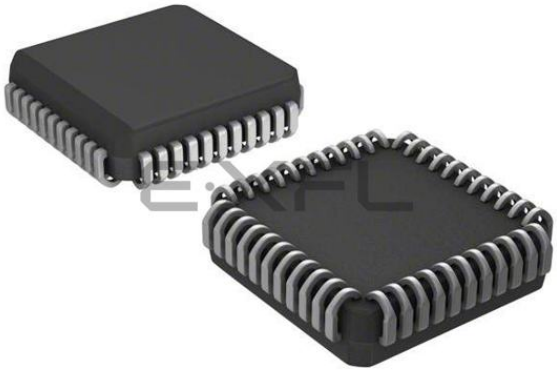


Welcome to E-XFL.COM

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	80C51
Core Size	8-Bit
Speed	60MHz
Connectivity	SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	34
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.6x16.6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at89c51rd2-slsim

The AT89C51RD2/ED2 retains all of the features of the Atmel 80C52 with 256 bytes of internal RAM, a 9-source 4-level interrupt controller and three timer/counters. The AT89C51ED2 provides 2048 bytes of EEPROM for nonvolatile data storage.

In addition, the AT89C51RD2/ED2 has a Programmable Counter Array, an XRAM of 1792 bytes, a Hardware Watchdog Timer, SPI interface, Keyboard, a more versatile serial channel that facilitates multiprocessor communication (EUART) and a speed improvement mechanism (X2 Mode).

The fully static design of the AT89C51RD2/ED2 allows to reduce system power consumption by bringing the clock frequency down to any value, including DC, without loss of data.

The AT89C51RD2/ED2 has 2 software-selectable modes of reduced activity and an 8-bit clock prescaler for further reduction in power consumption. In the Idle mode the CPU is frozen while the peripherals and the interrupt system are still operating. In the Power-down mode the RAM is saved and all other functions are inoperative.

The added features of the AT89C51RD2/ED2 make it more powerful for applications that need pulse width modulation, high speed I/O and counting capabilities such as alarms, motor control, corded phones, and smart card readers.

Table 1. Memory Size and I/O Pins

Package	Flash (Bytes)	XRAM (Bytes)	Total RAM (Bytes)	I/O
PLCC44/VQFP44/DIL40	64K	1792	2048	34
PLCC68/VQFP64	64K	1792	2048	50

Table 7. PCA SFRs (Continued)

Mnemo-nic	Add	Name	7	6	5	4	3	2	1	0
CCAP0H	FAh	PCA Compare Capture Module 0 H	CCAP0H7	CCAP0H6	CCAP0H5	CCAP0H4	CCAP0H3	CCAP0H2	CCAP0H1	CCAP0H0
CCAP1H	FBh	PCA Compare Capture Module 1 H	CCAP1H7	CCAP1H6	CCAP1H5	CCAP1H4	CCAP1H3	CCAP1H2	CCAP1H1	CCAP1H0
CCAP2H	FCh	PCA Compare Capture Module 2 H	CCAP2H7	CCAP2H6	CCAP2H5	CCAP2H4	CCAP2H3	CCAP2H2	CCAP2H1	CCAP2H0
CCAP3H	FDh	PCA Compare Capture Module 3 H	CCAP3H7	CCAP3H6	CCAP3H5	CCAP3H4	CCAP3H3	CCAP3H2	CCAP3H1	CCAP3H0
CCAP4H	FEh	PCA Compare Capture Module 4 H	CCAP4H7	CCAP4H6	CCAP4H5	CCAP4H4	CCAP4H3	CCAP4H2	CCAP4H1	CCAP4H0
CCAP0L	EAh	PCA Compare Capture Module 0 L	CCAP0L7	CCAP0L6	CCAP0L5	CCAP0L4	CCAP0L3	CCAP0L2	CCAP0L1	CCAP0L0
CCAP1L	EBh	PCA Compare Capture Module 1 L	CCAP1L7	CCAP1L6	CCAP1L5	CCAP1L4	CCAP1L3	CCAP1L2	CCAP1L1	CCAP1L0
CCAP2L	ECh	PCA Compare Capture Module 2 L	CCAP2L7	CCAP2L6	CCAP2L5	CCAP2L4	CCAP2L3	CCAP2L2	CCAP2L1	CCAP2L0
CCAP3L	EDh	PCA Compare Capture Module 3 L	CCAP3L7	CCAP3L6	CCAP3L5	CCAP3L4	CCAP3L3	CCAP3L2	CCAP3L1	CCAP3L0
CCAP4L	EEh	PCA Compare Capture Module 4 L	CCAP4L7	CCAP4L6	CCAP4L5	CCAP4L4	CCAP4L3	CCAP4L2	CCAP4L1	CCAP4L0

Table 8. Serial I/O Port SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
SCON	98h	Serial Control	FE/SM0	SM1	SM2	REN	TB8	RB8	TI	RI
SBUF	99h	Serial Data Buffer								
SADEN	B9h	Slave Address Mask								
SADDR	A9h	Slave Address								
BDRCON	9Bh	Baud Rate Control				BRR	TBCK	RBCK	SPD	SRC
BRL	9Ah	Baud Rate Reload								

Table 9. SPI Controller SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
SPCON	C3h	SPI Control	SPR2	SPEN	SSDIS	MSTR	CPOL	CPHA	SPR1	SPR0
SPSTA	C4h	SPI Status	SPIF	WCOL	SSERR	MODF				
SPDAT	C5h	SPI Data	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0

Table 10. Keyboard Interface SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
KBLS	9Ch	Keyboard Level Selector	KBLS7	KBLS6	KBLS5	KBLS4	KBLS3	KBLS2	KBLS1	KBLS0
KBE	9Dh	Keyboard Input Enable	KBE7	KBE6	KBE5	KBE4	KBE3	KBE2	KBE1	KBE0
KBF	9Eh	Keyboard Flag Register	KBF7	KBF6	KBF5	KBF4	KBF3	KBF2	KBF1	KBF0

Table 11. EEPROM data Memory SFR (AT89C51ED2 only)

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
EECON	D2h	EEPROM Data Control							EEE	EEBUSY

Table 12 shows all SFRs with their address and their reset value.

