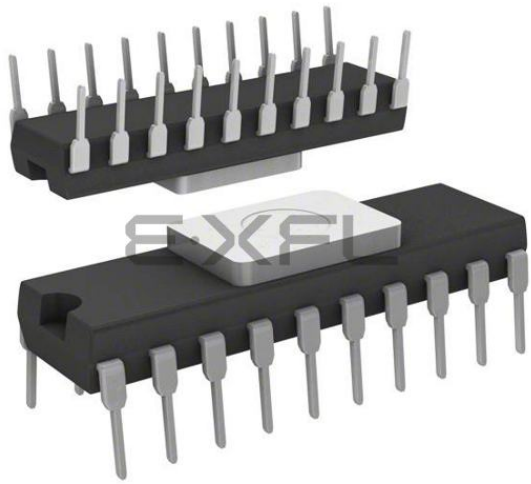


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ST6
Core Size	8-Bit
Speed	8MHz
Connectivity	SPI
Peripherals	LED, LVD, POR, WDT
Number of I/O	13
Program Memory Size	3.8KB (3.8K x 8)
Program Memory Type	EPROM, UV
EEPROM Size	128 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 6V
Data Converters	A/D 7x8b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	20-CDIP (0.300", 7.62mm) Window
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st62e60cf1

Table of Contents

1 GENERAL DESCRIPTION

1.1 INTRODUCTION

The ST62T53C, ST62T60C, ST62T63C and ST62E60C devices are low cost members of the ST62xx 8-bit HCMOS family of microcontrollers, which is targeted at low to medium complexity applications. All ST62xx devices are based on a building block approach: a common core is surrounded by a number of on-chip peripherals.

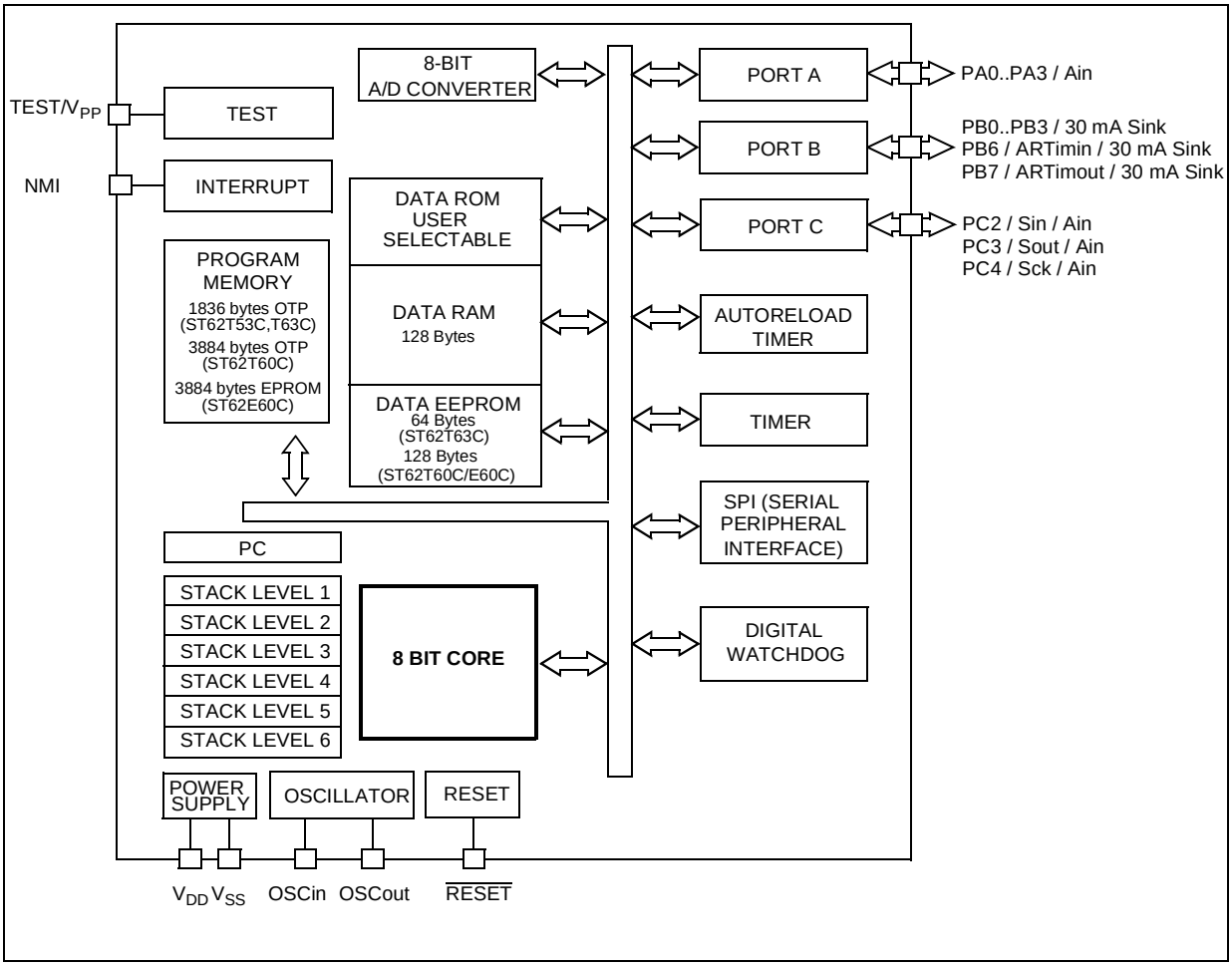
The ST62E60C is the erasable EPROM version of the ST62T60C device, which may be used to emulate the ST62T53C, ST62T60C and ST62T63C devices, as well as the respective ST6253C, ST6260B and ST6263B ROM devices.

OTP and EPROM devices are functionally identical. The ROM based versions offer the same functionality selecting as ROM options the options defined in the programmable option byte of the OTP/EPROM versions.

OTP devices offer all the advantages of user programmability at low cost, which make them the ideal choice in a wide range of applications where frequent code changes, multiple code versions or last minute programmability are required.

These compact low-cost devices feature a Timer comprising an 8-bit counter and a 7-bit programmable prescaler, an 8-bit Auto-Reload Timer, EEPROM data capability (except ST62T53C), a serial port communication interface, an 8-bit A/D Converter with 7 analog inputs and a Digital Watchdog timer, making them well suited for a wide range of automotive, appliance and industrial applications.

Figure 1. Block Diagram



MEMORY MAP (Cont'd)

1.3.3 Data Space

Data Space accommodates all the data necessary for processing the user program. This space comprises the RAM resource, the processor core and peripheral registers, as well as read-only data such as constants and look-up tables in OTP/EPROM.

1.3.3.1 Data ROM

All read-only data is physically stored in program memory, which also accommodates the Program Space. The program memory consequently contains the program code to be executed, as well as the constants and look-up tables required by the application.

The Data Space locations in which the different constants and look-up tables are addressed by the processor core may be thought of as a 64-byte window through which it is possible to access the read-only data stored in OTP/EPROM.

1.3.3.2 Data RAM/EEPROM

In ST62T53C, T60C, T63C and ST62E60C devices, the data space includes 60 bytes of RAM, the accumulator (A), the indirect registers (X), (Y), the short direct registers (V), (W), the I/O port registers, the peripheral data and control registers, the interrupt option register and the Data ROM Window register (DRW register).

Additional RAM and EEPROM pages can also be addressed using banks of 64 bytes located between addresses 00h and 3Fh.

1.3.4 Stack Space

Stack space consists of six 12-bit registers which are used to stack subroutine and interrupt return addresses, as well as the current program counter contents.

