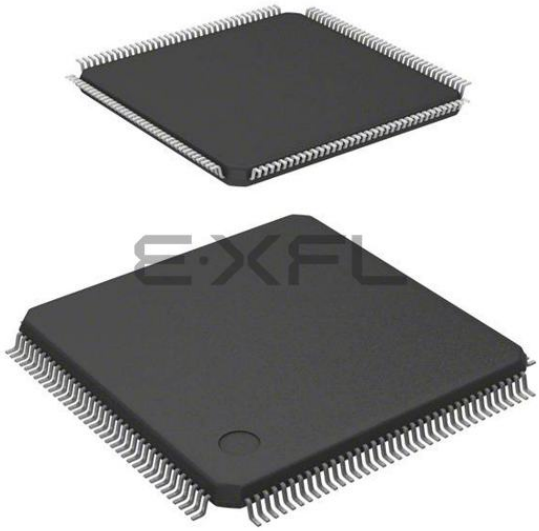


Welcome to E-XFL.COM

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	ST10
Core Size	16-Bit
Speed	40MHz
Connectivity	ASC, CANbus, EBI/EMI, I²C, SSC, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	111
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	10K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 24x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/f271-bag-t-tr

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

mapping. Accesses to this address range will send back the content of the Flash cell (by default FFFFh, blank value when the device is delivered)

2 *Accesses to the area will send back the value 009Bh.*

5.2.2 Modules structure

The IFLASH module is composed by a bank (Bank 0) of 256 Kbyte of Program Memory divided in 8 sectors (B0F0...B0F7). Bank 0 contains also a reserved sector named Test-Flash. The Addresses from 0x08 0000 to 0x08 FFFF are reserved for the Control Register Interface and other internal service memory space used by the Flash Program/Erase controller.

The following tables show the memory mapping of the Flash when it is accessed in read mode (*Table 4: Flash modules sectorization (Read operations)*), and when accessed in write or erase mode (*Table 5: Flash modules sectorization (Write operations or with ROMS1='1' or BootStrap mode)*): note that with this second mapping, the first four banks are remapped into code segment 1 (same as obtained setting bit ROMS1 in SYSCON register).

Table 4. Flash modules sectorization (Read operations)

Bank	Description	Addresses	Size	ST10 Bus size
B0	Bank 0 Flash 0 (B0F0)	0x0000 0000 - 0x0000 1FFF	8 KB	32-bit (I-BUS)
	Bank 0 Flash 1 (B0F1)	0x0000 2000 - 0x0000 3FFF	8 KB	
	Bank 0 Flash 2 (B0F2)	0x0000 4000 - 0x0000 5FFF	8 KB	
	Bank 0 Flash 3 (B0F3)	0x0000 6000 - 0x0000 7FFF	8 KB	
	Bank 0 Flash 4 (B0F4)	0x0001 8000 - 0x0001 FFFF	32 KB	
	Bank 0 Flash 5 (B0F5)	0x0002 0000 - 0x0002 FFFF	64 KB	
	Bank 0 Flash 6 (B0F6) (1)	*0x0003 0000 - 0x0003 FFFF*	*64 KB*	
	Bank 0 Flash 7 (B0F7) (1)	*0x0004 0000 - 0x0004 FFFF*	*64 KB*	

1. IFlash sectors B0F6 and B0F7 are not physically disabled. The corresponding area must be reserved by the application mapping. Accesses to this area will send back the content of the Flash cell (by default FFFFh: the erased state, if not modified by user).

Table 5. Flash modules sectorization (Write operations or with ROMS1='1' or BootStrap mode)

Bank	Description	Addresses	Size	ST10 Bus size
B0	Bank 0 Test-Flash (B0TF)	0x0000 0000 - 0x0000 1FFF	8 KB	32-bit (I-BUS)
	Bank 0 Flash 0 (B0F0)	0x0001 0000 - 0x0001 1FFF	8 KB	
	Bank 0 Flash 1 (B0F1)	0x0001 2000 - 0x0001 3FFF	8 KB	
	Bank 0 Flash 2 (B0F2)	0x0001 4000 - 0x0001 5FFF	8 KB	
	Bank 0 Flash 3 (B0F3)	0x0001 6000 - 0x0001 7FFF	8 KB	
	Bank 0 Flash 4 (B0F4)	0x0001 8000 - 0x0001 FFFF	32 KB	
	Bank 0 Flash 5 (B0F5)	0x0002 0000 - 0x0002 FFFF	64 KB	
	Bank 0 Flash 6 (B0F6) (1)	*0x0003 0000 - 0x0003 FFFF*	*64 KB*	
	Bank 0 Flash 7 (B0F7) (1)	*0x0004 0000 - 0x0004 FFFF*	*64 KB*	

Obsolete Product(s) - Obsolete Product(s)

5.4.4 Flash control register 1 high

The Flash Control Register 1 High (FCR1H), together with Flash Control Register 1 Low (FCR1L), is used to select the Sectors to Erase, or during any write operation to monitor the status of each Sector and Bank.

FCR1H (0x08 0006) FCR Reset value: 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
reserved							B0S	reserved							
							RS								

Table 10. Flash control register 1 high

Bit	Function
B0S	Bank 0 Status (IFLASH) During any erase operation, this bit is automatically modified and gives the status of the Bank 0. The meaning of B0S bit is given in the next Table 4 Banks (BxS) and Sectors (BxFy) Status bits meaning. This bit is automatically reset at the end of a erase operation if no errors are detected.

During any erase operation, this bit is automatically set and gives the status of the Bank 0. The meaning of B0Fy bit for Sector y of Bank 0 is given by the next Table 4 Banks (BxS) and Sectors (BxFy) Status bits meaning. These bits are automatically reset at the end of an erase operation if no errors are detected.

Table 11. Banks (BxS) and sectors (BxFy) status bits meaning

ERR	SUSP	B0S = 1 meaning	B0Fy = 1 meaning
1	-	Erase Error in Bank 0	Erase Error in Sector y of Bank 0
0	1	Erase Suspended in Bank 0	Erase Suspended in Sector y of Bank 0
0	0	Don't care	Don't care

5.4.5 Flash data register 0 low

The Flash Address Registers (FARH/L) and the Flash Data Registers (FDR1H/L-FDR0H/L) are used during the program operations to store Flash Address in which to program and Data to program.

FDR0L (0x08 0008) FCR Reset value: FFFFh

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIN15	DIN14	DIN13	DIN12	DIN11	DIN10	DIN9	DIN8	DIN7	DIN6	DIN5	DIN4	DIN3	DIN2	DIN1	DIN0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 12. Flash data register 0 low

Bit	Function
DIN(15:0)	Data Input 15:0 These bits must be written with the Data to program the Flash with the following operations: Word Program (32-bit), Double Word Program (64-bit) and Set Protection.

