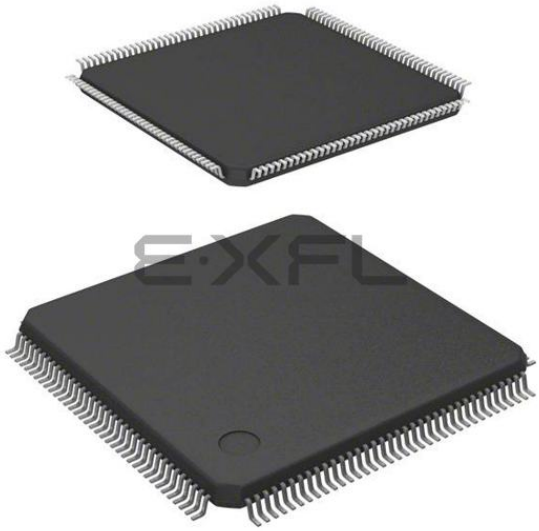


Welcome to E-XFL.COM

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	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	20K 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/f272-bag-t

Clock module, the Power-down consumption is dominated by the consumption of the oscillator amplifier itself.

A second on-chip oscillator amplifier circuit (32kHz) is implemented for low power modes: it can be used to provide the reference to the Real Time Clock counter (either in Power Down or Stand-by mode). Pin XTAL3 and XTAL4 replace a couple of VDD/VSS pins of ST10F269.

Possibility to re-program internal XBUS chip select window characteristics (XRAM2 window) is added.

5.4.11 Flash error register

Flash Error register, as well as all the other Flash registers, can be properly read only once LOCK bit of register FCR0L is low. Nevertheless, its content is updated when also BSY0 bit is reset as well; for this reason, it is definitively meaningful reading FER register content only when LOCK bit and BSY0 bit are cleared.

FER (0x8 0014h) FCR Reset value: 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
reserved							WPF	RESER	SEQER	reserved		10ER	PGER	ERER	ERR
							RC	RC	RC			RC	RC	RC	RC

Table 18. Flash error register

Bit	Function
ERR	Write Error This bit is automatically set when an error occurs during a Flash write operation or when a bad write operation setup is done. Once the error has been discovered and understood, ERR bit must be software reset.
ERER	Erase Error This bit is automatically set when an Erase error occurs during a Flash write operation. This error is due to a real failure of a Flash cell, that can no more be erased. This kind of error is fatal and the sector where it occurred must be discarded. This bit has to be software reset.
PGER	Program Error This bit is automatically set when a Program error occurs during a Flash write operation. This error is due to a real failure of a Flash cell, that can no more be programmed. The word where this error occurred must be discarded. This bit has to be software reset.
10ER	1 over 0 Error This bit is automatically set when trying to program at 1 bits previously set at 0 (this does not happen when programming the Protection bits). This error is not due to a failure of the Flash cell, but only flags that the desired data has not been written. This bit has to be software reset.
SEQER	Sequence Error This bit is automatically set when the control registers (FCR1H/L-FCR0H/L, FARH/L, FDR1H/L-FDR0H/L) are not correctly filled to execute a valid Write Operation. In this case no Write Operation is executed. This bit has to be software reset.
RESER	Resume Error This bit is automatically set when a suspended Program or Erase operation is not resumed correctly due to a protocol error. In this case the suspended operation is aborted. This bit has to be software reset.
WPF	Write Protection Flag This bit is automatically set when trying to program or erase in a sector write protected. In case of multiple Sector Erase, the not protected sectors are erased, while the protected sectors are not erased and bit WPF is set. This bit has to be software reset.

5.5 Protection strategy

The protection bits are stored in Non Volatile Flash cells inside IFLASH module, that are read once at reset and stored in 4 Volatile registers. Before they are read from the Non Volatile cells, all the available protections are forced active during reset.

The protections can be programmed using the Set Protection operation (see Flash Control Registers paragraph), that can be executed from all the internal or external memories except from the Flash itself.

Two kind of protections are available: write protections to avoid unwanted writings and access protections to avoid piracy. In next paragraphs all different level of protections are shown, and architecture limitations are highlighted as well.

5.5.1 Protection registers

The 4 Non Volatile Protection Registers are one time programmable for the user.

One register (FNVWPIR) is used to store the Write Protection fuses respectively for each sector IFLASH module. The other three Registers (FNVAPR0 and FNVAPR1L/H) are used to store the Access Protection fuses.

5.5.2 Flash non volatile write protection I register

FNVWPIR (0x08 DFB0) NVR Reset value: FFFFh

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
reserved								W0P7	W0P6	W0P5	W0P4	W0P3	W0P2	W0P1	W0P0
								RW	RW	RW	RW	RW	RW	RW	RW

Table 19. Flash non volatile write protection I register

Bit	Function
W0P(9:0)	Write Protection Bank 0 / Sectors 9-0 (IFLASH) These bits, if programmed at 0, disable any write access to the sectors of Bank 0 (IFLASH)

5.6 Write operation examples

In the following, examples for each kind of Flash write operation are presented.

Note: The write operation commands must be executed from another memory (internal RAM or external memory), as in ST10F269 device. In fact, due to IBus characteristics, it is not possible to perform write operation in Flash while fetching code from Flash.