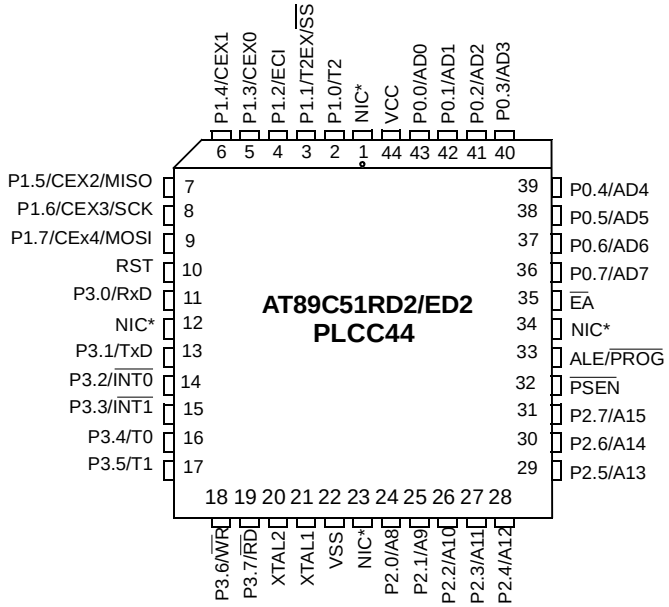
Table 12. SFR Mapping

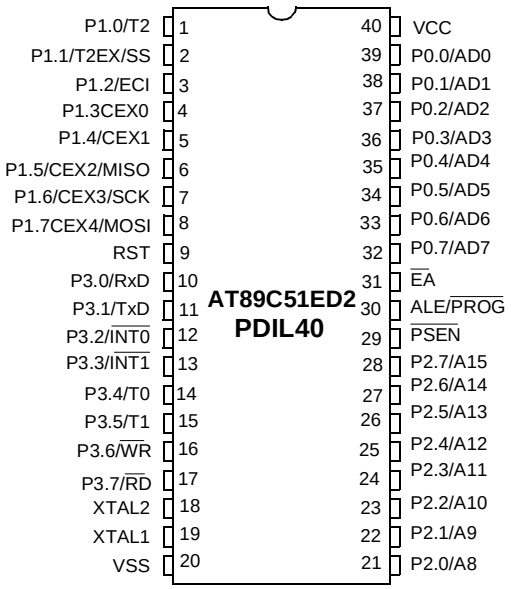
	Bit Addressable	**Non Bit Addressable**							
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8h		CH 0000 0000	CCAP0H XXXX XXXX	CCAP1H XXXX XXXX	CCAP2H XXXX XXXX	CCAP3H XXXX XXXX	CCAP4H XXXX XXXX		FFh
F0h	B 0000 0000								F7h
E8h	P5 bit addressable 1111 1111	CL 0000 0000	CCAP0L XXXX XXXX	CCAP1L XXXX XXXX	CCAP2L XXXX XXXX	CCAP3L XXXX XXXX	CCAP4L XXXX XXXX		EFh
E0h	ACC 0000 0000								E7h
D8h	CCON 00X0 0000	CMOD 00XX X000	CCAPM0 X000 0000	CCAPM1 X000 0000	CCAPM2 X000 0000	CCAPM3 X000 0000	CCAPM4 X000 0000		DFh
D0h	PSW 0000 0000	FCON XXXX 0000	EECON xxxx xx00						D7h
C8h	T2CON 0000 0000	T2MOD XXXX XX00	RCAP2L 0000 0000	RCAP2H 0000 0000	TL2 0000 0000	TH2 0000 0000			CFh
C0h	P4 1111 1111			SPCON 0001 0100	SPSTA 0000 0000	SPDAT XXXX XXXX		P5 byte Addressable 1111 1111	C7h
B8h	IPL0 X000 000	SADEN 0000 0000							BFh
B0h	P3 1111 1111	IEN1 XXXX X000	IPL1 XXXX X000	IPH1 XXXX X111				IPH0 X000 0000	B7h
A8h	IEN0 0000 0000	SADDR 0000 0000						CKCON1 XXXX XXX0	AFh
A0h	P2 1111 1111		AUXR1 0XXX X0X0				WDTRST XXXX XXXX	WDTPRG XXXX X000	A7h
98h	SCON 0000 0000	SBUF XXXX XXXX	BRL 0000 0000	BDRCON XXX0 0000	KBLS 0000 0000	KBE 0000 0000	KBF 0000 0000		9Fh
90h	P1 1111 1111							CKRL 1111 1111	97h
88h	TCON 0000 0000	TMOD 0000 0000	TL0 0000 0000	TL1 0000 0000	TH0 0000 0000	TH1 0000 0000	AUXR XX00 1000	CKCON0 0000 0000	8Fh
80h	P0 1111 1111	SP 0000 0111	DPL 0000 0000	DPH 0000 0000				PCON 00X1 0000	87h
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	