Obsolete Product(s) - Obsolete Product(s)

7.1 Multiplier-accumulator unit (MAC)

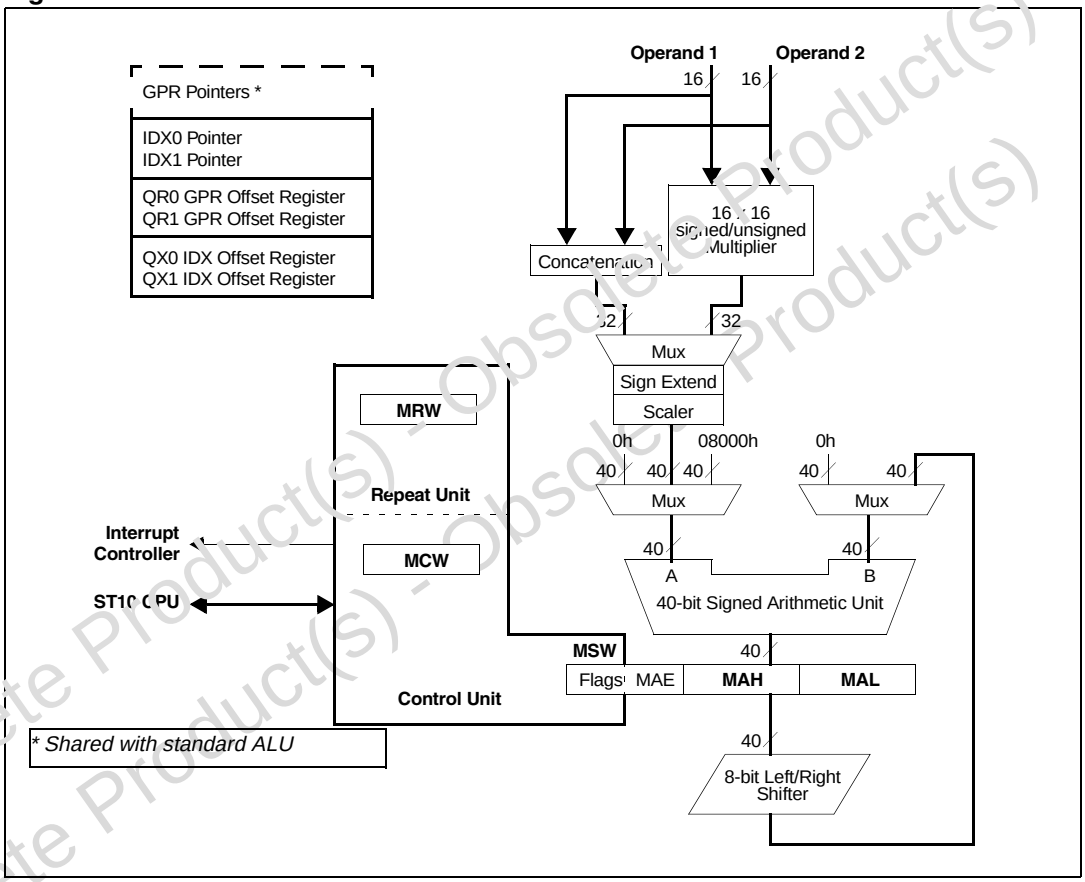
The MAC co-processor is a specialized co-processor added to the ST10 CPU Core in order to improve the performances of the ST10 Family in signal processing algorithms.

The standard ST10 CPU has been modified to include new addressing capabilities which enable the CPU to supply the new co-processor with up to 2 operands per instruction cycle.

This new co-processor (so-called MAC) contains a fast multiply-accumulate unit and a repeat unit.

The co-processor instructions extend the ST10 CPU instruction set with multiply, multiply-accumulate, 32-bit signed arithmetic operations.

Figure 8. MAC unit architecture



Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

8 External bus controller

All of the external memory accesses are performed by the on-chip external bus controller.

The EBC can be programmed to single chip mode when no external memory is required, or to one of four different external memory access modes:

- 16- / 18- / 20- / 24-bit addresses and 16-bit data, demultiplexed
- 16- / 18- / 20- / 24-bit addresses and 16-bit data, multiplexed
- 16- / 18- / 20- / 24-bit addresses and 8-bit data, multiplexed
- 16- / 18- / 20- / 24-bit addresses and 8-bit data, demultiplexed

In demultiplexed bus modes addresses are output on PORT1 and data is input / output on PORT0 or P0L, respectively. In the multiplexed bus modes both addresses and data use PORT0 for input / output.

Timing characteristics of the external bus interface (memory cycle time, memory tri-state time, length of ALE and read / write delay) are programmable giving the choice of a wide range of memories and external peripherals.

Up to four independent address windows may be defined (using register pairs ADDRSELx / BUSCONx) to access different resources and bus characteristics.

These address windows are arranged hierarchically where BUSCON4 overrides BUSCON3 and BUSCON2 overrides BUSCON1.

All accesses to locations not covered by these four address windows are controlled by BUSCON0. Up to five external $\overline{CS}$ signals (four windows plus default) can be generated in order to save external glue logic. Access to very slow memories is supported by a 'Ready' function.

A $\overline{HOLD}$ / $\overline{HLDA}$ protocol is available for bus arbitration which shares external resources with other bus masters.

The bus arbitration is enabled by setting bit HLDEN in register PSW. After setting HLDEN once, pins P6.7...P6.5 ($\overline{BREQ}$, $\overline{HLDA}$, $\overline{HOLD}$) are automatically controlled by the EBC. In master mode (default after reset) the $\overline{HLDA}$ pin is an output. By setting bit DP6.7 to'1' the slave mode is selected where pin $\overline{HLDA}$ is switched to input. This directly connects the slave controller to another master controller without glue logic.

For applications which require less external memory space, the address space can be restricted to 1 Mbyte, 256 Kbytes or to 64 Kbytes. Port 4 outputs all eight address lines if an address space of 16M Bytes is used, otherwise four, two or no address lines.

Chip select timing can be made programmable. By default (after reset), the $\overline{CSx}$ lines change half a CPU clock cycle after the rising edge of ALE. With the CSCFG bit set in the SYSCON register the $\overline{CSx}$ lines change with the rising edge of ALE.

The active level of the READY pin can be set by bit RDYPOL in the BUSCONx registers. When the READY function is enabled for a specific address window, each bus cycle within the window must be terminated with the active level defined by bit RDYPOL in the associated BUSCON register.