Moreover, direct addressing is not allowed for write accesses to IFlash control registers. This means that both address and data for a writing operation must be loaded in one of ST10 GPR register (R0...R15).

Write operation on IBus registers is 16 bit wide.

Example of indirect addressing mode

```
MOV  RWm, #ADDRESS;   /*Load Add in RWm*/
MOV  RWn, #DATA;      /*Load Data in RWn*/
MOV  [RWm], RWn;      /*Indirect addressing*/
```

Word program

Example: 32-bit Word Program of data 0xAAAAAAAA at address 0x025554

```
FCR0H|= 0x2000;       /*Set WPG in FCR0H*/
FARL  = 0x5554;       /*Load Add in FARL*/
FARH  = 0x0002;       /*Load Add in FARH*/
FDR0L = 0xAAAA;       /*Load Data in FDR0L*/
FDR0H = 0xAAAA;       /*Load Data in FDR0H*/
FCR0H|= 0x8000;       /*Operation start*/
```

Double word program

Example: Double Word Program (64-bit) of data 0x55AA55AA at address 0x035558 and data 0xAA55AA55 at address 0x03555C in IFLASH Module.

```
FCR0H   |= 0x1000;        /*Set DWPG/
FARL    = 0x5558;         /*Load Add in FARL*/
FARH    = 0x0003;         /*Load Add in FARH*/
FDR0L   = 0x55AA;         /*Load Data in FDR0L*/
FDR0H   = 0x55AA;         /*Load Data in FDR0H*/
FDR1L   = 0xAA55;         /*Load Data in FDR1L*/
FDR1H   = 0xAA55;         /*Load Data in FDR1H*/
FCR0H  |= 0x8000;         /*Operation start*/
```

Double Word Program is always performed on the Double Word aligned on a even Word: bit ADD2 of FARL is ignored.

Sector erase

Example: Sector Erase of sectors B0F1 and B0F0 of Bank 0 in IFLASH Module.

```
FCR0H  |= 0x0800;         /*Set SER in FCR0H*/
FCR1L  |= 0x0003;         /*Set B0F1, B0F0*/
FCR0H  |= 0x8000;         /*Operation start*/
```

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 ($\overline{BREQ}$, HLDA, $\overline{HOLD}$) are automatically controlled by the EBC. In master mode (default after reset) the HLDA pin is an output. By setting bit DP6.7 to’1’ the slave mode is selected where pin 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.

9 Interrupt system

The interrupt response time for internal program execution is from 78ns to 187.5ns at 64 MHz CPU clock.

The ST10F272 architecture supports several mechanisms for fast and flexible response to service requests that can be generated from various sources (internal or external) to the microcontroller. Any of these interrupt requests can be serviced by the Interrupt Controller or by the Peripheral Event Controller (PEC).

In contrast to a standard interrupt service where the current program execution is suspended and a branch to the interrupt vector table is performed, just one cycle is ‘stolen’ from the current CPU activity to perform a PEC service. A PEC service implies a single Byte or Word data transfer between any two memory locations with an additional increment of either the PEC source or destination pointer. An individual PEC transfer counter is implicitly decremented for each PEC service except when performing in the continuous transfer mode. When this counter reaches zero, a standard interrupt is performed to the corresponding source related vector location. PEC services are very well suited to perform the transmission or the reception of blocks of data. The ST10F272 has 8 PEC channels, each of them offers such fast interrupt-driven data transfer capabilities.

An interrupt control register which contains an interrupt request flag, an interrupt enable flag and an interrupt priority bit-field is dedicated to each existing interrupt source. Thanks to its related register, each source can be programmed to one of sixteen interrupt priority levels. Once starting to be processed by the CPU, an interrupt service can only be interrupted by a higher prioritized service request. For the standard interrupt processing, each of the possible interrupt sources has a dedicated vector location.

Software interrupts are supported by means of the ‘TRAP’ instruction in combination with an individual trap (interrupt) number.

Fast external interrupt inputs are provided to service external interrupts with high precision requirements. These fast interrupt inputs feature programmable edge detection (rising edge, falling edge or both edges).

Fast external interrupts may also have interrupt sources selected from other peripherals; for example the CANx controller receive signals (CANx_RxD) and I^2C serial clock signal can be used to interrupt the system.

Table 29 shows all the available ST10F272 interrupt sources and the corresponding hardware-related interrupt flags, vectors, vector locations and trap (interrupt) numbers:

Table 29. Interrupt sources

Source of Interrupt or PEC Service Request	Request Flag	Enable Flag	Interrupt Vector	Vector Location	Trap Number
CAPCOM Register 0	CC0IR	CC0IE	CC0INT	00’0040h	10h
CAPCOM Register 1	CC1IR	CC1IE	CC1INT	00’0044h	11h
CAPCOM Register 2	CC2IR	CC2IE	CC2INT	00’0048h	12h
CAPCOM Register 3	CC3IR	CC3IE	CC3INT	00’004Ch	13h
CAPCOM Register 4	CC4IR	CC4IE	CC4INT	00’0050h	14h
CAPCOM Register 5	CC5IR	CC5IE	CC5INT	00’0054h	15h

20 System reset

System reset initializes the MCU in a predefined state. There are six ways to activate a reset state. The system start-up configuration is different for each case as shown in *Table 49*.