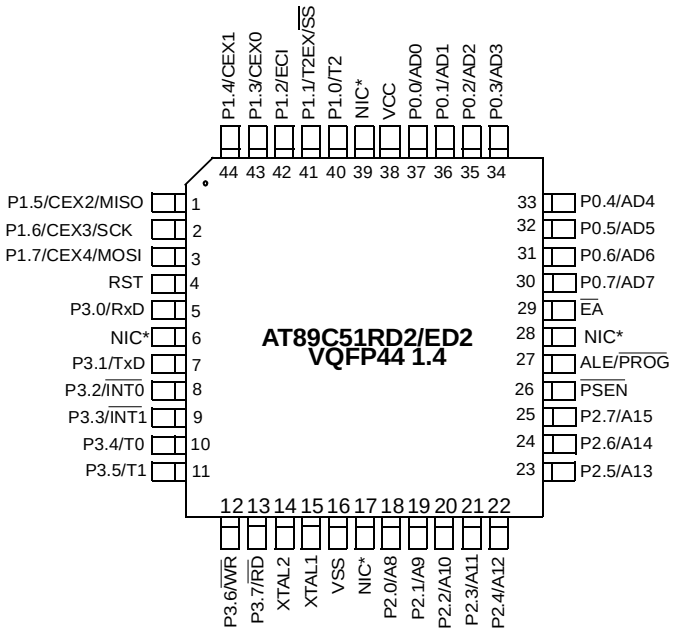
reserved

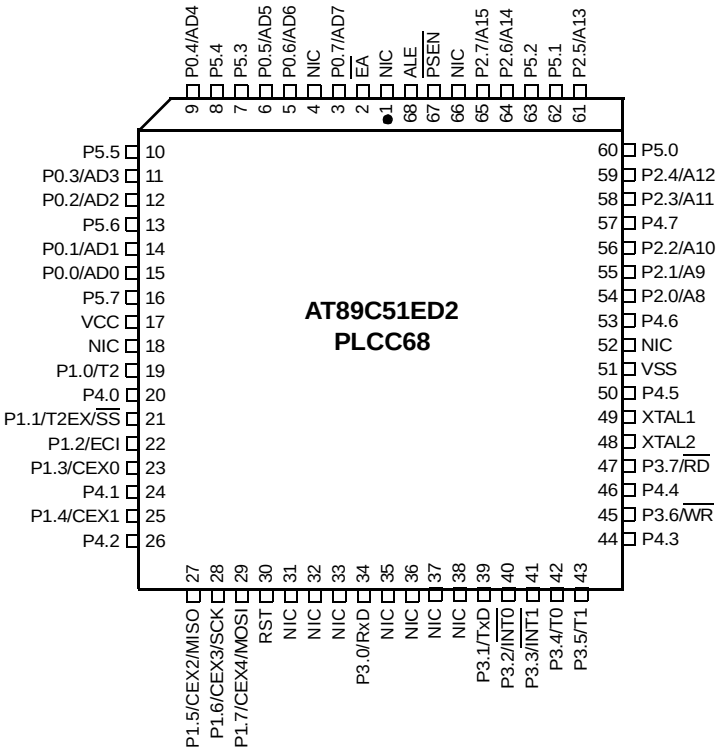
Pin Configurations

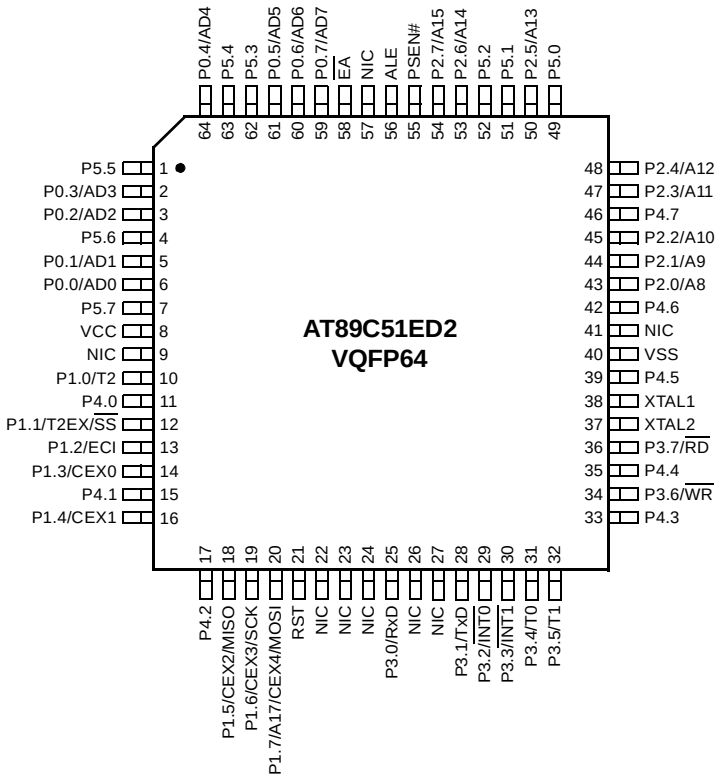
Figure 2. Pin Configurations











NIC: Not Internaly Connected

useful if external peripherals are mapped at addresses already used by the internal XRAM.

- With EXTRAM = 0, the XRAM is indirectly addressed, using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. An access to XRAM will not affect ports P0, P2, P3.6 (WR) and P3.7 (RD). For example, with EXTRAM = 0, MOVX @R0, # data where R0 contains 0A0H, accesses the XRAM at address 0A0H rather than external memory. An access to external data memory locations higher than the accessible size of the XRAM will be performed with the MOVX DPTR instructions in the same way as in the standard 80C51, with P0 and P2 as data/address busses, and P3.6 and P3.7 as write and read timing signals. Accesses to XRAM above 0FFH can only be done by the use of DPTR.
- With EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 80C51.MOVX @ Ri will provide an eight-bit address multiplexed with data on Port0 and any output port pins can be used to output higher order address bits. This is to provide the external paging capability. MOVX @DPTR will generate a sixteen-bit address. Port2 outputs the high-order eight address bits (the contents of DPH) while Port0 multiplexes the low-order eight address bits (DPL) with data. MOVX @ Ri and MOVX @DPTR will generate either read or write signals on P3.6 ($\overline{WR}$) and P3.7 ($\overline{RD}$).

The stack pointer (SP) may be located anywhere in the 256 bytes RAM (lower and upper RAM) internal data memory. The stack may not be located in the XRAM.

The M0 bit allows to stretch the XRAM timings; if M0 is set, the read and write pulses are extended from 6 to 30 clock periods. This is useful to access external slow peripherals.

Reset Output

Reset output can be generated by two sources:

- Internal POR/PFD
- Hardware watchdog timer

As detailed in Section “Hardware Watchdog Timer”, page 86, the WDT generates a 96-clock period pulse on the RST pin.

In order to properly propagate this pulse to the rest of the application in case of external capacitor or power-supply supervisor circuit, a 1 kΩ resistor must be added as shown Figure 11.