Table 1. Additional RAM/EEPROM Banks

Device	RAM	EEPROM
ST62T53C	1 x 64 bytes	-
ST62T60C/E60C	1 x 64 bytes	2 x 64 bytes
ST62T63C	1 x 64 bytes	1 x 64 bytes

Table 2ST62T53C, T60C, T63C and ST62E60C Data Memory Space

RAM and EEPROM	000h 03Fh
DATA ROM WINDOW AREA	040h 07Fh
X REGISTER	080h
Y REGISTER	081h
V REGISTER	082h
W REGISTER	083h
DATA RAM 60 BYTES	084h 0BFh
PORT A DATA REGISTER	0C0h
PORT B DATA REGISTER	0C1h
PORT C DATA REGISTER	0C2h
RESERVED	0C3h
PORT A DIRECTION REGISTER	0C4h
PORT B DIRECTION REGISTER	0C5h
PORT C DIRECTION REGISTER	0C6h
RESERVED	0C7h
INTERRUPT OPTION REGISTER	0C8h*
DATA ROM WINDOW REGISTER	0C9h*
RESERVED	0CAh 0CBh
PORT A OPTION REGISTER	0CCh
PORT B OPTION REGISTER	0CDh
PORT C OPTION REGISTER	0CEh
RESERVED	0CFh
A/D DATA REGISTER	0D0h
A/D CONTROL REGISTER	0D1h
TIMER PRESCALER REGISTER	0D2h
TIMER COUNTER REGISTER	0D3h
TIMER STATUS CONTROL REGISTER	0D4h
AR TIMER MODE CONTROL REGISTER	0D5h
AR TIMER STATUS/CONTROL REGISTER1	0D6h
AR TIMER STATUS/CONTROL REGISTER2	0D7h
WATCHDOG REGISTER	0D8h
AR TIMER RELOAD/CAPTURE REGISTER	0D9h
AR TIMER COMPARE REGISTER	0DAh
AR TIMER LOAD REGISTER	0DBh
OSCILLATOR CONTROL REGISTER	0DCh*
MISCELLANEOUS	0DDh
RESERVED	0DEh 0DFh
SPI DATA REGISTER	0E0h
SPI DIVIDER REGISTER	0E1h
SPI MODE REGISTER	0E2h
RESERVED	0E3h 0E7h
DATA RAM/EEPROM REGISTER	0E8h*
RESERVED	0E9h
EEPROM CONTROL REGISTER (except ST62T53C)	0EAh
RESERVED	0EBh 0FEh
ACCUMULATOR	0FFh

* WRITE ONLY REGISTER

MEMORY MAP (Cont'd)

1.3.5 Data Window Register (DWR)

The Data read-only memory window is located from address 0040h to address 007Fh in Data space. It allows direct reading of 64 consecutive bytes located anywhere in program memory, between address 0000h and 0FFFh (top memory address depends on the specific device). All the program memory can therefore be used to store either instructions or read-only data. Indeed, the window can be moved in steps of 64 bytes along the program memory by writing the appropriate code in the Data Window Register (DWR).

The DWR can be addressed like any RAM location in the Data Space, it is however a write-only register and therefore cannot be accessed using single-bit operations. This register is used to position the 64-byte read-only data window (from address 40h to address 7Fh of the Data space) in program memory in 64-byte steps. The effective address of the byte to be read as data in program memory is obtained by concatenating 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), as illustrated in Figure 6 below. For instance, when addressing location 0040h of the Data Space, with 0 loaded in the DWR register, the physical location addressed in program memory is 00h. The DWR register is not cleared on reset, therefore it must be written to prior to the first access to the Data read-only memory window area.

Data Window Register (DWR)

Address: 0C9h — Write Only

7							0
-	-	DWR5	DWR4	DWR3	DWR2	DWR1	DWR0

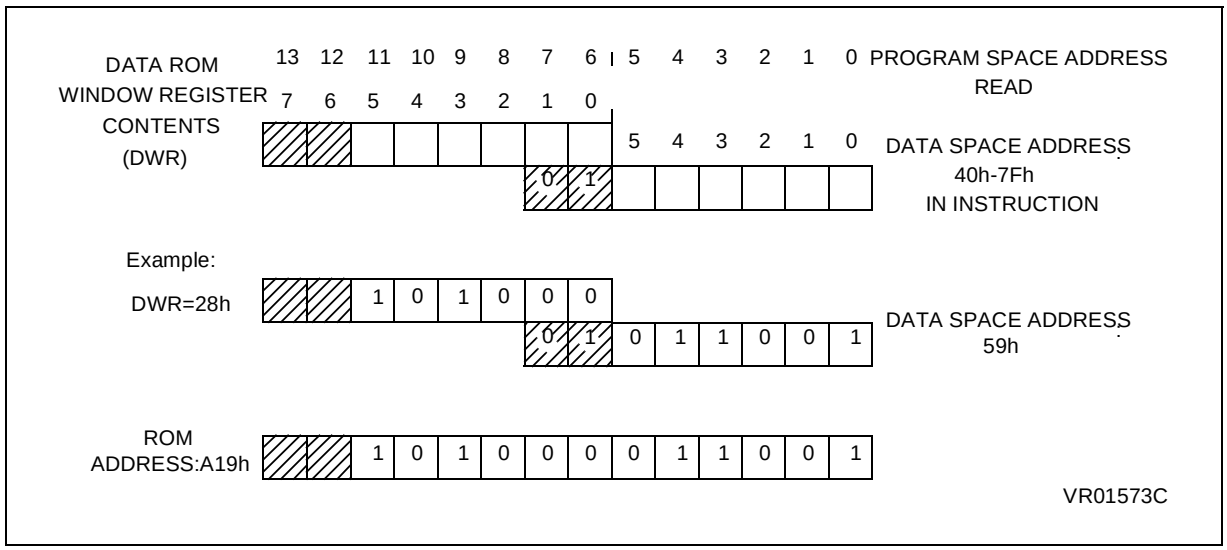
Bits 6, 7 = Not used.

Bit 5-0 = **DWR6-DWR0:** *Data read-only memory Window Register Bits.* These are the Data read-only memory Window bits that correspond to the upper bits of the data read-only memory space.

Caution: *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, the DWR contents should not be changed while executing an interrupt service routine, as the service routine cannot save and then restore the register's previous contents. If it is impossible to avoid writing to the DWR during the interrupt service routine, an image of the register must be saved in a RAM location, and each time the program writes to the DWR, it must also write to the image register. The image register must be written first so that, if an interrupt occurs between the two instructions, the DWR is not affected.

Figure 6Data read-only memory Window Memory Addressing



MEMORY MAP (Cont'd)

1.3.6 Data RAM/EEPROM Bank Register (DRBR)

Address: E8h — Write only

7							0
-	-	-	DRBR 4	-	-	DRBR 1	DRBR 0

Bit 7-5 = These bits are not used

Bit 4 - **DRBR4**. This bit, when set, selects RAM Page 2.

Bit 3-2 - Reserved. These bits are not used.

Bit 1 - **DRBR1**. This bit, when set, selects EEPROM Page 1, when available.

Bit 0 - **DRBR0**. This bit, when set, selects EEPROM Page 0, when available.

The selection of the bank is made by programming the Data RAM Bank Switch register (DRBR register) located at address E8h of the Data Space according to Table 1. No more than one bank should be set at a time.

The DRBR register can be addressed like a RAM Data Space at the address E8h; 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 bank of the Data Space. The bank number 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 an interrupt or a subroutine occurs.

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.

Care must also be taken not to change the E²PROM page (when available) when the parallel writing mode is set for the E²PROM, as defined in EECTL register.

Table 3Data RAM Bank Register Set-up

DRBR	ST62T53C	ST62T60C/E60C	ST62T63C
00	None	None	None
01	Not Available	EEPROM Page 0	EEPROM Page 0
02	Not Available	EEPROM Page 1	Not Available
08	Not Available	Not Available	Not Available
10h	RAM Page 2	RAM Page 2	RAM Page 2
other	Reserved	Reserved	Reserved

DIGITAL WATCHDOG (Cont'd)

These instructions test the C bit and Reset the MCU (i.e. disable the Watchdog) if the bit is set (i.e. if the Watchdog is active), thus disabling the Watchdog.

