions, that check the state of the watchdog and eventually reset the chip, are needed within the first 27 instructions after the reset. These instructions are:

```
jrz 0, WD, #+3
ldi WD, 0FDh
```

These instructions should be executed at the very beginning of the program.

If the watchdog is used, the watchdog register may be set to a low value during power-on reset that may generate a reset after a minimum of 28 instructions. To avoid this unwanted reset, the watchdog must be set to the desired value within the first 27 instructions. It is recommended to put this at the very beginning of the reset routine.

Figure 46. Digital Watchdog Working Principle



Alternatively the normal legal state can be checked with the following short routine:

```
ldi a, 0FEh
and a, WD
cpi a, 0FEh
jrz #+3
ldi WD, OFDh
```

This sequence is recommended for security applications, where possible stack confusion error loops must be avoided and the watchdog must only be refreshed after extensive checks.

8-BIT A/D CONVERTER

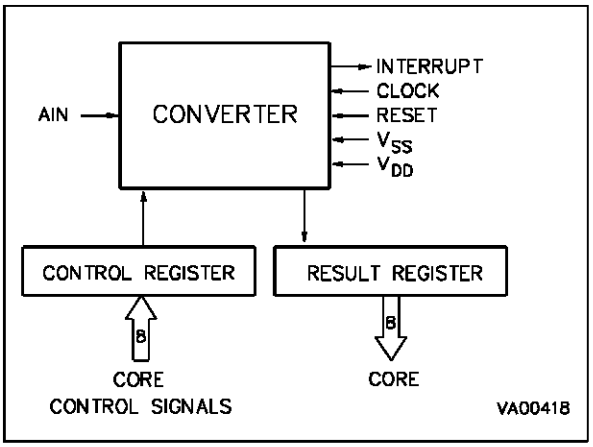
The A/D converter of ST6280 is an 8-bit analog to digital converter with up to 12 analog inputs (as alternate functions of I/O lines PB0-PB7, PC4-PC7) offering 8-bit resolution with total accuracy ±2 LSB and a typical conversion time of 70µs (clock frequency of 8MHz).

The A/D peripheral converts the input voltage by a process of successive approximations using a clock frequency derived from the oscillator with a division factor of twelve. With an oscillator clock frequency less than 1.2MHz, the A/D converter accuracy is decreased.

The selection of the pin signal that has to be converted is done by configuring the related I/O line as analog input through the I/O ports option and data registers (refer to I/O ports description for additional information). Only one I/O line must be configured as analog input at a time. The user must avoid the situation in which more than one I/O pin is selected to be analog input to avoid malfunction of the ST62xx.

The ADC uses two registers in the data space: the ADC data conversion register which stores the conversion result and the ADC control register used to program the ADC functions.

Figure 47. A/D Converter Block Diagram



A conversion is started by writing a “1” to the Start bit (STA) in the ADC control register. This automatically clears (resets to “0”) the End Of Conversion Bit (EOC). When a conversion has been finished this EOC bit is automatically set to “1” in order to flag that conversion is complete and that the data in the ADC data conversion register is valid. Each conversion has to be separately initiated by writing to the STA bit.

The STA bit is continually being scanned so that if the user sets it to “1” while a previous conversion is in progress then a new conversion is started before the previous one has been completed. The start bit (STA) is a write only bit, any attempt to read it will show a logical “0”.

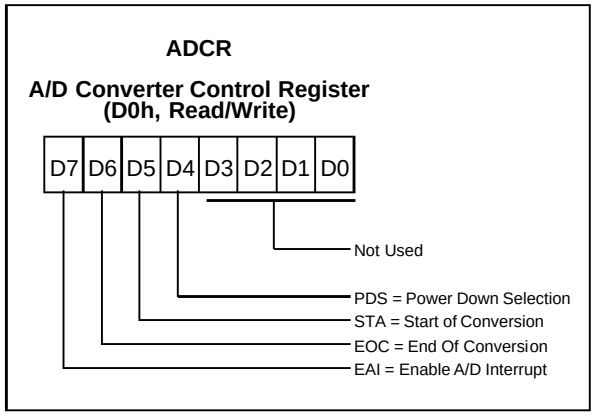
The A/D converter has a maskable interrupt associated to the end of conversion. This interrupt is associated to the interrupt vector #4 and occurs when the EOC bit is set, i.e. when a conversion is completed. The interrupt is masked using the EAI (interrupt mask) bit in the control register.

The power consumption of the device can be reduced by turning off the ADC peripheral. That is achieved when the PDS bit in the ADC control register is cleared to “0”. If PDS=“1”, the A/D is supplied and enabled for conversion. This bit must be set at least one instruction before the beginning of the conversion to allow the stabilization of the A/D converter. *This action is needed also before entering the STOP instruction as the A/D comparator is not automatically disabled by the STOP mode.*

During reset any conversion in progress is stopped, the control register is reset to all zeros and the A/D interrupt is masked (EAI=“0”).

8-BIT A/D CONVERTER (Continued)

Figure 48. A/D Converter Control Register



EAI. If this bit is set to one the A/D interrupt (vector #4) is enabled, when EAI=0 the interrupt is disabled.

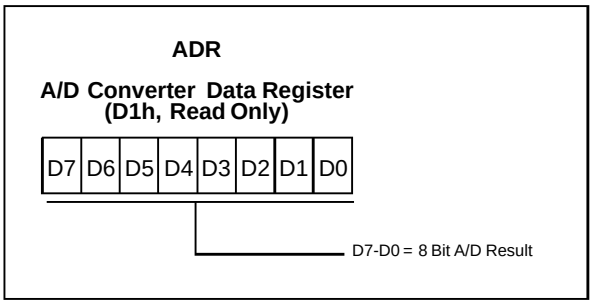
EOC. *Read Only*; This read only bit indicates when a conversion has been completed. This bit is automatically reset to zero when the STA bit is written. If the user is using the interrupt option then this bit can be used as an interrupt pending bit. Data in the data conversion register are valid only when this bit is set to one.

STA. *Write Only*, Writing a "1" in this bit will start a conversion on the selected channel and automatically reset to zero the EOC bit. If the bit is set again when a conversion is in progress, the present conversion is stopped and a new one will take place. This bit is write only, any attempt to read it will show a logical zero.

PDS. This bit activates the A/D converter if set to 1. Writing a zero into this bit will put the ADC in power down mode (idle mode).

D3-D0. Not used

Figure 49. A/D Converter Data Register



D7-D0. *Read Only*; These are the conversion result bits; the register is read only and stores the result of the last conversion. The contents of this register are valid only when EOC bit in the ADCR register is set to one (end-of-conversion).

Notes:

The ST62xx A/D converter does not feature a sample and hold. The analog voltage to be measured should therefore be stable during the conversion time. Variation should not exceed ±1/2 LSB for the best accuracy in measurement.

Since the ADC is on the same chip as the microprocessor the user should not switch heavily loaded output signals during conversion if high precision is needed. This is because such switching will affect the supply voltages which are used for comparisons.

A low pass filter can be used at the analog input pins to reduce input voltage variation during the conversion. For maximum accuracy, the impedance of the analog voltage sources should be less than 30kΩ while the impedance of the reference voltage should not exceed 2kΩ.

8-BIT A/D CONVERTER (Continued)

The accuracy of the conversion depends on the quality of the power supply voltages (V_{DD} and V_{SS}). The user must specially take care of applying regulated reference voltage on the V_{DD} and V_{SS} pins (the variation of the power supply voltage must be inferior to 5V/ms).

The converter can resolve the input voltage with an resolution of:

$$\frac{V_{DD} - V_{SS}}{256}$$

So if operating with a supply voltage of 5V the resolution is about 20mV.

The Input voltage (Ain) which has to be converted must be constant for 1µs before conversion and remain constant during the conversion.

The resolution of the conversion can be improved if the power supply voltage (V_{DD}) of the microcontroller becomes lower.

In order to optimize the resolution of the conversion, the user can configure the microcontroller in the WAIT mode because this mode allows the minimization of the noise disturbances and the variations of the power supply voltages due to the switching of the outputs. Nevertheless, it must be take care of executing the WAIT instruction as soon as possible after the beginning of the conversion because the execution of the WAIT instruction may provide a small variation of the V_{DD} voltage (the negative effect of this variation is minimized at the beginning of the conversion because the latter is less sensitive than the end of the conversion when the less significant bits are determined).

The best configuration from a accuracy point of view is the WAIT mode with the Timer and LCD driver stopped. Indeed, only the ADC peripheral and the oscillator are still working. The MCU has to be wake-up from the WAIT mode by the interrupt of the ADC peripheral at the end of the conversion. It must be noticed that the wake-up of the microcontroller could be done also with the interrupt of the TIMER, but in this case, the Timer is working and some noise could disturb the converter in terms of accuracy.

32KHz STAND-BY OSCILLATOR

The 32kHz stand-by oscillator allows the ST6280 to generate real time interrupts and to supply the clock to the LCD driver. This enables the ST6280 to provide real time functions with the LCD display capability. Figure 50 shows the 32kHz oscillator block diagram.

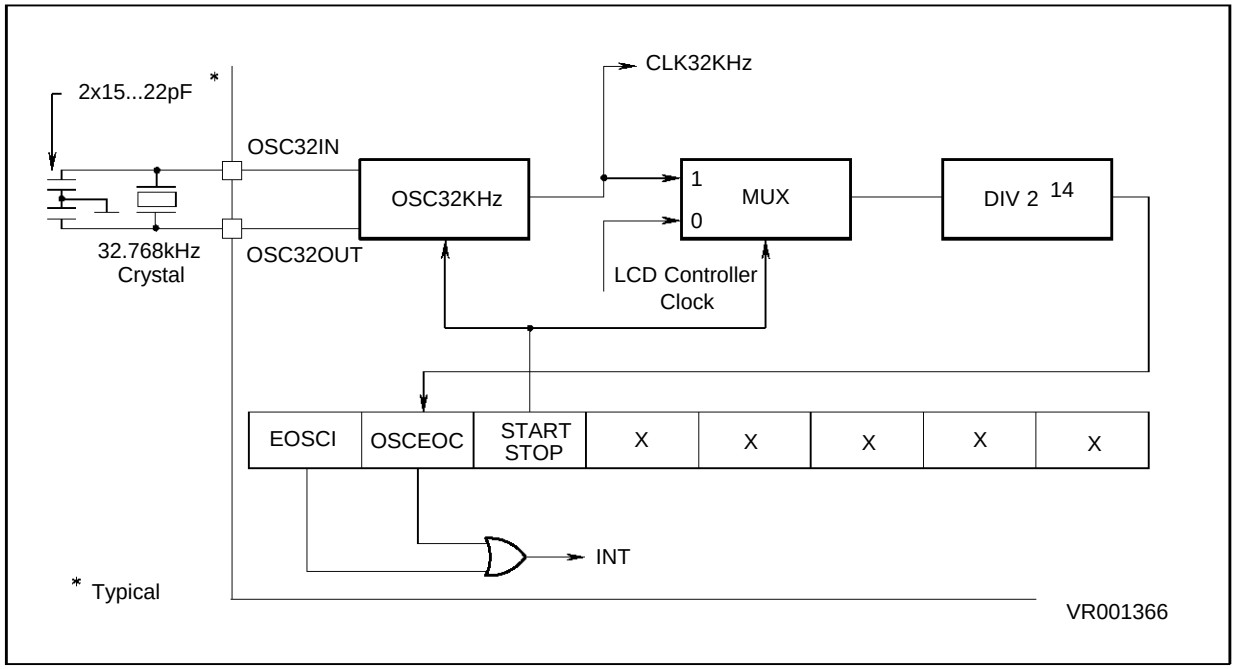
A 32.768kHz quartz crystal must be connected to the OSC32IN and OSC32OUT pins to perform the real time clock operation. Two external capacitors of 15-25pF (typical) each must be connected between the oscillator pins and ground. The 32kHz oscillator is managed by the dedicated status/control register located at address 0DBh.

When the 32kHz stand-by oscillator is stopped (bit 5 of the Status/Control register cleared) the divider chain is supplied with a clock signal synchronous with machine cycle ($f_{OSC}/13$), this produces an interrupt request every $13x2^{14}$ clock cycles (i.e. 26.624ms) with an 8MHz quartz crystal.

When the 32kHz stand-by oscillator is enabled (bit 5 of the Status/Control register set to one) the divider chain is directly supplied with the 32kHz oscillator. The 32kHz clock from the standby oscillator can also be used as the LCD clock. This allows operation of the LCD in STOP mode. The interrupt output of the 32kHz oscillator peripheral generates an interrupt request every half second (500ms) when a 32.768 kHz crystal is used. This can be used to perform a real time clock function when the MCU is in STOP mode.

This interrupt signal is “ORed” with the interrupt request signals of the two on-chip timers and connected to the low level sensitive interrupt input associated to the interrupt vector #3 (FF2h, FF3h). The interrupt request must to be cleared by user software before leaving the interrupt service routine. Discrimination between the three interrupts sources is made by polling the Status/Control registers of Timer 1, ARTimer and 32kHz oscillator.

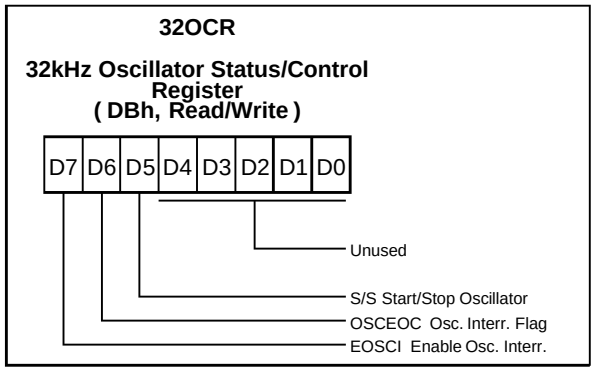
Figure 50. 32KHz Oscillator Block Diagram



32kHz STAND-BY OSCILLATOR (Continued)

32kHz Oscillator Status/Control Register

Figure 51. 32KHz Oscillator Register



EOSCI. Enable Oscillator Interrupt. This bit, when set, enables the 32kHz oscillator interrupt request.

OSCEOC. Oscillator Interrupt Flag. This bit indicates when the 32kHz oscillator has measured a 500ms elapsed time (providing a 32.768kHz quartz crystal is connected to the 32kHz oscillator dedicated pins). An interrupt request can be generated in relation to the state of EOSCI bit. This bit must be cleared by the user program before leaving the interrupt service routine.

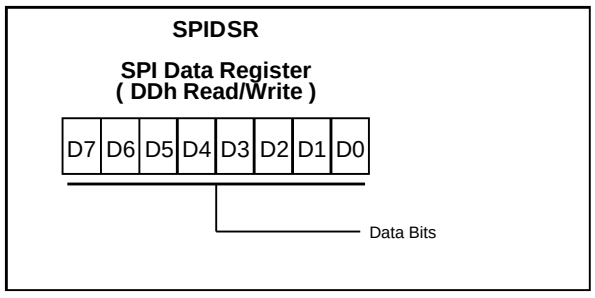
START/STOP. Oscillator Start/Stop bit. This bit, when set, enables the 32kHz stand-by oscillator and the free running divider chain is supplied by the 32kHz oscillator signal. When this bit is cleared to zero the divider chain is supplied with LCD controller clock signal.

This register is cleared after reset.

Note:

To achieve minimum power consumption in STOP mode (no system clock), the stand-by oscillator must be switched off (real time function not available) by clearing the Start/Stop in the oscillator status/control register.

Figure 52. SPI Data/Shift Register



SERIAL PERIPHERAL INTERFACE (SPI)

The ST6280 SPI is an optimized serial interface that supports a wide range of industry standard SPI specifications. The ST6280 SPI is controlled by small and simple user software to perform serial data exchange. The serial shift clock can be implemented either by software (using the bit-set and bit-reset instructions), with the on-chip timer 1 by externally connecting the SPI clock pin to the timer pin or by directly applying an external clock to the SPI.

The peripheral is composed by an 8-bit data/shift register (address DDh) and a 4-bit binary counter. The SCL, SIN and SOUT SPI data and clock signals are connected to the PA5, PA6 and PA7 I/O lines. With the 3 I/O pins, the SPI can operate in the following operating modes: Software SPI, S-BUS, I^2CBUS and as a standard serial I/O (clock, data, enable). An interrupt request can be generated after eight clock pulses. Figure 53 shows the SPI block diagram.

The PA5/SCL line clocks, on the falling edge, either the shift register and the counter. To allow SPI operation the PA5/SCL must be programmed as input, an external clock supplied to this pin will drive the SPI peripheral (slave mode).

If PA5/SCL is programmed as output, a clock signal can be generated by software, setting and resetting the port line by software (master mode).

The SCL clock signal is the shift clock for the SPI data/shift register. The PA6/SIN pin is the serial shift input and PA7/SOUT is the serial shift output. These two lines can be tied together to implement two wires protocols (I^2C bus, etc). When data is serialized, the MSB is the first bit. PA6/SIN has to be programmed as input. For serial output operation PA7/SOUT has to be programmed as open-drain output.

After 8 clock pulses (D7..D0) the output $\overline{Q4}$ (see Figure 53) of the 4-bit binary counter becomes low, disabling the clock from the counter and the data/shift register. Q4 (see Figure 53) enables the clock to generate an interrupt on every falling edge as long as no reset of the counter (processor write into the 8-bit data/shift register) takes place. After a processor reset the interrupt is disabled. The interrupt is active when writing data in the shift register (DDh) and desactivated when writing any data in the register SPI Interrupt Disable (C2h).

The generation of an interrupt to the Core provides information that new data is available (input mode) or that transmission is completed (output mode), allowing the Core to generate an acknowledge on the 9th clock pulse (I^2C bus).

SERIAL PERIPHERAL INTERFACE (Continued)

Since the SPI interrupt is connected to interrupt #1, the falling edge interrupt option should be selected by clearing to zero bit 6 of the interrupt option register (IOR, C8h).

After power on reset, or after writing the data/shift register, the counter is reset to zero and the clock is enabled. In this condition the data shift register is ready for reception. No start condition has to be detected. Through the user software the Core may pull down the SIN line (Acknowledge) and slow down the SCL, as long as it is needed to carry out data from the shift register.

I^2CBUS Master-Slave, Receiver-Transmitter

When pins SIN and SOUT are externally connected together it is possible to use the SPI as a receiver as well as a transmitter. With a simple software routine (by using bit-set and bit-reset on I/O line) a clock can be generated allowing I^2CBUS to work in master mode.

When implementing an I^2C bus protocol, the start condition can be detected by setting the processor into a “wait for start” condition by simply enabling the interrupt of the PA6/SIN I/O port. This frees the processor from polling the SIN and SCL lines. After the transmission/reception the processor has to poll for the STOP condition.

In slave mode the user software can slow down the SCL clock frequency by simply putting the SCL I/O line in output open-drain mode and writing a zero into the corresponding data register bit.

As it is possible to directly read the SIN pin directly through the port register, the software can detect a difference between internal data and external data (master mode). Similar condition can be applied to the clock.

The typical speed of transmission in I^2C master or slave mode is in the range of 10kHz.

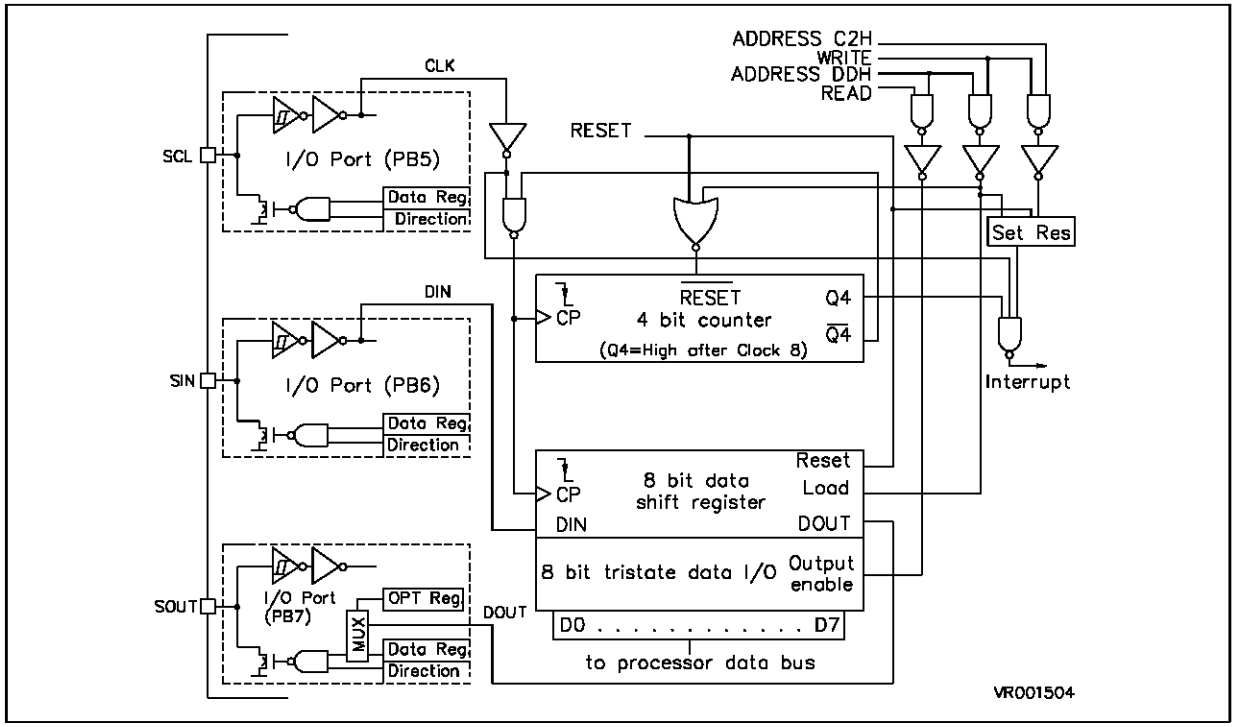
Three (Four) Wire Serial Bus

It is possible to use a single general purpose I/O pin (with the corresponding interrupt enabled) as a “chip enable” pin. SCL acts as active or passive clock pin, SIN as data in and SOUT as data out (four wire bus). SIN and SOUT can be connected together externally to implement three wire bus.