Figure 11. Recommended Reset Output Schematic

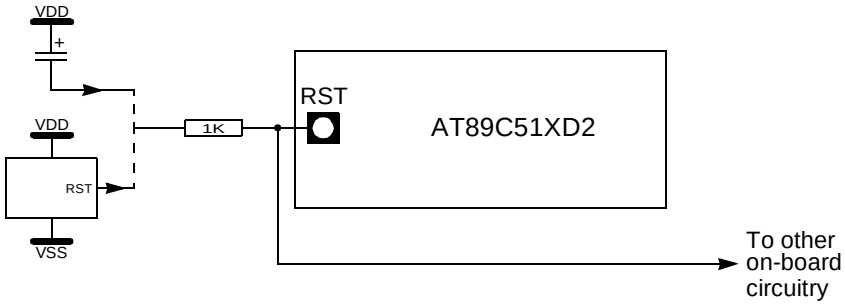
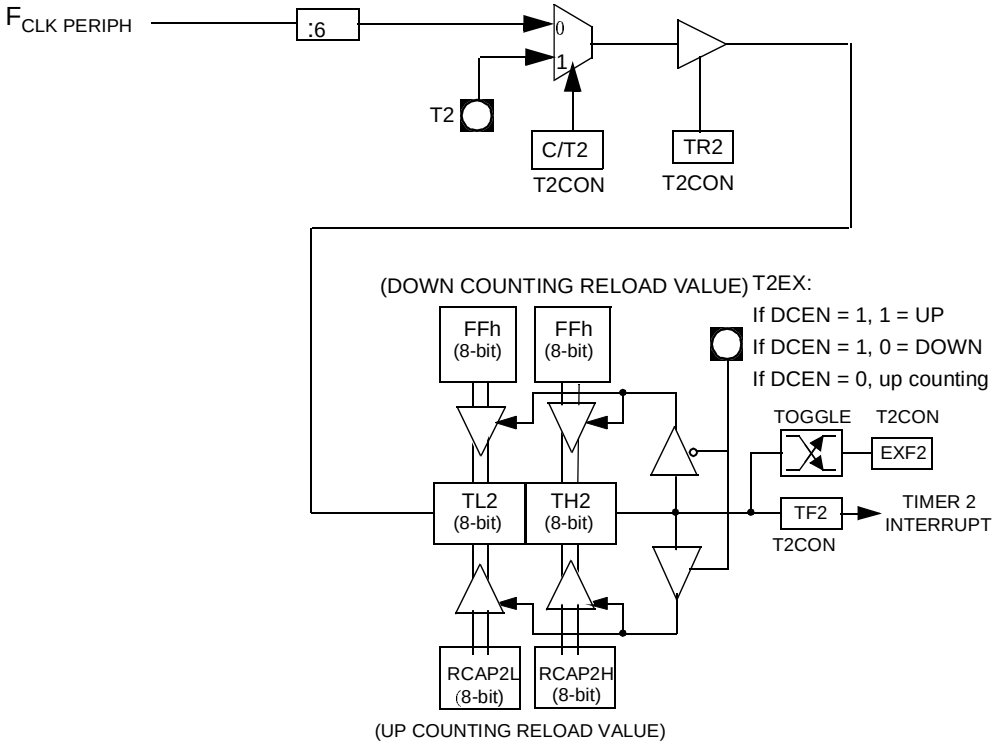


Figure 14. Auto-reload Mode Up/Down Counter (DCEN = 1)



Programmable Clock-output

In the clock-out mode, Timer 2 operates as a 50% duty-cycle, programmable clock generator (See Figure 15). The input clock increments TL2 at frequency $F_{CLK\ PERIPH}/2$. The timer repeatedly counts to overflow from a loaded value. At overflow, the contents of RCAP2H and RCAP2L registers are loaded into TH2 and TL2. In this mode, Timer 2 overflows do not generate interrupts. The formula gives the clock-out frequency as a function of the system oscillator frequency and the value in the RCAP2H and RCAP2L registers:

$$Clock\text{–}OutFrequency = \frac{F_{CLKPERIPH}}{4 \times (65536 - RCAP2H/RCAP2L)}$$

For a 16 MHz system clock, Timer 2 has a programmable frequency range of 61 Hz ($F_{CLK\ PERIPH}/2^{16}$) to 4 MHz ($F_{CLK\ PERIPH}/4$). The generated clock signal is brought out to T2 pin (P1.0).

Timer 2 is programmed for the clock-out mode as follows:

- Set T2OE bit in T2MOD register.
- Clear C/$\overline{T2}$ bit in T2CON register.
- Determine the 16-bit reload value from the formula and enter it in RCAP2H/RCAP2L registers.
- Enter a 16-bit initial value in timer registers TH2/TL2. It can be the same as the reload value or a different one depending on the application.
- To start the timer, set TR2 run control bit in T2CON register.

It is possible to use Timer 2 as a baud rate generator and a clock generator simultaneously. For this configuration, the baud rates and clock frequencies are not independent since both functions use the values in the RCAP2H and RCAP2L registers.

Table 27. CCAPnL Registers (n = 0 - 4)

CCAP0L - PCA Module 0 Compare/Capture Control Register Low (0EAh)

CCAP1L - PCA Module 1 Compare/Capture Control Register Low (0EBh)

CCAP2L - PCA Module 2 Compare/Capture Control Register Low (0ECh)

CCAP3L - PCA Module 3 Compare/Capture Control Register Low (0EDh)

CCAP4L - PCA Module 4 Compare/Capture Control Register Low (0EEh)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA Module n Compare/Capture Control** CCAPnL Value

Reset Value = 0000 0000b
Not bit addressable

Table 28. CH Register

CH - PCA Counter Register High (0F9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA counter** CH Value