In all modes, a minimum of 28 instructions are executed after activation, before the Watchdog can generate a Reset. Consequently, user software should load the watchdog counter within the first 27 instructions following Watchdog activation (software mode), or within the first 27 instructions executed following a Reset (hardware activation).

It should be noted that when the GEN bit is low (interrupts disabled), the NMI interrupt is active but cannot cause a wake up from STOP/WAIT modes.

Figure 19. A typical circuit making use of the EXERNAL STOP MODE CONTROL feature

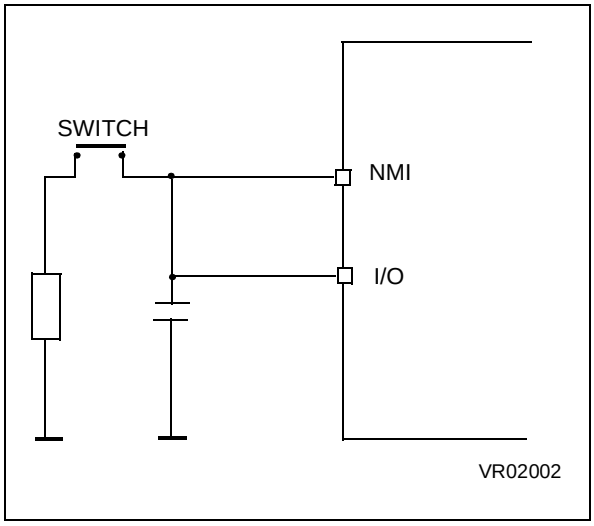
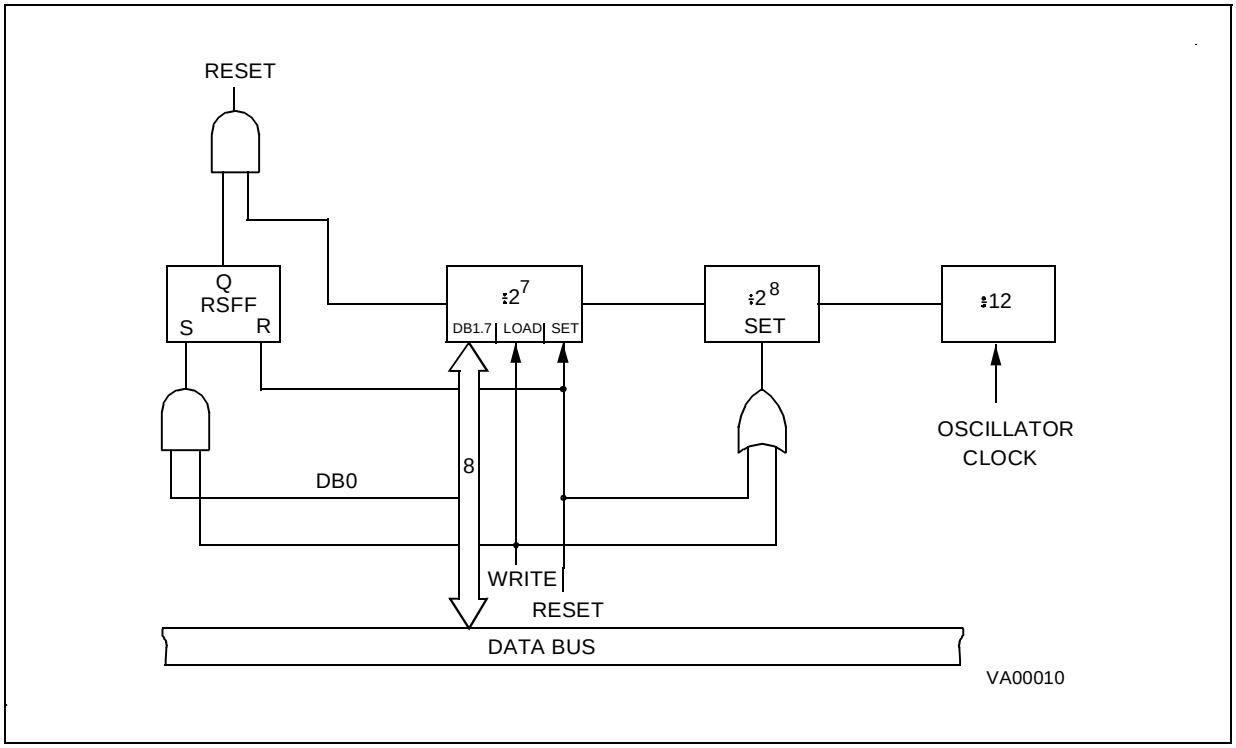


Figure 20. Digital Watchdog Block Diagram



INTERRUPTS (Cont'd)

3.4.2 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 cannot know the context and the time at which it occurred. As a result, the user should save all Data space registers which may be used within the interrupt routines. There are separate sets of processor flags for normal, interrupt and non-maskable interrupt modes, which are automatically switched and so do not need to be saved.

The following list summarizes the interrupt procedure:

MCU

– The interrupt is detected.
– The C and Z flags are replaced by the interrupt flags (or by the NMI flags).
– The PC contents are stored in the first level of the stack.
– The normal interrupt lines are inhibited (NMI still active).
– The first internal latch is cleared.
– The associated interrupt vector is loaded in the PC.

WARNING: In some circumstances, when a maskable interrupt occurs while the ST6 core is in NORMAL mode and especially during the execution of an "ldi IOR, 00h" instruction (disabling all maskable interrupts): if the interrupt arrives during the first 3 cycles of the "ldi" instruction (which is a 4-cycle instruction) the core will switch to interrupt mode BUT the flags CN and ZN will NOT switch to the interrupt pair CI and ZI.

User

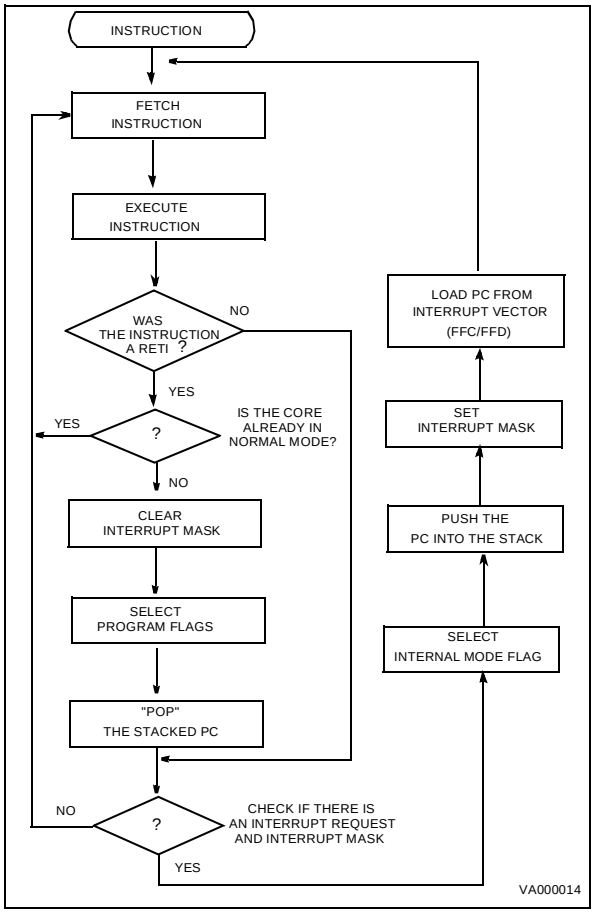
– User selected registers are saved within the interrupt service routine (normally on a software stack).
– The source of the interrupt is found by polling the interrupt flags (if more than one source is associated with the same vector).
– The interrupt is serviced.
– Return from interrupt (RETI)

MCU

– Automatically the MCU switches back to the normal flag set (or the interrupt flag set) and pops the previous PC value from the stack.