Note:

When the SPI is not used, the three I/O lines (SIN, SCL, SOUT) can be used as normal I/O, with the following limitation: bit SOUT cannot be used in open drain mode as this enables the shift register output to the port.
It is recommended, in order to avoid spurious interrupts from the SPI, to disable the SPI interrupt (the default state after reset) i.e. no write must be made to the 8-bit shift register (DDh). An explicit interrupt disable may be made in software by a dummy write to address C2h.

Figure 53. SPI Block Diagram



LCD CONTROLLER/DRIVER

The ST6280 includes an LCD driver designed for driving Dot-Matrix style multiplexed LCD panels. The multiplexing display modes are software selectable to cover a wide range of application requirements.
The driver has 56 segment outputs, 8 backplanes and 8 outputs selectable by software to be additional segment or backplane outputs, allowing drive of up to 56x16 = 896 display segments. Three multiplex modes (1/16, 1/11 and 1/8 MUX) and the LCD drive frequency can be selected by software.

When the 1/8 duty cycle multiplex MUX mode is selected, the backplane outputs 9-16 can be used as additional segment outputs allowing drive of up to 64x 8 = 512 segments.

Note:

In 1/11 duty cycle MUX mode, the inactive backplanes COM12 to COM16 show redundant signals and should be left open.

After reset the LCD driver is held in an inactive state and the LCD is switched off. After activation by a write operation to the LCD control register, the LCD driver reads automatically the data from the LCD RAM pages for display independently of the ST6 program and displays the data continuously on the segment outputs. When the display is turned off, all segment and backplane outputs are switched to ground, causing all the segments to be switched off regardless of the contents of the LCD RAM. LCD RAM memory which is not used for the LCD display can be used as standard RAM memory.

The LCD driver is managed by the LCD Mode/Control register located at data RAM address DCh. This register allows software control of the display mode, LCD clock and division ratio, the frame frequency and display on/off. It is cleared at reset and the display is turned off.

The scale factor of the clock prescaler can be fixed by software, therefore different frame frequencies can be defined.

The ST6280 oscillator can operate with various quartz crystal frequencies from 1 to 8.4 Mhz. Using the adequate crystal and programmable LCD clock division rate, an internal reference frequency of 32.768 kHz can be obtained. It is not recommended to select an internal frequency lower than 16 kHz as the clock supervisor circuit may switch off the LCD peripheral if the lower frequency is detected.

The LCD Driver clock can also be supplied by the 32kHz real-time oscillator, when the Stand-by oscillator function is selected, allowing display in low power conditions and real time clock operations.

Special care must be taken when entering the STOP mode. After switching the LCD clock source from the main oscillator to the 32kHz standby oscillator it must be guaranteed that enough clock pulses are delivered to complete the current frame before entering the STOP mode. Otherwise the LCD function will not be changed and the LCD will be switched OFF after entering the STOP mode.

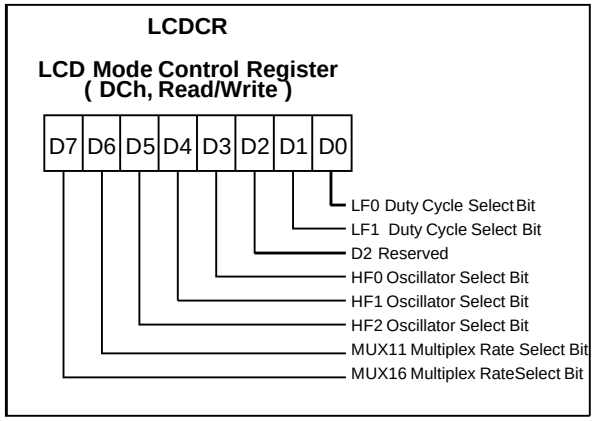
Table 14. Possible Display Configurations

Segments	Common	Size of display		
		1/16 MUX	1/11 MUX	1/8 MUX
56	8+8	56x16 896	56x11 616	64x8 512
48	8+8	48x16 768	48x11 528	56x8 448

LCD CONTROLLER/DRIVER (Continued)

LCD Mode Control Register

Figure 54. LCD Mode Control Register



MUX16, MUX11. Multiplex method control. These bits select the multiplex method used for the backplanes. If MUX16 is set, the display driver operates in 1/16 duty cycle MUX mode. If MUX11 is set, the display driver operates in 1/11 duty cycle MUX mode, the inactive backplanes COM12 to COM16 show redundant signals and should be left open. If both bits are set to zero, 1/8 duty cycle is selected and COM9-16 are switched to segment mode.

Table 15. Multiplex Rate And Operation Mode of COM9-16

MUX16	MUX11	MUX	COM 9-16
0	0	1 / 8	SEGMENT
1	0	1 / 11	COMMON
0	1	1 / 16	COMMON
1	1	————	not used

HF2, HF1, HF0. These bits allow the LCD controller clock to be supplied with the correct clock frequency when different high main oscillator frequencies are selected as system clock. The design of the LCD driver has been optimised to work at a frequency of 32.768kHz and it is recommended to use a clock frequency in this range. The programmable prescaler allows use of higher frequencies for the operation of the ST6280.
The 32kHz oscillator can be used to directly drive the display.

When the DISPLAY OFF function is selected, all segment and backplane outputs are switched to ground. All display dots are turned off, regardless of the contents of the LCD RAM.

Table 16. LCD Clock Selection

f_{OSC}	Prescaler	HF2	HF1	HF0	Function
8.388MHz	1 : 256	1	1	0	prescaler active
4.194MHz	1 : 128	1	0	1	prescaler active
2.097MHz	1 : 64	1	0	0	prescaler active
1.048MHz	1 : 32	0	1	1	prescaler active
		0	1	0	not used
		0	0	1	F_{EXT} (32kHz)
		0	0	0	DISPLAY OFF

To avoid incomplete frames of the LCD, the mode control bits do not immediately influence the LCD controller when the LCD control register is written. They are stored in a temporary register and change the LCD function only at the end of the frame.

D2. Reserved. This bit is reserved and must be set to “0”.

LF1, LF0. These bits select the four different LCD frame frequencies for each display mode. Values are given Table 17 providing the master clock is 32.768kHz.

Table 17. LCD Frame Frequency Selection

LF1	LF0	Ratio	Mode/Frame Frequency		
			1/16	1/11	1/8
1	1	1 : 1	256Hz	372Hz	512Hz
0	0	1 : 2	128Hz	186Hz	256Hz
1	0	1 : 3	85Hz	124Hz	170Hz
0	1	1 : 4	64Hz	93Hz	128Hz

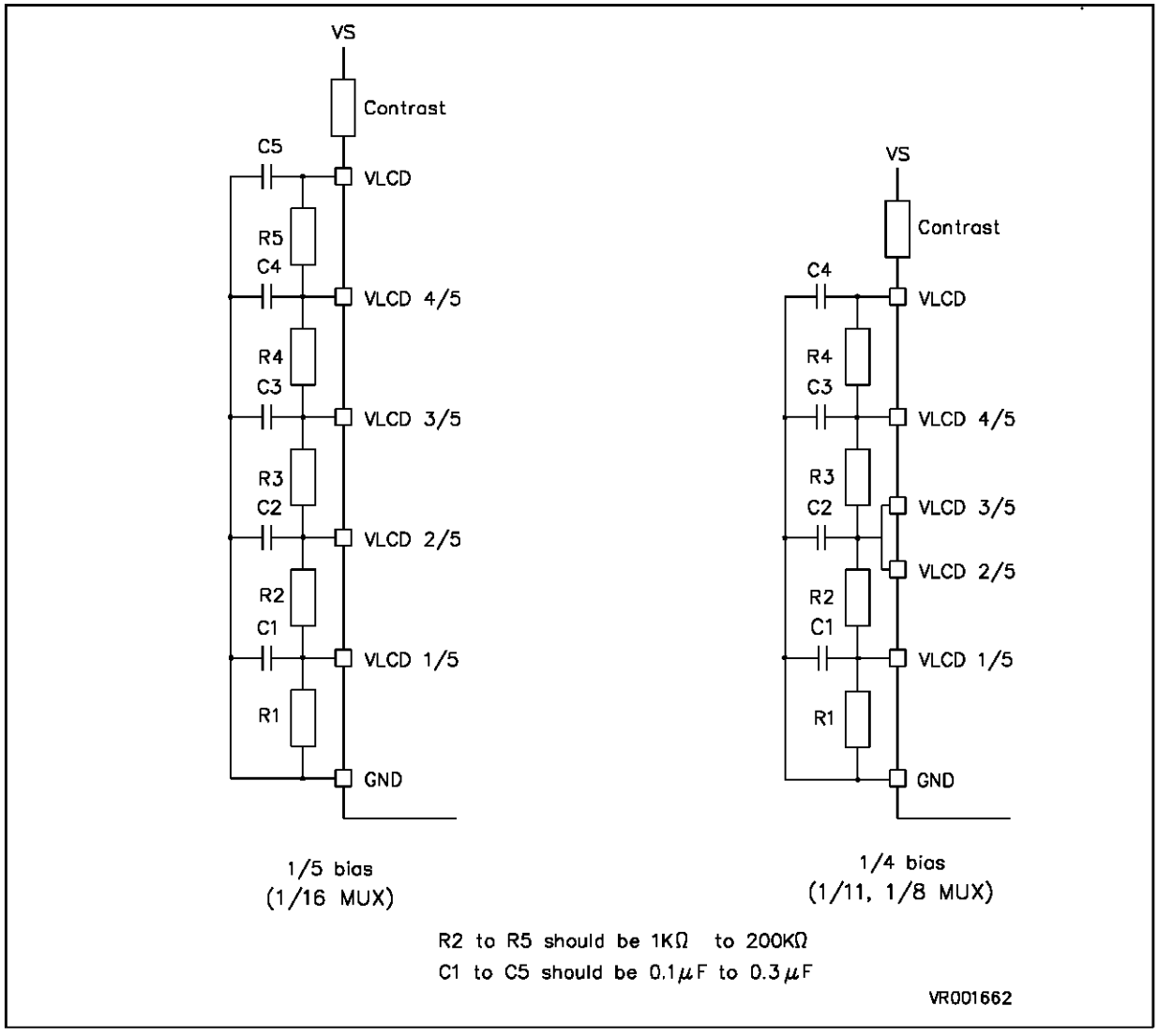
LCD CONTROLLER/DRIVER (Continued)

Driver Voltage Chain

The display voltage levels are supplied by an **external** resistor chain as shown in Figure 55. Two configurations with five or four display voltage levels can be chosen.

The resistors used must have good tolerance matching within 1% to avoid DC voltage levels on the liquid crystal device. DC levels trigger electrode reactions in the liquid crystal cell, causing a rapid deterioration of the display quality.

Figure 55. External Divider Chain



LCD CONTROLLER/DRIVER (Continued)

Address Mapping of the Display Segments.

The LCD RAM is located in the ST6280 data space in two pages of 64 bytes from addresses 00h to 3Fh. The LCD forms a matrix of 56 segment lines (columns) and 8 backplane lines (rows) or 48 segment lines and 11 or 16 backplane lines according to the chosen operating mode. Each bit of the LCD RAM is mapped to one dot of the LCD matrix, as described in Figure 56. If a bit is set, the corresponding LCD dot is switched on; if it is reset, the dot is switched off.

If 1/8 duty cycle mode is selected (56x8 dot matrix), only RAM page 1 is used for display data storage. In this case page 2 is completely free for common data storage.

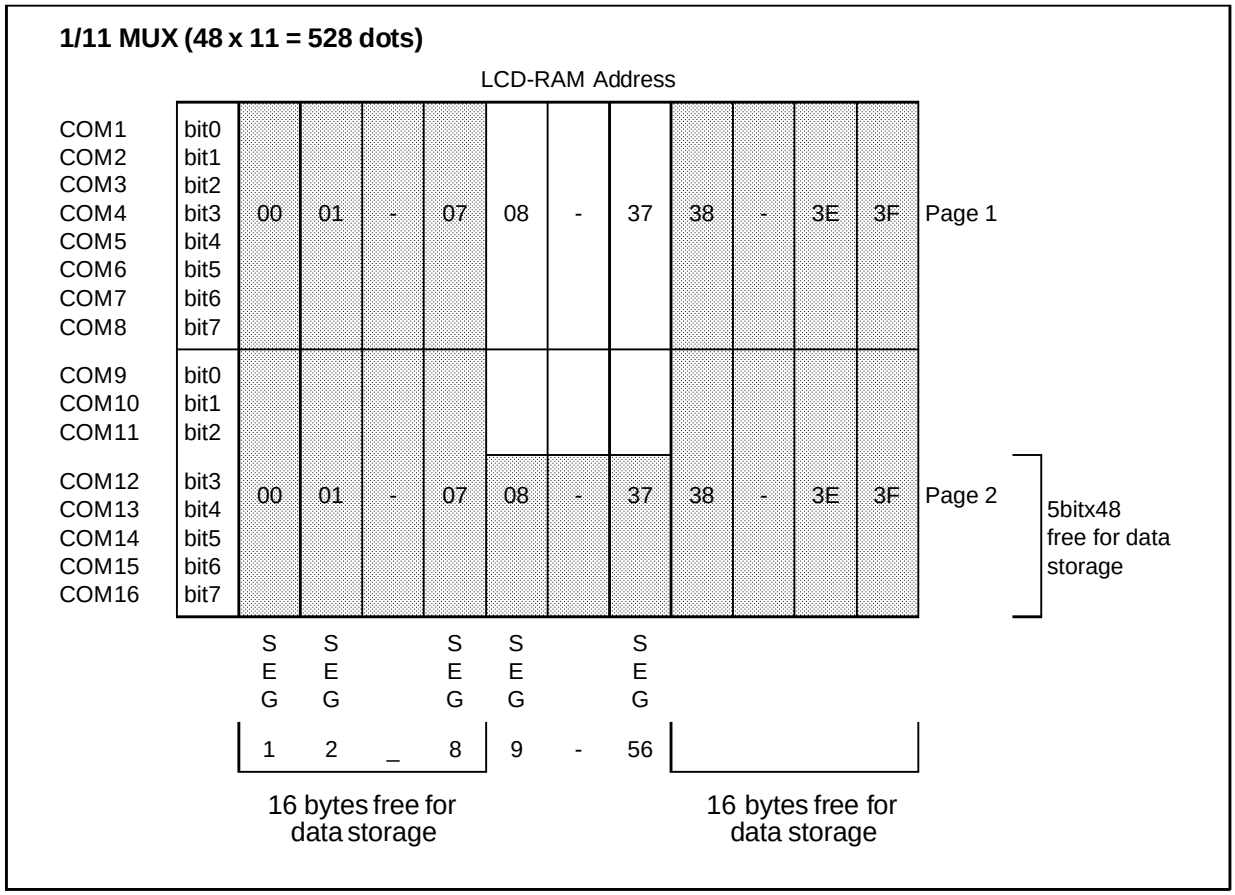
If 1/16 duty cycle mode is selected (48x16 dot matrix), RAM pages 1 and 2 are used for display data storage. In this case addresses 00 to 07 in both pages are free for common data storage.

If 1/11 duty cycle mode is selected (48x11 dot matrix), RAM pages 1 and 2 are used for display data storage. In this case addresses 00 to 07 in both pages and bits 3 to 7 in RAM Page 2 are free for common data storage.

In all display modes 16 bytes from address 38h to 3Fh in RAM Pages 1 and 2 are free for common data storage.

After reset, the LCD RAM is not initializated and contains arbitrary information. As the LCD control register is reset, the LCD is completely switched off.

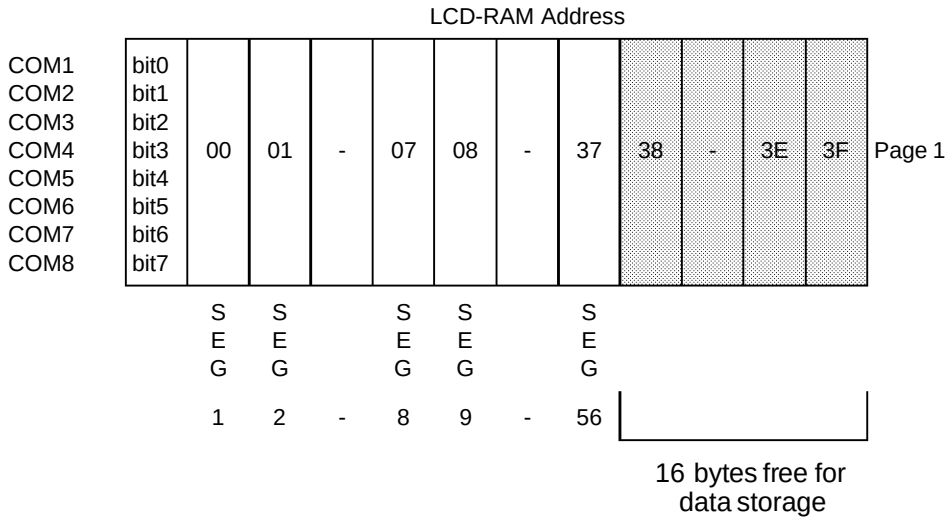
Figure 56. Addressing Mapping of the LCD RAM



LCD CONTROLLER/DRIVER (Continued)

Figure 57. Addressing Mapping of The LCD RAM (Continued)

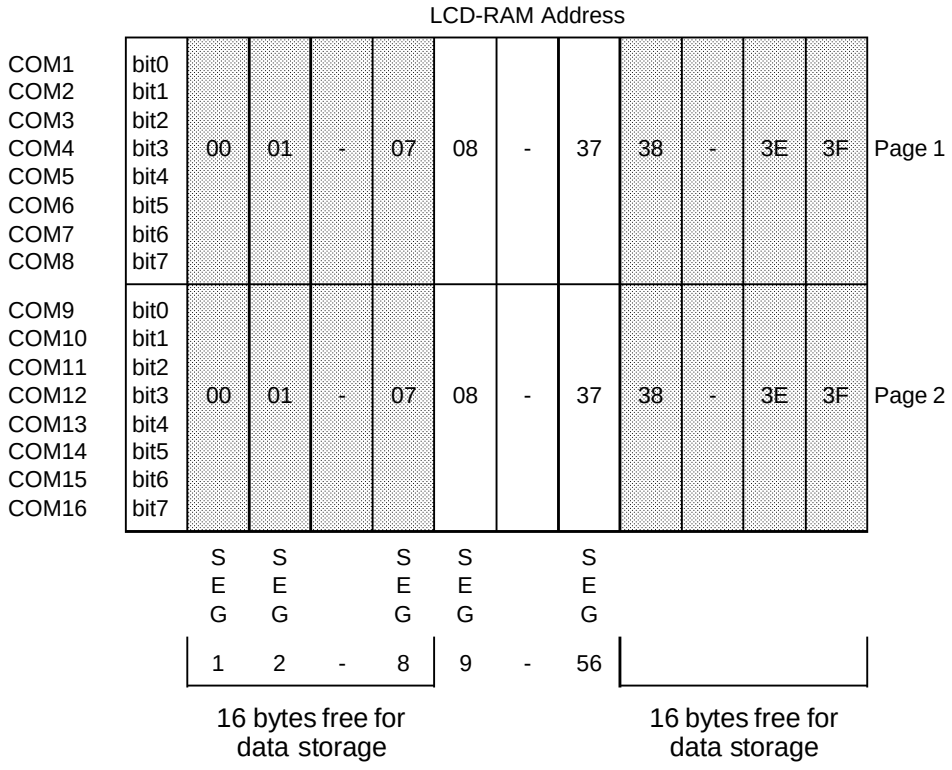
1/8 MUX (56 x 8 = 448 dots)



LCD RAM Page 2: 64 bytes free for data storage

Figure 58. Addressing Mapping of the LCD RAM (Continued)

1/16 MUX (48 x 16 = 768 dots)



SEGMENT AND COMMON SIGNALS

Each dot of the LCD dot matrix panel is turned on when the differential voltage between the segment signal and the common signal increases over a certain threshold, it is turned off when the voltage is below the threshold voltage. The common signals determine the select timing within a frame cycle (see Figure 59). The common signals have similar waveforms to the segments, but different phases.