Obsolete Product(s) - Obsolete Product(s)

Table 28. Interrupt sources (continued)

Source of Interrupt or PEC Service Request	Request Flag	Enable Flag	Interrupt Vector	Vector Location	Trap Number
CAPCOM Register 6	CC6IR	CC6IE	CC6INT	00'0058h	16h
CAPCOM Register 7	CC7IR	CC7IE	CC7INT	00'005Ch	17h
CAPCOM Register 8	CC8IR	CC8IE	CC8INT	00'0060h	18h
CAPCOM Register 9	CC9IR	CC9IE	CC9INT	00'0064h	19h
CAPCOM Register 10	CC10IR	CC10IE	CC10INT	00'0068h	1Ah
CAPCOM Register 11	CC11IR	CC11IE	CC11INT	00'006Ch	1Bh
CAPCOM Register 12	CC12IR	CC12IE	CC12INT	00'0070h	1Ch
CAPCOM Register 13	CC13IR	CC13IE	CC13INT	00'0074h	1Dh
CAPCOM Register 14	CC14IR	CC14IE	CC14INT	00'0078h	1Eh
CAPCOM Register 15	CC15IR	CC15IE	CC15INT	00'007Ch	1Fh
CAPCOM Register 16	CC16IR	CC16IE	CC16INT	00'00C0h	30h
CAPCOM Register 17	CC17IR	CC17IE	CC17INT	00'00C4h	31h
CAPCOM Register 18	CC18IR	CC18IE	CC18INT	00'00C8h	32h
CAPCOM Register 19	CC19IR	CC19IE	CC19INT	00'00CCh	33h
CAPCOM Register 20	CC20IR	CC20IE	CC20INT	00'00D0h	34h
CAPCOM Register 21	CC21IR	CC21IE	CC21INT	00'00D4h	35h
CAPCOM Register 22	CC22IR	CC22IE	CC22INT	00'00D8h	36h
CAPCOM Register 23	CC23IR	CC23IE	CC23INT	00'00DCh	37h
CAPCOM Register 24	CC24IR	CC24IE	CC24INT	00'00E0h	38h
CAPCOM Register 25	CC25IR	CC25IE	CC25INT	00'00E4h	39h
CAPCOM Register 26	CC26IR	CC26IE	CC26INT	00'00E8h	3Ah
CAPCOM Register 27	CC27IR	CC27IE	CC27INT	00'00ECh	3Bh
CAPCOM Register 28	CC28IR	CC28IE	CC28INT	00'00F0h	3Ch
CAPCOM Register 29	CC29IR	CC29IE	CC29INT	00'0110h	44h
CAPCOM Register 30	CC30IR	CC30IE	CC30INT	00'0114h	45h
CAPCOM Register 31	CC31IR	CC31IE	CC31INT	00'0118h	46h
CAPCOM Timer 0	T0IR	T0IE	T0INT	00'0080h	20h
CAPCOM Timer 1	T1IR	T1IE	T1INT	00'0084h	21h
CAPCOM Timer 7	T7IR	T7IE	T7INT	00'00F4h	3Dh
CAPCOM Timer 8	T8IR	T8IE	T8INT	00'00F8h	3Eh
GPT1 Timer 2	T2IR	T2IE	T2INT	00'0088h	22h
GPT1 Timer 3	T3IR	T3IE	T3INT	00'008Ch	23h
GPT1 Timer 4	T4IR	T4IE	T4INT	00'0090h	24h
GPT2 Timer 5	T5IR	T5IE	T5INT	00'0094h	25h

Obsolete Product(s) - Obsolete Product(s)

11.2 GPT2

The GPT2 module provides precise event control and time measurement. It includes two timers (T5, T6) and a capture/reload register (CAPREL). Both timers can be clocked with an input clock which is derived from the CPU clock via a programmable prescaler or with external signals. The count direction (up/down) for each timer is programmable by software or may additionally be altered dynamically by an external signal on a port pin (TxEUD). Concatenation of the timers is supported via the output toggle latch (T6OTL) of timer T6 which changes its state on each timer overflow/underflow.

The state of this latch may be used to clock timer T5, or it may be output on a port pin (T6OUT). The overflow / underflow of timer T6 can additionally be used to clock the CAPCOM timers T0 or T1, and to cause a reload from the CAPREL register. The CAPREL register may capture the contents of timer T5 based on an external signal transition on the corresponding port pin (CAPIN), and timer T5 may optionally be cleared after the capture procedure. This allows absolute time differences to be measured or pulse multiplication to be performed without software overhead.

The capture trigger (timer T5 to CAPREL) may also be generated upon transitions of GPT1 timer T3 inputs T3IN and/or T3EUD. This is advantageous when T3 operates in Incremental Interface Mode.

Table 36 and *Table 37* list the timer input frequencies, resolution and periods for each pre-scaler option at 40MHz and 64MHz CPU clock respectively.

Table 36. GPT2 timer input frequencies, resolutions and periods at 40 MHz

f_{CPU} = 40MHz	Timer Input Selection T5I / T6I							
	000b	001b	010b	011b	100b	101b	110b	111b
Pre-scaler factor	4	8	16	32	64	128	256	512
Input Freq	10MHz	5MHz	2.5MHz	1.25 MHz	625 kHz	312.5 kHz	156.25 kHz	78.125 kHz
Resolution	100ns	200ns	400ns	0.8µs	1.6µs	3.2µs	6.4µs	12.8µs
Period maximum	6.55ms	13.1ms	26.2ms	52.4ms	104.8ms	209.7ms	419.4ms	838.9ms

Table 37. GPT2 timer input frequencies, resolutions and periods at 64 MHz

f_{CPU} = 64MHz	Timer Input Selection T5I / T6I							
	000b	001b	010b	011b	100b	101b	110b	111b
Pre-scaler factor	4	8	16	32	64	128	256	512
Input Freq	16MHz	8MHz	4MHz	2MHz	1 kHz	500 kHz	250 kHz	128 kHz
Resolution	62.5ns	125ns	250ns	0.5µs	1.0µs	2.0µs	4.0µs	8.0µs
Period maximum	4.1ms	8.2ms	16.4ms	32.8ms	65.5ms	131.1ms	262.1ms	524.3ms

Obsolete Product(s) - Obsolete Product(s)

13 Parallel ports

13.1 Introduction

The ST10F271 MCU provides up to 111 I/O lines with programmable features. These capabilities bring very flexible adaptation of this MCU to wide range of applications.

ST10F271 has nine groups of I/O lines gathered as follows:

- Port 0 is a two time 8-bit port named P0L (Low as less significant byte) and P0H (high as most significant byte)
- Port 1 is a two time 8-bit port named P1L and P1H
- Port 2 is a 16-bit port
- Port 3 is a 15-bit port (P3.14 line is not implemented)
- Port 4 is a 8-bit port
- Port 5 is a 16-bit port input only
- Port 6, Port 7 and Port 8 are 8-bit ports

These ports may be used as general purpose bidirectional input or output, software controlled with dedicated registers.

For example, the output drivers of six of the ports (2, 3, 4, 6, 7, 8) can be configured (bit-wise) for push-pull or open drain operation using ODPx registers.