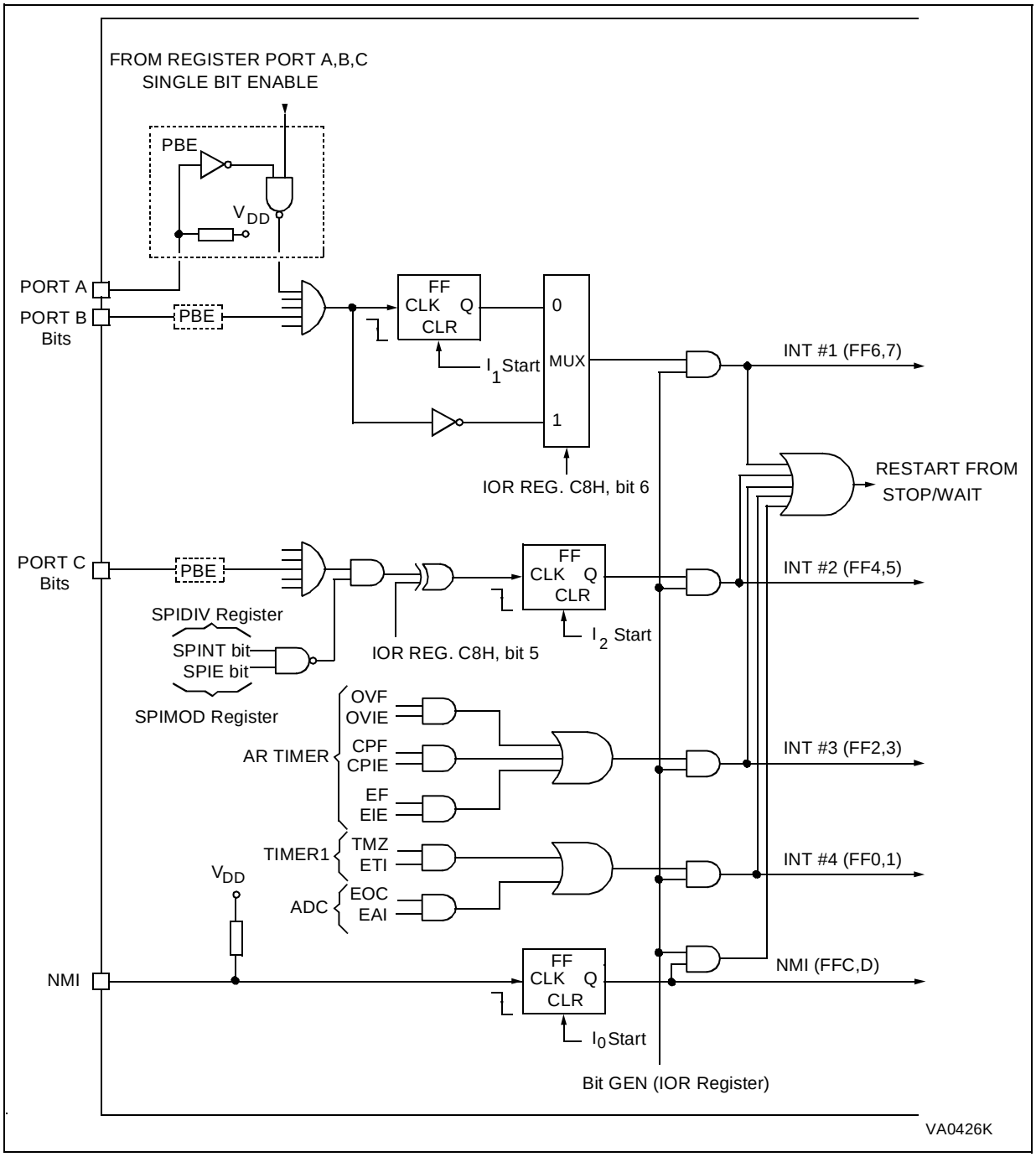
The interrupt routine usually begins by the identifying the device which generated the interrupt request (by polling). The user should save the registers which are used within the interrupt routine in a software stack. After the RETI instruction is executed, the MCU returns to the main routine.

Figure 21. Interrupt Processing Flow Chart



INTERRUPTS (Cont'd)

Figure 22. Interrupt Block Diagram



I/O PORTS (Cont'd)

4.1.2 Safe I/O State Switching Sequence

Switching the I/O ports from one state to another should be done in a sequence which ensures that no unwanted side effects can occur. The recommended safe transitions are illustrated in Figure 24. All other transitions are potentially risky and should be avoided when changing the I/O operating mode, as it is most likely that undesirable side-effects will be experienced, such as spurious interrupt generation or two pins shorted together by the analog multiplexer.

Single bit instructions (SET, RES, INC and DEC) should be used with great caution on Ports Data registers, since these instructions make an implicit read and write back of the entire register. In port input mode, however, the data register reads from the input pins directly, and not from the data register latches. Since data register information in input mode is used to set the characteristics of the input pin (interrupt, pull-up, analog input), these may be unintentionally reprogrammed depending on the state of the input pins. As a general rule, it is better to limit the use of single bit instructions on data registers to when the whole (8-bit) port is in output mode. In the case of inputs or of mixed inputs and outputs, it is advisable to keep a copy of the data register in RAM. Single bit instructions may then be used on the RAM copy, after which the whole copy register can be written to the port data register:

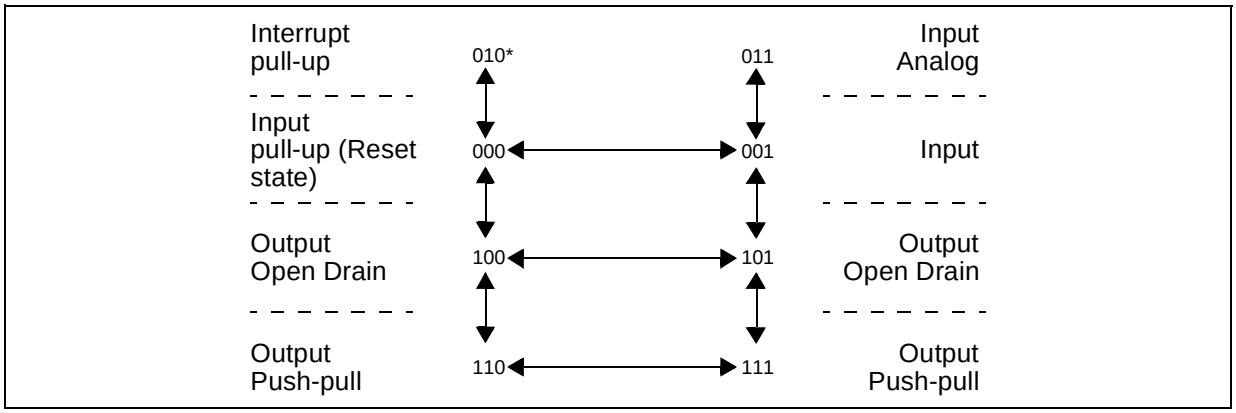
```
SET bit, datacopy
LD a, datacopy
LD DRA, a
```

Warning: Care must also be taken to not use instructions that act on a whole port register (INC, DEC, or read operations) when all 8 bits are not available on the device. Unavailable bits must be masked by software (AND instruction).

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/Os in input mode with well-defined logic levels.