Each common signal shows a high signal amplitude (VLCD-VSS) only at the correspoding section of a frame time. At the other sections of the frame, the signal amplitude is low (4/5VLCD-1/5VLCD). A dot can be turned-on only at phases with high signal amplitude.

In 1/16 duty cycle mode, one frame is divided into 16 sections, and each section is divided into two phases, phase 0 and 1. In 1/11 and 1/8 duty cycle modes, the number of sections is reduced to 11 or 8. This means the waveform pattern repeats faster in 1/11 and 1/8 duty cycle modes than 1/16 mode and the average voltage and the ON/OFF duty cycle on a selected pin is higher than in 1/16 mode. This results in a better contrast of the display.

Figure 59. Waveforms on LCD Outputs, 1/5 bias

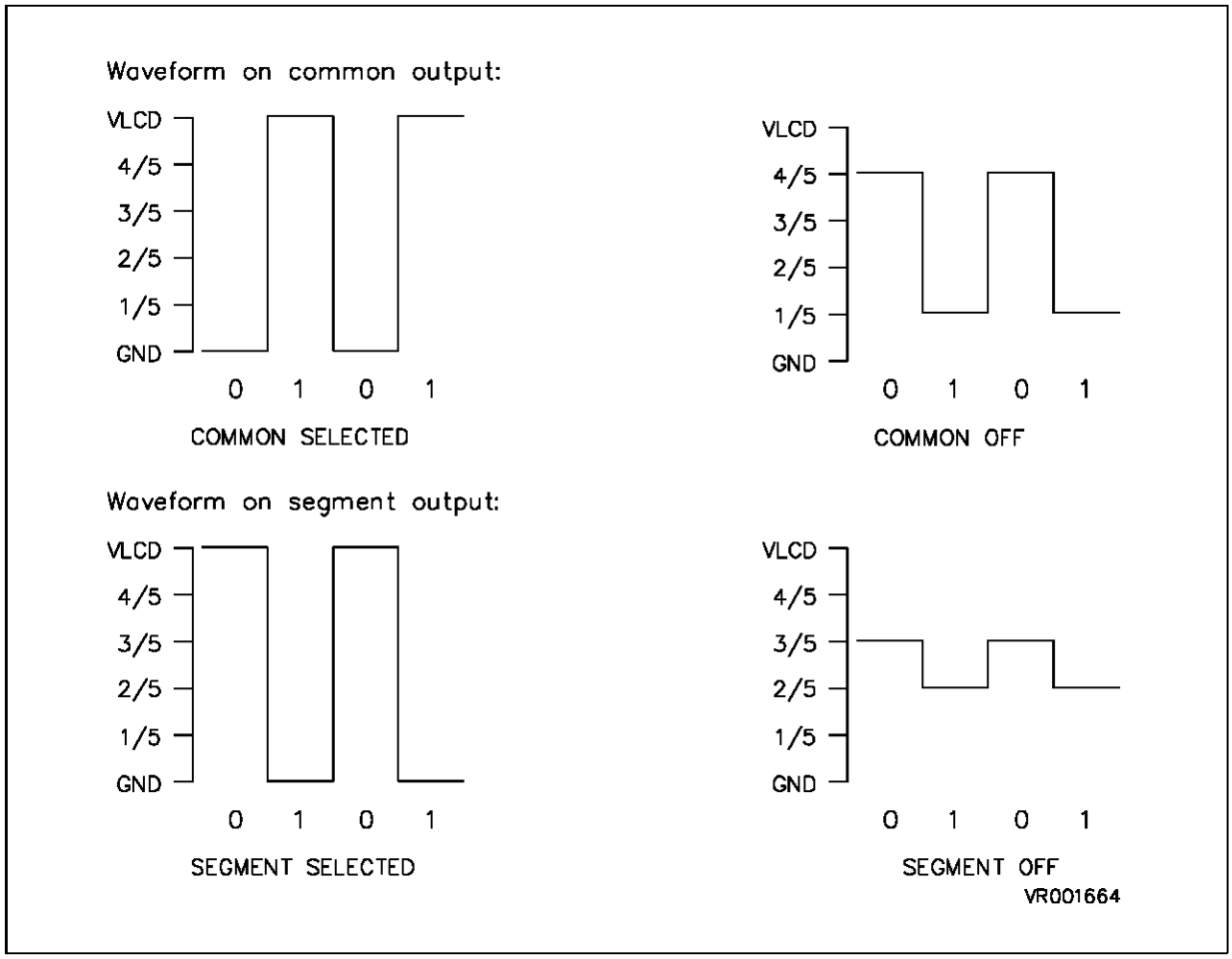
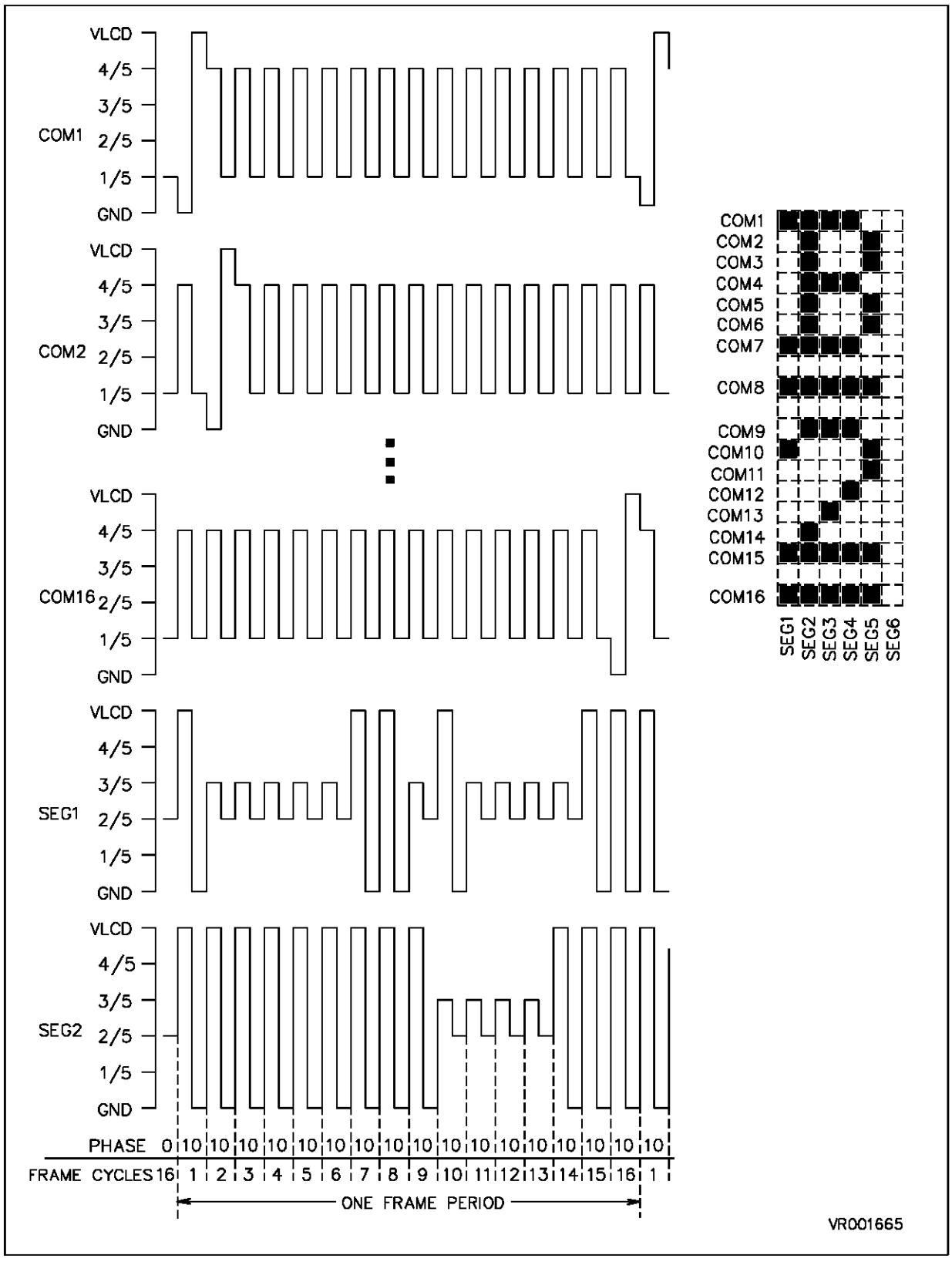


Figure 60. Waveforms on LCD Display, 1/16 Multiplex, 1/5 Bias

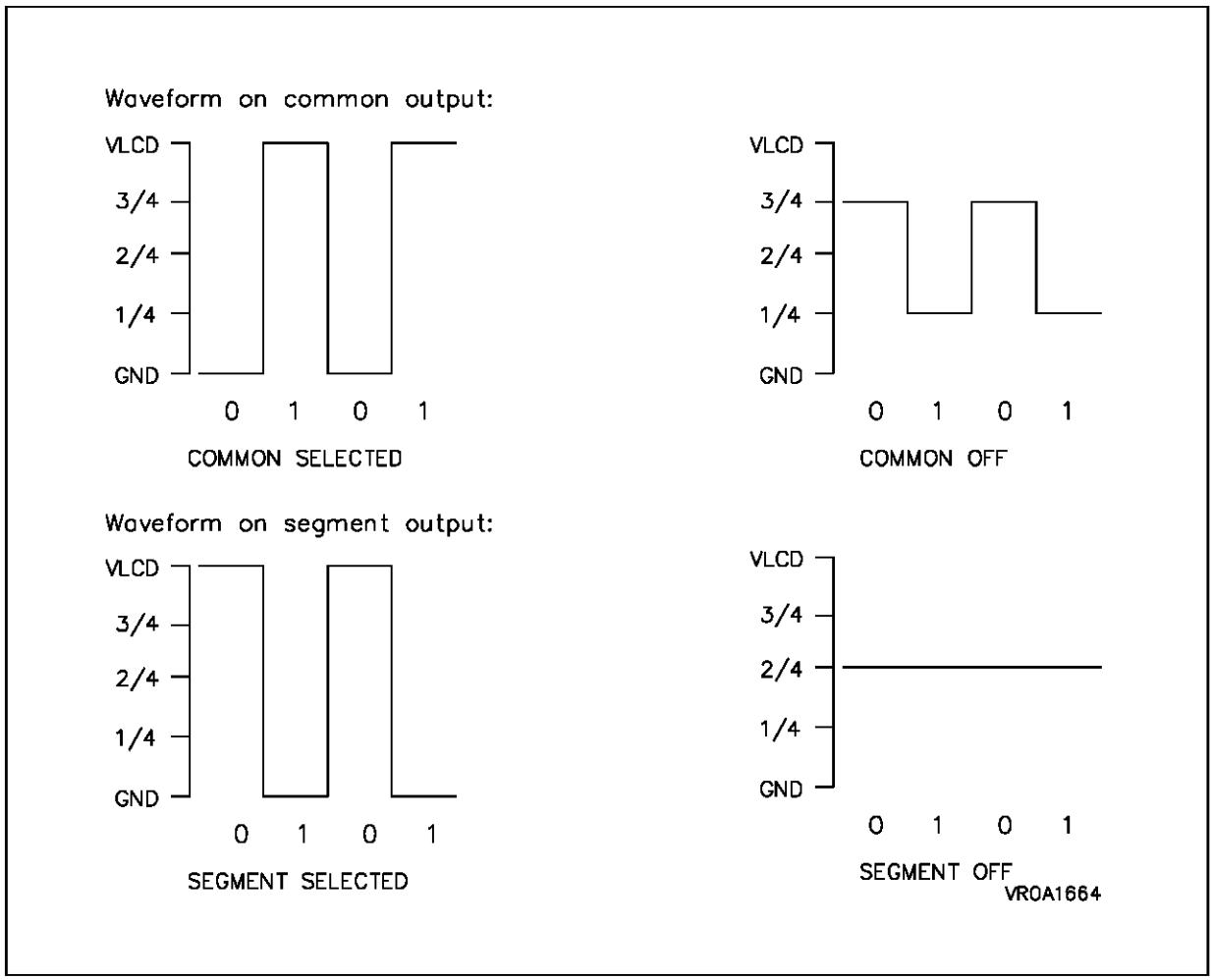


SEGMENT AND COMMON SIGNALS (Continued)

Working with 1/4 bias

Depending on the selected display material, the operating mode and the display voltage VLCD, it is possible to reduce the LCD bias resistor chain to 4 resistors and to operate in 1/4 bias mode.

Figure 61. Waveforms on LCD Outputs, 1/4 bias



Notes:

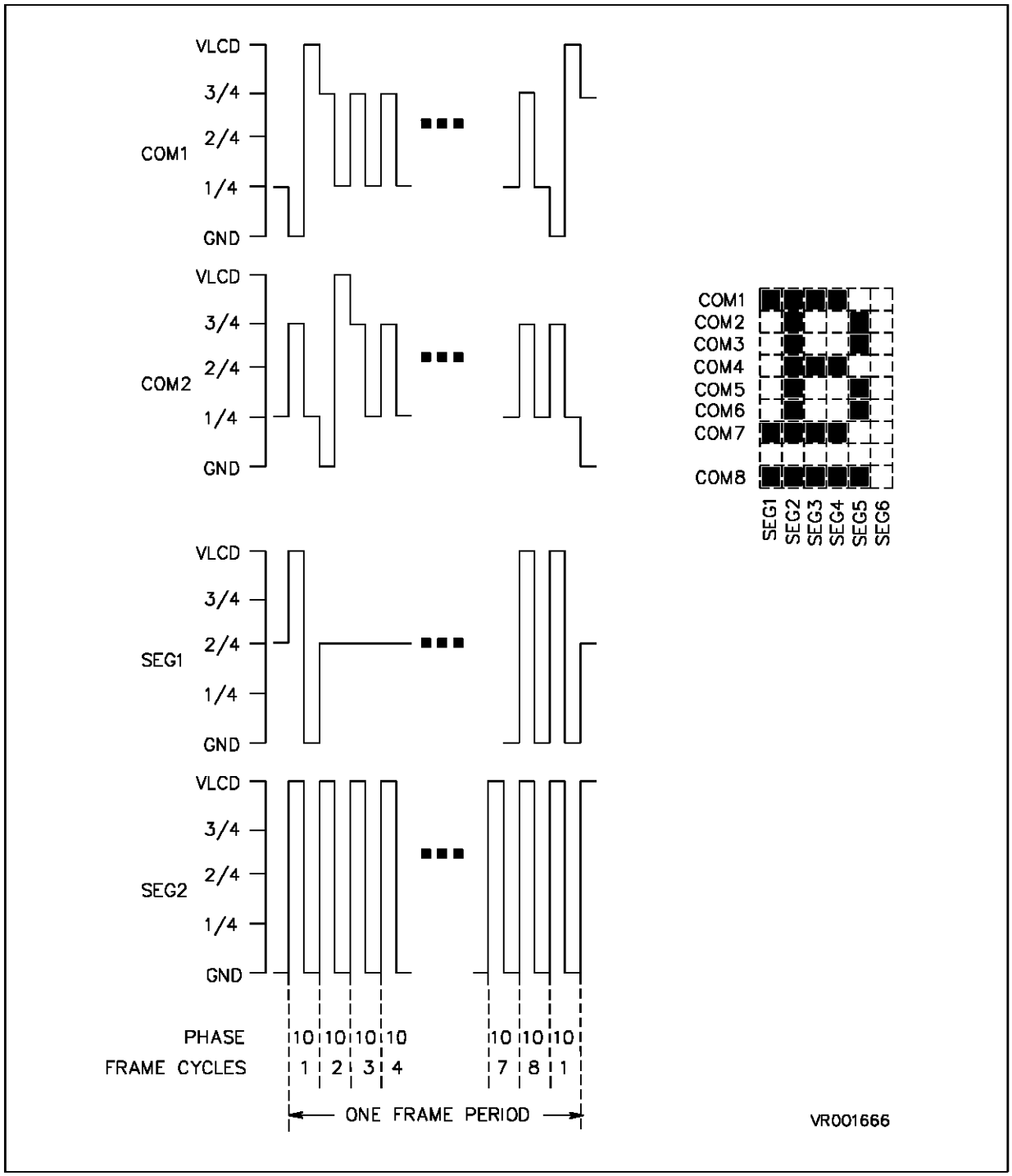
In STOP mode no clock is available for the LCD controller from the main oscillator. If the 32kHz oscillator is activated the LCD can also operate in STOP mode. If the stand-by oscillator is not active, the LCD controller is switched off when STOP instruction is executed; this mode has to be selected to reach the lowest power consumption.

A missing LCD clock (no oscillator active, broken crystal, etc.) is detected by a clock supervisor circuit that switches all the segments and backplane lines to ground to avoid destructive DC levels at the LCD. If the LCD clock is not missing, but is too slow (for example due to an incorrect setting of the clock selection lines in the LCD control register), the LCD will be switched off periodically. This situation must be avoided.

The LCD function change is only effective at the end of a frame. For this reason special care has to be taken when entering the STOP mode. After switching the LCD clock source from the main oscillator to the 32kHz stand-by oscillator it must be guaranteed that enough clock pulses are delivered to complete the current frame before entering the STOP mode. Otherwise the LCD function will not be changed and the LCD is switched off after entering the STOP mode.

When using a 1/16 MUX rate, a 1/5 bias is usually necessary. When operating with a 1/11 or 1/8 MUX rate, 1/4 or 1/5 bias can be chosen depending on the selected LCD.

Figure 62. Waveforms on LCD Display, 1/8 Multiplex, 1/4 Bias



SOFTWARE DESCRIPTION

The ST62xx software has been designed to fully use the hardware in the most efficient way possible while keeping byte usage to a minimum, in short to provide byte efficient programming capability. The ST62xx core has the ability to set or clear any register or RAM location bit of the Data space with a single instruction. Furthermore, the program may branch to a selected address depending on the status of any bit of the Data space. The carry bit is stored with the value of the bit when the SET or RES instruction is processed.

Addressing Modes

The ST62xx core has nine addressing modes which are described in the following paragraphs. The ST62xx core uses three different address spaces: Program space, Data space, and Stack space. Program space contains the instructions which are to be executed, plus the data for immediate mode instructions. Data space contains the Accumulator, the X,Y,V and W registers, peripheral and Input/Output registers, the RAM locations and Data ROM locations (for storage of tables and constants). Stack space contains six 12-bit RAM cells used to stack the return addresses for subroutines and interrupts.

Immediate. In the immediate addressing mode, the operand of the instruction follows the opcode location. As the operand is a ROM byte, the immediate addressing mode is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

Direct. In the direct addressing mode, the address of the byte that is processed by the instruction is stored in the location that follows the opcode. Direct addressing allows the user to directly address the 256 bytes in Data Space memory with a single two-byte instruction.

Short Direct. The core can address the four RAM registers X,Y,V,W (locations 80h, 81h, 82h, 83h) in the short-direct addressing mode . In this case, the instruction is only one byte and the selection of the location to be processed is contained in the opcode. Short direct addressing is a subset of the direct addressing mode. (Note that 80h and 81h are also indirect registers).

Extended. In the extended addressing mode, the 12-bit address needed to define the instruction is obtained by concatenating the four less significant bits of the opcode with the byte following the opcode. The instructions (JP, CALL) that use the extended addressing mode are able to branch to any address of the 4K bytes Program space.

An extended addressing mode instruction is two-byte long.

Program Counter Relative. The relative addressing mode is only used in conditional branch instructions. The instruction is used to perform a test and, if the condition is true, a branch with a span of -15 to +16 locations around the address of the relative instruction. If the condition is not true, the instruction that follows the relative instruction is executed. The relative addressing mode instruction is one-byte long. The opcode is obtained in adding the three most significant bits that characterize the kind of the test, one bit that determines whether the branch is a forward (when it is 0) or backward (when it is 1) branch and the four less significant bits that give the span of the branch (0h to Fh) that must be added or subtracted to the address of the relative instruction to obtain the address of the branch.

Bit Direct. In the bit direct addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode points to the address of the byte in which the specified bit must be set or cleared. Thus, any bit in the 256 locations of Data space memory can be set or cleared.