Table 49. Reset event definition

Reset source	Flag	RPD status	Conditions
Power-on reset	PONR	Low	Power-on
Asynchronous Hardware reset	LHWR	Low	$t_{\overline{RSTIN}}$ > [1]
Synchronous Long Hardware reset		High	$t_{\overline{RSTIN}}$ > (1032 + 12) TCL + max(4 TCL, 500ns)
Synchronous Short Hardware reset	SHWR	High	$t_{\overline{RSTIN}}$ > max(4 TCL, 500ns) $t_{\overline{RSTIN}}$ ≤ (1032 + 12) TCL + max(4 TCL, 500ns)
Watchdog Timer reset	WDTR	[3]	WDT overflow
Software reset	SWR	[3]	SRST instruction execution

1) $\overline{RSTIN}$ pulse should be longer than 500ns (Filter) and than settling time for configuration of Port0.

2) See next *Section 20.1* for more details on minimum reset pulse duration.

3) The RPD status has no influence unless Bidirectional Reset is activated (bit BDRSTEN in SYSCON): RPD low inhibits the Bidirectional reset on SW and WDT reset events, that is $\overline{RSTIN}$ is not activated (refer to *Section 20.4*, *Section 20.5* and *Section 20.6*).

20.1 Input filter

On $\overline{RSTIN}$ input pin an on-chip RC filter is implemented. It is sized to filter all the spikes shorter than 50ns. On the other side, a valid pulse shall be longer than 500ns to grant that ST10 recognizes a reset command. In between 50ns and 500ns a pulse can either be filtered or recognized as valid, depending on the operating conditions and process variations.

For this reason all minimum durations mentioned in this Chapter for the different kind of reset events shall be carefully evaluated taking into account of the above requirements.

In particular, for Short Hardware Reset, where only 4 TCL is specified as minimum input reset pulse duration, the operating frequency is a key factor. Examples:

- For a CPU clock of 64 MHz, 4 TCL is 31.25ns, so it would be filtered. In this case the minimum becomes the one imposed by the filter (that is 500ns).
- For a CPU clock of 4 MHz, 4 TCL is 500ns. In this case the minimum from the formula is coherent with the limit imposed by the filter.

Short and long synchronous reset

Once the first maximum 16 TCL are elapsed (4+12TCL), the internal reset sequence starts. It is 1024 TCL cycles long: at the end of it, and after other 8TCL the level of RSTIN is sampled (after the filter, see RSTF in the drawings): if it is already at high level, only Short Reset is flagged (Refer to *Section 19* 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 RSTIN pin, and when Bidirectional reset is enabled, the RSTIN pin is held low by the internal circuitry. At the end of the 1024 TCL cycles, the RTSIN pin is released, but due to the presence of the input analog filter the internal input reset signal (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 (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 (RSTF sampled High means Short reset, 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 3-4TCL after the internal RSTF signal becomes high.

The same behavior just described, occurs also when unidirectional reset is selected and RSTIN pin is held low till the end of the internal sequence (exactly 1024TCL + 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 RSTIN level becomes high. Besides, it is internally prolonged by the FLASH initialization when EA=1 (internal memory selected). Then, the code execution restarts. The system configuration is latched from Port0, and ALE, RD and WR/WRL pins are driven to their inactive level. The ST10F272 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 *Figure 20* and *Figure 21* 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. *Figure 22* and *Figure 23* reports the

Table 53. List of special function registers (continued)