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

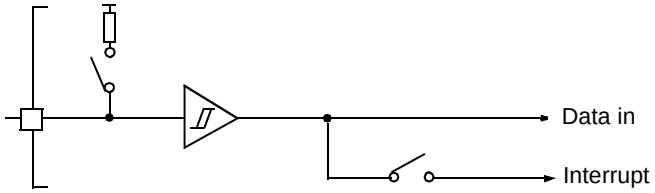
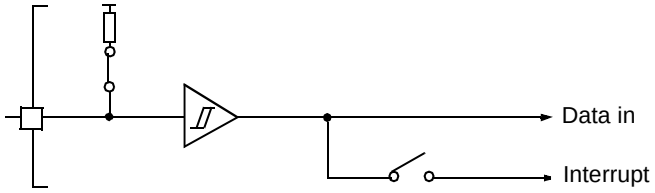
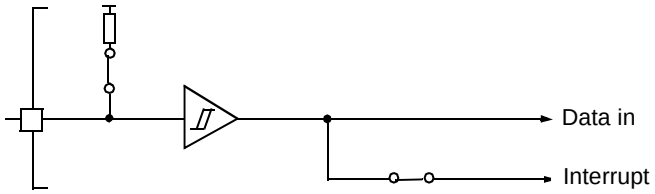
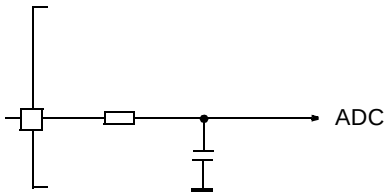
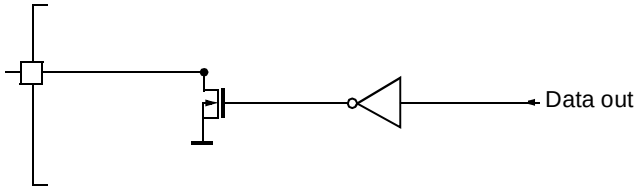
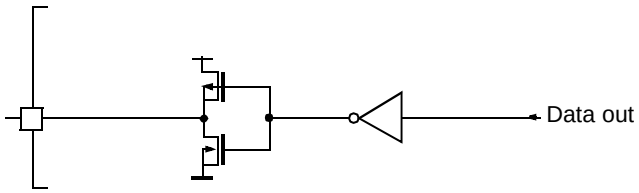
Figure 24. Diagram showing Safe I/O State Transitions



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

I/O PORTS (Cont'd)

Table 11I/O Port Option Selections

MODE	AVAILABLE ON(1)	SCHEMATIC
Input	PA0-PA3 PB0-PB3, PB6-PB7 PC2-PC4	
Input with pull up	PA0-PA3 PB0-PB3, PB6-PB7 PC2-PC4	
Input with pull up with interrupt	PA0-PA3 PB0-PB3, PB6-PB7 PC2-PC4	
Analog Input	PA0-PA3 PC2-PC4	
Open drain output 5mA Open drain output 30mA	PA0-PA3 PC2-PC4 PB0-PB3, PB6-PB7	
Push-pull output 5mA Push-pull output 30mA	PA0-PA3 PC2-PC4 PB0-PB3, PB6-PB7	

Note 1. Provided the correct configuration has been selected.

TIMER (Cont'd)

4.2.1 Timer Operation

The Timer prescaler is clocked by the prescaler clock input ($f_{INT} \div 12$).

The user can select the desired prescaler division ratio through the PS2, PS1, PS0 bits. When the TCR count reaches 0, it sets the TMZ bit in the TSCR. The TMZ bit can be tested under program control to perform a timer function whenever it goes high.

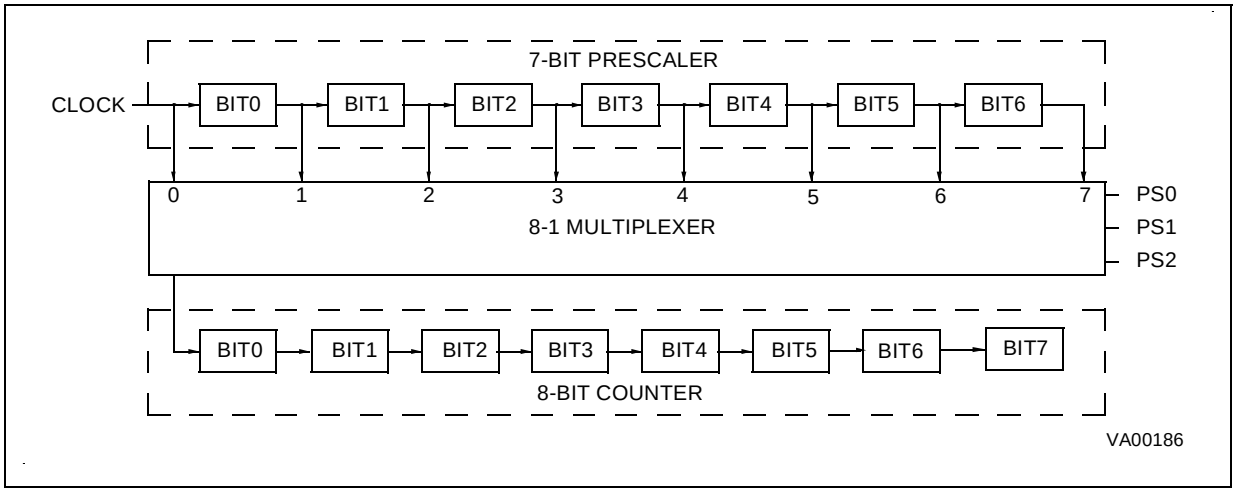
4.2.2 Timer Interrupt

When the counter register decrements to zero with the ETI (Enable Timer Interrupt) bit set to one, an interrupt request associated with Interrupt Vector #4 is generated. When the counter decrements to zero, the TMZ bit in the TSCR register is set to one.

4.2.3 Application Notes

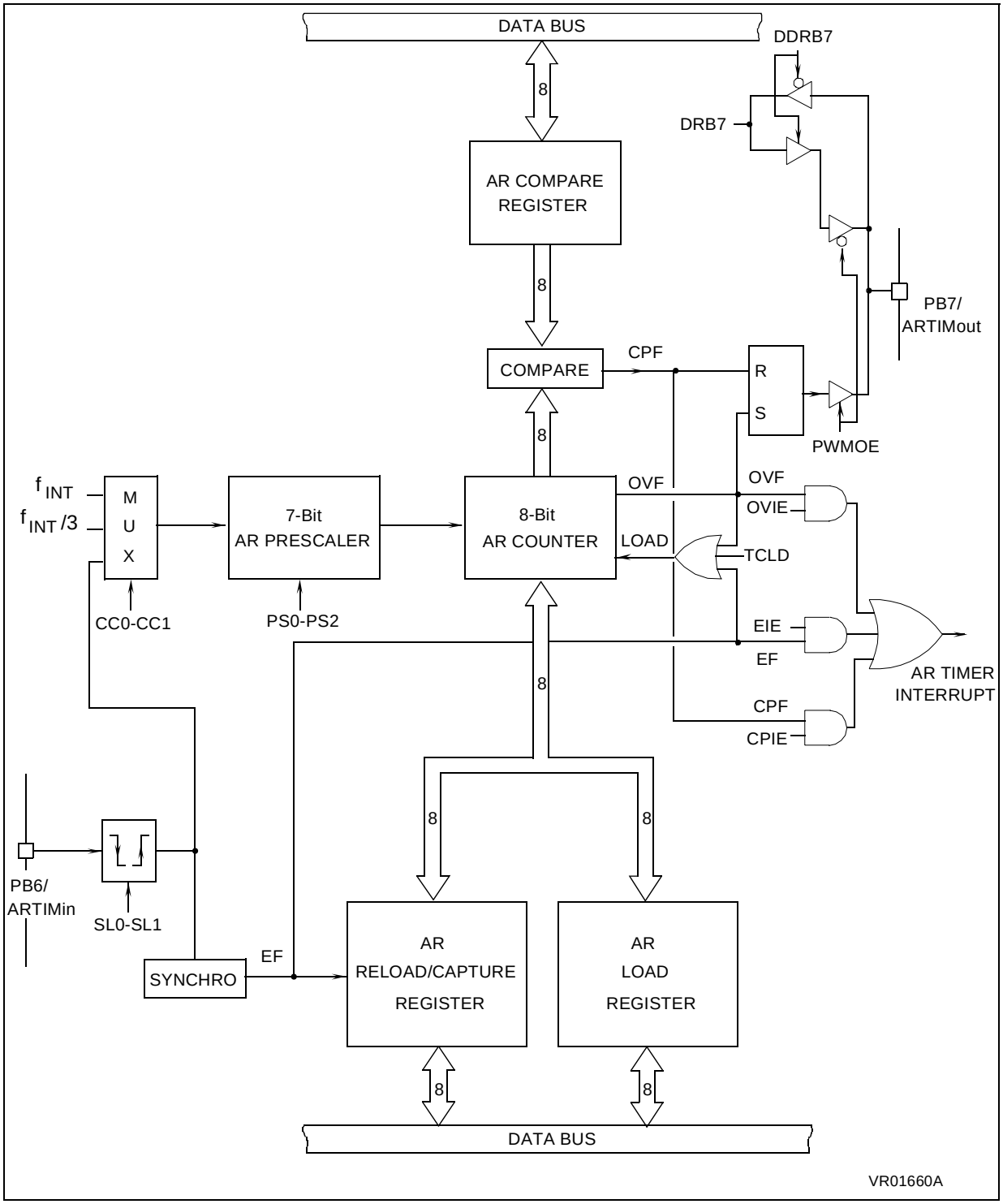
TMZ is set when the counter reaches zero; however, it may also be set by writing 00h in the TCR register or by setting bit 7 of the TSCR register. The TMZ bit must be cleared by user software when servicing the timer interrupt to avoid undesired interrupts when leaving the interrupt service routine. After reset, the 8-bit counter register is loaded with 0FFh, while the 7-bit prescaler is loaded with 07Fh, and the TSCR register is cleared. This means that the Timer is stopped (PSI="0") and the timer interrupt is disabled.