The input threshold levels are programmable (TTL/CMOS) for all the ports. The logic level of a pin is clocked into the input latch once per state time, regardless whether the port is configured for input or output. The threshold is selected with PICON and XPICON registers control bits.

A write operation to a port pin configured as an input causes the value to be written into the port output latch, while a read operation returns the latched state of the pin itself. A read-modify-write operation reads the value of the pin, modifies it, and writes it back to the output latch.

Writing to a pin configured as an output (DPx.y='1') causes the output latch and the pin to have the written value, since the output buffer is enabled. Reading this pin returns the value of the output latch. A read-modify-write operation reads the value of the output latch, modifies it, and writes it back to the output latch, thus also modifying the level at the pin.

I/O lines support an alternate function which is detailed in the following description of each port.

Obsolete Product(s) - Obsolete Product(s)

13.2 I/O's special features

13.2.1 Open drain mode

Some of the I/O ports of ST10F271 support the open drain capability. This programmable feature may be used with an external pull-up resistor, in order to get an AND wired logical function.

This feature is implemented for ports P2, P3, P4, P6, P7 and P8 (see respective sections), and is controlled through the respective Open Drain Control Registers ODPx.

20.2 Asynchronous reset

An asynchronous reset is triggered when $\overline{\text{RSTIN}}$ pin is pulled low while RPD pin is at low level. Then the ST10F271 is immediately (after the input filter delay) forced in reset default state. It pulls low $\overline{\text{RSTOUT}}$ pin, it cancels pending internal hold states if any, it aborts all internal/external bus cycles, it switches buses (data, address and control signals) and I/O pin drivers to high-impedance, it pulls high Port0 pins.

Note: If an asynchronous reset occurs during a read or write phase in internal memories, the content of the memory itself could be corrupted: to avoid this, synchronous reset usage is strongly recommended.

Power-on reset

The asynchronous reset must be used during the power-on of the device. Depending on crystal or resonator frequency, the on-chip oscillator needs about 1ms to 10ms to stabilize (Refer to Electrical Characteristics Section), with an already stable V_{DD}. The logic of the ST10F271 does not need a stabilized clock signal to detect an asynchronous reset, so it is suitable for power-on conditions. To ensure a proper reset sequence, the $\overline{\text{RSTIN}}$ pin and the RPD pin must be held at low level until the device clock signal is stabilized and the system configuration value on Port0 is settled.

At Power-on it is important to respect some additional constraints introduced by the start-up phase of the different embedded modules.

In particular the on-chip voltage regulator needs at least 1ms to stabilize the internal 1.8V for the core logic: this time is computed from when the external reference (V_{DD}) becomes stable (inside specification range, that is at least 4.5V). This is a constraint for the application hardware (external voltage regulator): the $\overline{\text{RSTIN}}$ pin assertion shall be extended to guarantee the voltage regulator stabilization.

A second constraint is imposed by the embedded FLASH. When booting from internal memory, starting from $\overline{\text{RSTIN}}$ releasing, it needs a maximum of 1ms for its initialization: before that, the internal reset (RST signal) is not released, so the CPU does not start code execution in internal memory.

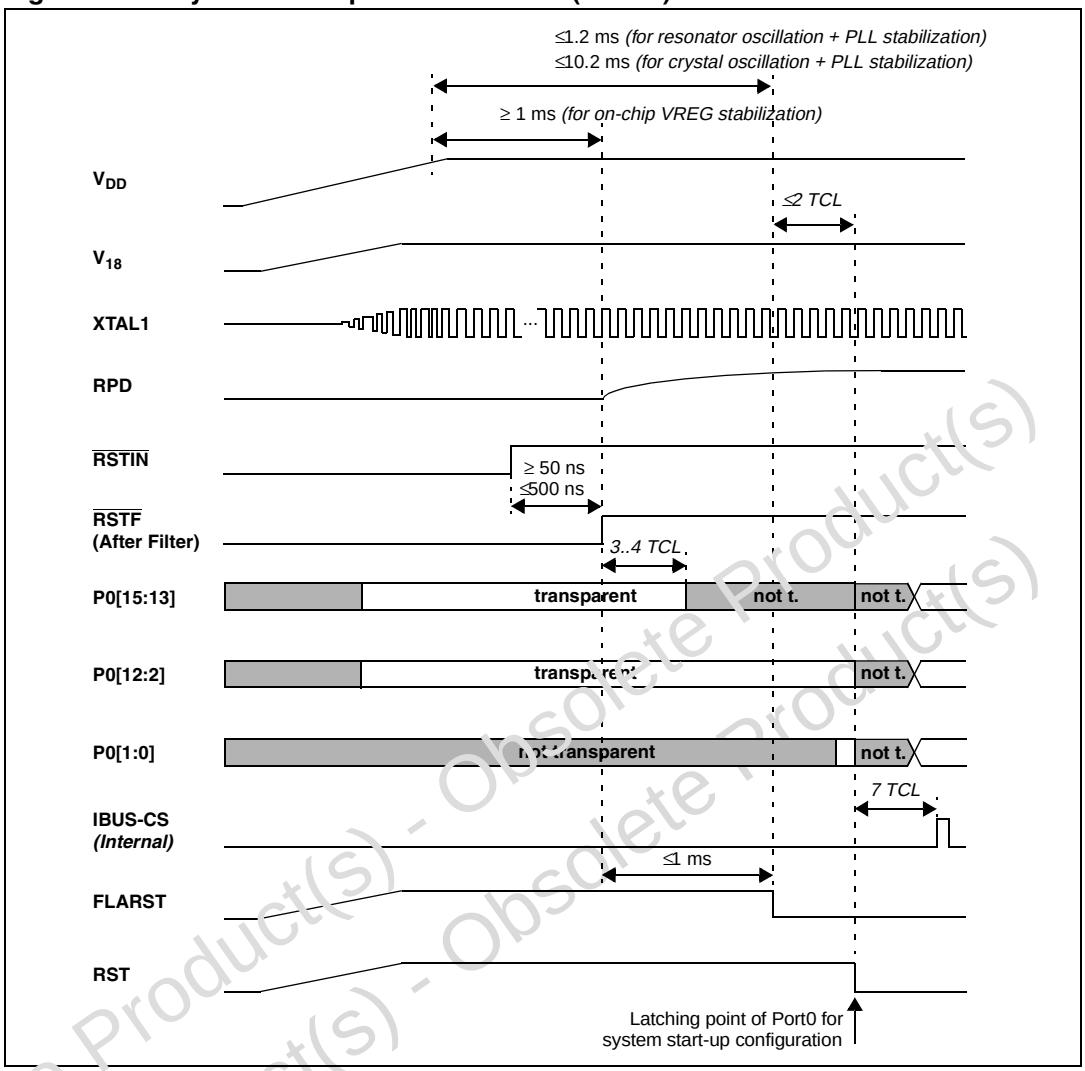
Note: This is not true if external memory is used (pin $\overline{EA}$ held low during reset phase). In this case, once $\overline{RSTIN}$ pin is released, and after few CPU clock (Filter delay plus 3...8 TCL), the internal reset signal RST is released as well, so the code execution can start immediately after. Obviously, an eventual access to the data in internal Flash is forbidden before its initialization phase is completed: an eventual access during starting phase will return FFFFh (just at the beginning), while later 009Bh (an illegal opcode trap can be generated).