Name	Physical address	8-bit address	Description	Reset value
PW1	FE32h	19h	PWM module pulse width register 1	0000h
PW2	FE34h	1Ah	PWM module pulse width register 2	0000h
PW3	FE36h	1Bh	PWM module pulse width register 3	0000h
PWMCON0**b**	FF30h	98h	PWM module control register 0	0000h
PWMCON1**b**	FF32h	99h	PWM module control register 1	0000h
PWMIC**b**	F17Eh**E**	BFh	PWM module interrupt control register	- - 00h
QR0	F004h**E**	02h	MAC unit offset register r0	0000h
QR1	F006h**E**	03h	MAC unit offset register R1	0000h
QX0	F000h**E**	00h	MAC unit offset register X0	0000h
QX1	F002h**E**	01h	MAC unit offset register X1	0000h
RP0H**b**	F108h**E**	84h	System start-up configuration register (read only)	- - XXh
S0BG	FEB4h	5Ah	Serial channel 0 baud rate generator reload register	0000h
S0CON**b**	FFB0h	D8h	Serial channel 0 control register	0000h
S0EIC**b**	FF70h	B8h	Serial channel 0 error interrupt control register	- - 00h
S0RBUF	FEB2h	59h	Serial channel 0 receive buffer register (read only)	- - XXh
S0RIC**b**	FF6Eh	B7h	Serial channel 0 receive interrupt control register	- - 00h
S0TBIC**b**	F19Ch**E**	CEh	Serial channel 0 transmit buffer interrupt control reg.	- - 00h
S0TBUF	FEB0h	58h	Serial channel 0 transmit buffer register (write only)	0000h
S0TIC**b**	FF6Ch	B6h	Serial channel 0 transmit interrupt control register	- - 00h
SP	FE12h	09h	CPU system stack pointer register	FC00h
SSCBR	F0B4h**E**	5Ah	SSC Baud rate register	0000h
SSCCON**b**	FFB2h	D9h	SSC control register	0000h
SSCEIC**b**	FF76h	BBh	SSC error interrupt control register	- - 00h
SSCRB	F0B2h**E**	59h	SSC receive buffer (read only)	XXXXh
SSCRIC**b**	FF74h	BAh	SSC receive interrupt control register	- - 00h
SSCTB	F0B0h**E**	58h	SSC transmit buffer (write only)	0000h
SSCTIC**b**	FF72h	B9h	SSC transmit interrupt control register	- - 00h
STKOV	FE14h	0Ah	CPU stack overflow pointer register	FA00h
STKUN	FE16h	0Bh	CPU stack underflow pointer register	FC00h
SYSCON**b**	FF12h	89h	CPU system configuration register	0xx0h [1]
T0	FE50h	28h	CAPCOM timer 0 register	0000h
T01CON**b**	FF50h	A8h	CAPCOM timer 0 and timer 1 control register	0000h
T0IC**b**	FF9Ch	CEh	CAPCOM timer 0 interrupt control register	- - 00h
T0REL	FE54h	2Ah	CAPCOM timer 0 reload register	0000h

Table 53. List of special function registers (continued)

Name	Physical address	8-bit address	Description	Reset value
T1	FE52h	29h	CAPCOM timer 1 register	0000h
T1IC**b**	FF9Eh	CFh	CAPCOM timer 1 interrupt control register	- - 00h
T1REL	FE56h	2Bh	CAPCOM timer 1 reload register	0000h
T2	FE40h	20h	GPT1 timer 2 register	0000h
T2CON**b**	FF40h	A0h	GPT1 timer 2 control register	0000h
T2IC**b**	FF60h	B0h	GPT1 timer 2 interrupt control register	- - 00h
T3	FE42h	21h	GPT1 timer 3 register	0000h
T3CON**b**	FF42h	A1h	GPT1 timer 3 control register	0000h
T3IC**b**	FF62h	B1h	GPT1 timer 3 interrupt control register	- - 00h
T4	FE44h	22h	GPT1 timer 4 register	0000h
T4CON**b**	FF44h	A2h	GPT1 timer 4 control register	0000h
T4IC**b**	FF64h	B2h	GPT1 timer 4 interrupt control register	- - 00h
T5	FE46h	23h	GPT2 timer 5 register	0000h
T5CON**b**	FF46h	A3h	GPT2 timer 5 control register	0000h
T5IC**b**	FF66h	B3h	GPT2 timer 5 interrupt control register	- - 00h
T6	FE48h	24h	GPT2 timer 6 register	0000h
T6CON**b**	FF48h	A4h	GPT2 timer 6 control register	0000h
T6IC**b**	FF68h	B4h	GPT2 timer 6 interrupt control register	- - 00h
T7	F050h**E**	28h	CAPCOM timer 7 register	0000h
T78CON**b**	FF20h	90h	CAPCOM timer 7 and 8 control register	0000h
T7IC**b**	F17Ah**E**	BDh	CAPCOM timer 7 interrupt control register	- - 00h
T7REL	F054h**E**	2Ah	CAPCOM timer 7 reload register	0000h
T8	F052h**E**	29h	CAPCOM timer 8 register	0000h
T8IC**b**	F17Ch**E**	BEh	CAPCOM timer 8 interrupt control register	- - 00h
T8REL	F056h**E**	2Bh	CAPCOM timer 8 reload register	0000h
TFR **b**	FFACh	D6h	Trap Flag register	0000h
WDT	FEAEh	57h	Watchdog timer register (read only)	0000h
WDTCON**b**	FFAEh	D7h	Watchdog timer control register	00xxh [2]
XADRS3	F01Ch**E**	0Eh	XPER address select register 3	800Bh
XP0IC**b**	F186h**E**	C3h	See *Section 9.1*	- - 00h [3]
XP1IC**b**	F18Eh**E**	C7h	See *Section 9.1*	- - 00h [3]
XP2IC**b**	F196h**E**	CBh	See *Section 9.1*	- - 00h [3]
XP3IC**b**	F19Eh**E**	CFh	See *Section 9.1*	- - 00h [3]

Table 54. List of XBus registers (continued)