Figure 27. Timer Working Principle



AUTO-RELOAD TIMER (Cont'd)

Figure 28. AR Timer Block Diagram



AUTO-RELOAD TIMER (Cont'd)

Capture Mode with PWM Generation. In this mode, the AR counter operates as a free running 8-bit counter fed by the prescaler output. The counter is incremented on every clock rising edge.

An 8-bit capture operation from the counter to the ARRC register is performed on every active edge on the ARTIMin pin, when enabled by Edge Control bits SL0, SL1 in the ARSC1 register. At the same time, the External Flag, EF, in the ARSC0 register is set and an external interrupt request is generated if the External Interrupt Enable bit, EIE, in the ARMC register, is set. The EF flag must be reset by user software.

Each ARTC overflow sets ARTIMout, while a match between the counter and ARCP (Compare Register) resets ARTIMout and sets the compare flag, CPF. A compare interrupt request is generated if the related compare interrupt enable bit, CPIE, is set. A PWM signal is generated on ARTIMout. The CPF flag must be reset by user software.

The frequency of the generated signal is determined by the prescaler setting. The duty cycle is determined by the ARCP register.

Initialization and reading of the counter are identical to the auto-reload mode (see previous description).

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

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

In Capture mode, the allowed clock sources are the internal clock and the internal clock divided by 3; the external ARTIMin input pin should not be used as a clock source.

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

Load on External Input. The counter operates as a free running 8-bit counter fed by the prescaler. the count is incremented on every clock rising edge.

Each counter overflow sets the ARTIMout pin. A match between the counter and ARCP (Compare Register) resets the ARTIMout pin and sets the compare flag, CPF. A compare interrupt request is generated if the related compare interrupt enable bit, CPIE, is set. A PWM signal is generated on ARTIMout. The CPF flag must be reset by user software.

Initialization of the counter is 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 counter, whether the counter is running or not.

Notes:

The allowed AR Timer clock sources are the following:

AR Timer Mode	Clock Sources
Auto-reload mode	f_{INT}, $f_{INT/3}$, ARTIMin
Capture mode	f_{INT}, $f_{INT/3}$
Capture/Reset mode	f_{INT}, $f_{INT/3}$
External Load mode	f_{INT}, $f_{INT/3}$

The clock frequency should not be modified while the counter is counting, since the counter may be set to an unpredictable value. For instance, the multiplexer setting should not be modified while the counter is counting.

Loading of the counter by any means (by auto-reload, through ARLR, ARRC or by the Core) resets the prescaler at the same time.

Care should be taken when both the Capture interrupt and the Overflow interrupt are used. Capture and overflow are asynchronous. If the capture occurs when the Overflow Interrupt Flag, OVF, is high (between counter overflow and the flag being reset by software, in the interrupt routine), the External Interrupt Flag, EF, may be cleared simultaneusly without the interrupt being taken into account.

The solution consists in resetting the OVF flag by writing 06h in the ARSC0 register. The value of EF is not affected by this operation. If an interrupt has occured, it will be processed when the MCU exits from the interrupt routine (the second interrupt is latched).

4.5 SERIAL PERIPHERAL INTERFACE (SPI)

The SPI peripheral is an optimized synchronous serial interface with programmable transmission modes and master/slave capabilities supporting a wide range of industry standard SPI specifications. The SPI interface may also implement asynchronous data transfer, in which case processor overhead is limited to data transfer from or to the shift register on an interrupt driven basis.

The SPI may be controlled by simple user software to perform serial data exchange with low-cost external memory, or with serially controlled peripherals to drive displays, motors or relays.

The SPI's shift register is simultaneously fed by the Sin pin and feeds the Sout pin, thus transmission and reception are essentially the same process. Suitable setting of the number of bits in the data frame can allow filtering of unwanted leading data bits in the incoming data stream.

The SPI comprises an 8-bit Data/Shift Register, DSR, a Divide register, DIV, a Mode Control Register MOD, and a Miscellaneous register, MISCR.

The SPI may be operated either in Master mode or in Slave mode.

Master mode is defined by the synchronous serial clock being supplied by the MCU, by suitably programming the clock divider (DIV register). Slave mode is defined by the serial clock being supplied externally on the SCK pin by the external Master device.

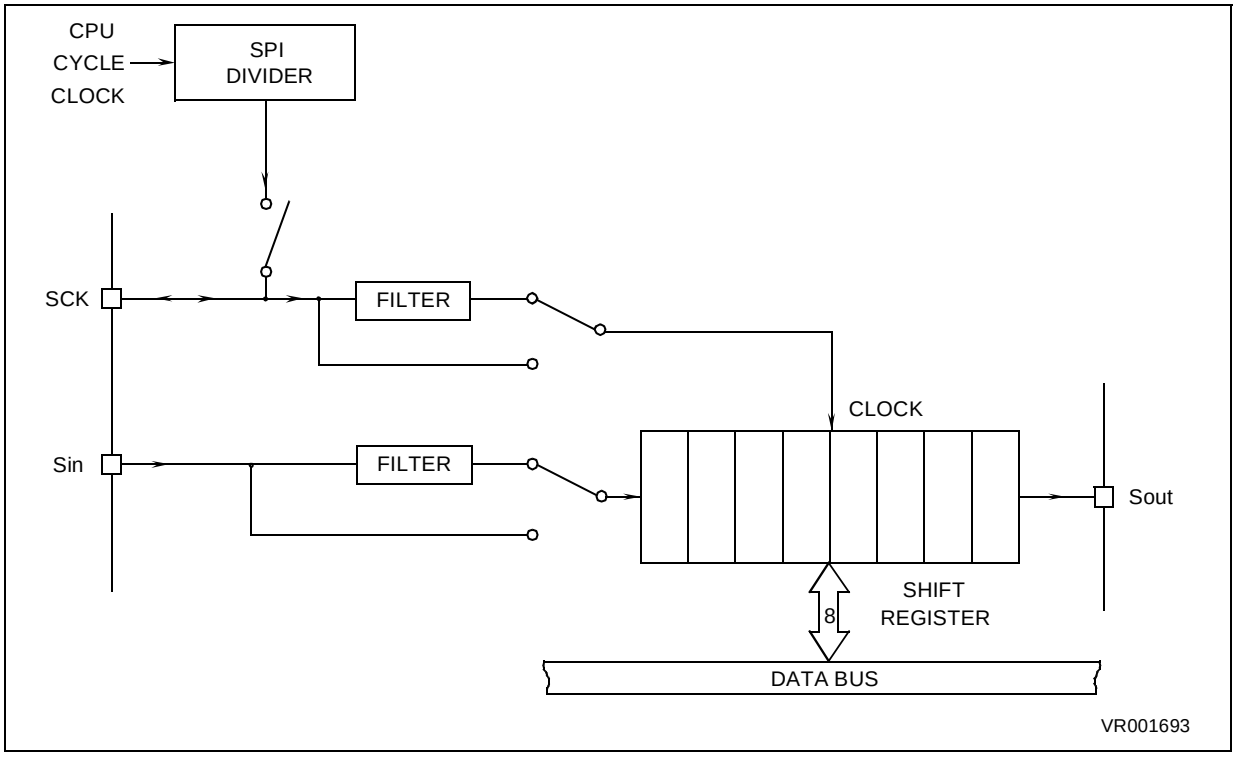
For maximum versatility the SPI may be programmed to sample data either on the rising or on the falling edge of SCK, with or without phase shift (clock Polarity and Phase selection).