Reset Value = 0000 0000b
Not bit addressable

Table 29. CL Register

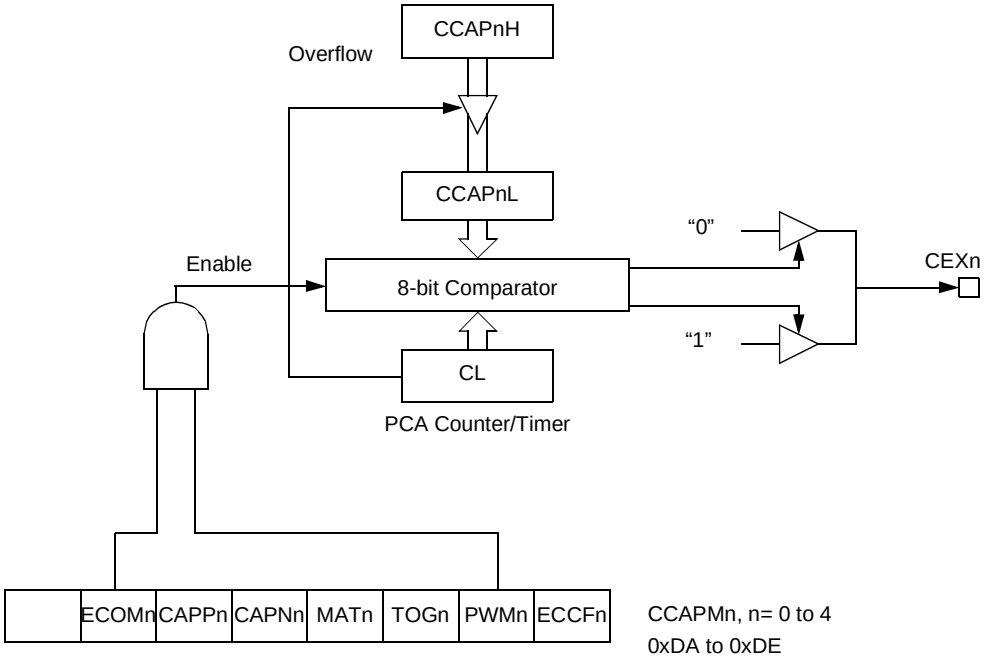
CL - PCA Counter Register Low (0E9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA Counter** CL Value

Reset Value = 0000 0000b
Not bit addressable

Figure 21. PCA PWM Mode



PCA Watchdog Timer

An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed. Figure 19 shows a diagram of how the watchdog works. The user pre-loads a 16-bit value in the compare registers. Just like the other compare modes, this 16-bit value is compared to the PCA timer value. If a match is allowed to occur, an internal reset will be generated. This will not cause the RST pin to be driven high.

In order to hold off the reset, the user has three options:

1. Periodically change the compare value so it will never match the PCA timer.
2. Periodically change the PCA timer value so it will never match the compare values.
3. Disable the watchdog by clearing the WDTE bit before a match occurs and then re-enable it.

Power Management

Introduction

Two power reduction modes are implemented in the AT89C51RD2/ED2. The Idle mode and the Power-Down mode. These modes are detailed in the following sections. In addition to these power reduction modes, the clocks of the core and peripherals can be dynamically divided by 2 using the X2 mode detailed in Section “Enhanced Features”, page 17.

Idle Mode

Idle mode is a power reduction mode that reduces the power consumption. In this mode, program execution halts. Idle mode freezes the clock to the CPU at known states while the peripherals continue to be clocked. The CPU status before entering Idle mode is preserved, i.e., the program counter and program status word register retain their data for the duration of Idle mode. The contents of the SFRs and RAM are also retained. The status of the Port pins during Idle mode is detailed in Table 59.

Entering Idle Mode

To enter Idle mode, set the IDL bit in PCON register (see Table 60). The AT89C51RD2/ED2 enters Idle mode upon execution of the instruction that sets IDL bit. The instruction that sets IDL bit is the last instruction executed.

Note: If IDL bit and PD bit are set simultaneously, the AT89C51RD2/ED2 enters Power-Down mode. Then it does not go in Idle mode when exiting Power-Down mode.

Exiting Idle Mode

There are two ways to exit Idle mode:

1. Generate an enabled interrupt.
   - Hardware clears IDL bit in PCON register which restores the clock to the CPU. Execution resumes with the interrupt service routine. Upon completion of the interrupt service routine, program execution resumes with the instruction immediately following the instruction that activated Idle mode. The general purpose flags (GF1 and GF0 in PCON register) may be used to indicate whether an interrupt occurred during normal operation or during Idle mode. When Idle mode is exited by an interrupt, the interrupt service routine may examine GF1 and GF0.
2. Generate a reset.
   - A logic high on the RST pin clears IDL bit in PCON register directly and asynchronously. This restores the clock to the CPU. Program execution momentarily resumes with the instruction immediately following the instruction that activated the Idle mode and may continue for a number of clock cycles before the internal reset algorithm takes control. Reset initializes the AT89C51RD2/ED2 and vectors the CPU to address C:0000h.

Note: During the time that execution resumes, the internal RAM cannot be accessed; however, it is possible for the Port pins to be accessed. To avoid unexpected outputs at the Port pins, the instruction immediately following the instruction that activated Idle mode should not write to a Port pin or to the external RAM.

Power-Down Mode

The Power-Down mode places the AT89C51RD2/ED2 in a very low power state. Power-Down mode stops the oscillator, freezes all clock at known states. The CPU status prior to entering Power-Down mode is preserved, i.e., the program counter, program status word register retain their data for the duration of Power-Down mode. In addition,

3. Generate an enabled external Keyboard interrupt (same behavior as external interrupt).

Note: During the time that execution resumes, the internal RAM cannot be accessed; however, it is possible for the Port pins to be accessed. To avoid unexpected outputs at the Port pins, the instruction immediately following the instruction that activated the Power-Down mode should not write to a Port pin or to the external RAM.

Note: Exit from power-down by reset redefines all the SFRs, but does not affect the internal RAM content.