Name	Physical address	Description	Reset value
CAN2IF1MC	EE1Ch	CAN2: IF1 message control	0000h
CAN2IF2A1	EE48h	CAN2: IF2 arbitration 1	0000h
CAN2IF2A2	EE4Ah	CAN2: IF2 arbitration 2	0000h
CAN2IF2CM	EE42h	CAN2: IF2 command mask	0000h
CAN2IF2CR	EE40h	CAN2: IF2 command request	0001h
CAN2IF2DA1	EE4Eh	CAN2: IF2 data A 1	0000h
CAN2IF2DA2	EE50h	CAN2: IF2 data A 2	0000h
CAN2IF2DB1	EE52h	CAN2: IF2 data B 1	0000h
CAN2IF2DB2	EE54h	CAN2: IF2 data B 2	0000h
CAN2IF2M1	EE44h	CAN2: IF2 mask 1	FFFFh
CAN2IF2M2	EE46h	CAN2: IF2 mask 2	FFFFh
CAN2IF2MC	EE4Ch	CAN2: IF2 message control	0000h
CAN2IP1	EEA0h	CAN2: interrupt pending 1	0000h
CAN2IP2	EEA2h	CAN2: interrupt pending 2	0000h
CAN2IR	EE08h	CAN2: interrupt register	0000h
CAN2MV1	EEB0h	CAN2: message valid 1	0000h
CAN2MV2	EEB2h	CAN2: message valid 2	0000h
CAN2ND1	EE90h	CAN2: new data 1	0000h
CAN2ND2	EE92h	CAN2: new data 2	0000h
CAN2SR	EE02h	CAN2: status register	0000h
CAN2TR	EE0Ah	CAN2: test register	00x0h
CAN2TR1	EE80h	CAN2: transmission request 1	0000h
CAN2TR2	EE82h	CAN2: Transmission request 2	0000h
I2CCCR1	EA06h	I2C clock control register 1	0000h
I2CCCR2	EA0Eh	I2C clock control register 2	0000h
I2CCR	EA00h	I2C control register	0000h
I2CDR	EA0Ch	I2C data register	0000h
I2COAR1	EA08h	I2C own address register 1	0000h
I2COAR2	EA0Ah	I2C own address register 2	0000h
I2CSR1	EA02h	I2C status register 1	0000h
I2CSR2	EA04h	I2C status register 2	0000h
RTCAH	ED14h	RTC alarm register high byte	XXXXh
RTCAL	ED12h	RTC alarm register low byte	XXXXh
RTCCON	ED00H	RTC control register	000Xh

IDMANUF (F07Eh / 3Fh) ESFR Reset Value: 0403h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MANUF											0	0	0	1	1
R											R				

Table 56. IDMANUF

Bit	Function
MANUF	Manufacturer identifier 020h: STMicroelectronics manufacturer (JTAG worldwide normalization).

IDCHIP (F07Ch / 3Eh) ESFR Reset Value: 110Xh

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDCHIP												REVID			
R												R			

Table 57. IDCHIP

Bit	Function
IDCHIP	Device identifier 110h: ST10F272 identifier (272).
REVID	Device revision identifier Xh: According to revision number.

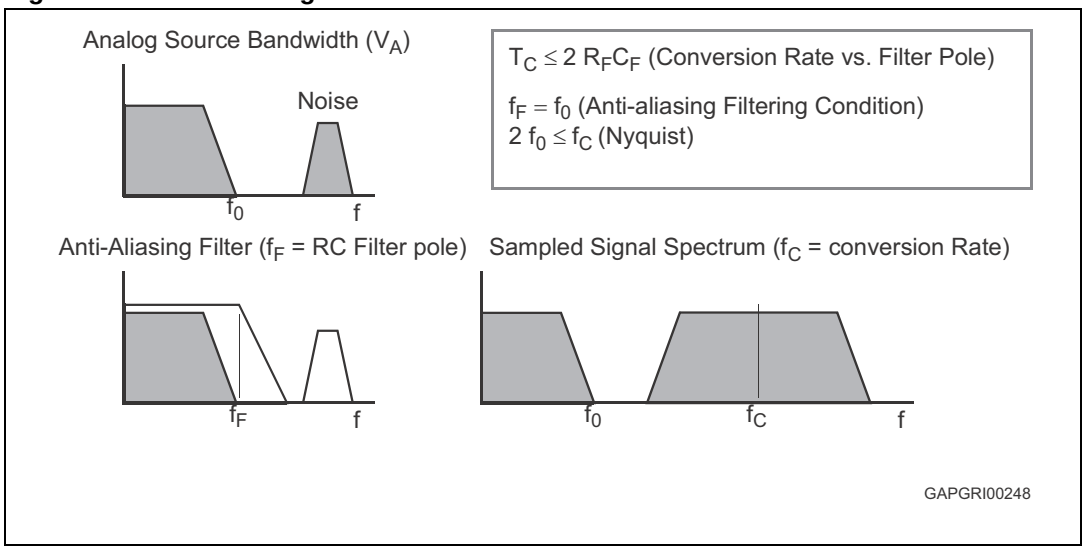
IDMEM (F07Ah / 3Dh) ESFR Reset Value: 3040h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMTYP				MEMSIZE											
R				R											

Table 58. IDMEM

Bit	Function
MEMSIZE	Internal memory size Internal memory size is 4 x (MEMSIZE) (in Kbyte) 040h for 256 Kbytes (ST10F272)
MEMTYP	Internal memory type ‘0h’: ROM-Less ‘1h’: (M) ROM memory ‘2h’: (S) Standard Flash memory ‘3h’: (H) High performance Flash memory (ST10F272) ‘4h...Fh’: Reserved