Bit Test & Branch. The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit test and branch instruction is three-byte long. The bit identification and the tested condition are included in the opcode byte. The address of the byte to be tested follows immediately the opcode in the Program space. The third byte is the jump displacement, which is in the range of -126 to +129. This displacement can be determined using a label, which is converted by the assembler.

Indirect. In the indirect addressing mode, the byte processed by the register-indirect instruction is at the address pointed by the content of one of the indirect registers, X or Y (80h,81h). The indirect register is selected by the bit 4 of the opcode. A register indirect instruction is one byte long.

Inherent. In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. These instructions are one byte long.

SOFTWARE DESCRIPTION (Continued)

Instruction Set

The ST62xx core has a set of 40 basic instructions. When these instructions are combined with nine addressing modes, 244 usable opcodes can be obtained. They can be divided into six different types: load/store, arithmetic/logic, conditional branch, control instructions, jump/call, bit manipulation. The following paragraphs describe the different types.

All the instructions within a given type are presented in individual tables.

Load & Store. These instructions use one,two or three bytes in relation with the addressing mode. One operand is the Accumulator for LOAD and the other operand is obtained from data memory using one of the addressing modes.

For Load Immediate one operand can be any of the 256 data space bytes while the other is always immediate data.

Table 18. Load & Store Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
LD A, X	Short Direct	1	4	Δ	*
LD A, Y	Short Direct	1	4	Δ	*
LD A, V	Short Direct	1	4	Δ	*
LD A, W	Short Direct	1	4	Δ	*
LD X, A	Short Direct	1	4	Δ	*
LD Y, A	Short Direct	1	4	Δ	*
LD V, A	Short Direct	1	4	Δ	*
LD W, A	Short Direct	1	4	Δ	*
LD A, rr	Direct	2	4	Δ	*
LD rr, A	Direct	2	4	Δ	*
LD A, (X)	Indirect	1	4	Δ	*
LD A, (Y)	Indirect	1	4	Δ	*
LD (X), A	Indirect	1	4	Δ	*
LD (Y), A	Indirect	1	4	Δ	*
LDI A, #N	Immediate	2	4	Δ	*
LDI rr, #N	Immediate	3	4	*	*

Notes:
X,Y. Indirect Register Pointers, V & W Short Direct Registers
. Immediate data (stored in ROM memory)
rr. Data space register
Δ . Affected
* . Not Affected

SOFTWARE DESCRIPTION (Continued)

Arithmetic and Logic. These instructions are used to perform the arithmetic calculations and logic operations. In AND, ADD, CP, SUB instructions one operand is always the accumulator while the other can be either a data space memory content or an immediate value in relation with the addressing mode. In CLR, DEC, INC instructions the operand can be any of the 256 data space addresses. In COM, RLC, SLA the operand is always the accumulator.

Table 19. Arithmetic & Logic Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
ADD A, (X)	Indirect	1	4	Δ	Δ
ADD A, (Y)	Indirect	1	4	Δ	Δ
ADD A, rr	Direct	2	4	Δ	Δ
ADDI A, #N	Immediate	2	4	Δ	Δ
AND A, (X)	Indirect	1	4	Δ	*
AND A, (Y)	Indirect	1	4	Δ	*
AND A, rr	Direct	2	4	Δ	*
ANDI A, #N	Immediate	2	4	Δ	*
CLR A	Short Direct	2	4	Δ	Δ
CLR rr	Direct	3	4	*	*
COM A	Inherent	1	4	Δ	Δ
CP A, (X)	Indirect	1	4	Δ	Δ
CP A, (Y)	Indirect	1	4	Δ	Δ
CP A, rr	Direct	2	4	Δ	Δ
CPI A, #N	Immediate	2	4	Δ	Δ
DEC X	Short Direct	1	4	Δ	*
DEC Y	Short Direct	1	4	Δ	*
DEC V	Short Direct	1	4	Δ	*
DEC W	Short Direct	1	4	Δ	*
DEC A	Direct	2	4	Δ	*
DEC rr	Direct	2	4	Δ	*
DEC (X)	Indirect	1	4	Δ	*
DEC (Y)	Indirect	1	4	Δ	*
INC X	Short Direct	1	4	Δ	*
INC Y	Short Direct	1	4	Δ	*
INC V	Short Direct	1	4	Δ	*
INC W	Short Direct	1	4	Δ	*
INC A	Direct	2	4	Δ	*
INC rr	Direct	2	4	Δ	*
INC (X)	Indirect	1	4	Δ	*
INC (Y)	Indirect	1	4	Δ	*
RLC A	Inherent	1	4	Δ	Δ
SLA A	Inherent	2	4	Δ	Δ
SUB A, (X)	Indirect	1	4	Δ	Δ
SUB A, (Y)	Indirect	1	4	Δ	Δ
SUB A, rr	Direct	2	4	Δ	Δ
SUBI A, #N	Immediate	2	4	Δ	Δ

Notes:
X,Y. Indirect Register Pointers, V & W Short Direct Registers
. Immediate data (stored in ROM memory)
rr. Data space register
Δ. Affected
* . Not Affected

SOFTWARE DESCRIPTION (Continued)

Conditional Branch. The branch instructions achieve a branch in the program when the selected condition is met.

Bit Manipulation Instructions. These instructions can handle any bit in data space memory. One group either sets or clears. The other group (see Conditional Branch) performs the bit test branch operations. These instructions should be used very carefully with I/O ports data registers.

Control Instructions. The control instructions control the MCU operations during program execution.

Jump and Call. These two instructions are used to perform long (12-bit) jumps or subroutines call inside the whole program space. Refer to Table 23.

Table 20. Conditional Branch Instructions

Instruction	Branch If	Bytes	Cycles	Flags	
				Z	C
JRC e	C = 1	1	2	*	*
JRNC e	C = 0	1	2	*	*
JRZ e	Z = 1	1	2	*	*
JRNZ e	Z = 0	1	2	*	*
JRR b, rr, ee	Bit = 0	3	5	*	Δ
JRS b, rr, ee	Bit = 1	3	5	*	Δ

Notes:
b. 3-bit address
e. 5 bit signed displacement in the range -15 to +16
ee. 8 bit signed displacement in the range -126 to +129
rr. Data space register
Δ . Affected
* . Not Affected

Table 21. Bit Manipulation Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
SET b,rr	Bit Direct	2	4	*	*
RES b,rr	Bit Direct	2	4	*	*

Notes:
b. 3-bit address;
rr. Data space register;
* . Not Affected

Table 22. Control Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
NOP	Inherent	1	2	*	*
RET	Inherent	1	2	*	*
RETI	Inherent	1	2	Δ	Δ
STOP (1)	Inherent	1	2	*	*
WAIT	Inherent	1	2	*	*

Notes:
1. This instruction is deactivated and a WAIT is automatically executed instead of a STOP if the Watchdog function is selected.
Δ . Affected
* . Not Affected

Table 23. Jump & Call Instructions

Instruction	Addressing Mode	Bytes	Cycles	Flags	
				Z	C
CALL abc	Extended	2	4	*	*
JP abc	Extended	2	4	*	*

Notes:
abc.12-bit address;
* . Not Affected

SOFTWARE DESCRIPTION (Continued)

Opcode Map Summary. The following table contains an opcode map for the instructions used on the MCU.

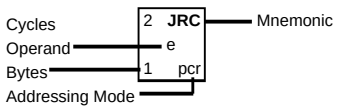
LOW / HI	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	LOW / HI
0 0000	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b0,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **LD** a,(x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b0,rr 2 b.d	2 **JRZ** e 1 pcr	4 **LDI** rr,nn 3 imm	2 **JRC** e 1 pcr	4 **LD** a,(y) 1 ind	0 0000
1 0001	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b0,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **INC** x 1 sd	2 **JRC** e 1 prc	4 **LDI** a,nn 2 imm	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b0,rr 2 b.d	2 **JRZ** e 1 pcr	4 **DEC** x 1 sd	2 **JRC** e 1 pcr	4 **LD** a,rr 2 dir	1 0001
2 0010	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b4,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **CP** a,(x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b4,rr 2 b.d	2 **JRZ** e 1 pcr	4 **COM** a 1 inh	2 **JRC** e 1 pcr	4 **CP** a,(y) 1 ind	2 0010
3 0011	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b4,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **LD** a,x 1 sd	2 **JRC** e 1 prc	4 **CPI** a,nn 2 imm	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b4,rr 2 b.d.	2 **JRZ** e 1 pcr	4 **LD** x,a 1 sd	2 **JRC** e 1 pcr	4 **CP** a,rr 2 dir	3 0011
4 0100	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b2,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **ADD** a,(x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b2,rr 2 b.d	2 **JRZ** e 1 pcr	2 **RETI** 1 inh	2 **JRC** e 1 pcr	4 **ADD** a,(y) 1 ind	4 0100
5 0101	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b2,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **INC** y 1 sd	2 **JRC** e 1 prc	4 **ADDI** a,nn 2 imm	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b2,rr 2 b.d	2 **JRZ** e 1 pcr	4 **DEC** y 1 sd	2 **JRC** e 1 pcr	4 **ADD** a,rr 2 dir	5 0101
6 0110	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b6,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **INC** (x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b6,rr 2 b.d	2 **JRZ** e 1 pcr	2 **STOP** 1 inh	2 **JRC** e 1 pcr	4 **INC** (y) 1 ind	6 0110
7 0111	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b6,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **LD** a,y 1 sd	2 **JRC** e 1 prc	#	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b6,rr 2 b.d	2 **JRZ** e 1 pcr	4 **LD** y,a 1 sd	2 **JRC** e 1 pcr	4 **INC** rr 2 dir	7 0111
8 1000	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b1,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **LD** (x),a 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b1,rr 2 b.d	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 pcr	4 **LD** (y),a 1 ind	8 1000
9 1001	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b1,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **INC** v 1 sd	2 **JRC** e 1 prc	#	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b1,rr 2 b.d	2 **JRZ** e 1 pcr	4 **DEC** v 1 sd	2 **JRC** e 1 pcr	4 **LD** rr,a 2 dir	9 1001
A 1010	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b5,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **AND** a,(x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b5,rr 2 b.d	2 **JRZ** e 1 pcr	4 **RLC** a 1 inh	2 **JRC** e 1 pcr	4 **AND** a,(y) 1 ind	A 1010
B 1011	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b5,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **LD** a,v 1 sd	2 **JRC** e 1 prc	4 **ANDI** a,nn 2 imm	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b5,rr 2 b.d	2 **JRZ** e 1 pcr	4 **LD** v,a 1 sd	2 **JRC** e 1 pcr	4 **AND** a,rr 2 dir	B 1011
C 1100	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b3,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **SUB** a,(x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b3,rr 2 b.d	2 **JRZ** e 1 pcr	2 **RET** 1 inh	2 **JRC** e 1 pcr	4 **SUB** a,(y) 1 ind	C 1100
D 1101	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b3,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **INC** w 1 sd	2 **JRC** e 1 prc	4 **SUBI** a,nn 2 imm	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b3,rr 2 b.d	2 **JRZ** e 1 pcr	4 **DEC** w 1 sd	2 **JRC** e 1 pcr	4 **SUB** a,rr 2 dir	D 1101
E 1110	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRR** b7,rr,ee 3 bt	2 **JRZ** e 1 pcr	#	2 **JRC** e 1 prc	4 **DEC** (x) 1 ind	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **RES** b7,rr 2 b.d	2 **JRZ** e 1 pcr	2 **WAIT** 1 inh	2 **JRC** e 1 pcr	4 **DEC** (y) 1 ind	E 1110
F 1111	2 **JRNZ** e 1 pcr	4 **CALL** abc 2 ext	2 **JRNC** e 1 pcr	5 **JRS** b7,rr,ee 3 bt	2 **JRZ** e 1 pcr	4 **LD** a,w 1 sd	2 **JRC** e 1 prc	#	2 **JRNZ** e 1 pcr	4 **JP** abc 2 ext	2 **JRNC** e 1 pcr	4 **SET** b7,rr 2 b.d	2 **JRZ** e 1 pcr	4 **LD** w,a 1 sd	2 **JRC** e 1 pcr	4 **DEC** rr 2 dir	F 1111

Abbreviations for Addressing Modes:

dir	Direct
sd	Short Direct
imm	Immediate
inh	Inherent
ext	Extended
b.d	Bit Direct
bt	Bit Test
pcr	Program Counter Relative
ind	Indirect

Legend:

#	Indicates Illegal Instructions
e	5 Bit Displacement
b	3 Bit Address
rr	1byte dataspace address
nn	1 byte immediate data
abc	12 bit address
ee	8 bit Displacement



ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

This product contains devices to protect the inputs against damage due to high static voltages, however it is advised to take normal precaution to avoid application of any voltage higher than maximum rated voltages.
For proper operation it is recommended that V_I and V_O must be higher than V_{SS} and smaller than V_{DD}. Reliability is enhanced if unused inputs are connected to an appropriated logic voltage level (V_{DD} or V_{SS}).

Power Considerations. The average chip-junction temperature, Tj, in Celsius can be obtained from :

$Tj = T_A + PD \times RthJA$

Where : T_A = Ambient Temperature.

RthJA = Package thermal resistance (junction-to ambient).

PD = Pint + Pport.

Pint = $I_{DD} \times V_{DD}$ (chip internal power).

Pport = Port power dissipation (determinated by the user).

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.3 to 7.0	V
V_{LCD}	Display Voltage	-0.3 to 11.0	V
V_I	Input Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
V_O	Output Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
I_O	Current Drain per Pin Excluding V_{DD} & V_{SS}	± 10	mA
IV_{DD}	Total Current into V_{DD} (source)	50	mA
IV_{SS}	Total Current out of V_{SS} (sink)	50	mA
Tj	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-60 to 150	°C

Notes :
- Stresses above those listed as "absolute maximum ratings" may cause permanent damage to the device . This is a stress rating only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
- (1). Within these limits, clamping diodes are guaranteed not to be conductive. Voltage outside these limits are authorised as long as injection current is kept within the specification.

THERMAL CHARACTERISTIC

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
RthJA	Thermal Resistance	PQFP100		70		°C/W

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
T_A	Operating Temperature	1 Suffix Version 6 Suffix Version	0 -40		70 85	°C
V_{DD}	Operating Supply Voltage (1)		4.5		6	V
V_{LCD}	Display Voltage (1)		3		10	V
V_{DD}	RAM Retention Voltage (1)		2			V

RECOMMENDED OPERATING CONDITIONS (Continued)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency (1)(4)	$V_{DD} \geq 4.5V$	0.01		8.4	MHz
I_{INJ+}	Pin Injection Current (positive) Digital Input (2) Analog Input (3)	V_{DD} = 4.5 to 5.5V			+5	mA
I_{INJ-}	Pin Injection Current (negative) Digital Input (2) Analog Input	V_{DD} = 4.5 to 5.5V			-5	mA

Notes :
1. An oscillator frequency above 1MHz is recommended for reliable A/D results.
2. A current of ± 5mA can be forced on each pin of the digital section without affecting the functional behaviour of the device. For a positive current injected into one pin, a part of this current (~ 10%) can be expected to flow from the neighbouring pins. A current of -5mA can be forced on one input of the analog section at a time (or -2.5mA for all inputs at a time) without affecting the conversion.
3. If a total current of +1mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 1mA, all the conversion is resulting shifted by +1LSB. If a total positive current of +5mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 5mA, all the conversion is resulting shifted by +2LSB.
4. Operation below 0.01 MHz is possible but requires increased supply current.

EEPROM INFORMATION

The ST62xx EEPROM single poly process has been specially developped to achieve 300,000 Write/Erase cycles and a 10 year data retention.

DC ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
V_{IL}	Input Low Level Voltage	RESET, NMI, TIMER Pin			$0.3V_{DD}$	V
V_{IH}	Input High Level Voltage	TIMER Pin	$0.80V_{DD}$			V
		RESET, NMI Pin	$0.70V_{DD}$			V
I_{IL} I_{IH}	Input Leakage Current	RESET Pin V_{DD} =5V $V_{IN} = V_{DD}$ (1) $V_{IN} = V_{DD}$ (2) $V_{IN} = V_{SS}$ (5)			10 1 50	µA mA µA
V_{OL}	Low Level Output Voltage	TIMER, I_{OL} = 5.0mA			$0.3V_{DD}$	V
V_{OH}	High Level Output Voltage	TIMER, I_{OL} = -5.0mA	$0.65V_{DD}$			V

Notes on next page

DC ELECTRICAL CHARACTERISTICS (Continued)
(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
R_{PU}	Pull-up Resistor	V_{IN}=0V V_{DD}=5V NMI	50	100	200	kΩ
		RESET	200	300	500	kΩ
I_{IL} I_{IH}	Input Leakage Current	TIMER $V_{IN} = V_{DD}$ or V_{SS}		0.1	1.0	µA
I_{IL} I_{IH}	Input Leakage Current	NMI VDD = 5.V $V_{IN} = V_{SS}$ (5) $V_{IN} = V_{DD}$			100 1.0	µA
I_{DD}	Supply Current RUN Mode	f_{OSC} = 8MHz, I_{LOAD} = 0mA V_{DD} = 5.0V		4	7	mA
	Supply Current WAIT Mode (4)	f_{OSC} = 8MHz, I_{LOAD} = 0mA V_{DD} = 5.0V		1	3	mA
	Supply Current RESET Mode	f_{OSC} = 8MHz, $V_{RESET} = V_{SS}$		1	7	mA
	Supply Current STOP Mode (3)(4)	I_{LOAD} = 0mA V_{DD} = 5.0V		1	10	µA

Notes :
1. No Watchdog Reset activated.
2. Reset generated by Watchdog.
3. When the watchdog function is actvated the STOP instruction is deactivated. WAIT instruction is automatically executed.
4. All on-chip peripherals in OFF state
5. Pull-up resistor

AC ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency (2)	$V_{DD} \geq 4.5V$	0.01		8.4	MHz
t_{SU}	Oscillator Start-up Time	$C_{L1} = C_{L2}$ = 22pF - crystal		5	20	ms
t_{SR}	Supply Rise Time	10% to 90%	0.01		100	
t_{REC}	Supply Recovery Time (1)		100			
T_W	Minimum Pulse Width	NMI Pin V_{DD} = 5V	100			ns
		RESET Pin	100			ns
T_{WEE}	EEPROM Write Time	T_A = 25°C One Byte T_A = 85°C One Byte		5 15	10 19	ms
Endurance	EEPROM WRITE/ERASE Cycles	Q_A L_{OT} Acceptance Criteria	300.000	> 1 million		cycles
Retention	EEPROM Data Retention	T_A = 55°C	10			years
C_{IN}	Input Capacitance	All Inputs Pins			10	pF
C_{OUT}	Output Capacitance	All Outputs Pins			10	pF

Notes:
1. Period for which V_{DD} has to be connected or at 0V to allow internal Reset function at next power-up.
2. Operation below 0.01 MHz is possible but requires increased supply current.

I/O PORTS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
V_{IL}	Input Low Level Voltage	I/O Pins			$0.3V_{DD}$	V
V_{IH}	Input High Level Voltage	I/O Pins	$0.7V_{DD}$			V
V_{OL}	Low Level Output Voltage	V_{DD} = 5.0V I_{OL} = 10µA, All I/O Pins I_{OL} = 5mA, Standard I/O I_{OL} =10mA, PA/PC0-PC3 I_{OL} =20mA, PA/PC0-PC3			0.1 0.8 0.8 1.3	V
V_{OH}	High Level Output Voltage	I/O Pins, I_O = -10µA (source)	V_{DD}-0.1			V
		I/O Pins, I_{OL} = -V_{DD}x1mA V_{DD} = 5.0V	$0.6xV_{DD}$			V
I_{IL} I_{IH}	Input Leakage Current	I/O Pins,(1)		0.1	1.0	µA
R_{PU}	Pull-up Resistor	I/O Pins V_{IN} = 0V, V_{DD} = 5.0V	50	100	200	KΩ

Note 1. Pull-up resistor off

SPI ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified, V_{DD} = 5.0V)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
F_{CL}	Clock Frequency	applied on PB5/SCL			1	MHz
t_{SU}	Set-up Time	applied on PB6/Sin		50		ns
t_h	Hold Time	applied on PB6/Sin		100		ns

A/D CONVERTER CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
Res	Resolution (3)			8		Bit
A_{TOT}	Total Accuracy (3)	f_{OSC} > 1.2 MHz f_{OSC} > 32kHz			± 2 ±4	LSB
t_C (1)	Conversion Time	f_{OSC} = 8MHz		70		µs
V_{AN}	Conversion Range		V_{SS}		V_{DD}	V
ZIR	Zero Input Reading	Conversion result when $V_{IN} = V_{SS}$	00			Hex
FSR	Full Scale Reading	Conversion result when $V_{IN} = V_{DD}$			FF	Hex
AD_I	Analog Input Current During Conversion	V_{DD}= 4.5V			1.0	µA
AC_{IN} (2)	Analog Input Capacitance			2	5	pF
ASI	Analog Source Impedance (4)				30	kΩ

Notes:

1. With oscillator frequencies less than 1MHz, the A/D Converter accuracy is decreased.
2. Excluding Pad Capacitance
3. Noise at V_{DD}, V_{SS} ≤ 10m
4. A value higher than 30kΩ may disturb the conversion if the ADC is switched between different I/O pins connected to the Converter. The reason is that the ADC input capacitance (10pF) has to be charged before the beginning of the conversion. If the serial input impedance is high, the stabilisation time of the input voltage is non-negligible versus the total conversion time.
   If the input impedance is higher than 30kΩ a small decoupling capacitance can be added to the ADCin pins, and a short delay can be introduced by software, between the I/O switching and the beginning of the conversion.
   If the ADC is always connected to the same I/O pin, the ADC input capacitance is always loaded and the serial impedance can be higher. Its maximum value has just to be small enough not to disturb the input voltage during the conversion (1nA each 3µs through the I/O during the conversion).