Table 59. Pin Conditions in Special Operating Modes

Mode	Port 0	Port 1	Port 2	Port 3	Port 4	ALE	PSEN#
Reset	Floating	High	High	High	High	High	High
Idle (internal code)	Data	Data	Data	Data	Data	High	High
Idle (external code)	Floating	Data	Data	Data	Data	High	High
Power-Down (internal code)	Data	Data	Data	Data	Data	Low	Low
Power-Down (external code)	Floating	Data	Data	Data	Data	Low	Low

Reduced EMI Mode

The ALE signal is used to demultiplex address and data buses on port 0 when used with external program or data memory. Nevertheless, during internal code execution, ALE signal is still generated. In order to reduce EMI, ALE signal can be disabled by setting AO bit.

The AO bit is located in AUXR register at bit location 0. As soon as AO is set, ALE is no longer output but remains active during MOVX and MOVC instructions and external fetches. During ALE disabling, ALE pin is weakly pulled high.

Table 65. AUXR Register

AUXR - Auxiliary Register (8Eh)

7	6	5	4	3	2	1	0
DPU	**-**	**M0**	**XRS2**	**XRS1**	**XRS0**	**EXTRAM**	**AO**

Bit Number	Bit Mnemonic	Description
7	DPU	**Disable Weak Pull-up** Cleared by software to activate the permanent weak pull-up (default) Set by software to disable the weak pull-up (reduce power consumption)
6	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
5	M0	**Pulse length** Cleared to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 6 clock periods (default). Set to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 30 clock periods.
4	XRS2	**XRAM Size** XRS2 XRS1 XRS0 XRAM size 0 0 0 256 bytes 0 0 1 512 bytes 0 1 0 768 bytes(default) 0 1 1 1024 bytes 1 0 0 1792 bytes
3	XRS1	
2	XRS0	
1	EXTRAM	**EXTRAM bit** Cleared to access internal XRAM using MOVX @ Ri/ @ DPTR. Set to access external memory. Programmed by hardware after Power-up regarding Hardware Security Byte (HSB), default setting, XRAM selected.
0	AO	**ALE Output bit** Cleared, ALE is emitted at a constant rate of 1/6 the oscillator frequency (or 1/3 if X2 mode is used) (default). Set, ALE is active only during a MOVX or MOVC instruction is used.

Reset Value = XX00 10'HSB. XRAM'0b
Not bit addressable

Flash Registers and Memory Map

The AT89C51RD2/ED2 Flash memory uses several registers for its management:

- Hardware register can only be accessed through the parallel programming modes which are handled by the parallel programmer.
- Software registers are in a special page of the Flash memory which can be accessed through the API or with the parallel programming modes. This page, called "Extra Flash Memory", is not in the internal Flash program memory addressing space.

Hardware Register

The only hardware register of the AT89C51RD2/ED2 is called Hardware Byte or Hardware Security Byte (HSB).

Table 67. Hardware Security Byte (HSB)

7	6	5	4	3	2	1	0
X2	**BLJB**	**-**	**-**	**XRAM**	**LB2**	**LB1**	**LB0**

Bit Number	Bit Mnemonic	Description
7	X2	**X2 Mode** Programmed (‘0’ value) to force X2 mode (6 clocks per instruction) after reset. Unprogrammed (‘1’ Value) to force X1 mode, Standard Mode, after reset (Default).
6	BLJB	**Boot Loader Jump Bit** Unprogrammed (‘1’ value) to start the user’s application on next reset at address 0000h. Programmed (‘0’ value) to start the boot loader at address F800h on next reset (Default).
5	-	**Reserved**
4	-	**Reserved**
3	XRAM	**XRAM config bit (only programmable by programmer tools)** Programmed to inhibit XRAM. Unprogrammed, this bit to valid XRAM (Default).
2-0	LB2-0	**User Memory Lock Bits (only programmable by programmer tools)** See Table 68

Boot Loader Jump Bit (BLJB)

One bit of the HSB, the BLJB bit, is used to force the boot address:

- When this bit is programmed (‘0’ value) the boot address is F800h.
- When this bit is unprogrammed (‘1’ value) the boot address is 0000h.

By default, this bit is programmed and the ISP is enabled.

Flash Memory Lock Bits

The three lock bits provide different levels of protection for the on-chip code and data when programmed as shown in Table 68.

Table 68. Program Lock Bits

Program Lock Bits				
Security Level	**LB0**	**LB1**	**LB2**	**Protection Description**
1	U	U	U	No program lock features enabled.
2	P	U	U	MOVC instruction executed from external program memory is disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further parallel programming of the on chip code memory is disabled. ISP and software programming with API are still allowed.
3	X	P	U	Same as 2, also verify code memory through parallel programming interface is disabled.
4	X	X	P	Same as 3, also external execution is disabled (Default).

Note: U: Unprogrammed or "one" level.
P: Programmed or "zero" level.
X: Do not care
WARNING: Security level 2 and 3 should only be programmed after Flash and code verification.

These security bits protect the code access through the parallel programming interface. They are set by default to level 4. The code access through the ISP is still possible and is controlled by the "software security bits" which are stored in the extra Flash memory accessed by the ISP firmware.

To load a new application with the parallel programmer, a chip erase must first be done. This will set the HSB in its inactive state and will erase the Flash memory. The part reference can always be read using Flash parallel programming modes.

Default Values

The default value of the HSB provides parts ready to be programmed with ISP:

- BLJB: Programmed force ISP operation.
- X2: Unprogrammed to force X1 mode (Standard Mode).
- XRAM: Unprogrammed to valid XRAM
- LB2-0: Security level four to protect the code from a parallel access with maximum security.

Software Registers

Several registers are used in factory and by parallel programmers. These values are used by Atmel ISP.

These registers are in the "Extra Flash Memory" part of the Flash memory. This block is also called "XAF" or eXtra Array Flash. They are accessed in the following ways:

- Commands issued by the parallel memory programmer.
- Commands issued by the ISP software.
- Calls of API issued by the application software.

Several software registers are described in Table 69.