Figure 41. Anti-aliasing filter and conversion rate



The considerations above lead to impose new constraints to the external circuit, to reduce the accuracy error due to the voltage drop on C_S; from the two charge balance equations above, it is simple to derive the following relation between the ideal and real sampled voltage on C_S:

Equation 8

$$\frac{V_A}{V_{A2}} = \frac{C_{P1} + C_{P2} + C_F}{C_{P1} + C_{P2} + C_F + C_S}$$

From this formula, in the worst case (when V_A is maximum, that is for instance 5V), assuming to accept a maximum error of half a count (~2.44mV), it is immediately evident a constraints on C_F value:

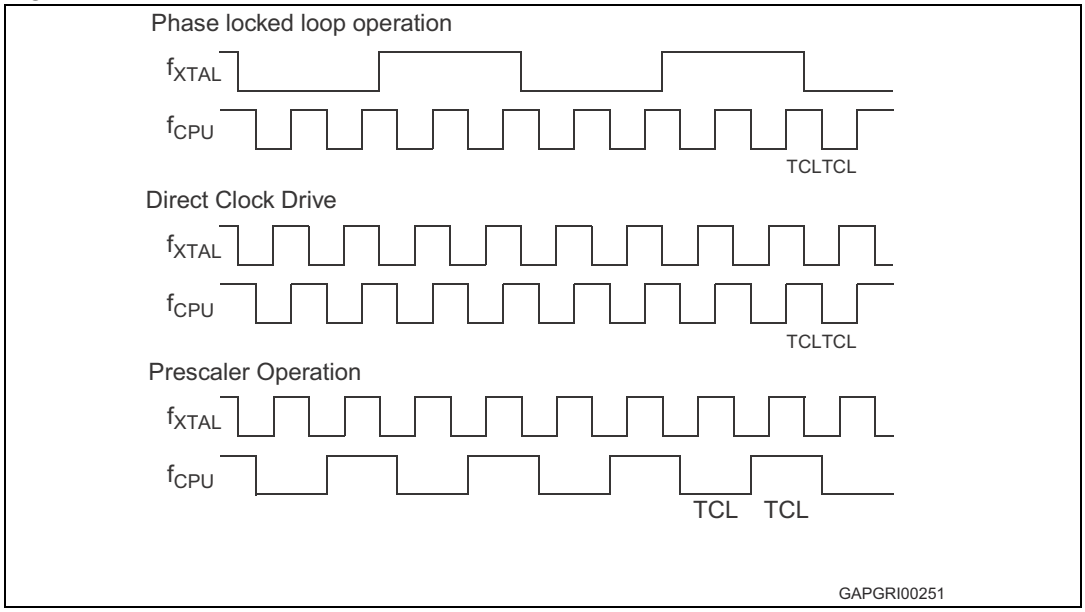
Equation 9

$$C_F > 2048\ C_S$$

In the next section an example of how to design the external network is provided, assuming some reasonable values for the internal parameters and making hypothesis on the characteristics of the analog signal to be sampled.

The mechanism used to generate the CPU clock is selected during reset by the logic levels on pins P0.15-13 (P0H.7-5).

Figure 44. Generation mechanisms for the CPU clock



24.8.3 Clock generation modes

Table 69 associates the combinations of these three bits with the respective clock generation mode.

Table 69. On-chip clock generator selections

P0.15-13 (P0H.7-5)			CPU Frequency $f_{CPU} = f_{XTAL} \times F$	External Clock Input Range [1) 3)]	Notes
1	1	1	F_{XTAL} x 4	4 to 8MHz	Default configuration
1	1	0	F_{XTAL} x 3	5.3 to 8MHz	
1	0	1	F_{XTAL} x 8	4 to 8MHz	
1	0	0	F_{XTAL} x 5	6.4 to 8MHz	
0	1	1	F_{XTAL} x 1	1 to 64MHz	Direct Drive (oscillator bypassed) [2)]
0	1	0	F_{XTAL} x 10	4 to 6.4MHz	
0	0	1	F_{XTAL} / 2	4 to 8MHz	CPU clock via prescaler [3)]
0	0	0	F_{XTAL} x 16	4MHz	

1. The external clock input range refers to a CPU clock range of 1...64 MHz. Besides, the PLL usage is limited to 4-8MHz. All configurations need a crystal (or ceramic resonator) to generate the CPU clock through the internal oscillator amplifier (apart from Direct Drive): vice versa, the clock can be forced through an external clock source only in Direct Drive mode (on-chip oscillator amplifier disabled, so no crystal or resonator can be used).
2. The maximum depends on the duty cycle of the external clock signal: when 64MHz is used, 50% duty cycle shall be granted (low phase = high phase = 7.8ns); when 32MHz is selected a 25% duty cycle can be

24.8.15 External memory bus timing

The following sections include the External Memory Bus timings. The given values are computed for a maximum CPU clock of 40MHz.

Obviously, when higher CPU clock frequency is used (up to 64MHz), some numbers in the timing formulas become zero or negative which, in most cases is not acceptable or not meaningless at all. In these cases, it is necessary to relax the speed of the bus setting properly t_A, t_C and t_F.

Note: All External Memory Bus Timings and SSC Timings reported in the following tables are granted by Design Characterization and not fully tested in production.

24.8.16 Multiplexed bus

V_{DD} = 5V ± 10%, V_{SS} = 0V, T_A = –40 to +125°C, CL = 50pF,
ALE cycle time = 6 TCL + $2t_A$ + t_C + t_F (75ns at 40MHz CPU clock without wait states)

Table 78. Multiplexed 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_7	CC	Address hold after ALE	4 + t_A	–	TCL – 8.5 + t_A	–	ns
t_8	CC	ALE falling edge to $\overline{RD}$, $\overline{WR}$ (with RW-delay)	4 + t_A	–	TCL – 8.5 + t_A	–	ns
t_9	CC	ALE falling edge to $\overline{RD}$, $\overline{WR}$ (no RW-delay)	– 8.5 + t_A	–	– 8.5 + t_A	–	ns
t_{10}	CC	Address float after $\overline{RD}$, $\overline{WR}$ (with RW-delay)	–	6	–	6	ns
t_{11}	CC	Address float after $\overline{RD}$, $\overline{WR}$ (no RW-delay)	–	18.5	–	TCL + 6	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_{19}	SR	Data float after $\overline{RD}$	–	16.5 + t_F	–	2TCL – 8.5 + t_F	ns
t_{22}	CC	Data valid to $\overline{WR}$	10 + t_C	–	2TCL – 15 + t_C	–	ns
t_{23}	CC	Data hold after $\overline{WR}$	4 + t_F	–	2TCL – 8.5 + t_F	–	ns
t_{25}	CC	ALE rising edge after $\overline{RD}$, $\overline{WR}$	15 + t_F	–	2TCL – 10 + t_F	–	ns
t_{27}	CC	Address/Unlatched $\overline{CS}$ hold after $\overline{RD}$, $\overline{WR}$	10 + t_F	–	2TCL – 15 + t_F	–	ns
t_{38}	CC	ALE falling edge to Latched $\overline{CS}$	– 4 – t_A	10 – t_A	– 4 – t_A	10 – t_A	ns

Figure 53. External memory cycle: Demultiplexed bus, with/without r/w delay, normal ALE

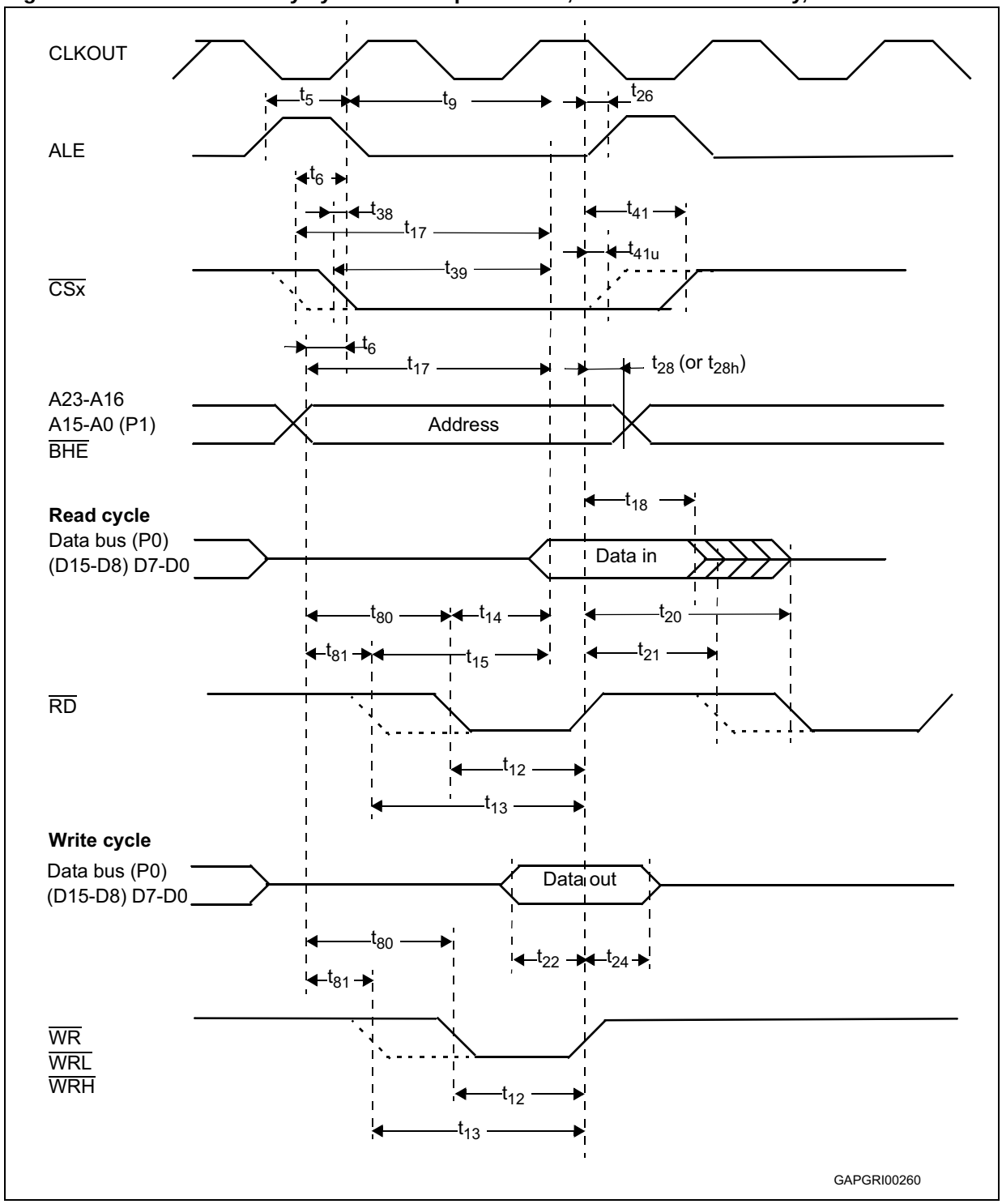
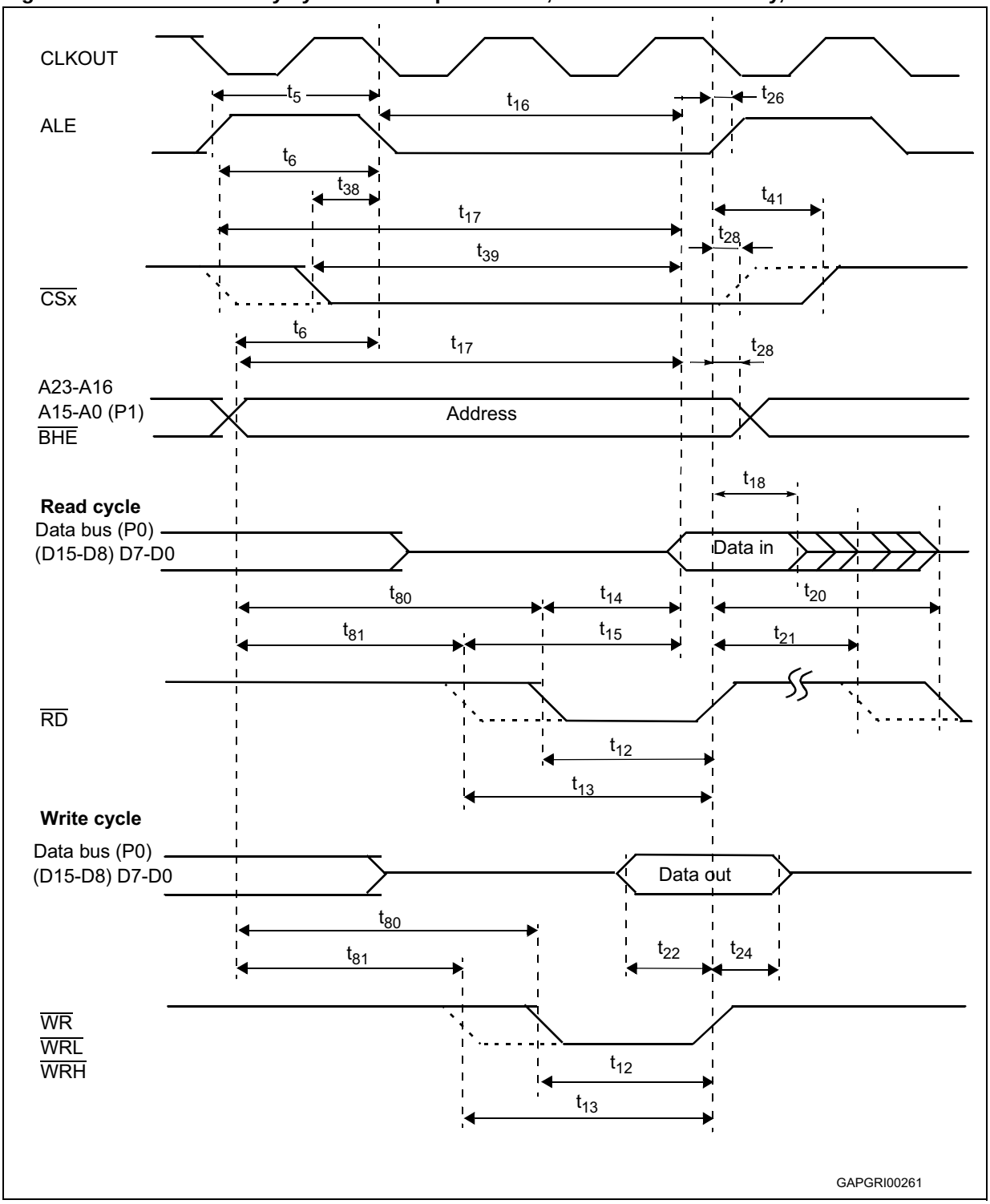


Figure 54. Exteral memory cycle: Demultiplexed bus, with/without r/w delay, extended ALE



Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

ST PRODUCTS ARE NOT DESIGNED OR AUTHORIZED FOR USE IN: (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS; (B) AERONAUTIC APPLICATIONS; (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER’S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR “AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL” INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.
Information in this document supersedes and replaces all information previously supplied.
The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com