TIMER1 CHARACTERISTICS
(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
t_{RES}	Resolution		$\frac{12}{f_{OSC}}$			s
f_{IN}	Input Frequency on PA4/TIM1 Pin				$\frac{f_{OSC}}{8}$	MHz
t_W	Pulse Width at PA4/TIM1 Pin	V_{DD} = 3.0V $V_{DD} \geq$ 4.5V	1 125			µs ns

AR TIMER CHARACTERISTICS
(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
t_{RES}	Resolution		$\frac{1}{f_{OSC}}$			s
f_{ARin}	Input Frequency on ARTIMin pin	STOP Mode RUN and WAIT Modes			2 $\frac{f_{OSC}}{4}$	MHz MHz
t_W	Pulse Width at ARTIMin Pin	V_{DD} = 3.0V (1) $V_{DD} \geq$ 4.5V	125 125			ns ns

Note 1. If low voltage option chosen

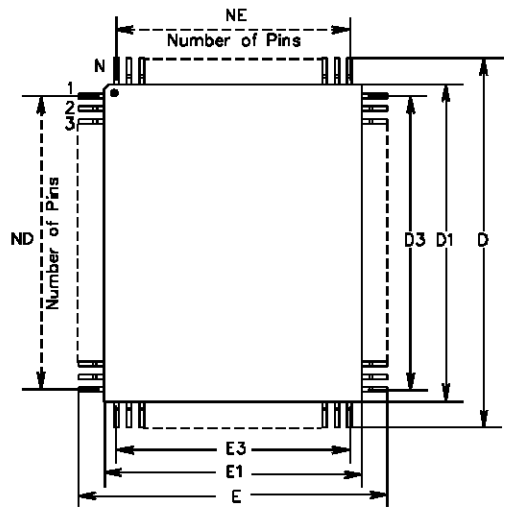
LCD ELECTRICAL CHARACTERISTICS
(T_A= -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{FR}	Frame Frequency	f_{OSC} = 1, 2, 4, 8 MHz	64		512	Hz
V_{OS}	DC Offset Voltage(1)	$V_{LCD} = V_{DD}$, no load			50	mV
V_{OH}	COM High Level, Output Voltage	I = 100µA, V_{LCD} = 5V	4.5			V
V_{OL}	COM Low Level, Output Voltage	I = 100µA, V_{LCD} = 5V			0.5	V
V_{OH}	SEG High Level, Output Voltage	I = 50µA, V_{LCD} = 5V	4.5			V
V_{OL}	SEG Low Level, Output Voltage	I = 50µA, V_{LCD} = 5V			0.5	V
V_{LCD}	Display Voltage		3		10	V

Note 1. The DC offset voltage refers to all segment and common outputs. It is the difference between the measured voltage value and nominal value for every voltage level. Ri of voltage meter must be greater than or equal to 10MΩ.

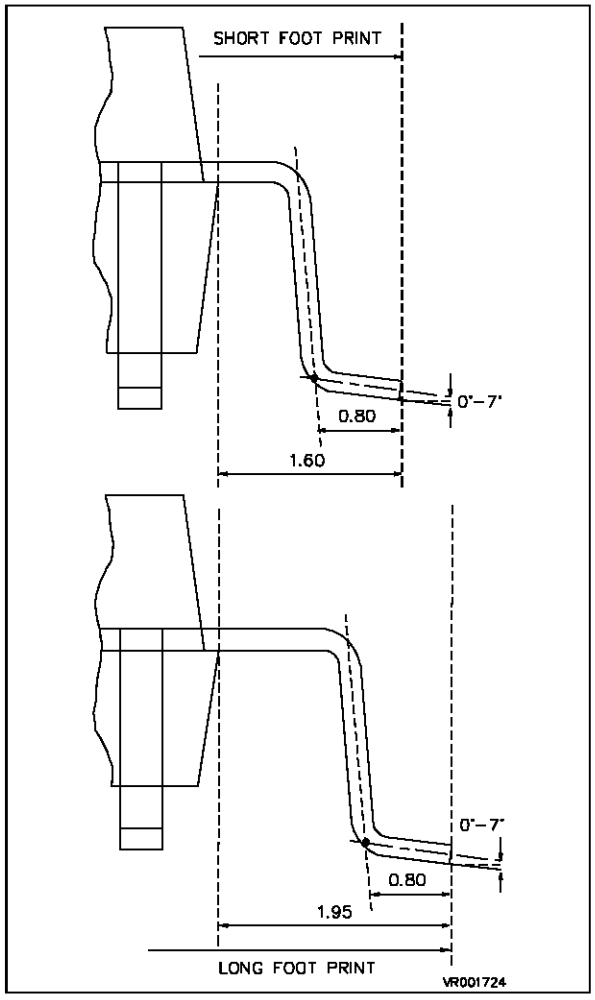
PACKAGE MECHANICAL DATA

Figure 63. ST6280 100 Pin Plastic Quad Flat Pack Package

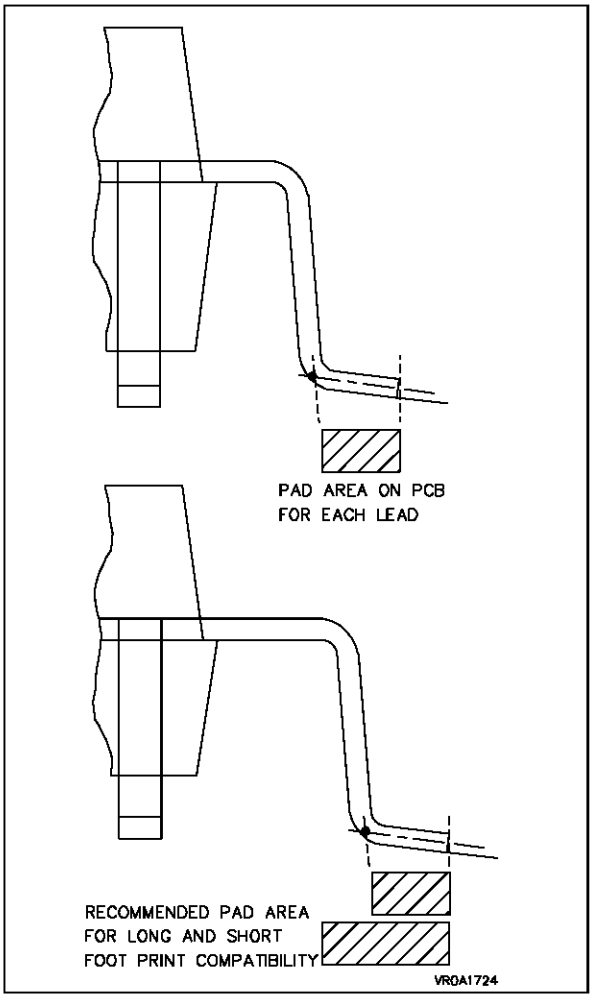


Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			3.30			0.130
A2	2.55	2.80	3.05	0.100	0.110	0.120
D	23.00	23.20	23.40	0.906	0.913	0.921
D1	19.90	20.00	20.10	0.783	0.787	0.791
D3		18.85			0.742	
E	17.00	17.20	17.40	0.669	0.677	0.685
E1	13.90	14.00	14.10	0.547	0.551	0.555
E3		12.35			0.486	
e		0.65			0.026	
Number of Pins						
ND	30					
NE	20					
N	100					

Short/Long Footprint Measurement



Short/Long Footprint recommended Padding



ORDERING INFORMATION

The following chapter deals with the procedure for transfer the Program/Data ROM codes to SGS-THOMSON.

Communication of the ROM Codes. To communicate the contents of Program/Data ROM memories to SGS-THOMSON, the customer has to send :

– one file in INTEL INTELLEC 8/MDS FORMAT (in an MS-DOS 5" diskette) for the PROGRAM Memory

– one file in INTEL INTELLEC 8/MDS FORMAT (in a MS-DOS 5" diskette) for the EEPROM initial content (this file is optional)

– a filled Option List form as described in the OPTION LIST paragraph.

The program ROM should respect the ROM Memory Map as in Table 24.

The ROM code must be generated with ST6 assembler. Before programming the EPROM, the buffer of the EPROM programmer must be filled with FFh.

Table 24. ROM Memory Map

ROM Page	Device Address	Description
Page 0	0000h-007Fh 0080h-07FFh	Reserved User ROM
Page 1 "STATIC"	0800h-0F9Fh 0FA0h-0FEFh 0FF0h-0FF7h 0FF8h-0FFBh 0FFCh-0FFDh 0FFEh-0FFFh	User ROM Reserved Interrupt Vectors Reserved NMI Vector Reset Vector
Page 2	0000h-000Fh 0010h-07FFh	Reserved User ROM
Page 3	0000h-000Fh 0010h-07FFh	Reserved User ROM

Note : EPROM addresses are related to the ROM file to be processed.

Customer EEPROM Initial Contents :
Format

a. The content should be written into an INTEL INTELLEC format file.

b. In the case of 128 bytes of EEPROM, the starting address in 000h and the end in 7Fh.

c. Undefined or don't care bytes should have the content FFh.

Listing Generation & Verification. When SGS-THOMSON receives the Codes, they are compared and a computer listing is generated from them. This listing refers exactly to the mask that will be used to produce the microcontroller. Then the listing is returned to the customer that must thoroughly check, complete, sign and return it to SGS-THOMSON. The signed list constitutes a part of the contractual agreement for the creation of the customer mask. SGS-THOMSON sales organization will provide detailed information on contractual points.

ORDERING INFORMATION TABLE

Sales Types	Temperature Range	Package
ST6280Q1/XX	0 to + 70°C	PQFP100
ST6280Q6/XX	-40 to + 85°C	PQFP100

Note : "XX" is the ROM code identifier allocated by SGS-THOMSON after receipt of all required options and the related ROM file.

ST6280 MICROCONTROLLER OPTION LIST

Customer . . . . . .
Address . . . . . .

Contact . . . . . .
Phone No . . . . . .
Reference . . . . . .

SGS-THOMSON Microelectronics references

Device
[ ] ST6280

Package
[ ] Plastic Quad Flat Package

Temperature Range
[ ] 0°C to + 70°C [ ] -40°C to + 85°C

Special Marking
[ ] No
[ ] Yes "_ _ _ _ _ _ _ _ _ _"
Authorized characters are Letters, digits, '.', '–', '/' and spaces only.
For special marking one line with 10 characters maximum is possible.

Power supply:
[ ] Standard (4.5V to 6V)

Comments :
- Number of LCD segments used :
- Number of LCD backplanes used :
- Multiplexing rate:

Note :

Signature

Date

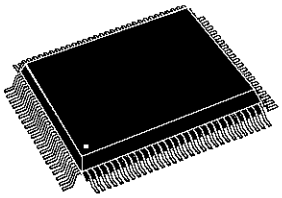
NOTES:

SGS-THOMSON MICROELECTRONICS

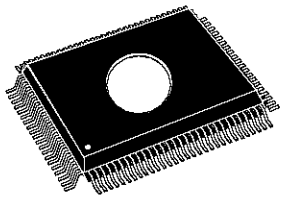
ST62E80
ST62T80

8-BIT EPROM HCMOS MCU WITH DOT MATRIX LCD DRIVER EEPROM AND A/D CONVERTER

- 4.5 to 6V supply operating range
- 8.4MHz Maximum Clock Frequency
- -40 to +85°C Operating Temperture Range
- Run, Wait & Stop Modes
- 5 different interrupt vectors
- Look-up table capability in EPROM
  User EPROM: 8192 bytes
  Data RAM: 192 bytes
  LCD RAM: 128 bytes
  EEPROM: 128 bytes
- PQFP100 and CQFP100-W packages
- 12 fully software programmable I/O as:
  - Input with/without pull-up resistor
  - Input with interrupt generation
  - Open-Drain or Push-pull outputs
  - Analog Inputs
- 10 I/O lines can sink up to 20mA for direct LED or TRIAC driving and have SPI alternate functions
- One 8-bit counter with 7-bit programmable prescalers (Timer 1)
- One 8-bit auto-reload timer with 7-bit programmable prescaler (Timer 2)
- Software activated digital watchdog
- 8-bit A/D converter with up to 12 analog inputs
- 8-bit synchronous serial peripheral interface (SPI)
- LCD driver with 48 segment outputs, 8 backplane outputs and 8 software selectable segment/backplane outputs able to drive up to 48x16 (768) or 56x8 (448) LCD segments.
- 32kHz oscillator for stand-by LCD operation
- One external not maskable interrupt
- 9 powerful addressing modes
- The accumulator, the X, Y, V & W registers, the port and peripherals data & control registers are addressed in the data space as RAM locations.
- The ST62E80 is the EPROM version, ST62T80 is the OTP version, fully compatible with ST6280 ROM version.



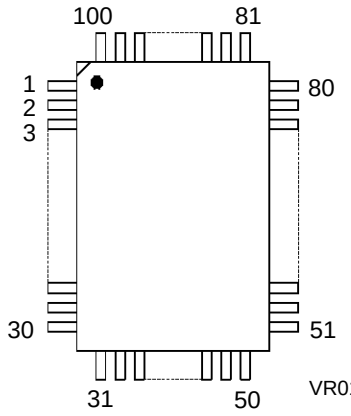
PQFP100



CQFP100-W

(Ordering Information at the end of the datasheet)

Figure 1. 100 Pin Quad Flat Pack (QFP) Package Pinout



ST62E80/T80 Pin Description

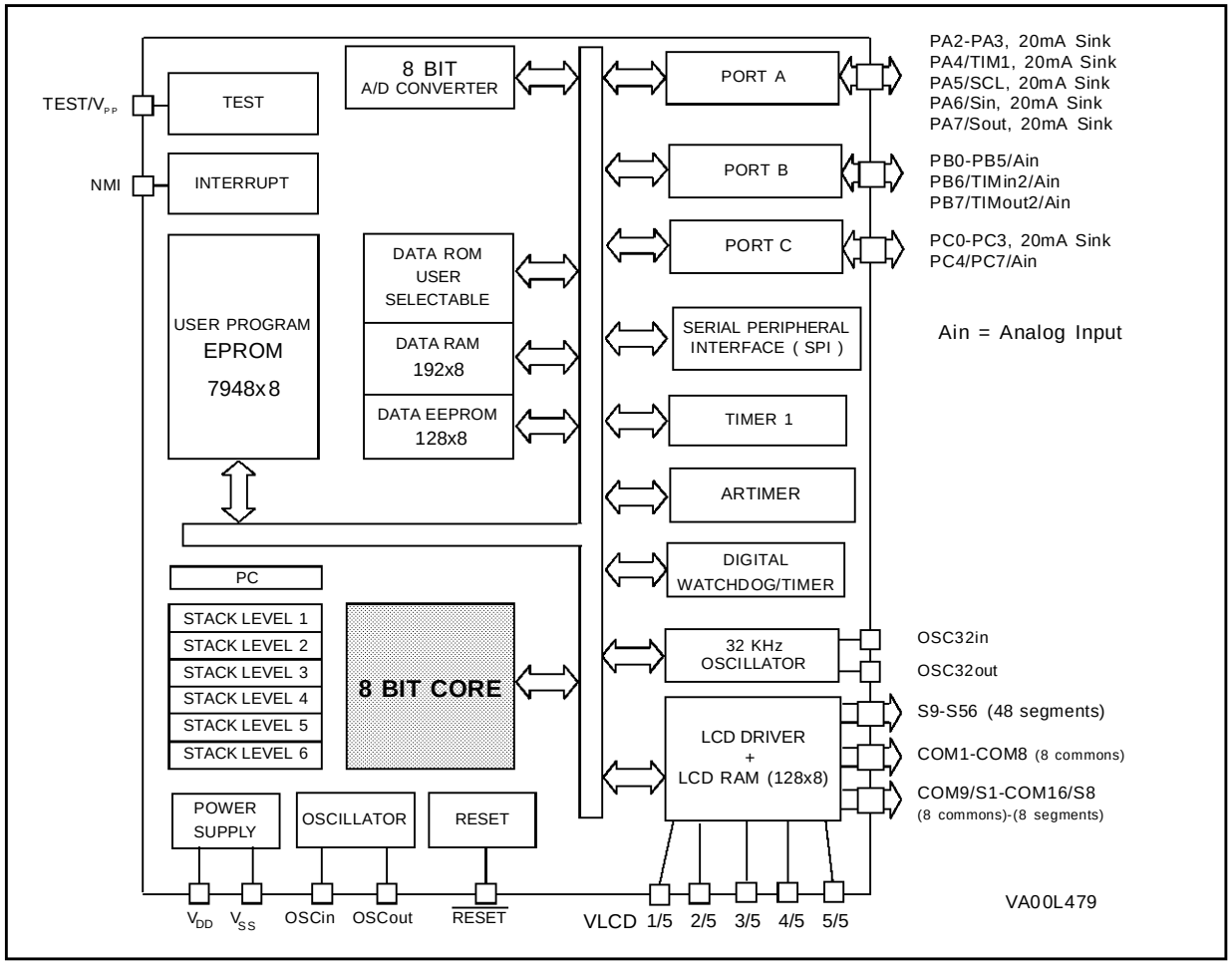
Pin number	Pin name	Pin number	Pin name	Pin number	Pin name	Pin number	Pin name
1	S39	31	PC7/Ain	51	PA3 (1)	81	S19
2	S40	32	PC6/Ain	52	PA2 (1)	82	S20
3	S41	33	PC5/Ain	53	OSC32out	83	S21
4	S42	34	PC4/Ain	54	OSC32in	84	S22
5	S43	35	PC3 (1)	55	COM1	85	S23
6	S44	36	PC2 (1)	56	COM2	86	S24
7	S45	37	PC1 (1)	57	COM3	87	S25
8	S46	38	PC0 (1)	58	COM4	88	S26
9	S47	39	NMI	59	COM5	89	S27
10	S48	40	V_{DD}	60	COM6	90	S28
11	S49	41	V_{SS}	61	COM7	91	S29
12	S50	42	VLCD	62	COM8	92	S30
13	S51	43	VLCD4/5	63	COM9/S1	93	S31
14	S52	44	VLCD3/5	64	COM10/S2	94	S32
15	S53	45	VLCD2/5	65	COM11/S3	95	S33
16	S54	46	VLCD1/5	66	COM12/S4	96	S34
17	S55	47	PA7/Sout (1)	67	COM13/S5	97	S35
18	S56	48	PA6/Sin (1)	68	COM14/S6	98	S36
19	PB7/TIMout2 /Ain	49	PA5/SCL (1)	69	COM15/S7	99	S37
		50	PA4/TIM1 (1)	70	COM16/S8	100	S38
20	PB6/TIMin2 /Ain			71	S9		
				72	S10		
21	PB5/Ain			73	S11		
22	PB4/Ain			74	S12		
23	PB3/Ain			75	S13		
24	PB2/Ain			76	S14		
25	PB1/Ain			77	S15		
26	PB0/Ain			78	S16		
27	TEST/V_{PP}			79	S17		
28	OSCout			80	S18		
29	OSCin						
30	$\overline{\text{RESET}}$						

Note 1: 20mA SINK

GENERAL DESCRIPTION

The ST62E80, ST62T80 microcontrollers are members of the 8-bit HCMOS ST62xx family, a series of devices oriented to low-medium complexity applications. They are the EPROM/OTP versions of the ST6280 ROM device and are suitable for prototyping and low-volume production. All ST62xx members are based on a building block approach: a common core is associated with a combination of on-chip peripherals (macrocells). The macrocells of the ST6280 family are: an advanced LCD driver/controller with 48 segments, 8 backplanes and 8 software selectable segment/backplane outputs able to drive up to 48×16 (768) or 56×8 (448) segments, one 8 bit reload timer with 7 bit programmable prescaler (Timer 2), one 8 bit standard timer/counter with a 7-bit software programmable prescaler (Timer 1), the digital watchdog timer (DWD), an 8-bit A/D Converter with up to 12 analog inputs, a 32kHz Oscillator, and an 8-bit synchronous serial peripheral interface (SPI). In addition this device offers 128 bytes of EEPROM for storage of non-volatile data. Thanks to these peripherals the ST6280 family is well suited for general purpose, automotive, security, appliance and industrial applications.

Figure 2. ST62E80/T80 Block Diagram



PIN DESCRIPTION

V_{DD} and **V_{SS}**. Power is supplied to the MCU using these two pins. V_{DD} is power and V_{SS} is the ground connection.

OSCin and OSCout. These pins are internally connected with the on-chip oscillator circuit. A quartz crystal or a ceramic resonator can be connected between these two pins in order to allow the correct operation of the MCU with various stability/cost trade-offs. OSCin is the input pin, OSCout is the output pin. An external clock signal can be applied to OSCin.

$\overline{\text{RESET}}$. The active low $\overline{\text{RESET}}$ pin is used to restart the microcontroller at the beginning of its program. The $\overline{\text{RESET}}$ pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

TEST. TEST must be held at V_{SS} for normal operation (an internal pull-down resistor selects normal operating mode if TEST pin is not connected).

NMI. The NMI pin provides the capability for asynchronous applying an external top priority interrupt to the MCU. This pin is falling edge sensitive. The NMI pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

PA4/TIM1. This pin can be used as Timer 1 I/O pin. In input mode it is connected to the timer prescaler input and acts as the external timer clock or as the control gate for the internal timer clock. In the output mode the timer pin outputs the timer data bit when a time out occurs.
To use this pin as Timer output the I/O pin has to be programmed as open-drain output. To use this pin as Timer input the I/O pin has to be programmed as input.

PB6/TIMIN2, PB7/TIMOUT2. These pins are the Input and Output pins of the Autoreload Timer. The timer input pin ARTIMin is connected to port line PB6. To use the line as timer input function, PB6 has to be programmed as input with or without pull-up. The timer output pin is connected to the port line PB7. A dedicated bit in the ARTIMER mode control register sets the line as timer output function ARTIMout.

PA2-PA7. These 6 lines are organized as one I/O port (A). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. In output mode these lines can also sink 20mA for direct LED or triac driving. PA5-PA7 can also be used as respectively Clock, Data in and Data out pins for the on-chip SPI to carry the synchronous serial I/O signals. PA4 can also be used as the TIMER 1 I/O pin.

PB0-PB7. These 8 lines are organized as one I/O port (B). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, open-drain or push-pull output or as an analog input for the A/D converter. PB6 is also connected to the ARTIMER input function while PB7 can act as the ARTIMER output.

PC0-PC3, PC4-PC7. These 8 lines are organized as one I/O port (C). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. PC0-PC3 can also sink 20mA for direct LED or triac driving while PC4-PC7 can be programmed as analog inputs for the A/D converter.

COM1-COM8. These eight pins are the LCD peripheral common outputs. They are the outputs of the on-chip backplane voltage generator which is used for multiplexing the LCD segment lines.

S9-S56. These pins are the 48 LCD peripheral driver outputs of the ST6280. Segments S1-S8 are multiplexed with COM9-COM16 and their function is software selectable.

COM9/S1-COM16/S8. These pins are the 8 multiplexed common/segment lines. Under software selected control, they can act as LCD common outputs allowing a 48×16 dot matrix operation, or they can act as segment outputs allowing 56×8 dot matrix operation.

VLCD1/5-VLCD5/5. Resistor network nodes for determining the intermediate LCD voltage levels on COM1-COM8/COM16 and S1/S8-S56 pins during multiplex operation.

OSC32in and OSC32out. These pins are internally connected with the on-chip 32kHz oscillator circuit. A 32.768kHz quartz crystal can be connected between these two pins if it is necessary to provide the LCD stand-by clock and real time interrupt. OSC32in is the input pin, OSC32out is the output pin.

ST62E80/ST62T80 EPROM/OTP DESCRIPTION.

The ST62E80 is the EPROM version of the ST6280 ROM product. It is intended for use during the development of an application, and for pre-production and small volume production. The ST62T80 OTP has the same characteristics. They both include EPROM memory instead of the ROM memory of the ST6280, and so the program and constants of the program can be easily modified by the user with the ST62E80 EPROM programming board of from SGS-THOMSON.

From a user point of view (with the following exceptions) the ST62E80, ST62T80 products have exactly the same software and hardware features of the ROM version.

On the ST62E80, all the 8192 bytes of PROGRAM memory are available for the user, as all the EPROM memory can be erased by exposure to UV light. On the ST62T80 (OTP) device a reserved area for test purposes exists, as for the ST6280 ROM device. In order to avoid any discrepancy between program functionality when using the EPROM, OTP and ROM it is recommended not to use these reserved areas, even when using the ST62E80.

Other than these exceptions, the ST62E80, ST62T80 parts are fully compatible with the ROM ST6280 equivalent, this datasheet thus provides only information specific to the EPROM based devices.

THE READER IS ASKED TO REFER TO THE DATASHEET OF THE ST6280 ROM-BASED DEVICE FOR FURTHER DETAILS.

EPROM ERASING

The EPROM of the windowed package of the ST62E80 may be erased by exposure to Ultra Violet light.

The erasure characteristic of the ST62E80 EPROM is such that erasure begins when the memory is exposed to light with wave lengths shorter than approximately 4000Å. It should be noted that sunlight and some types of fluorescent lamps have wavelengths in the range 3000-4000Å. It is thus recommended that the window of the ST62E80 package be covered by an opaque label to prevent unintentional erasure problems when testing the application in such an environment.
The recommended erasure procedure of the ST62E80 EPROM is exposure to short wave ultraviolet light which has wavelength 2537Å. The integrated dose (i.e. UV intensity x exposure time) for erasure should be a minimum of 15 W-sec/cm^2. The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with 12000µW/cm^2 power rating. The ST62E80 should be placed within 2.5 cm (1 inch) of the lamp tubes during erasure.

ST62T80 OTP Memory Map

ROM Page	Device Address	Description
Page 0	0000h-007Fh 0080h-07FFh	Reserved User ROM
Page 1 “STATIC”	0800h-0F9Fh 0FA0h-0FEFh 0FF0h-0FF7h 0FF8h-0FFBh 0FFCh-0FFDh 0FFEh-0FFFh	User ROM Reserved Interrupt Vectors Reserved NMI Vector Reset Vector
Page 2	0000h-000Fh 0010h-07FFh	Reserved User ROM
Page 3	0000h-000Fh 0010h-07FFh	Reserved User ROM

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

This product contains devices to protect the inputs against damage due to high static voltages, however it is advised to take normal precaution to avoid application of any voltage higher than maximum rated voltages.

For proper operation it is recommended that V_I and V_O must be higher than V_{SS} and smaller V_{DD}. Reliability is enhanced if unused inputs are connected to an appropriated logic voltage level (V_{DD} or V_{SS}).

Power Considerations. The average chip-junction temperature, Tj, in Celsius can be obtained from :

$Tj = T_A + PD \times RthJA$

Where : T_A = Ambient Temperature.

RthJA = Package thermal resistance (junction-to ambient).

PD = Pint + Pport.

Pint = $I_{DD} \times V_{DD}$ (chip internal power).

Pport = Port power dissipation (determinated by the user).

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.3 to 7.0	V
V_{LCD}	Display Voltage	-0.3 to 11.0	V
V_I	Input Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
V_O	Output Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
I_O	Current Drain per Pin Excluding V_{DD} & V_{SS}	± 10	mA
IV_{DD}	Total Current into V_{DD} (source)	50	mA
IV_{SS}	Total Current out of V_{SS} (sink)	50	mA
Tj	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-60 to 150	°C

Notes :

- Stresses above those listed as “absolute maximum ratings” may cause permanent damage to the device . This is a stress rating only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
- (1). Within these limits, clamping diodes are guaranteed not to be conductive. Voltage outside these limits are authorised as long as injection current is kept within the specification.

THERMAL CHARACTERISTIC

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
RthJA	Thermal Resistance	PQFP100 CQPF100-W		70		°C/W

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
T_A	Operating Temperature	1 suffix Version 6 suffix Version	0 -40		70 85	°C
V_{DD}	Operating Supply Voltage(1)		4.5		6	V
V_{LCD}	Display Voltage(1)		3		10	V
V_{DD}	RAM Retention Voltage(1)		2			V

RECOMMENDED OPERATING CONDITIONS (Continued)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency(1)(4)	V_{DD} = 4.5V	0.01		8.4	MHz
I_{INJ+}	Pin Injection Current (positive) Digital Input (2) Analog Inputs(3)	V_{DD} = 4.5 to 5.5V			+5	mA
I_{INJ-}	Pin Injection Current (negative) Digital Input (2) Analog Input	V_{DD} = 4.5 to 5.5V			-5	mA

Notes :

1.An oscillator frequency above 1MHz is recommended for reliable A/D results.

2.A current of ± 5mA can be forced on each pin of the digital section without affecting the functional behaviour of the device. For a positive current injected into one pin, a part of this current (~ 10%) can be expected to flow from the neighbouring pins. A current of – 5mA can be forced on one input of the analog section at a time (or – 2.5mA for all inputs at a time) without affecting the conversion.

3.If a total current of + 1mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 1mA, all the conversion is resulting shifted of + 1 LSB. If a total positive current of + 5mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 5mA, all the conversion is resulting shifted of + 2 LSB.

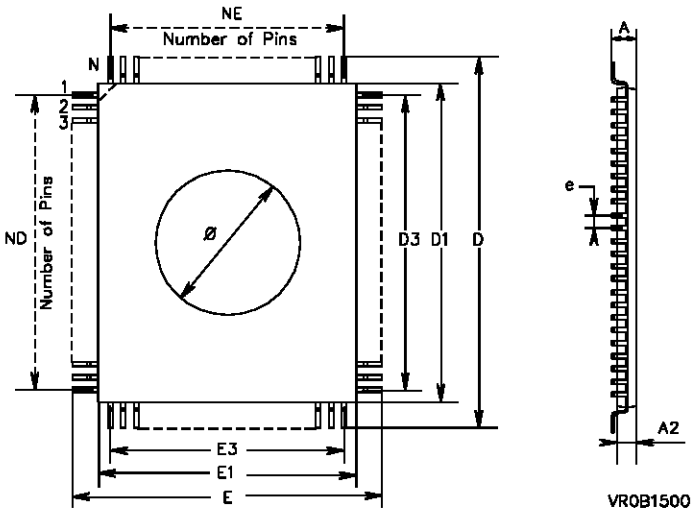
4.Operation below 0.01 MHz is possible but requires increased supply current.

EEPROM INFORMATION

The ST62XX EEPROM single poly process has been specially developed to achieve 300.000 Write/Erase cycles and a 10 years data retention.

PACKAGE MECHANICAL DATA

Figure 3. ST62E80 100 Pin Ceramic Quad Flat Package



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A		3.55	2.80		0.140	
A2		3.40			0.133	
D		23.90			0.941	
D1		20.00			0.787	
D3		18.85			0.742	
E		17.90			0.705	
E1		14.00			0.551	
E3		12.35			0.486	
Ø		7.62			0.3	
e		0.65			0.026	
Number of Pins						
ND	30					
NE	20					
N	100					

ORDERING INFORMATION

ORDERING INFORMATION TABLE

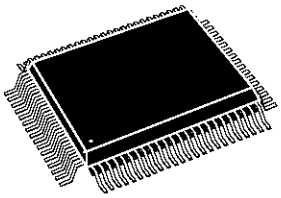
Sales Types	Memory Type	Temperature Range	Package
ST62E80G1	8K EPROM 128 bytes EEPROM	Tested at 25°C only	CQFP100-W
ST62T80	Under development. Please contact your local SGS-THOMSON office.		

SGS-THOMSON MICROELECTRONICS

ST6285

8-BIT HCMOS MCU WITH DOT MATRIX LCD DRIVER AND A/D CONVERTER

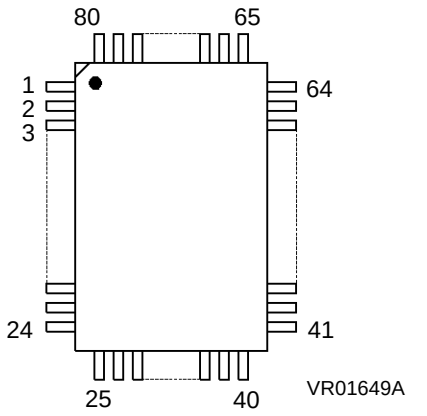
- 4.5 to 6V supply operating range
- 8.4MHz Maximum Clock Frequency
- -40 to +85°C Operating Temperture Range
- Run, Wait & Stop Modes
- 5 different interrupt vectors
- Look-up table capability in EPROM
  User EPROM: 8192 bytes
  Reserved ROM: 244 bytes
  Data RAM: 192 bytes
  LCD RAM: 96 bytes
- PQFP80 package
- 8 fully software programmable I/O as:
  – Input with/without pull-up resistor
  – Input with interrupt generation
  – Open-Drain or Push-pull outputs
  – Analog Inputs
- 4 I/O lines can sink up to 20mA for direct LED or TRIAC driving and have SPI alternate functions
- One 8-bit counter with 7-bit programmable prescalers (Timer 1)
- Software activated digital watchdog
- 8-bit A/D converter with up to 12 analog inputs
- 8-bit synchronous serial peripheral interface (SPI)
- LCD driver with 40 segment outputs, 8 backplane outputs and 8 software selectable segment/backplane outputs able to drive up to 40x16 (640) or 48x8 (384) LCD segments.
- One external not maskable interrupt
- 9 powerful addressing modes
- The accumulator, the X, Y, V & W registers, the port and peripherals data & control registers are addressed in the data space as RAM locations.
- The ST62E85 is the EPROM version, ST62T85 is the OTP version, fully compatible with ST6285 ROM version.



PQFP80

(Ordering Information at the end of the datasheet)

Figure 1. 80 Pin Quad Flat Pack (QFP) Package Pinout



ST6285 Pin Description

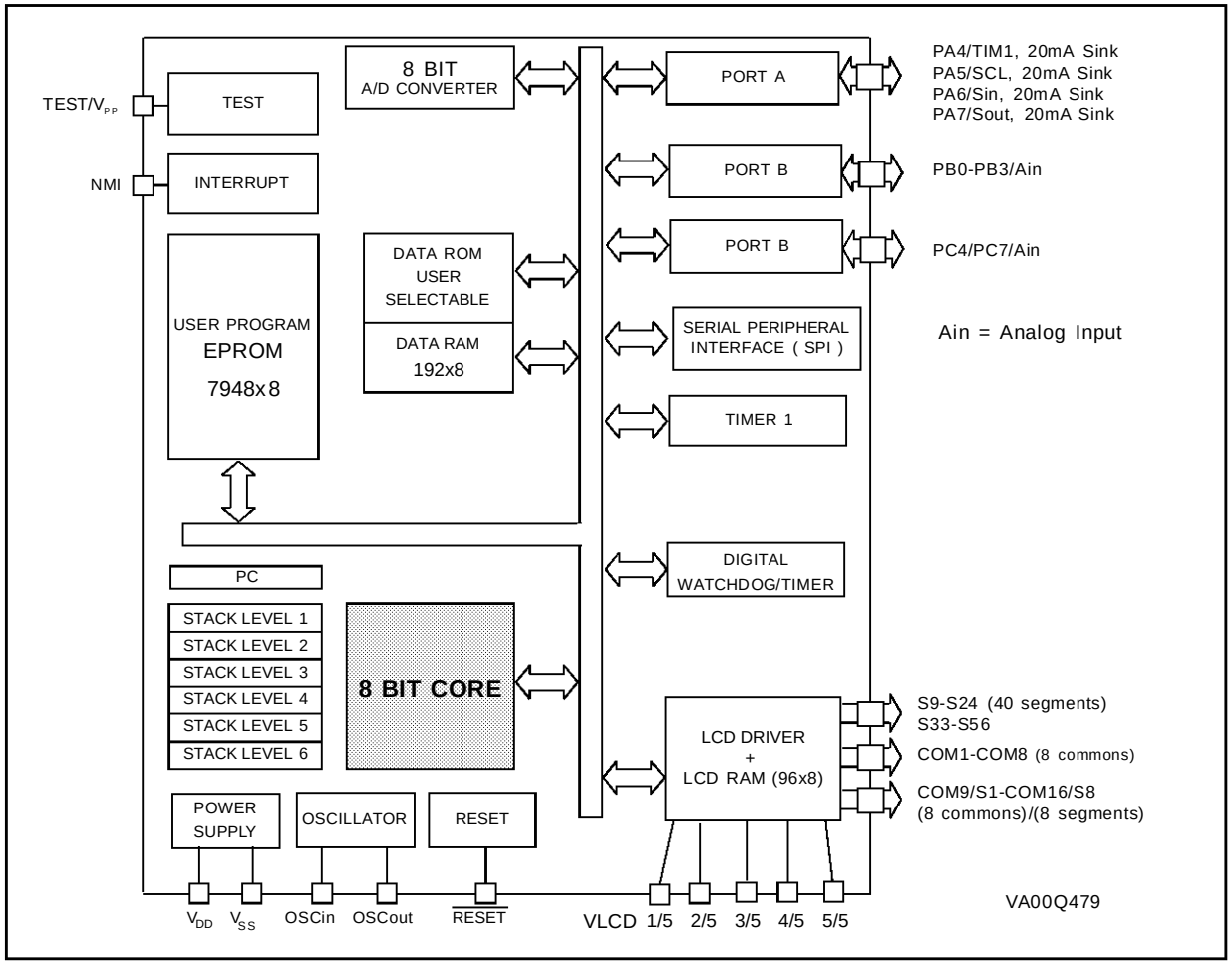
Pin number	Pin name	Pin number	Pin name	Pin number	Pin name	Pin number	Pin name
1	S41	25	PC7	41	COM1	65	S17
2	S42	26	PC6	42	COM2	66	S18
3	S43	27	PC5	43	COM3	67	S19
4	S44	28	PC4	44	COM4	68	S20
5	S45	29	NMI	45	COM5	69	S21
6	S46	30	V_{DD}	46	COM6	70	S22
7	S47	31	V_{SS}	47	COM7	71	S23
8	S48	32	VLCD	48	COM8	72	S24
9	S49	33	VLCD4/5	49	COM9/S1	73	S33
10	S50	34	VLCD3/5	50	COM10/S2	74	S34
11	S51	35	VLCD2/5	51	COM11/S3	75	S35
12	S52	36	VLCD1/5	52	COM12/S4	76	S36
13	S53	37	PA7/Sout (1)	53	COM13/S5	77	S37
14	S54	38	PA6/Sin (1)	54	COM14/S6	78	S38
15	S55	39	PA5/SCL (1)	55	COM15/S7	79	S39
16	S56	40	PA4/TIM1 (1)	56	COM16/S8	80	S40
17	PB3			57	S9		
18	PB2			58	S10		
19	PB1			59	S11		
20	PB0			60	S12		
21	TEST/V_{PP}			61	S13		
22	OSCout			62	S14		
23	OSCin			63	S15		
24	$\overline{\text{RESET}}$			64	S16		

Note 1: 20mA SINK

GENERAL DESCRIPTION

The ST6285 microcontroller is a member of the 8-bit HCMOS ST62xx family, a series of devices oriented to low-medium complexity applications. All ST62xx members are based on a building block approach: a common core is associated with a combination of on-chip peripherals (macrocells). The macrocells of the ST6285 family are: an advanced LCD driver/controller with 40 segments, 8 backplanes and 8 software selectable segment/backplane outputs able to drive up to 40×16 (640) or 48×8 (384) segments, one 8 bit standard timer/counter with a 7-bit software programmable prescaler (Timer 1), the digital watchdog timer (DWD), an 8-bit A/D Converter with up to 12 analog inputs and an 8-bit synchronous serial peripheral interface (SPI). Thanks to these peripherals the ST6285 family is well suited for general purpose, automotive, security, appliance and industrial applications. The ST62E85 EPROM version is available for prototyping and low-volume production, an OTP version is also available.

Figure 2. ST6285 Block Diagram



PIN DESCRIPTION

V_{DD} and **V_{SS}**. Power is supplied to the MCU using these two pins. V_{DD} is power and V_{SS} is the ground connection.

OSCin and OSCout. These pins are internally connected with the on-chip oscillator circuit. A quartz crystal or a ceramic resonator can be connected between these two pins in order to allow the correct operation of the MCU with various stability/cost trade-offs. OSCin is the input pin, OSCout is the output pin. An external clock signal can be applied to OSCin.

$\overline{\text{RESET}}$. The active low $\overline{\text{RESET}}$ pin is used to restart the microcontroller at the beginning of its program. The $\overline{\text{RESET}}$ pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

TEST. TEST must be held at V_{SS} for normal operation (an internal pull-down resistor selects normal operating mode if TEST pin is not connected).

NMI. The NMI pin provides the capability for asynchronous applying an external top priority interrupt to the MCU. This pin is falling edge sensitive. The NMI pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

PA4/TIM1. This pin can be used as Timer 1 I/O pin. In input mode it is connected to the timer prescaler input and acts as the external timer clock or as the control gate for the internal timer clock. In the output mode the timer pin outputs the timer data bit when a time out occurs.
To use this pin as Timer output the I/O pin has to be programmed as open-drain output. To use this pin as Timer input the I/O pin has to be programmed as input.

PA4-PA7. These 4 lines are organized as one I/O port (A). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. In output mode these lines can also sink 20mA for direct LED or triac driving. PA5-PA7 can also be used as respectively Clock, Data in and Data out pins for the on-chip SPI to carry the synchronous serial I/O signals. PA4 can also be used as the TIMER 1 I/O pin.

PB0-PB3. These 4 lines are organized as one I/O port (B). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, open-drain or push-pull output or as an analog input for the A/D converter.

PC4-PC7. These 4 lines are organized as one I/O port (C). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. PC4-PC7 can be programmed as analog inputs for the A/D converter. Port C has schmitt trigger inputs and a 5mA drive capability in output mode.

COM1-COM8. These eight pins are the LCD peripheral common outputs. They are the outputs of the on-chip backplane voltage generator which is used for multiplexing the LCD segment lines.

S9/S24-S33/S56. These pins are the 40 LCD peripheral driver outputs of the ST62E85, ST62T85. Segments S1-S8 are multiplexed with COM9-COM16 and their function is software selectable.

COM9/S1-COM16/S8. These pins are the 8 multiplexed common/segment lines. Under software selected control, they can act as LCD common outputs allowing a 40×16 dot matrix operation, or they can act as segment outputs allowing 48×8 dot matrix operation.

VLCD1/5-VLCD5/5. Resistor network nodes for determining the intermediate display voltage levels on COM1-COM8/COM16 and S1/S8-S56 pins during multiplex operation.

ST6285 DESCRIPTION

With the following exceptions, the ST6285 has the same software and hardware features as the ST6280:

1 - LCD RAM: The accessible segments are segments 9 to 24 and 33 to 56.

2 - I/O: To prevent floating input or uncontrolled I/O interrupt, the port bit PA0-PA3, PB4-PB7 must be programmed as push-pull output.

3 - Data Memory Space: Write 40h at the address DFh of the data memory space (desabled EEPROM).

4 - Do not access data space locations: C7, CD, D9, DA, DB, DF, E5 to FE.

5 - EEPROM and Auto-reload Timer: Not available.

THE READER IS ASKED TO REFER TO THE DATASHEET OF THE ST6280 FOR FURTHER DETAILS.

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

This product contains devices to protect the inputs against damage due to high static voltages, however it is advised to take normal precaution to avoid application of any voltage higher than maximum rated voltages.

For proper operation it is recommended that VI and VO must be higher than VSS and smaller than VDD. Reliability is enhanced if unused inputs are connected to an appropriated logic voltage level (VDD or VSS).

Power Considerations. The average chip-junction temperature, Tj, in Celsius can be obtained from :

$Tj = T_A + PD \times RthJA$

Where: T_A = Ambient Temperature.

RthJA = Package thermal resistance (junction-to ambient).

PD = Pint + Pport.

Pint = $I_{DD} \times V_{DD}$ (chip internal power).

Pport = Port power dissipation (determinated by the user).

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.3 to 7.0	V
V_{LCD}	Display Voltage	-0.3 to 11.0	V
V_I	Input Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
V_O	Output Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
I_O	Current Drain per Pin Excluding V_{DD} & V_{SS}	± 10	mA
IV_{DD}	Total Current into V_{DD} (source)	50	mA
IV_{SS}	Total Current out of V_{SS} (sink)	50	mA
Tj	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-60 to 150	°C

Notes :
- Stresses above those listed as "absolute maximum ratings" may cause permanent damage to the device . This is a stress rating only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
- (1). Within these limits, clamping diodes are guaranteed not to be conductive. Voltage outside these limits are authorised as long as injection current is kept within the specification.

THERMAL CHARACTERISTIC

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
RthJA	Thermal Resistance	PQFP80		70		°C/W

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
T_A	Operating Temperature	1 Suffix Version 6 Suffix Version	0 -40		70 85	°C
V_{DD}	Operating Supply Voltage (1)		4.5		6	V
V_{LCD}	Display Voltage (1)		3		10	V
V_{DD}	RAM Retention Voltage (1)		2			V

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency (1)(4)	$V_{DD} \geq 4.5V$	0.01		8.4	MHz
I_{INJ+}	Pin Injection Current (positive) Digital Input (2) Analog Input (3)	V_{DD} = 4.5 to 5.5V			+5	mA
I_{INJ-}	Pin Injection Current (negative) Digital Input (2) Analog Input	V_{DD} = 4.5 to 5.5V			-5	mA

Notes :
1. An oscillator frequency above 1MHz is recommended for reliable A/D results.
2. A current of ± 5mA can be forced on each pin of the digital section without affecting the functional behaviour of the device. For a positive current injected into one pin, a part of this current (~ 10%) can be expected to flow from the neighbouring pins. A current of -5mA can be forced on one input of the analog section at a time (or -2.5mA for all inputs at a time) without affecting the conversion.
3. If a total current of +1mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 1mA, all the conversion is resulting shifted by +1LSB. If a total positive current of +5mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 5mA, all the conversion is resulting shifted by +2LSB.
4. Operation below 0.01 MHz is possible but requires increased supply current.

DC ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
V_{IL}	Input Low Level Voltage	RESET, NMI, TIMER Pin			$0.3V_{DD}$	V
V_{IH}	Input High Level Voltage	TIMER Pin	$0.80V_{DD}$			V
		RESET, NMI Pin	$0.70V_{DD}$			V
I_{IL} I_{IH}	Input Leakage Current	RESET Pin V_{DD} =5V $V_{IN} = V_{DD}$ (1) $V_{IN} = V_{DD}$ (2) $V_{IN} = V_{SS}$ (5)			10 1 50	µA mA µA
V_{OL}	Low Level Output Voltage	TIMER, I_{OL} = 5.0mA			$0.3V_{DD}$	V
V_{OH}	High Level Output Voltage	TIMER, I_{OL} = -5.0mA	$0.65V_{DD}$			V

Notes: on next page

DC ELECTRICAL CHARACTERISTICS (Continued)
(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
R_{PU}	Pull-up Resistor	V_{IN}=0V V_{DD}=5V NMI	50	100	200	kΩ
		RESET	200	300	500	kΩ
I_{IL} I_{IH}	Input Leakage Current	TIMER $V_{IN} = V_{DD}$ or V_{SS}		0.1	1.0	µA
I_{IL} I_{IH}	Input Leakage Current	NMI VDD = 5.V $V_{IN} = V_{SS}$ (5) $V_{IN} = V_{DD}$			100 1.0	µA
I_{DD}	Supply Current RUN Mode	f_{OSC} = 8MHz, I_{LOAD} = 0mA V_{DD} = 5.0V		4	7	mA
	Supply Current WAIT Mode (4)	f_{OSC} = 8MHz, I_{LOAD} = 0mA V_{DD} = 5.0V		1	3	mA
	Supply Current RESET Mode	f_{OSC} = 8MHz, $V_{RESET} = V_{SS}$		1	7	mA
	Supply Current STOP Mode (3)(4)	I_{LOAD} = 0mA V_{DD} = 5.0V		1	10	µA

Notes :
1. No Watchdog Reset activated.
2. Reset generated by Watchdog.
3. When the watchdog function is actvated the STOP instruction is deactivated. WAIT instruction is automatically executed.
4. All on-chip peripherals in OFF state
5. Pull-up resistor

AC ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency (2)	$V_{DD} \geq 4.5V$	0.01		8.4	MHz
t_{SU}	Oscillator Start-up Time	$C_{L1} = C_{L2}$ = 22pF - crystal		5	20	ms
t_{SR}	Supply Rise Time	10% to 90%	0.01		100	
t_{REC}	Supply Recovery Time (1)		100			
T_W	Minimum Pulse Width	NMI Pin V_{DD} = 5V	100			ns
		RESET Pin	100			ns
C_{IN}	Input Capacitance	All Inputs Pins			10	pF
C_{OUT}	Output Capacitance	All Outputs Pins			10	pF

Notes:
1. Period for which V_{DD} has to be connected or at 0V to allow internal Reset function at next power-up.
2. Operation below 0.01 MHz is possible but requires increased supply current.

I/O PORTS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
V_{IL}	Input Low Level Voltage	I/O Pins			$0.3V_{DD}$	V
V_{IH}	Input High Level Voltage	I/O Pins	$0.7V_{DD}$			V
V_{OL}	Low Level Output Voltage	V_{DD} = 5.0V I_{OL} = 10µA, All I/O Pins I_{OL} = 5mA, Standard I/O I_{OL} =10mA, PA/PC0-PC3 I_{OL} =20mA, PA/PC0-PC3			0.1 0.8 0.8 1.3	V
V_{OH}	High Level Output Voltage	I/O Pins, I_O = -10µA (source)	V_{DD}-0.1			V
		I/O Pins, I_{OL} = -V_{DD}x1mA V_{DD} = 5.0V	$0.6xV_{DD}$			V
I_{IL} I_{IH}	Input Leakage Current	I/O Pins,(1)		0.1	1.0	µA
R_{PU}	Pull-up Resistor	I/O Pins V_{IN} = 0V, V_{DD} = 5.0V	50	100	200	KΩ

Note 1. Pull-up resistor off

SPI ELECTRICAL CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified, V_{DD}=5.0V)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
F_{CL}	Clock Frequency	applied on PB5/SCL			1	MHz
t_{SU}	Set-up Time	applied on PB6/Sin		50		ns
t_h	Hold Time	applied on PB6/Sin		100		ns

A/D CONVERTER CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
Res	Resolution (3)			8		Bit
A_{TOT}	Total Accuracy (3)	f_{OSC} > 1.2 MHz f_{OSC} > 32kHz			± 2 ±4	LSB
t_C (1)	Conversion Time	f_{OSC} = 8MHz		70		µs
V_{AN}	Conversion Range		V_{SS}		V_{DD}	V
ZIR	Zero Input Reading	Conversion result when $V_{IN} = V_{SS}$	00			Hex
FSR	Full Scale Reading	Conversion result when $V_{IN} = V_{DD}$			FF	Hex
AD_I	Analog Input Current During Conversion	V_{DD}= 4.5V			1.0	µA
AC_{IN} (2)	Analog Input Capacitance			2	5	pF
ASI	Analog Source Impedance (4)				30	kΩ

Notes:
1. With oscillator frequencies less than 1MHz, the A/D Converter accuracy is decreased.
2. Excluding Pad Capacitance
3. Noise at V_{DD},V_{SS} ≤ 10m
4. A value higher than 30kΩ may disturb the conversion if the ADC is switched between different I/O pins connected to the Converter. The reason is that the ADC input capacitance (10pF) has to be charged before the beginning of the conversion. If the serial input impedance is high, the stabilisation time of the input voltage is non-negligible versus the total conversion time.
   If the input impedance is higher than 30kΩ a small decoupling capacitance can be added to the ADCin pins, and a short delay can be introduced by software, between the I/O switching and the beginning of the conversion.
   If the ADC is always connected to the same I/O pin, the ADC input capacitance is always loaded and the serial impedance can be higher. Its maximum value has just to be small enough not to disturb the input voltage during the conversion (1nA each 3µs through the I/O during the conversion).

TIMER 1 CHARACTERISTICS

(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
t_{RES}	Resolution		$\frac{12}{f_{OSC}}$			s
f_{IN}	Input Frequency on PA4/TIM1 Pin				$\frac{f_{OSC}}{8}$	MHz
t_W	Pulse Width at PA4/TIM1 Pin	V_{DD} = 3.0V $V_{DD} \geq$ 4.5V	1 125			µs ns

LCD ELECTRICAL CHARACTERISTICS

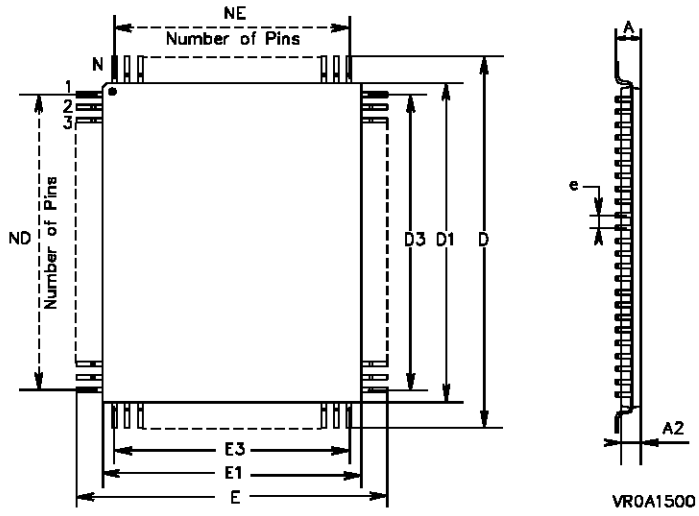
(T_A = -40 to +85°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{FR}	Frame Frequency	f_{OSC} = 1, 2, 4, 8 MHz	64		512	Hz
V_{OS}	DC Offset Voltage(1)	$V_{LCD} = V_{DD}$, no load			50	mV
V_{OH}	COM High Level, Output Voltage	I = 100µA, V_{LCD} = 5V	4.5			V
V_{OL}	COM Low Level, Output Voltage	I = 100µA, V_{LCD} = 5V			0.5	V
V_{OH}	SEG High Level, Output Voltage	I = 50µA, V_{LCD} = 5V	4.5			V
V_{OL}	SEG Low Level, Output Voltage	I = 50µA, V_{LCD} = 5V			0.5	V
V_{LCD}	Display Voltage		3		10	V

Note 1. The DC offset voltage refers to all segment and common outputs. It is the difference between the measured voltage value and nominal value for every voltage level. Ri of voltage meter must be greater than or equal to 10MΩ.

PACKAGE MECHANICAL DATA

Figure 3. ST6285 80 Pin Plastic Quad Flat Package



Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A			3.30			0.130
A2	2.55	2.80	3.05	0.100	0.110	0.120
D	23.00	23.20	23.40	0.906	0.913	0.921
D1	19.90	20.00	20.10	0.783	0.787	0.791
D3		18.40			0.724	
E	17.00	17.20	17.40	0.669	0.677	0.685
E1	13.90	14.00	14.10	0.547	0.551	0.555
E3		12.00			0.472	
e		0.80			0.032	
Number of Pins						
ND	24					
NE	16					
N	80					

ORDERING INFORMATION

The following chapter deals with the procedure for transfer the Program/Data ROM codes to SGS-THOMSON.

Communication of the ROM Codes. To communicate the contents of Program/Data ROM memories to SGS-THOMSON, the customer has to send :

– one file in INTEL INTELLEC 8/MDS FORMAT (in an MS-DOS 5" diskette) for the PROGRAM Memory

– one file in INTEL INTELLEC 8/MDS FORMAT (in a MS-DOS 5" diskette) for the EEPROM initial content (this file is optional)

– a filled Option List form as described in the OPTION LIST paragraph.

The program ROM should respect the ROM Memory Map as in the following table.

The ROM code must be generated with ST6 assembler. Before programming the EPROM, the buffer of the EPROM programmer must be filled with FFh.

ROM Memory MAP

ROM Page	Device Address	Description
Page 0	0000h-007Fh 0080h-07FFh	Reserved User ROM
Page 1 "STATIC"	0800h-0F9Fh 0FA0h-0FEFh 0FF0h-0FF7h 0FF8h-0FFBh 0FFCh-0FFDh 0FFEh-0FFFh	User ROM Reserved Interrupt Vectors Reserved NMI Vector Reset Vector
Page 2	0000h-000Fh 0010h-07FFh	Reserved User ROM
Page 3	0000h-000Fh 0010h-07FFh	Reserved User ROM

Note : EPROM addresses are related to the ROM file to be processed.

Listing Generation & Verification.

When SGS-THOMSON receives the Codes, they are compared and a computer listing is generated from them. This listing refers exactly to the mask that will be used to produce the microcontroller. Then the listing is returned to the customer that must thoroughly check, complete, sign and return it to SGS-THOMSON. The signed list constitutes a part of the contractual agreement for the creation of the customer mask. SGS-THOMSON sales organization will provide detailed information on contractual points.

ORDERING INFORMATION TABLE

Sales Types	Temperature Range	Package
ST6285Q1/XX	0 to + 70°C	PQFP80
ST6285Q6/XX	-40 to + 85°C	PQFP80

Note : "XX" is the ROM code identifier allocated by SGS-THOMSON after receipt of all required options and the related ROM file.

ST6285 MICROCONTROLLER OPTION LIST

Customer . . . . . .
Address . . . . . .

Contact . . . . . .
Phone No . . . . . .
Reference . . . . . .

SGS-THOMSON Microelectronics references

Device
[ ] ST6285

Package
[ ] Plastic Quad Flat Package

Temperature Range
[ ] 0°C to + 70°C [ ] -40°C to + 85°C

Special Marking
[ ] No
[ ] Yes "_ _ _ _ _ _ _ _ _ _"
Authorized characters are Letters, digits, '.', '–', '/' and spaces only.
For special marking one line with 10 characters maximum is possible.

Power supply:
[ ] Standard (4.5V to 6V)

Comments :
- Number of LCD segments used :
- Number of LCD backplanes used :
- Multiplexing rate:

Note :

Signature

Date

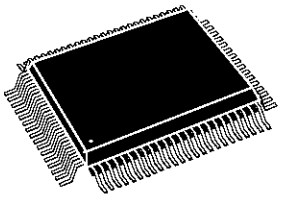
NOTES:

SGS-THOMSON MICROELECTRONICS

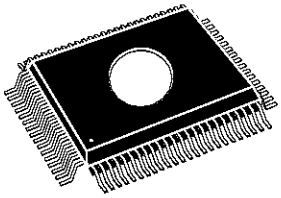
ST62E85
ST62T85

8-BIT EPROM HCMOS MCU WITH DOT MATRIX LCD DRIVER AND A/D CONVERTER

- 4.5 to 6V supply operating range
- 8.4MHz Maximum Clock Frequency
- -40 to +85°C Operating Temperture Range
- Run, Wait & Stop Modes
- 5 different interrupt vectors
- Look-up table capability in EPROM
  User EPROM: 8192 bytes
  Data RAM: 192 bytes
  LCD RAM: 96 bytes
- PQFP80 and CQFP80-W packages
- 8 fully software programmable I/O as:
  – Input with/without pull-up resistor
  – Input with interrupt generation
  – Open-Drain or Push-pull outputs
  – Analog Inputs
- 4 I/O lines can sink up to 20mA for direct LED or TRIAC driving and have SPI alternate functions
- One 8-bit counter with 7-bit programmable prescalers (Timer 1)
- Software activated digital watchdog
- 8-bit A/D converter with up to 12 analog inputs
- 8-bit synchronous serial peripheral interface (SPI)
- LCD driver with 40 segment outputs, 8 backplane outputs and 8 software selectable segment/backplane outputs able to drive up to 40x16 (640) or 48x8 (384) LCD segments.
- One external not maskable interrupt
- 9 powerful addressing modes
- The accumulator, the X, Y, V & W registers, the port and peripherals data & control registers are addressed in the data space as RAM locations.
- The ST62E85 is the EPROM version, ST62T85 is the OTP version, fully compatible with ST6285 ROM version.



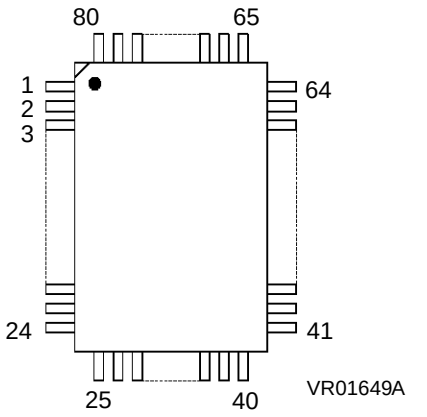
PQFP80



CQFP80-W

(Ordering Information at the end of the datasheet)

Figure 1. 80 Pin Quad Flat Pack (QFP) Package Pinout



ST62E85 Pin Description

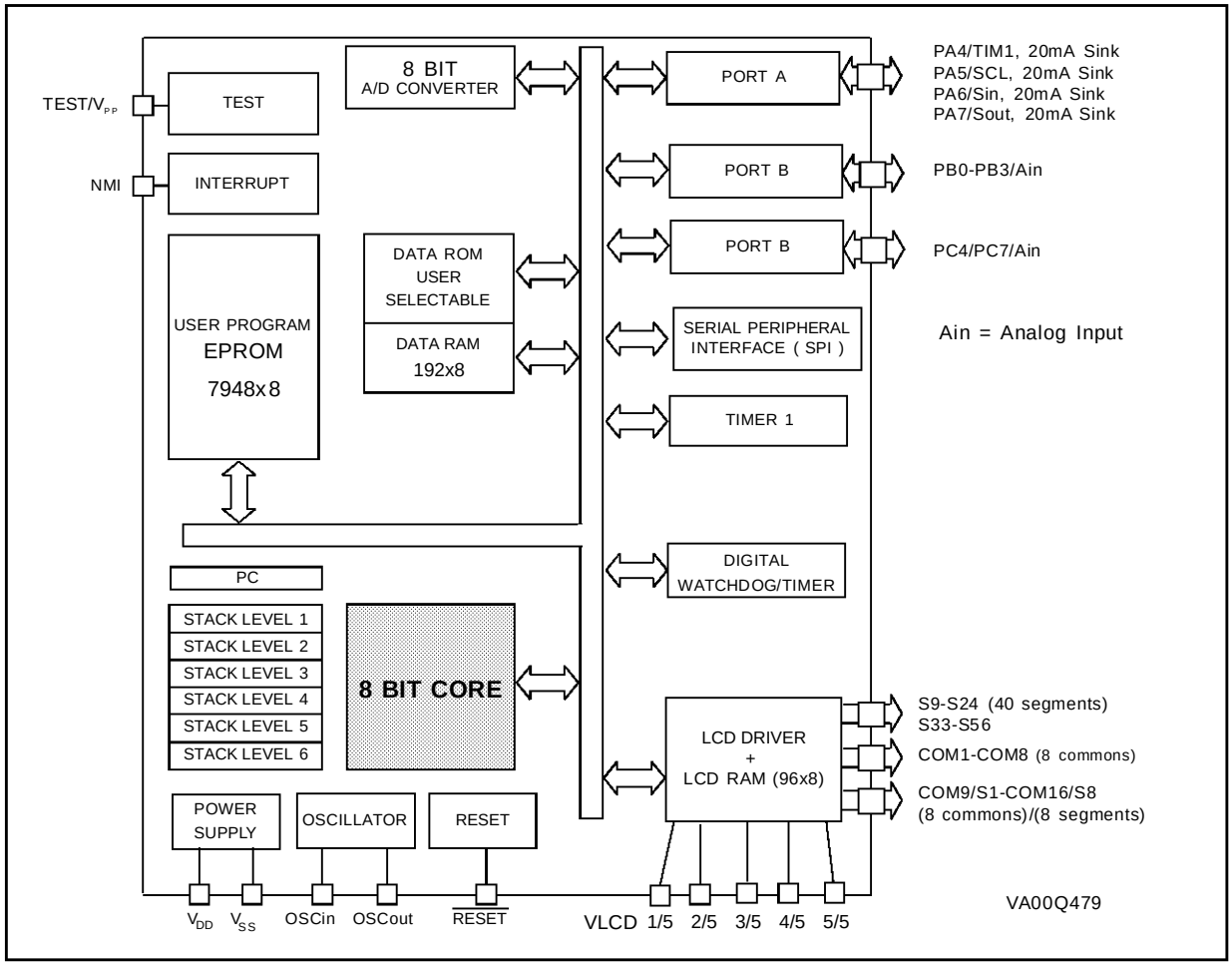
Pin number	Pin name	Pin number	Pin name	Pin number	Pin name	Pin number	Pin name
1	S41	25	PC7	41	COM1	65	S17
2	S42	26	PC6	42	COM2	66	S18
3	S43	27	PC5	43	COM3	67	S19
4	S44	28	PC4	44	COM4	68	S20
5	S45	29	NMI	45	COM5	69	S21
6	S46	30	V_{DD}	46	COM6	70	S22
7	S47	31	V_{SS}	47	COM7	71	S23
8	S48	32	VLCD	48	COM8	72	S24
9	S49	33	VLCD4/5	49	COM9/S1	73	S33
10	S50	34	VLCD3/5	50	COM10/S2	74	S34
11	S51	35	VLCD2/5	51	COM11/S3	75	S35
12	S52	36	VLCD1/5	52	COM12/S4	76	S36
13	S53	37	PA7/Sout (1)	53	COM13/S5	77	S37
14	S54	38	PA6/Sin (1)	54	COM14/S6	78	S38
15	S55	39	PA5/SCL (1)	55	COM15/S7	79	S39
16	S56	40	PA4/TIM1 (1)	56	COM16/S8	80	S40
17	PB3			57	S9		
18	PB2			58	S10		
19	PB1			59	S11		
20	PB0			60	S12		
21	TEST/V_{PP}			61	S13		
22	OSCout			62	S14		
23	OSCin			63	S15		
24	$\overline{\text{RESET}}$			64	S16		

Note 1: 20mA SINK

GENERAL DESCRIPTION

The ST62E85, ST62T85 microcontrollers are members of the 8-bit HCMOS ST62xx family, a series of devices oriented to low-medium complexity applications. They are the EPROM/OTP versions of the ST6285 ROM device and are suitable for prototyping and low-volume production. All ST62xx members are based on a building block approach: a common core is associated with a combination of on-chip peripherals (macrocells). The macrocells of the ST6285 family are: an advanced LCD driver/controller with 40 segments, 8 backplanes and 8 software selectable segment/backplane outputs able to drive up to 40×16 (640) or 48×8 (384) segments, one 8 bit standard timer/counter with a 7-bit software programmable prescaler (Timer 1), the digital watchdog timer (DWD), an 8-bit A/D Converter with up to 12 analog inputs and an 8-bit synchronous serial peripheral interface (SPI). Thanks to these peripherals the ST6285 family is well suited for general purpose, automotive, security, appliance and industrial applications.

Figure 2. ST62E85/T85 Block Diagram



PIN DESCRIPTION

V_{DD} and **V_{SS}**. Power is supplied to the MCU using these two pins. V_{DD} is power and V_{SS} is the ground connection.

OSCIN and OSCOUT. These pins are internally connected with the on-chip oscillator circuit. A quartz crystal or a ceramic resonator can be connected between these two pins in order to allow the correct operation of the MCU with various stability/cost trade-offs. OSCin is the input pin, OSCout is the output pin. An external clock signal can be applied to OSCin.

$\overline{\text{RESET}}$. The active low $\overline{\text{RESET}}$ pin is used to restart the microcontroller at the beginning of its program. The $\overline{\text{RESET}}$ pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

TEST/. TEST must be held at V_{SS} for normal operation (an internal pull-down resistor selects normal operating mode if TEST pin is not connected).

NMI. The NMI pin provides the capability for asynchronous applying an external top priority interrupt to the MCU. This pin is falling edge sensitive. The NMI pin is provided with an on-chip pull-up resistor and schmitt trigger input characteristics.

PA4/TIM1. This pin can be used as Timer 1 I/O pin. In input mode it is connected to the timer prescaler input and acts as the external timer clock or as the control gate for the internal timer clock. In the output mode the timer pin outputs the timer data bit when a time out occurs.
To use this pin as Timer output the I/O pin has to be programmed as open-drain output. To use this pin as Timer input the I/O pin has to be programmed as input.

PA4-PA7. These 4 lines are organized as one I/O port (A). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. In output mode these lines can also sink 20mA for direct LED or triac driving. PA5-PA7 can also be used as respectively Clock, Data in and Data out pins for the on-chip SPI to carry the synchronous serial I/O signals. PA4 can also be used as the TIMER 1 I/O pin.

PB0-PB3. These 4 lines are organized as one I/O port (B). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, open-drain or push-pull output or as an analog input for the A/D converter.

PC4-PC7. These 4 lines are organized as one I/O port (C). Each line may be configured under software control as an input with or without internal pull-up resistor, an interrupt generating input with pull-up resistor, or an open-drain or push-pull output. PC4-PC7 can be programmed as analog inputs for the A/D converter. Port C has schmitt trigger inputs and a 5mA drive capability in output mode.

COM1-COM8. These eight pins are the LCD peripheral common outputs. They are the outputs of the on-chip backplane voltage generator which is used for multiplexing the LCD segment lines.

S9/S24-S33/S56. These pins are the 40 LCD peripheral driver outputs of the ST62E85, ST62T85. Segments S1-S8 are multiplexed with COM9-COM16 and their function is software selectable.

COM9/S1-COM16/S8. These pins are the 8 multiplexed common/segment lines. Under software selected control, they can act as LCD common outputs allowing a 40×16 dot matrix operation, or they can act as segment outputs allowing 48×8 dot matrix operation.

VLCD1/5-VLCD5/5. Resistor network nodes for determining the intermediate display voltage levels on COM1-COM8/COM16 and S1/S8-S56 pins during multiplex operation.

ST62E85/ST62T85 EPROM/OTP DESCRIPTION.

The ST62E85 is the EPROM version of the ST6285 ROM product. It is intended for use during the development of an application, and for pre-production and small volume production. The ST62T85 OTP has the same characteristics. They both include EPROM memory instead of the ROM memory of the ST6285, and so the program and constants of the program can be easily modified by the user with the ST62E85 EPROM programming board of from SGS-THOMSON.

From a user point of view (with the following exceptions) the ST62E85, ST62T85 products have exactly the same software and hardware features of the ROM version.

On the ST62E85, all the 8192 bytes of PROGRAM memory are available for the user, as all the EPROM memory can be erased by exposure to UV light. On the ST62T85 (OTP) device a reserved area for test purposes exists, as for the ST6285 ROM device. In order to avoid any discrepancy between program functionality when using the EPROM, OTP and ROM it is recommended not to use these reserved areas, even when using the ST62E85.

Notes on programming:

In order to emulate exactly the ST6285 features with the ST62E85 and ST62T85, some software precautions have to be taken:

1. LCD RAM: The accessible segments are segments 9 to 24 and 33 to 56.

2. I/O: To prevent floating input or uncontrolled I/O interrupt, the port bit PA0-PA3, PB4-PB7 must be programmed as push-pull output.

3. Data Memory Space: write 40h at the address DFh of the Data Memory Space (disabled EEPROM).

4. Do not access data space locations: C7, CD, D9, DA, DB, DF, E5 to FE.

Other than these exceptions, the ST62E85, ST62T80 parts are fully compatible with the ROM ST6280 equivalent, this datasheet thus provides only information specific to the EPROM based devices.

THE READER IS ASKED TO REFER TO THE DATASHEET OF THE ST6285 ROM-BASED DEVICE FOR FURTHER DETAILS.

EPROM ERASING

The EPROM of the windowed package of the ST62E85 may be erased by exposure to Ultra Violet light.

The erasure characteristic of the ST62E85 EPROM is such that erasure begins when the memory is exposed to light with wave lengths shorter than approximately 4000Å. It should be noted that sunlight and some types of fluorescent lamps have wavelengths in the range 3000-4000Å. It is thus recommended that the window of the ST62E85 package be covered by an opaque label to prevent unintentional erasure problems when testing the application in such an environment.
The recommended erasure procedure of the ST62E85 EPROM is exposure to short wave ultraviolet light which has wavelength 2537Å. The integrated dose (i.e. UV intensity x exposure time) for erasure should be a minimum of 15 W-sec/cm^2. The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with 12000µW/cm^2 power rating. The ST62E85 should be placed within 2.5 cm (1 inch) of the lamp tubes during erasure.

ST62T80 OTP Memory Map

ROM Page	Device Address	Description
Page 0	0000h-007Fh 0080h-07FFh	Reserved User ROM
Page 1 “STATIC”	0800h-0F9Fh 0FA0h-0FEFh 0FF0h-0FF7h 0FF8h-0FFBh 0FFCh-0FFDh 0FFEh-0FFFh	User ROM Reserved Interrupt Vectors Reserved NMI Vector Reset Vector
Page 2	0000h-000Fh 0010h-07FFh	Reserved User ROM
Page 3	0000h-000Fh 0010h-07FFh	Reserved User ROM

Note : EPROM addresses are related to the ROM file to be processed.

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

This product contains devices to protect the inputs against damage due to high static voltages, however it is advised to take normal precaution to avoid application of any voltage higher than maximum rated voltages.

For proper operation it is recommended that VI and VO must be higher than V_{SS} and smaller than V_{DD}. Reliability is enhanced if unused inputs are connected to an appropriated logic voltage level (V_{DD} or V_{SS}).

Power Considerations. The average chip-junction temperature, Tj, in Celsius can be obtained from :

$Tj = T_A + PD \times RthJA$

Where: T_A = Ambient Temperature.

RthJA = Package thermal resistance (junction-to ambient).

PD = Pint + Pport.

Pint = $I_{DD} \times V_{DD}$ (chip internal power).

Pport = Port power dissipation (determinated by the user).

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.3 to 7.0	V
V_{LCD}	Display Voltage	-0.3 to 11.0	V
V_I	Input Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
V_O	Output Voltage	V_{SS} - 0.3 to V_{DD} + 0.3 (1)	V
I_O	Current Drain per Pin Excluding V_{DD} & V_{SS}	± 10	mA
IV_{DD}	Total Current into V_{DD} (source)	50	mA
IV_{SS}	Total Current out of V_{SS} (sink)	50	mA
Tj	Junction Temperature	150	°C
T_{STG}	Storage Temperature	-60 to 150	°C

Notes :
- Stresses above those listed as "absolute maximum ratings" may cause permanent damage to the device . This is a stress rating only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.
- (1). Within these limits, clamping diodes are guaranteed not to be conductive. Voltage outside these limits are authorised as long as injection current is kept within the specification.

THERMAL CHARACTERISTICS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
RthJA	Thermal Resistance	PQFP80		70		°C/W

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
T_A	Operating Temperature	1 Suffix Version 6 Suffix Version	0 -40		70 85	°C
V_{DD}	Operating Supply Voltage (1)		4.5		6	V
V_{LCD}	Display Voltage (1)		3		10	V
V_{DD}	RAM Retention Voltage (1)		2			V

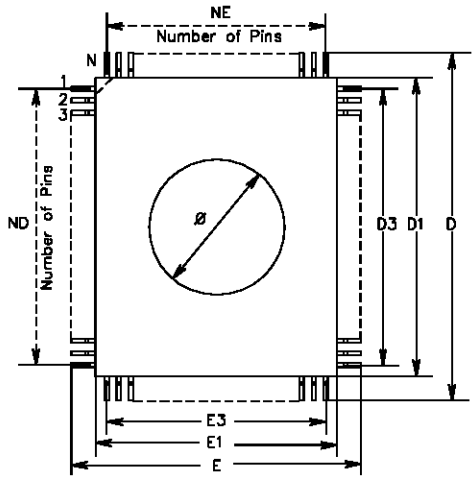
RECOMMENDED OPERATING CONDITIONS (Continued)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
f_{OSC}	Oscillator Frequency (1)(4)	$V_{DD} \geq 4.5V$	0.01		8.4	MHz
I_{INJ+}	Pin Injection Current (positive) Digital Input (2) Analog Input (3)	V_{DD} = 4.5 to 5.5V			+5	mA
I_{INJ-}	Pin Injection Current (negative) Digital Input (2) Analog Input	V_{DD} = 4.5 to 5.5V			-5	mA

Notes :
1. An oscillator frequency above 1MHz is recommended for reliable A/D results.
2. A current of ± 5mA can be forced on each pin of the digital section without affecting the functional behaviour of the device. For a positive current injected into one pin, a part of this current (~ 10%) can be expected to flow from the neighbouring pins. A current of -5mA can be forced on one input of the analog section at a time (or -2.5mA for all inputs at a time) without affecting the conversion.
3. If a total current of +1mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 1mA, all the conversion is resulting shifted by +1LSB. If a total positive current of +5mA is flowing into the single analog channel or if the total current flowing into all the analog inputs is of 5mA, all the conversion is resulting shifted by +2LSB.
4. Operation below 0.01 MHz is possible but requires increased supply current.

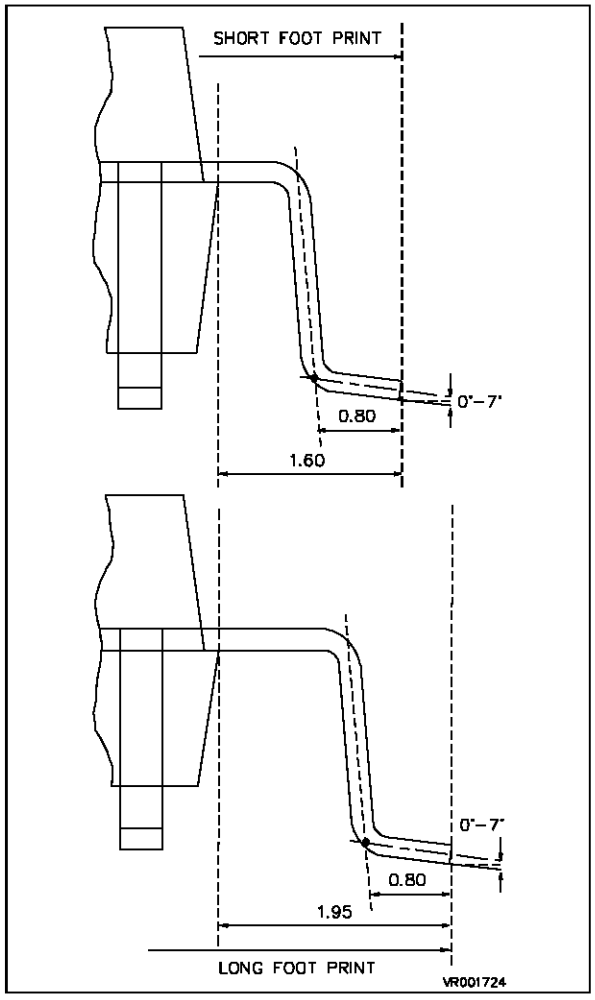
PACKAGE MECHANICAL DATA

Figure 3. ST62E85 80 Pin Ceramic Quad Flat Package

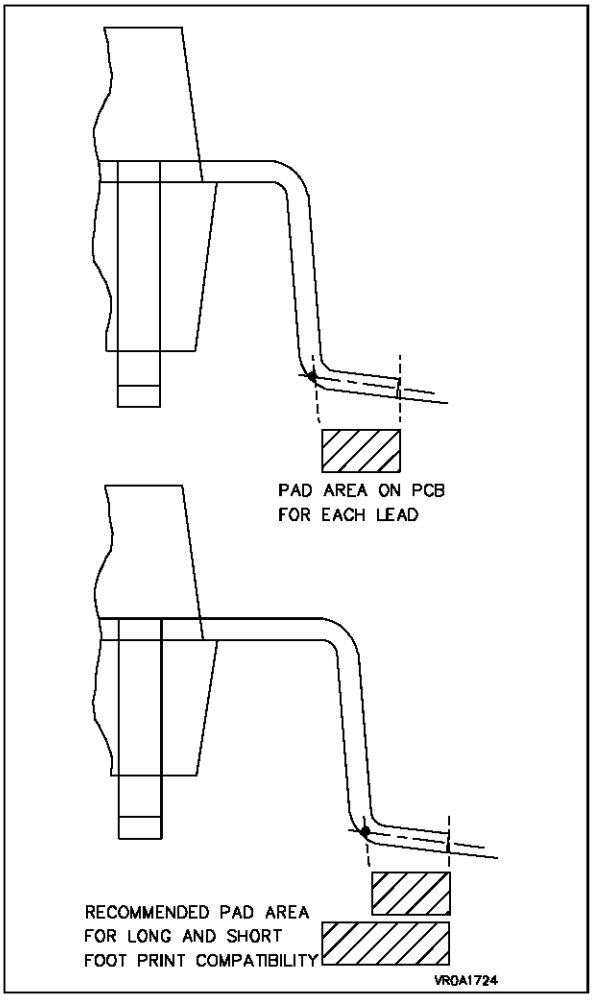


Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A		3.55			0.140	
A2		3.40			0.133	
D		23.90			0.941	
D1		20.00			0.787	
D3		18.40			0.724	
E		17.90			0.705	
E1		14.00			0.551	
E3		12.00			0.472	
Ø		7.62			0.3	
e		0.80			0.032	
Number of Pins						
ND	24					
NE	16					
N	80					

Short/Long Footprint Measurement



Short/Long Footprint recommended Padding



ORDERING INFORMATION

ORDERING INFORMATION TABLE

Sales Types	Memory Type	Temperature Range	Package
ST62E85G1	8K EPROM	Tested at 25°C only	CQFP80-W
ST62T85	Under development. Please contact your local SGS-THOMSON office.		