At Power-on, the $\overline{\text{RSTIN}}$ pin shall be tied low for a minimum time that includes also the start-up time of the main oscillator (t_{STUP} = 1ms for resonator, 10ms for crystal) and PLL synchronization time (t_{PSUP} = 200µs): this means that if the internal FLASH is used, the $\overline{\text{RSTIN}}$ pin could be released before the main oscillator and PLL are stable to recover some time in the start-up phase (FLASH initialization only needs stable V_{18}, but does not need stable system clock since an internal dedicated oscillator is used).

Warning: It is recommended to provide the external hardware with a current limitation circuitry. This is necessary to avoid permanent damages of the device during the power-on transient, when the capacitance on V_{18} pin is charged. For the on-chip voltage regulator functionality 10nF are

Obsolete Product(s) - Obsolete Product(s)

Figure 17. Asynchronous power-on RESET ($\overline{EA}$ = 1)



Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

Short and long synchronous reset

Once the first maximum 16 TCL are elapsed (4+12 TCL), the internal reset sequence starts. It is 1024 TCL cycles long: at the end of it, and after other 8 TCL the level of $\overline{\text{RSTIN}}$ is sampled (after the filter, see $\overline{\text{RSTF}}$ in the drawings): if it is already at high level, only Short Reset is flagged (Refer to *Chapter 19: Watchdog timer* for details on reset flags); if it is recognized still low, the Long reset is flagged as well. The major difference between Long and Short reset is that during the Long reset, also P0(15:13) become transparent, so it is possible to change the clock options.

Warning: In case of a short pulse on $\overline{\text{RSTIN}}$ pin, and when Bidirectional reset is enabled, the $\overline{\text{RSTIN}}$ pin is held low by the internal circuitry. At the end of the 1024 TCL cycles, the $\overline{\text{RTSIN}}$ pin is released, but due to the presence of the input analog filter the internal input reset signal ($\overline{\text{RSTF}}$ in the drawings) is released later (from 50 to 500ns). This delay is in parallel with the additional 8 TCL, at the end of which the internal input reset line ($\overline{\text{RSTF}}$) is sampled, to decide if the reset event is Short or Long. In particular:

- If 8 TCL > 500ns (F_{CPU} < 8 MHz), the reset event is always recognized as Short
- If 8 TCL < 500ns (F_{CPU} > 8 MHz), the reset event could be recognized either as Short or Long, depending on the real filter delay (between 50 and 500ns) and the CPU frequency ($\overline{\text{RSTF}}$ sampled High means Short reset, $\overline{\text{RSTF}}$ sampled Low means Long reset). Note that in case a Long Reset is recognized, once the 8 TCL are elapsed, the P0(15:13) pins becomes transparent, so the system clock can be re-configured. The port returns not transparent 3-4 TCL after the internal $\overline{\text{RSTF}}$ signal becomes high.

The same behavior just described, occurs also when unidirectional reset is selected and $\overline{\text{RSTIN}}$ pin is held low till the end of the internal sequence (exactly 1024 TCL + max 16 TCL) and released exactly at that time.

Note: When running with CPU frequency lower than 40 MHz, the minimum valid reset pulse to be recognized by the CPU (4 TCL) could be longer than the minimum analog filter delay (50ns); so it might happen that a short reset pulse is not filtered by the analog input filter, but on the other hand it is not long enough to trigger a CPU reset (shorter than 4 TCL): this would generate a FLASH reset but not a system reset. In this condition, the FLASH answers always with FFFFh, which leads to an illegal opcode and consequently a trap event is generated.

Exit from synchronous reset state

The reset sequence is extended until $\overline{\text{RSTIN}}$ level becomes high. Besides, it is internally prolonged by the FLASH initialization when $\overline{\text{EA}}$=1 (internal memory selected). Then, the code execution restarts. The system configuration is latched from Port0, and ALE, $\overline{\text{RD}}$ and $\overline{\text{WR}}$/$\overline{\text{WRL}}$ pins are driven to their inactive level. The ST10F271 starts program execution from memory location 00'0000h in code segment 0. This starting location will typically point to the general initialization routine. Timing of synchronous reset sequence are summarized in Figures *21* and *22* where a Short Reset event is shown, with particular highlighting on the fact that it can degenerate into Long Reset: the two figures show the behavior when booting from internal or external memory respectively. Figures *23* and *24* reports the timing of a typical synchronous Long Reset, again when booting from internal or external memory.

Obsolete Product(s) - Obsolete Product(s)

Refer to next Figures *25* and *26* for unidirectional SW reset timing, and to Figures *27*, *28* and *29* for bidirectional.

20.5 Watchdog timer reset

When the watchdog timer is not disabled during the initialization, or serviced regularly during program execution, it will overflow and trigger the reset sequence.

Unlike hardware and software resets, the watchdog reset completes a running external bus cycle if this bus cycle either does not use $\overline{\text{READY}}$, or if $\overline{\text{READY}}$ is sampled active (low) after the programmed wait states.