Bootloader Functionality

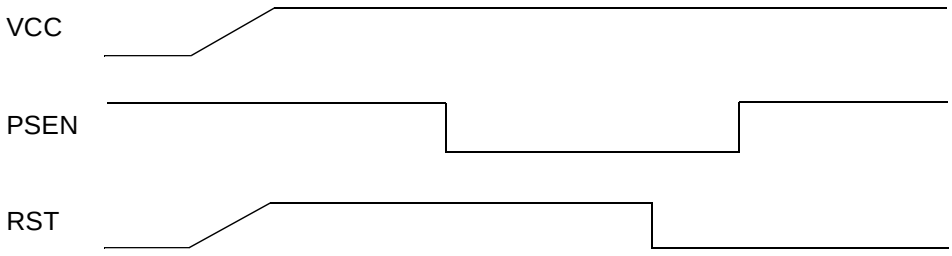
The bootloader can be activated by two means: Hardware conditions or regular boot process.

The Hardware conditions (EA = 1, PSEN = 0) during the Reset# falling edge force the on-chip bootloader execution. This allows an application to be built that will normally execute the end user's code but can be manually forced into default ISP operation.

As PSEN is a an output port in normal operating mode after reset, user application should take care to release PSEN after falling edge of reset signal. The hardware conditions are sampled at reset signal falling edge, thus they can be released at any time when reset input is low.

To ensure correct microcontroller startup, the PSEN pin should not be tied to ground during power-on (See Figure 43).

Figure 43. Hardware conditions typical sequence during power-on.



The on-chip bootloader boot process is shown Figure 44.

Table 72. Bootloader Process Description

	Purpose
Hardware Conditions	The Hardware Conditions force the bootloader execution whatever BLJB, BSB and SBV values.
BLJB	The Boot Loader Jump Bit forces the application execution. BLJB = 0 => Bootloader execution BLJB = 1 => Application execution The BLJB is a fuse bit in the Hardware Byte. It can be modified by hardware (programmer) or by software (API). Note: The BLJB test is performed by hardware to prevent any program execution.
SBV	The Software Boot Vector contains the high address of customer bootloader stored in the application. SBV = FCh (default value) if no customer bootloader in user Flash. Note: The customer bootloader is called by JMP [SBV]00h instruction.

ISP Protocol Description

Physical Layer

The UART used to transmit information has the following configuration:

- Character: 8-bit data
- Parity: none
- Stop: 2 bits
- Flow control: none
- Baudrate: autobaud is performed by the bootloader to compute the baudrate chosen by the host.

Frame Description

The Serial Protocol is based on the Intel Hex-type records.

Intel Hex records consist of ASCII characters used to represent hexadecimal values and are summarized below.

Figure 45. Intel Hex Type Frame

Record Mark ':'	Reclen	Load Offset	Record Type	Data or Info	Checksum
1-byte	1-byte	2-bytes	1-byte	n-bytes	1-byte

- Record Mark:

Record Mark is the start of frame. This field must contain ':'.

- Reclen:

Reclen specifies the number of bytes of information or data which follows the Record Type field of the record.

- Load Offset:

Load Offset specifies the 16-bit starting load offset of the data bytes, therefore this field is used only for Data Program Record (see Section "ISP Commands Summary").

- Record Type:

Record Type specifies the command type. This field is used to interpret the remaining information within the frame. The encoding for all the current record types is described in Section "ISP Commands Summary".

- Data/Info:

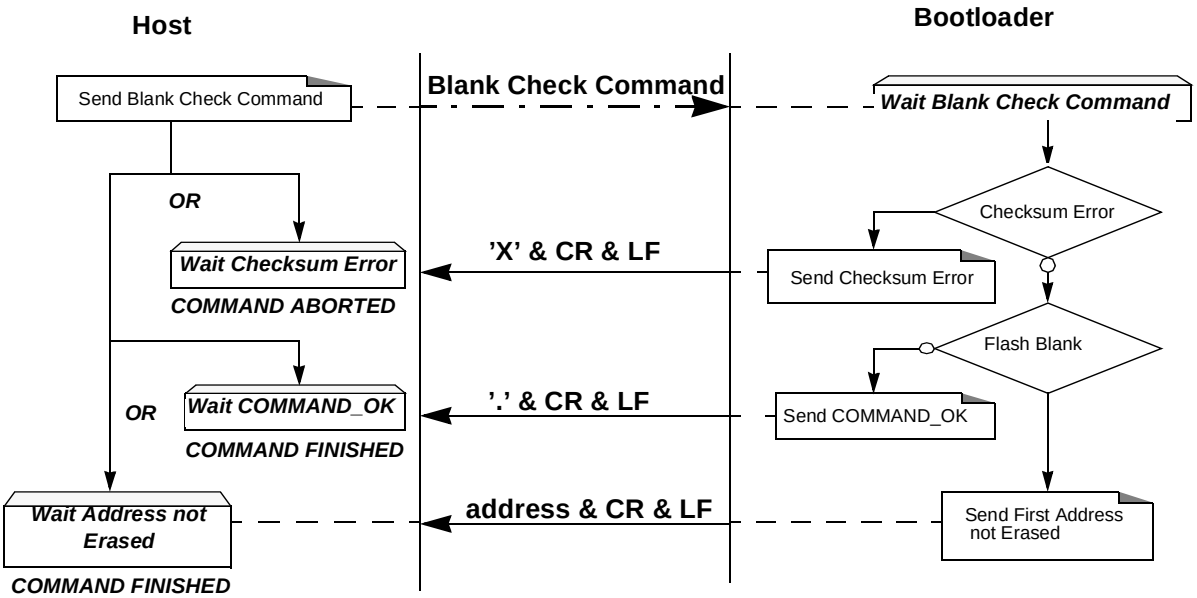
Data/Info is a variable length field. It consists of zero or more bytes encoded as pairs of hexadecimal digits. The meaning of data depends on the **Record Type**.

- Checksum:

The two's complement of the 8-bit bytes that result from converting each pair of ASCII hexadecimal digits to one byte of binary, and including the **Reclen** field to and including the last byte of the **Data/Info** field. Therefore, the sum of all the ASCII pairs in a record after converting to binary, from the **Reclen** field to and including the **Checksum** field, is zero.