The Sin, Sout and SCK signals are connected as alternate I/O pin functions.

For serial input operation, Sin must be configured as an input. For serial output operation, Sout is selected as an output by programming Bit 0 of the Miscellaneous Register: clearing this bit will set the pin as a standard I/O line, while setting the bit will select the Sout function.

An interrupt request may be associated with the end of a transmission or reception cycle; this is defined by programming the number of bits in the data frame and by enabling the interrupt. This request is associated with interrupt vector #2, and can be masked by programming the SPIE bit of the MOD register. Since the SPI interrupt is "ORed" with the port interrupt source, an interrupt flag bit is available in the DIV register allowing discrimination of the interrupt request.

Figure 31. SPI Block Diagram



SERIAL PERIPHERAL INTERFACE SPI (Cont'd)

SPI DIV Register (DIV)

Address: E1h — Read/Write

Reset status: 00h

7							0
SPINT	DOV6	DIV5	DIV4	DIV3	CD2	CD1	CD0

The SPIDIV register defines the transmission rate and frame format and contains the interrupt flag.

Bits CD0-CD2, DIV3-DIV6 are read/write while SPINT can be read and cleared only. Write access is not allowed if SPRUN in the MOD register is set.

Bit 7 = **SPINT**: *Interrupt Flag.* If SPIE bit=1, SPINT is automatically set to one by the SPI at the end of a transmission or reception and an interrupt request can be generated depending on the state of the interrupt mask bit in the MOD control register. This bit is write and read and must be cleared by user software at the end of the interrupt service routine.

Bit 6-3 = **DIV6-DIV3**: *Burst Mode Bit Clock Period Selection.* Define the number of shift register bits that are transmitted or received in a frame. The available selections are listed in Table 16. The normal maximum setting is 8 bits, since the shift register is 8 bits wide. Note that by setting a greater number of bits, in conjunction with the SPIN bit in the MOD register, unwanted data bits may be filtered from the data stream.

Bit 2-0 = **CD2-CD0**: *Base/Bit Clock Rate Selection*. Define the division ratio between the core clock (f_{INT} divided by 13) and the clock supplied to the Shift Register in Master mode.

Table 15. Base/Bit Clock Ratio Selection

CD2-CD0			Divide Ratio (decimal)
0	0	0	Divide by 1
0	0	1	Divide by 2
0	1	0	Divide by 4
0	1	1	Divide by 8
1	0	0	Divide by 16
1	0	1	Divide by 32
1	1	0	Divide by 64
1	1	1	Divide by 256

Note: For example, when an 8MHz CPU clock is used, asynchronous operation at 9600 Baud is possible (8MHz/13/64). Other Baud rates are available by proportionally selecting division factors depending on CPU clock frequency.

Data setup time on Sin is typically 250ns min, while data hold time is typically 50ns min.

DIV6-DIV3				Number of bits sent	
0	0	0	0	Reserved (not to be used)	
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	
1	0	0	1	9	)
1	0	1	0	10	|
1	0	1	1	11	| Refer to the
1	1	0	0	12	| description of the
1	1	0	1	13	| DIV6-DIV3 bits in
1	1	1	0	14	| the DIV Register
1	1	1	1	15	)

SPI Data/Shift Register (SPIDSR)

Address: E0h — Read/Write

Reset status: XXh

7							0
D7	D6	D5	D4	D3	D2	D1	D0

SPIDSR is read/write, however write access is not allowed if the SPRUN bit of Mode Control register is set to one.

Data is sampled into SPDSR on the SCK edge determined by the CPOL and CPHA bits. The affect of these setting is shown in the following diagrams.

The Shift Register transmits and receives the Most Significant Bit first.

Bit 7-0 = **DSR7-DSR0**: *Data Bits.* These are the SPI shift register data bits.

Miscellaneous Register (MISCR)

Address: DDh — Write only

Reset status: xxxxxxxb

7							0
-	-	-	-	-	-	-	D0

Bit 7-1 = **D7-D1**: *Reserved.*

Bit 0 = **D0**: *Bit 0.* This bit, when set, selects the Sout pin as the SPI output line. When this bit is cleared, Sout acts as a standard I/O line.

Opcode Map Summary (Continued)

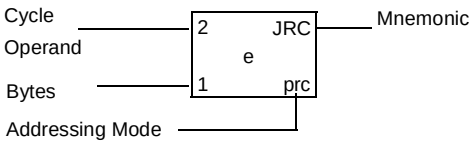
LOW / HI	**8 1000**	**9 1001**	**A 1010**	**B 1011**	**C 1100**	**D 1101**	**E 1110**	**F 1111**	LOW / HI
0 0000	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 RES b0,rr 2 b.d	2 JRZ e 1 pcr	4 LDI rr,nn 3 imm	2 JRC e 1 prc	4 LD a,(y) 1 ind	**0 0000**
1 0001	2 JRNZ e 1 pcr	4 JP abc 2 ext	2 JRNC e 1 pcr	4 SET b0,rr 2 b.d	2 JRZ e 1 pcr	4 DEC x 1 sd	2 JRC e 1 prc	4 LD a,rr 2 dir	**1 0001**
2 0010	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	2 JRC e 1 prc	4 CP a,(y) 1 ind	**2 0010**
3 0011	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 prc	4 CP a,rr 2 dir	**3 0011**
4 0100	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 prc	4 ADD a,(y) 1 ind	**4 0100**
5 0101	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 prc	4 ADD a,rr 2 dir	**5 0101**
6 0110	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 prc	4 INC (y) 1 ind	**6 0110**
7 0111	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 prc	4 INC rr 2 dir	**7 0111**
8 1000	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 prc	4 LD (y),a 1 ind	**8 1000**
9 1001	2 RNZ 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 prc	4 LD rr,a 2 dir	**9 1001**
A 1010	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 RCL a 1 inh	2 JRC e 1 prc	4 AND a,(y) 1 ind	**A 1010**
B 1011	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 prc	4 AND a,rr 2 dir	**B 1011**
C 1100	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 prc	4 SUB a,(y) 1 ind	**C 1100**
D 1101	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 prc	4 SUB a,rr 2 dir	**D 1101**
E 1110	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 prc	4 DEC (y) 1 ind	**E 1110**
F 1111	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 prc	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



6 ELECTRICAL CHARACTERISTICS

6.1 ABSOLUTE MAXIMUM RATINGS

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

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

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

Tj=TA + PD x RthJA

Where:TA = Ambient Temperature.

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

PD = Pint + Pport.

Pint =IDD x VDD (chip internal power).

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

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.3 to 7.0	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
IV_{DD}	Total Current into V_{DD} (source)	80	mA
IV_{SS}	Total Current out of V_{SS} (sink)	100	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 guarantee to be not conductive. Voltages outside these limits are authorised as long as injection current is kept within the specification.