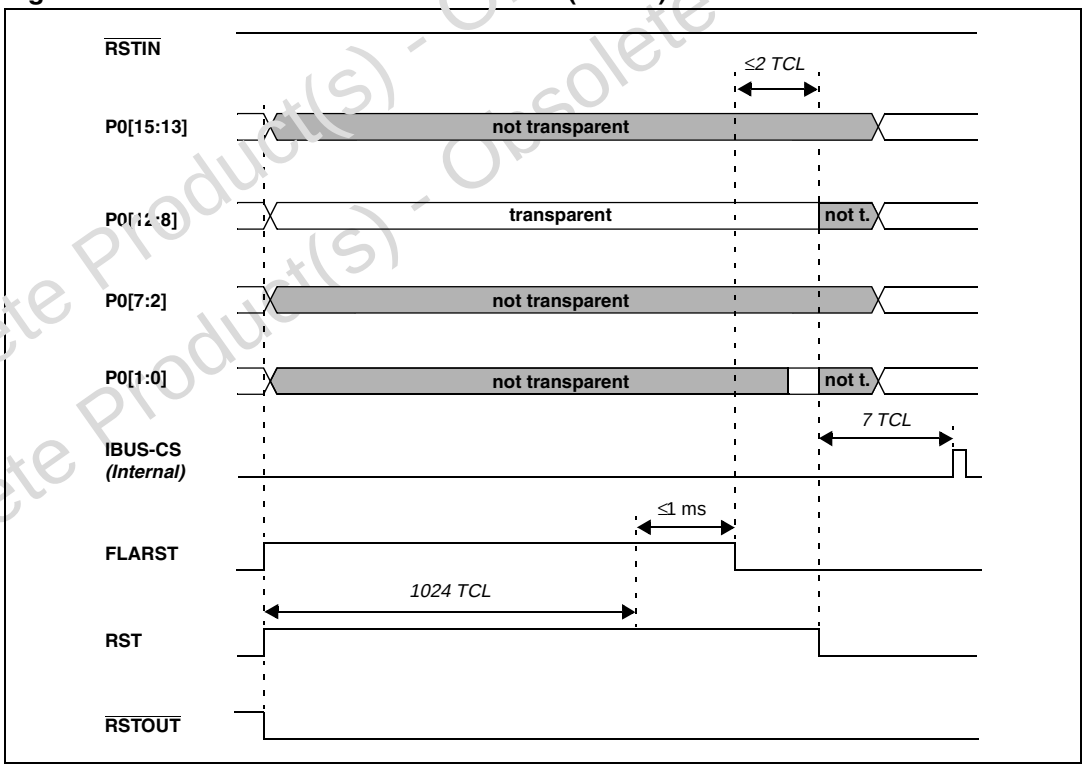
When $\overline{\text{READY}}$ is sampled inactive (high) after the programmed wait states the running external bus cycle is aborted. Then the internal reset sequence is started.

Bit P0.12...P0.8 are latched at the end of the reset sequence and bit P0.7...P0.2 are cleared (that is written at '1').

A Watchdog reset is always taken as synchronous: there is no influence on Watchdog Reset behavior with RPD status. In case Bidirectional Reset is selected, a Watchdog Reset event pulls $\overline{\text{RSTIN}}$ pin low: this occurs only if RPD is high; if RPD is low, $\overline{\text{RSTIN}}$ pin is not pulled low even though Bidirectional Reset is selected.

Refer to next Figures *25* and *26* for unidirectional SW reset timing, and to Figures *27*, *28* and *29* for bidirectional.

Figure 25. SW / WDT unidirectional RESET ($\overline{\text{EA}}$ = 1)



Obsolete Product(s) - Obsolete Product(s)

21 Power reduction modes

Three different power reduction modes with different levels of power reduction have been implemented in the ST10F271. In Idle mode only CPU is stopped, while peripheral still operate. In Power Down mode both CPU and peripherals are stopped. In Stand-by mode the main power supply (V_{DD}) can be turned off while a portion of the internal RAM remains powered via V_{STBY} dedicated power pin.

Idle and Power Down modes are software activated by a protected instruction and are terminated in different ways as described in the following sections.

Stand-by mode is entered simply removing V_{DD}, holding the MCU under reset state.

*Note: All external bus actions are completed before Idle or Power Down mode is entered. However, Idle or Power Down mode is **not** entered if READY is enabled, but has not been activated (driven low for negative polarity, or driven high for positive polarity) during the last bus access.*

21.1 Idle mode

Idle mode is entered by running IDLE protected instruction. The CPU operation is stopped and the peripherals still run.

Idle mode is terminate by any interrupt request. Whatever the interrupt is serviced or not, the instruction following the IDLE instruction will be executed after return from interrupt (RETI) instruction, then the CPU resumes the normal program.

21.2 Power down mode

Power Down mode starts by running PWRDN protected instruction. Internal clock is stopped, all MCU parts are on hold including the watchdog timer. The only exception could be the Real Time Clock if opportunely programmed and one of the two oscillator circuits as a consequence (either the main or the 32 kHz on-chip oscillator).

When Real Time Clock module is used, when the device is in Power Down mode a reference clock is needed. In this case, two possible configurations may be selected by the user application according to the desired level of power reduction:

- A 32 kHz crystal is connected to the on-chip low-power oscillator (pins XTAL3 / XTAL4) and running. In this case the main oscillator is stopped when Power Down mode is entered, while the Real Time Clock continue counting using 32 kHz clock signal as reference. The presence of a running low-power oscillator is detected after the Power-on: this clock is immediately assumed (if present, or as soon as it is detected) as reference for the Real Time Clock counter and it will be maintained forever (unless specifically disabled via software).
- Only the main oscillator is running (XTAL1 / XTAL2 pins). In this case the main oscillator is not stopped when Power Down is entered, and the Real Time Clock continue counting using the main oscillator clock signal as reference.

There are two different operating Power Down modes: protected mode and interruptible mode.

Obsolete Product(s) - Obsolete Product(s)

Table 52. List of special function registers (continued)

Name	Physical address	8-bit address	Description	Reset value
XPERCON **b**	F024h **E**	12h	XPER configuration register	- - 05h
ZEROS **b**	FF1Ch	8Eh	Constant value 0's register (read only)	0000h

Note: 1. The system configuration is selected during reset. SYSCON reset value is 0000 0xx0 x000 0000b.

2. Reset Value depends on different triggered reset event.

3. The XPnIC Interrupt Control Registers control interrupt requests from integrated X-Bus peripherals. Some software controlled interrupt requests may be generated by setting the XPnIR bits (of XPnIC register) of the unused X-Peripheral nodes.

23.2 XBus registers

The following table lists all XBus registers which are implemented in the ST10F271 ordered by their name.

Note: The XBus registers are not bit-addressable.

Table 53. List of XBus registers

Name	Physical address	Description	Reset value
CAN1BRPER	EF0Ch	CAN1: BRP extension register	0000h
CAN1BTR	EF06h	CAN1: Bit timing register	2301h
CAN1CR	EF00h	CAN1: CAN control register	0001h
CAN1EC	EF04h	CAN1: error counter	0000h
CAN1IF1A1	EF18h	CAN1: IF1 arbitration 1	0000h
CAN1IF1A2	EF1Ah	CAN1: IF1 arbitration 2	0000h
CAN1IF1CM	EF12h	CAN1: IF1 command mask	0000h
CAN1IF1CR	EF10h	CAN1: IF1 command request	0001h
CAN1IF1DA1	EF1Eh	CAN1: IF1 data A 1	0000h
CAN1IF1DA2	EF20h	CAN1: IF1 data A 2	0000h
CAN1IF1DB1	EF22h	CAN1: IF1 data B 1	0000h
CAN1IF1DB2	EF24h	CAN1: IF1 data B 2	0000h
CAN1IF1M1	EF14h	CAN1: IF1 mask 1	FFFFh
CAN1IF1M2	EF16h	CAN1: IF1 mask 2	FFFFh
CAN1IF1MC	EF1Ch	CAN1: IF1 message control	0000h
CAN1IF2A1	EF48h	CAN1: IF2 arbitration 1	0000h
CAN1IF2A2	EF4Ah	CAN1: IF2 arbitration 2	0000h
CAN1IF2CM	EF42h	CAN1: IF2 command mask	0000h

Obsolete Product(s) - Obsolete Product(s)

23.3 Flash registers ordered by name

The following table lists all Flash Control Registers which are implemented in the ST10F271 ordered by their name. These registers are physically mapped on the IBus, except for XFVTAUR0, which is mapped on XBus.

Note: These registers are not bit-addressable.

Table 54. List of flash registers

Name	Physical address	Description	Reset value
FARH	0x0008 0012	Flash address register - high	0000h
FARL	0x0008 0010	Flash address register - low	0000h
FCR0H	0x0008 0002	Flash control register 0 - high	0000h
FCR0L	0x0008 0000	Flash control register 0 - low	0000h
FCR1H	0x0008 0006	Flash control register 1 - high	0000h
FCR1L	0x0008 0004	Flash control register 1 - low	0000h
FDR0H	0x0008 000A	Flash data register 0 - high	FFFFh
FDR0L	0x0008 0008	Flash data register 0 - low	FFFFh
FDR1H	0x0008 000E	Flash data register 1 - high	FFFFh
FDR1L	0x0008 000C	Flash data register 1 - low	FFFFh
FER	0x0008 0014	Flash error register	0000h
FNVAPR0	0x0008 DFB8	Flash non volatile access protection reg.0	ACFFh
FNVAPR1H	0x0008 DFBE	Flash non volatile access protection reg.1 - high	FFFFh
FNVAPR1L	0x0008 DFBC	Flash non volatile access protection reg.1 - low	FFFFh
FNVWPIR	0x000E DFB0	Flash non volatile protection I register	FFFFh
XFVTAUR0	0x0000 EB50	XBus Flash volatile temporary access unprotection register 0	FFFFh

23.4 Identification registers

The ST10F271 have four Identification registers, mapped in ESFR space. These registers contain:

- A manufacturer identifier
- A chip identifier with its revision
- A internal Flash and size identifier
- Programming voltage description

Note: As the ST10F271 device is supported with the silicon of the ST10F272 (commercial version of the same product), the identification registers provide the values corresponding to the ST10F272 device.

Obsolete Product(s) - Obsolete Product(s)

24.6 Flash characteristics

V_{DD} = 5 V ± 10%, V_{SS} = 0 V

Table 64. Flash characteristics

Parameter	Typical	Maximum		Unit	Notes
	T_A = 25°C	T_A = 125°C			
	0 cycles(1)	0 cycles(1)	100k cycles		
Word Program (32-bit) (2)	35	80	290	µs	–
Double Word Program (64-bit)(2))	60	150	570	µs	–
Bank 0 Program (256K) (Double Word Program)	1.6	2.0	3.9	s	–
Sector Erase (8K)	0.6 0.5	0.9 0.8	1.0 0.9	s	not preprogrammed preprogrammed
Sector Erase (32K)	1.1 0.8	2.0 1.8	2.7 2.5	s	not preprogrammed preprogrammed
Sector Erase (64K)	1.7 1.3	3.7 3.3	5.1 4.7	s	not preprogrammed preprogrammed
Bank 0 Erase (256K) (3)	5.6 4.0	13.6 11.9	19.2 17.5	s	not preprogrammed preprogrammed
Recovery from Power-Down (t_{PD})	–	40	40	µs	(4)
Program Suspend Latency (4)	–	10	10	µs	
Erase Suspend Latency (4)	–	30	30	µs	
Erase Suspend Request Rate (4)	20	20	20	ms	Min delay between 2 requests
Set Protection (4)	40	90	300	µs	

1. The figures are given after about 100 cycles due to testing routines (0 cycles at the final customer).
2. Word and Double Word Programming times are provided as average values derived from a full sector programming time: absolute value of a Word or Double Word Programming time could be longer than the average value.
3. Bank Erase is obtained through a multiple Sector Erase operation (setting bits related to all sectors of the Bank). As ST10F271 implements only one bank, the Bank Erase operation is equivalent to Module and Chip Erase operations.
4. Not 100% tested, guaranteed by Design Characterization.

Obsolete Product(s) - Obsolete Product(s)

24.8.4 Prescaler operation

When pins P0.15-13 (P0H.7-5) equal '001' during reset, the CPU clock is derived from the internal oscillator (input clock signal) by a 2:1 prescaler.
The frequency of f_{CPU} is half the frequency of f_{XTAL} and the high and low time of f_{CPU} (i.e. the duration of an individual TCL) is defined by the period of the input clock f_{XTAL}.

The timings listed in the AC Characteristics that refer to TCL therefore can be calculated using the period of f_{XTAL} for any TCL.

Note that if the bit OWDDIS in SYSCON register is cleared, the PLL runs on its free-running frequency and delivers the clock signal for the Oscillator Watchdog. If bit OWDDIS is set, then the PLL is switched off.

24.8.5 Direct drive

When pins P0.15-13 (P0H.7-5) equal '011' during reset the on-chip phase locked loop is disabled, the on-chip oscillator amplifier is bypassed and the CPU clock is directly driven by the input clock signal on XTAL1 pin.

The frequency of CPU clock (f_{CPU}) directly follows the frequency of f_{XTAL} so the high and low time of f_{CPU} (i.e. the duration of an individual TCL) is defined by the duty cycle of the input clock f_{XTAL}.

Therefore, the timings given in this chapter refer to the minimum TCL. This minimum value can be calculated by the following formula:

$$TCL_{min} = 1/f_{XTAL} \times DC_{min}$$

DC = duty cycle

For two consecutive TCLs, the deviation caused by the duty cycle of f_{XTAL} is compensated, so the duration of 2TCL is always $1/f_{XTAL}$.

The minimum value TCL_{min} has to be used only once for timings that require an odd number of TCLs (1,3 ...). Timings that require an even number of TCLs (2,4,...) may use the formula:

$$2TCL = 1/f_{XTAL}$$

The address float timings in Multiplexed bus mode (t_{11} and t_{45}) use the maximum duration of TCL ($TCL_{max} = 1/f_{XTAL} \times DC_{max}$) instead of TCL_{min}.
Similarly to what happen for Prescaler Operation, if the bit OWDDIS in SYSCON register is cleared, the PLL runs on its free-running frequency and delivers the clock signal for the Oscillator Watchdog. If bit OWDDIS is set, then the PLL is switched off.

24.8.6 Oscillator watchdog (OWD)

An on-chip watchdog oscillator is implemented in the ST10F271. This feature is used for safety operation with external crystal oscillator (available only when using direct drive mode with or without prescaler, so the PLL is not used to generate the CPU clock multiplying the frequency of the external crystal oscillator). This watchdog oscillator operates as following.

The reset default configuration enables the watchdog oscillator. It can be disabled by setting the OWDDIS (bit 4) of SYSCON register.

When the OWD is enabled, the PLL runs at its free-running frequency, and it increments the watchdog counter. On each transition of external clock, the watchdog counter is cleared. If

Obsolete Product(s) - Obsolete Product(s)

an external clock failure occurs, then the watchdog counter overflows (after 16 PLL clock cycles).

The CPU clock signal will be switched to the PLL free-running clock signal, and the oscillator watchdog Interrupt Request is flagged. The CPU clock will not switch back to the external clock even if a valid external clock exits on XTAL1 pin. Only a hardware reset (or bidirectional Software / Watchdog reset) can switch the CPU clock source back to direct clock input.

When the OWD is disabled, the CPU clock is always the external oscillator clock (in Direct Drive or Prescaler Operation) and the PLL is switched off to decrease consumption supply current.

24.8.7 Phase Locked Loop (PLL)

For all other combinations of pins P0.15-13 (P0H.7-5) during reset the on-chip phase locked loop is enabled and it provides the CPU clock (see *Table 68*). The PLL multiplies the input frequency by the factor F which is selected via the combination of pins P0.15-13 (f_{CPU} = f_{XTAL} x F). With every F'th transition of f_{XTAL} the PLL circuit synchronizes the CPU clock to the input clock. This synchronization is done smoothly, so the CPU clock frequency does not change abruptly.

Due to this adaptation to the input clock the frequency of f_{CPU} is constantly adjusted so it is locked to f_{XTAL}. The slight variation causes a jitter of f_{CPU} which also effects the duration of individual TCLs.

The timings listed in the AC Characteristics that refer to TCLs therefore must be calculated using the minimum TCL that is possible under the respective circumstances.

The real minimum value for TCL depends on the jitter of the PLL. The PLL tunes f_{CPU} to keep it locked on f_{XTAL}. The relative deviation of TCL is the maximum when it is referred to one TCL period.

This is especially important for bus cycles using wait states and e.g. for the operation of timers, serial interfaces, etc. For all slower operations and longer periods (e.g. pulse train generation or measurement, lower Baud rates, etc.) the deviation caused by the PLL jitter is negligible. Refer to next *Section 24.8.9: PLL Jitter* for more details.

24.8.8 Voltage Controlled Oscillator

The ST10F271 implements a PLL which combines different levels of frequency dividers with a Voltage Controlled Oscillator (VCO) working as frequency multiplier. In the following table, a detailed summary of the internal settings and VCO frequency is reported.

Table 69. Internal PLL divider mechanism

P0.15-13 (P0H.7-5)			XTAL Frequency	Input Prescaler	PLL		Output Prescaler	CPU Frequency $f_{CPU} = f_{XTAL}$ x F
					Multiply by	Divide by		
1	1	1	4 to 8MHz	F_{XTAL} / 4	64	4	–	F_{XTAL} x 4
1	1	0	5.3 to 8MHz [1]	F_{XTAL} / 4	48	4	–	F_{XTAL} x 3
1	0	1	4 to 8MHz	F_{XTAL} / 4	64	2	–	F_{XTAL} x 8
1	0	0	6.4 to 8MHz [1]	F_{XTAL} / 4	40	2	–	F_{XTAL} x 5
0	1	1	1 to 64MHz	–	PLL bypassed		–	F_{XTAL} x 1

Obsolete Product(s) - Obsolete Product(s)

24.8.17 Demultiplexed bus

V_{DD} = 5V ± 10%, V_{SS} = 0V, T_A = –40 to +125°C, CL = 50pF,
ALE cycle time = 4 TCL + $2t_A$ + t_C + t_F (50ns at 40MHz CPU clock without wait states).

Table 78. Demultiplexed bus timings

Symbol		Parameter	F_{CPU} = 40 MHz TCL = 12.5 ns		Variable CPU Clock 1/2 TCL = 1 to 64MHz		Unit
			min.	max.	min.	max.	
t_5	CC	ALE high time	4 + t_A	–	TCL – 8.5 + t_A	–	ns
t_6	CC	Address setup to ALE	1.5 + t_A	–	TCL – 11 + t_A	–	ns
t_{80}	CC	Address/Unlatched $\overline{CS}$ setup to $\overline{RD}$, $\overline{WR}$ (with RW-delay)	12.5 + $2t_A$	–	2TCL – 12.5 + + $2t_A$	–	ns
t_{81}	CC	Address/Unlatched $\overline{CS}$ setup to $\overline{RD}$, $\overline{WR}$ (no RW-delay)	0.5 + $2t_A$	–	TCL – 12 + $2t_A$	–	ns
t_{12}	CC	$\overline{RD}$, $\overline{WR}$ low time (with RW-delay)	15.5 + t_C	–	2TCL – 9.5 + t_C	–	ns
t_{13}	CC	$\overline{RD}$, $\overline{WR}$ low time (no RW-delay)	28 + t_C	–	3TCL – 9.5 + t_C	–	ns
t_{14}	SR	$\overline{RD}$ to valid data in (with RW-delay)	–	6 + t_C	–	2TCL – 19 + t_C	ns
t_{15}	SR	$\overline{RD}$ to valid data in (no RW-delay)	–	18.5 + t_C	–	3TCL – 19 + t_C	ns
t_{16}	SR	ALE low to valid data in	–	17.5 + t_A + + t_C	–	3TCL – 20 + + t_A + t_C	ns
t_{17}	SR	Address/Unlatched $\overline{CS}$ to valid data in	–	20 + $2t_A$ + + t_C	–	4TCL – 30 + + $2t_A$ + t_C	ns
t_{18}	SR	Data hold after $\overline{RD}$ rising edge	0	–	0	–	ns
t_{20}	SR	Data float after $\overline{RD}$ rising edge (with RW-delay) [1]	–	16.5 + t_F	–	2TCL – 8.5 + + t_F + $2t_A$	ns
t_{21}	SR	Data float after $\overline{RD}$ rising edge (no RW-delay) [1]	–	4 + t_F	–	TCL – 8.5 + + t_F + $2t_A$	ns
t_{22}	CC	Data valid to $\overline{WR}$	10 + t_C	–	2TCL – 15 + t_C	–	ns
t_{24}	CC	Data hold after $\overline{WR}$	4 + t_F	–	TCL – 8.5 + t_F	–	ns
t_{26}	CC	ALE rising edge after $\overline{RD}$, $\overline{WR}$	–10 + t_F	–	–10 + t_F	–	ns
t_{28}	CC	Address/Unlatched $\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$ [2]	0 + t_F	–	0 + t_F	–	ns
t_{28h}	CC	Address/Unlatched $\overline{CS}$ hold after $\overline{WRH}$	– 5 + t_F	–	– 5 + t_F	–	ns

Obsolete Product(s) - Obsolete Product(s) - Obsolete Product(s)