Blank Check Command Description

Figure 49. Blank Check Flow



Example

Blank Check ok

```
HOST          : 05 0000 04 0000 7FFF 01 78
BOOTLOADER    : 05 0000 04 0000 7FFF 01 78 . CR LF
```

Blank Check ok at address xxxx

```
HOST          : 05 0000 04 0000 7FFF 01 78
BOOTLOADER    : 05 0000 04 0000 7FFF 01 78 xxxx CR LF
```

Blank Check with checksum error

```
HOST          : 05 0000 04 0000 7FFF 01 70
BOOTLOADER    : 05 0000 04 0000 7FFF 01 70 X CR LF CR LF
```

API Call Description

The IAP allows to reprogram a microcontroller on-chip Flash memory without removing it from the system and while the embedded application is running.

The user application can call some Application Programming Interface (API) routines allowing IAP. These API are executed by the bootloader.

To call the corresponding API, the user must use a set of Flash_api routines which can be linked with the application.

Example of Flash_api routines are available on the Atmel web site on the software application note:

C Flash Drivers for the AT89C51RD2/ED2

The API calls description and arguments are shown in Table 76.

Process

The application selects an API by setting R1, ACC, DPTR0 and DPTR1 registers.

All calls are made through a common interface “USER_CALL” at the address FFF0h.

The jump at the USER_CALL must be done by LCALL instruction to be able to come-back in the application.

Before jump at the USER_CALL, the bit ENBOOT in AUXR1 register must be set.

Constraints

The interrupts are not disabled by the bootloader.

Interrupts must be disabled by user prior to jump to the USER_CALL, then re-enabled when returning.

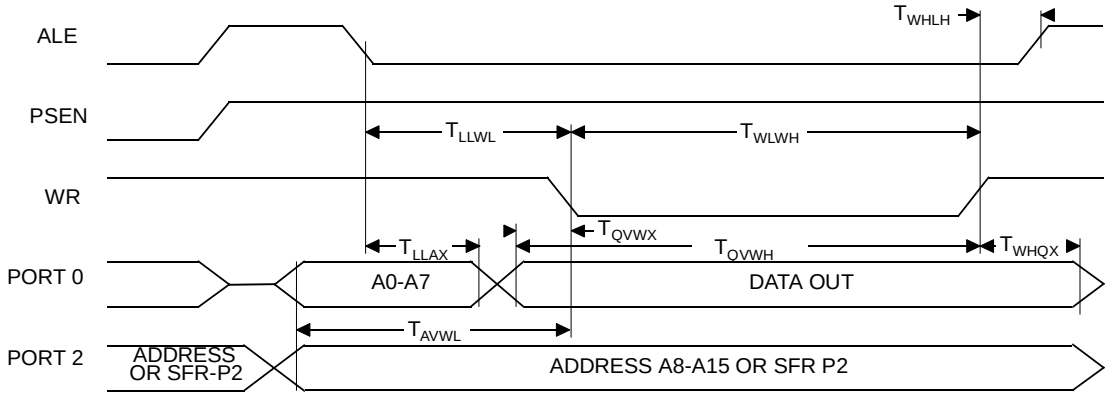
Interrupts must also be disabled before accessing EEPROM Data then re-enabled after.

The user must take care of hardware watchdog before launching a Flash operation.

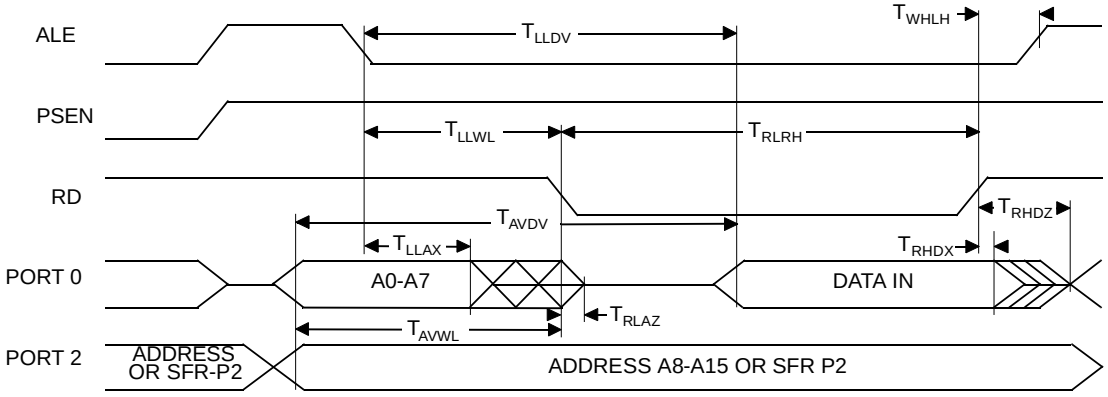
Table 76. API Call Summary

Command	R1	A	DPTR0	DPTR1	Returned Value	Command Effect
READ MANUF ID	00h	XXh	0000h	XXh	ACC = Manufacturer Id	Read Manufacturer identifier
READ DEVICE ID1	00h	XXh	0001h	XXh	ACC = Device Id 1	Read Device identifier 1
READ DEVICE ID2	00h	XXh	0002h	XXh	ACC = Device Id 2	Read Device identifier 2
READ DEVICE ID3	00h	XXh	0003h	XXh	ACC = Device Id 3	Read Device identifier 3
ERASE BLOCK	01h	XXh	DPH = 00h	00h	ACC = DPH	Erase block 0
			DPH = 20h			Erase block 1
			DPH = 40h			Erase block 2
			DPH = 80h			Erase block 3
			DPH = C0h			Erase block 4
PROGRAM DATA BYTE	02h	Vaue to write	Address of byte to program	XXh	ACC = 0: DONE	Program up one data byte in the on-chip flash memory.

External Data Memory Write Cycle



External Data Memory Read Cycle



Serial Port Timing - Shift Register Mode

Table 83. Symbol Description

Symbol	Parameter
T_{XLXL}	Serial port clock cycle time
T_{QVHX}	Output data set-up to clock rising edge
T_{XHQX}	Output data hold after clock rising edge
T_{XHDX}	Input data hold after clock rising edge
T_{XHDV}	Clock rising edge to input data valid