DC ELECTRICAL CHARACTERISTICS (Cont'd)

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

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
V_{up}	LVD Threshold in power-on		V_{dn} +50 mV	4.1	4.3	V
V_{dn}	LVD threshold in powerdown		3.6	3.8	V_{up} -50 mV	V
V_{OL}	Low Level Output Voltage All Output pins	V_{DD}= 5.0V; I_{OL} = +10µA V_{DD}= 5.0V; I_{OL} = + 5mA V_{DD}= 5.0V; I_{OL} = + 10mAv			0.1 0.8 1.2	V
	Low Level Output Voltage 30 mA Sink I/O pins	V_{DD}= 5.0V; I_{OL} = +10µA V_{DD}= 5.0V; I_{OL} = +10mA V_{DD}= 5.0V; I_{OL} = +20mA V_{DD}= 5.0V; I_{OL} = +30mA			0.1 0.8 1.3 2.0	
V_{OH}	High Level Output Voltage All Output pins	V_{DD}= 5.0V; I_{OH} = -10µA V_{DD}= 5.0V; I_{OH} = -5.0mA	4.9 3.5			V
I_{DD}	Supply Current in STOP Mode, with LVD disabled(*)	I_{LOAD}=0mA V_{DD}=5.0V			10	µA

Note:
(*) All Peripherals in stand-by.

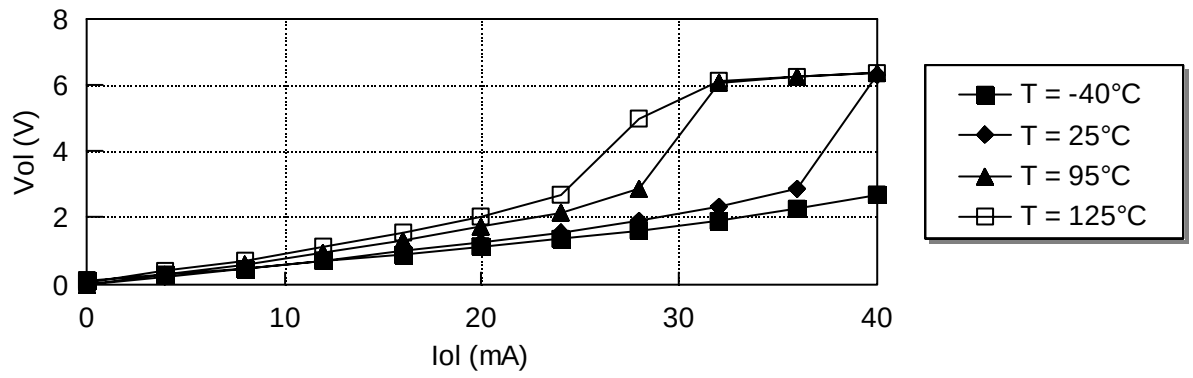
6.4 AC ELECTRICAL CHARACTERISTICS

(T_A = -40 to +125°C unless otherwise specified)

Symbol	Parameter	Test Conditions	Value			Unit
			Min.	Typ.	Max.	
t_{REC}	Supply Recovery Time (1)		100			ms
T_{WEE}	EEPROM Write Time	T_A = 25°C T_A = 85°C T_A = 125°C		5 10 20	10 20 30	ms
Endurance (2)	EEPROM WRITE/ERASE Cycle	Q_A L_{OT} Acceptance (25°C)	300,000	1 million		cycles
Retention	EEPROM Data Retention	T_A = 55°C	10			years
f_{LFAO}	Internal frequency with LFAO active		200	400	800	kHz
f_{OSG}	Internal Frequency with OSG enabled2)	V_{DD} = 3V V_{DD} = 3.6V V_{DD} = 4.5V V_{DD} = 6V	1 1 2 2		f_{OSC}	MHz
f_{RC}	Internal frequency with RC oscillator and OSG disabled2) 3)	VDD=5.0V (Except 626xB ROM) R=47kΩ R=100kΩ R=470kΩ	 4 2.7 800	 5 3.2 850	 5.8 3.5 900	 MHz MHz kHz
		VDD=5.0V (626xB ROM) R=10kΩ R=27kΩ R=67kΩ R=100kΩ	 6.3 4.7 2.8 2.2	 8.2 5.9 3.6 2.8	 9.8 7 4.3 3.4	 MHz MHz MHz MHz
C_{IN}	Input Capacitance	All Inputs Pins			10	pF
C_{OUT}	Output Capacitance	All Outputs Pins			10	pF

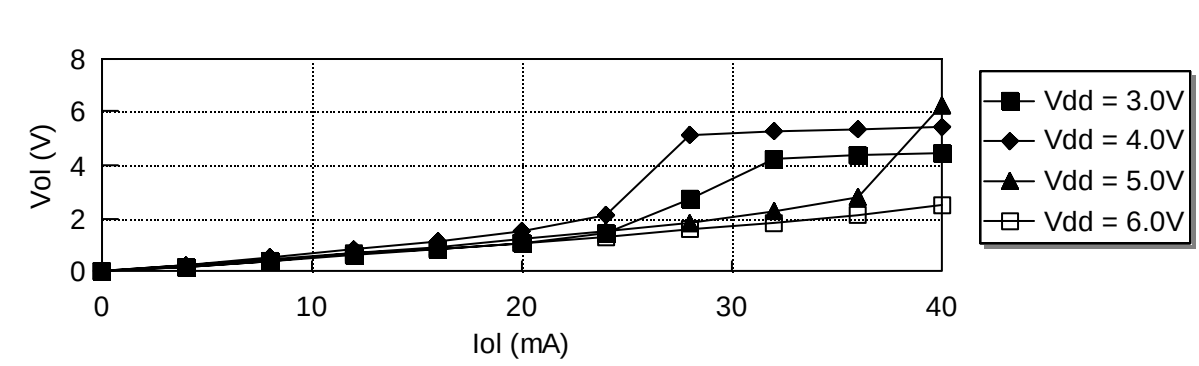
Notes:
1. Period for which V_{DD} has to be connected at 0V to allow internal Reset function at next power-up.
2 An oscillator frequency above 1MHz is recommended for reliable A/D results.
3. Measure performed with OSCin pin soldered on PCB, with an around 2pF equivalent capacitance.

Figure 37. Vol versus Iol on all I/O port at Vdd=5V



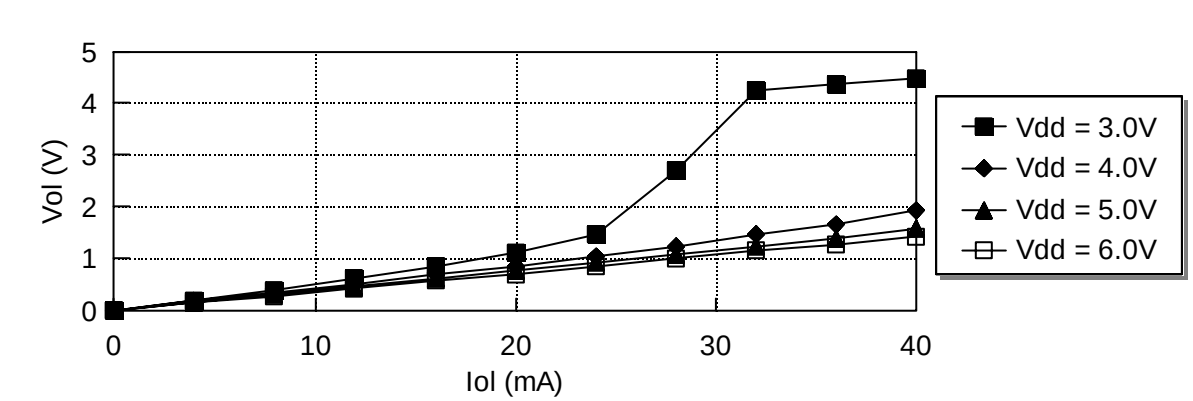
This curves represents typical variations and is given for guidance only

Figure 38. Vol versus Iol on all I/O port at T=25°C



This curves represents typical variations and is given for guidance only

Figure 39. Vol versus Iol for High sink (30mA) I/Oports at T=25°C



This curves represents typical variations and is given for guidance only

Notes: