**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 | ST7 |
| Core Size | 8-Bit |
| Speed | 8MHz |
| Connectivity | LINbusSCI, SPI |
| Peripherals | LVD, Motor Control PWM, POR, PWM, WDT |
| Number of I/O | 44 |
| Program Memory Size | 48KB (48K x 8) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 1.5K x 8 |
| Voltage - Supply (Vcc/Vdd) | 3.8V ~ 5.5V |
| Data Converters | A/D 16x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 64-LQFP |
| Supplier Device Package | - |
| Purchase URL | https://www.e-xfl.com/product-detail/stmicroelectronics/st7fmc2r7t6tr |

# 3 REGISTER & MEMORY MAP

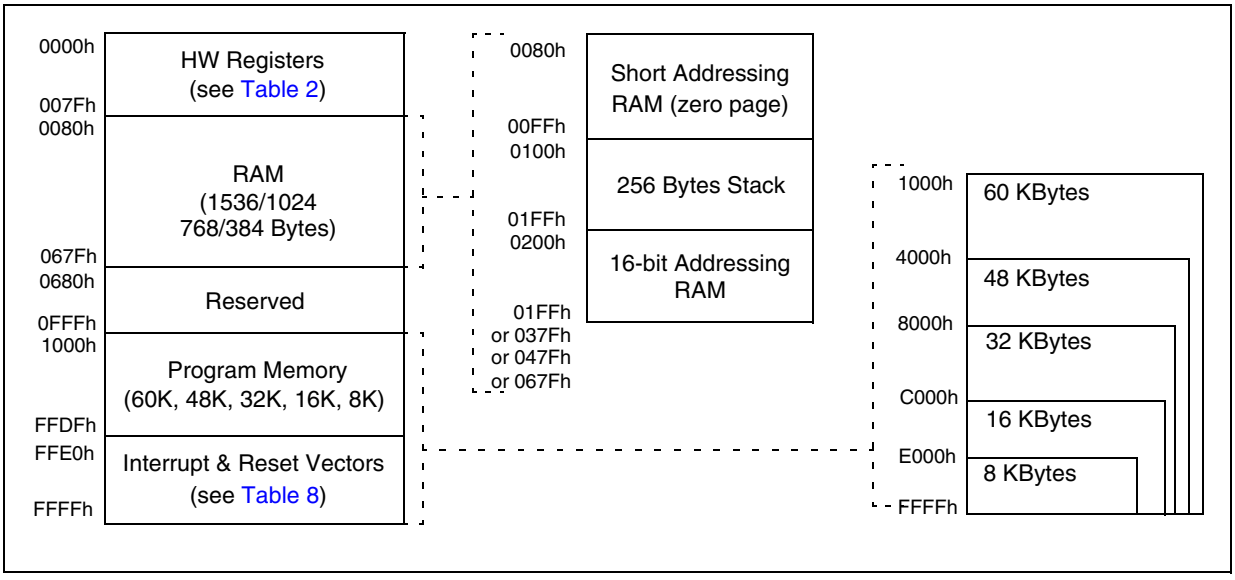
As shown in Figure 8, the MCU is capable of addressing 64K bytes of memories and I/O registers.

The available memory locations consist of 128 bytes of register locations, up to 2Kbytes of RAM and up to 60Kbytes of user program memory. The RAM space includes up to 256 bytes for the stack from 0100h to 01FFh.

The highest address bytes contain the user reset and interrupt vectors.

**IMPORTANT:** Memory locations marked as “Reserved” must never be accessed. Accessing a reserved area can have unpredictable effects on the device.

**Figure 8. Memory Map**




As shown in Figure 9, the MCU is capable of addressing 64K bytes of memories and I/O registers.

The available memory locations consist of 128 bytes of register locations, up to 1536 bytes of RAM and up to 60 Kbytes of user program memory. The RAM space includes up to 256 bytes for the stack from 0100h to 01FFh.

The highest address bytes contain the user reset and interrupt vectors.

## MAIN CLOCK CONTROLLER WITH REAL-TIME CLOCK (Cont'd)

### 6.4.5 Low Power Modes

| Mode | Description |
|---|---|
| Wait | No effect on MCC/RTC peripheral.<br>MCC/RTC interrupt cause the device to exit from Wait mode. |
| Active-halt | No effect on MCC/RTC counter (OIE bit is set), the registers are frozen.<br>MCC/RTC interrupt cause the device to exit from Active-halt mode. |
| Halt | MCC/RTC counter and registers are frozen.<br>MCC/RTC operation resumes when the MCU is woken up by an interrupt with "exit from Halt" capability. |

### 6.4.6 Interrupts

The MCC/RTC interrupt event generates an interrupt if the OIE bit of the MCCSR register is set and the interrupt mask in the CC register is not active (RIM instruction).

| Interrupt Event | Event Flag | Enable Control Bit | Exit from Wait | Exit from Halt |
|---|---|---|---|---|
| Time base overflow event | OIF | OIE | Yes | No [1] |

**Note**:

The MCC/RTC interrupt wakes up the MCU from Active-halt mode, not from Halt mode.

### 6.4.7 Register Description

**MCC CONTROL/STATUS REGISTER (MCCSR)**

Read/Write

Reset Value: 0000 0000 (00h)

| 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|
| MCO | CP1 | CP0 | SMS | TB1 | TB0 | OIE | OIF |

Bit 7 = **MCO** *Main clock out selection*
This bit enables the MCO alternate function on the PF0 I/O port. It is set and cleared by software.
0: MCO alternate function disabled (I/O pin free for general-purpose I/O)
1: MCO alternate function enabled ($f_{OSC2}$ on I/O port)

**Note**: To reduce power consumption, the MCO function is not active in Active-halt mode.

Bit 6:5 = **CP[1:0]** *CPU clock prescaler*
These bits select the CPU clock prescaler which is applied in the different slow modes. Their action is conditioned by the setting of the SMS bit. These two bits are set and cleared by software

| $f_{CPU}$ in Slow mode | CP1 | CP0 |
|---|---|---|
| $f_{OSC2}$ / 2 | 0 | 0 |
| $f_{OSC2}$ / 4 | 0 | 1 |
| $f_{OSC2}$ / 8 | 1 | 0 |
| $f_{OSC2}$ / 16 | 1 | 1 |

Bit 4 = **SMS** *Slow mode select*
This bit is set and cleared by software.
0: Normal mode. $f_{CPU} = f_{OSC2}$
1: Slow mode. $f_{CPU}$ is given by CP1, CP0
See Section 8.2 SLOW MODE and Section 6.4 MAIN CLOCK CONTROLLER WITH REAL-TIME CLOCK AND BEEPER (MCC/RTC) for more details.

Bit 3:2 = **TB[1:0]** *Time base control*

These bits select the programmable divider time base. They are set and cleared by software.

| Counter Prescaler | Time Base | | TB1 | TB0 |
|---|---|---|---|---|
| | $f_{OSC2}$ =4MHz | $f_{OSC2}$=8MHz | | |
| 16000 | 4ms | 2ms | 0 | 0 |
| 32000 | 8ms | 4ms | 0 | 1 |
| 80000 | 20ms | 10ms | 1 | 0 |
| 200000 | 50ms | 25ms | 1 | 1 |

A modification of the time base is taken into account at the end of the current period (previously set) to avoid an unwanted time shift. This allows to use this time base as a real-time clock.

Bit 1 = **OIE** *Oscillator interrupt enable*
This bit set and cleared by software.
0: Oscillator interrupt disabled
1: Oscillator interrupt enabled
This interrupt can be used to exit from Active-halt mode.
When this bit is set, calling the ST7 software HALT instruction enters the Active-halt power saving mode.

**INTERRUPTS** (Cont'd)

**Table 7. Dedicated Interrupt Instruction Set**

| Instruction | New Description | Function/Example | I1 | H | I0 | N | Z | C |
|---|---|---|---|---|---|---|---|---|
| HALT | Entering Halt mode | | 1 | | 0 | | | |
| IRET | Interrupt routine return | Pop CC, A, X, PC | I1 | H | I0 | N | Z | C |
| JRM | Jump if I1:0=11 (level 3) | I1:0=11 ? | | | | | | |
| JRNM | Jump if I1:0<>11 | I1:0<>11 ? | | | | | | |
| POP CC | Pop CC from the Stack | Mem => CC | I1 | H | I0 | N | Z | C |
| RIM | Enable interrupt (level 0 set) | Load 10 in I1:0 of CC | 1 | | 0 | | | |
| SIM | Disable interrupt (level 3 set) | Load 11 in I1:0 of CC | 1 | | 1 | | | |
| TRAP | Software trap | Software NMI | 1 | | 1 | | | |
| WFI | Wait for interrupt | | 1 | | 0 | | | |

**Note**: During the execution of an interrupt routine, the HALT, POPCC, RIM, SIM and WFI instructions change the current software priority up to the next IRET instruction or one of the previously mentioned instructions.

**Table 8. Interrupt Mapping**

| N° | Source Block | Description | Register Label | Priority Order | Exit from Halt[1)] | Address Vector |
|---|---|---|---|---|---|---|
| | RESET | Reset | N/A | Highest Priority ↓ Lowest Priority | yes | FFFEh-FFFFh |
| | TRAP | Software interrupt | | | no | FFFCh-FFFDh |
| 0 | MCES | Motor Control Emergency Stop or Speed error interrupt | MISR MCRC | | no | FFFAh-FFFBh |
| 1 | MCC/RTC CSS | Main clock controller time base interrupt Safe oscillator activation interrupt | MCCSR SICSR | | yes | FFF8h-FFF9h |
| 2 | ei0 | External interrupt port | N/A | | yes | FFF6h-FFF7h |
| 3 | ei1 | External interrupt port | | | yes | FFF4h-FFF5h |
| 4 | ei2 | External interrupt port | | | yes | FFF2h-FFF3h |
| 5 | MTC | Event U or Current Loop or Sampling Out | MISR/MCONF | | no | FFF0h-FFF1h |
| 6 | | Event R or Event Z | MISR | | no | FFEEh-FFEFh |
| 7 | | Event C or Event D | | | no | FFECh-FFEDh |
| 8 | SPI | SPI peripheral interrupts | SPICSR | | yes | FFEAh-FFEBh |
| 9 | TIMER A | TIMER A peripheral interrupts | TASR | | no | FFE8h-FFE9h |
| 10 | TIMER B | TIMER B peripheral interrupts | TBSR | | no | FFE6h-FFE7h |
| 11 | LINSCI™ | LINSCI™ Peripheral interrupts | SCISR | | no | FFE4h-FFE5h |
| 12 | AVD/ ADC | Auxiliary Voltage detector interrupt ADC End of conversion interrupt | SICSR ADCSR | | yes | FFE2h-FFE3h |
| 13 | PWM ART | PWM ART overflow interrupt PWM ART input capture interrupts | ARTCSR ARTICCSR | | no | FFE0h-FFE1h |

**Note 1.** Valid for Halt and Active-halt modes except for the MCC/RTC or CSS interrupt source which exits from Active-halt mode only.

**EXTERNAL INTERRUPT CONTROL REGISTER (EICR)** (Cont'd)

- ei0 (port D6..4)

| IS31 | IS30 | External Interrupt Sensitivity | |
|---|---|---|---|
| | | IPA bit =0 | IPA bit =1 |
| 0 | 0 | Falling edge & low level | Rising edge & high level |
| 0 | 1 | Rising edge only | Falling edge only |
| 1 | 0 | Falling edge only | Rising edge only |
| 1 | 1 | Rising and falling edge | |

- ei0 (port D3..1)

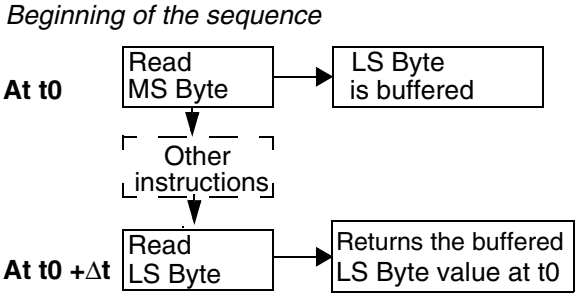
| IS31 | IS30 | External Interrupt Sensitivity |
|---|---|---|
| 0 | 0 | Falling edge & low level |
| 0 | 1 | Rising edge only |
| 1 | 0 | Falling edge only |
| 1 | 1 | Rising and falling edge |

These 2 bits can be written only when I1 and I0 of the CC register are both set to 1 (level 3).

Bit 0 = **IPA** *Interrupt polarity for port D*
This bit is used to invert the sensitivity of the port D [6:4] external interrupts. It can be set and cleared by software only when I1 and I0 of the CC register are both set to 1 (level 3).
0: No sensitivity inversion
1: Sensitivity inversion

**16-BIT TIMER** (Cont'd)

**16-bit read sequence:** (from either the Counter Register or the Alternate Counter Register).




The user must read the MS Byte first, then the LS Byte value is buffered automatically.

This buffered value remains unchanged until the 16-bit read sequence is completed, even if the user reads the MS Byte several times.

After a complete reading sequence, if only the CLR register or ACLR register are read, they return the LS Byte of the count value at the time of the read.

Whatever the timer mode used (input capture, output compare, one pulse mode or PWM mode) an overflow occurs when the counter rolls over from FFFFh to 0000h then:

– The TOF bit of the SR register is set.

– A timer interrupt is generated if:

  – TOIE bit of the CR1 register is set and

  – I bit of the CC register is cleared.

If one of these conditions is false, the interrupt remains pending to be issued as soon as they are both true.

Clearing the overflow interrupt request is done in two steps:

1. Reading the SR register while the TOF bit is set.
2. An access (read or write) to the CLR register.

**Notes:** The TOF bit is not cleared by accesses to ACLR register. The advantage of accessing the ACLR register rather than the CLR register is that it allows simultaneous use of the overflow function and reading the free running counter at random times (for example, to measure elapsed time) without the risk of clearing the TOF bit erroneously.

The timer is not affected by Wait mode.

In Halt mode, the counter stops counting until the mode is exited. Counting then resumes from the previous count (Device awakened by an interrupt) or from the reset count (Device awakened by a Reset).

#### 10.3.3.2 External Clock

The external clock (where available) is selected if CC0=1 and CC1=1 in CR2 register.

The status of the EXEDG bit in the CR2 register determines the type of level transition on the external clock pin EXTCLK that will trigger the free running counter.

The counter is synchronised with the falling edge of the internal CPU clock.

A minimum of four falling edges of the CPU clock must occur between two consecutive active edges of the external clock; thus the external clock frequency must be less than a quarter of the CPU clock frequency.

**16-BIT TIMER** (Cont'd)

**Notes:**

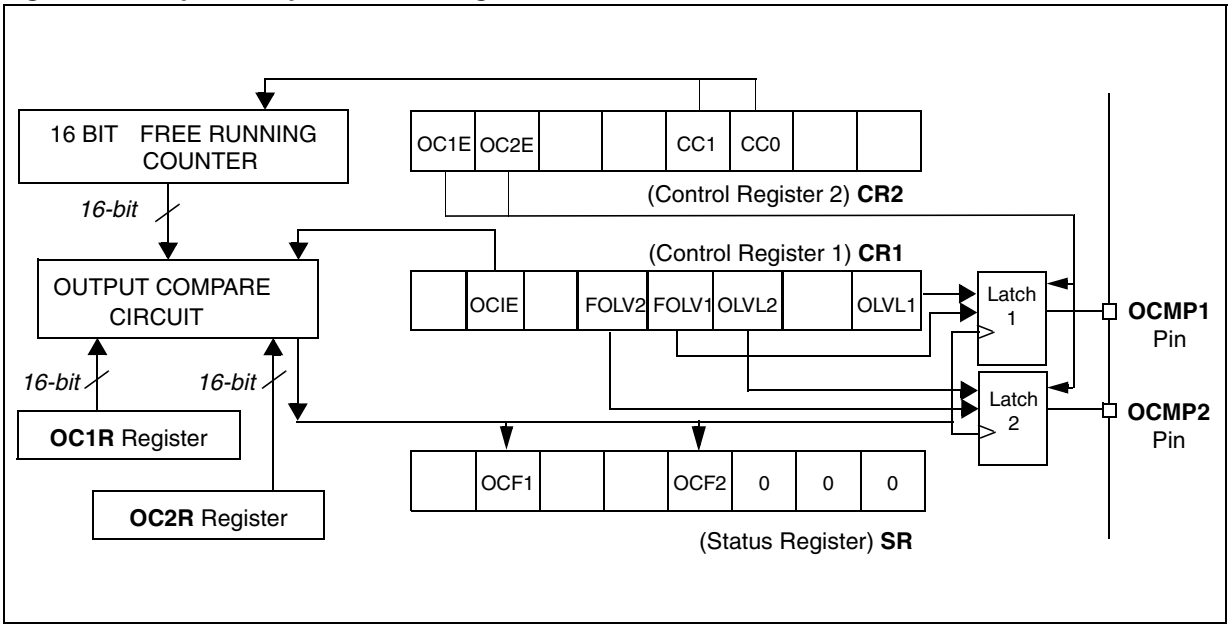
1. After a processor write cycle to the OC*i*HR register, the output compare function is inhibited until the OC*i*LR register is also written.
2. If the OC*i*E bit is not set, the OCMP*i* pin is a general I/O port and the OLVL*i* bit will not appear when a match is found but an interrupt could be generated if the OCIE bit is set.
3. In both internal and external clock modes, OCFi and OCMPi are set while the counter value equals the OCiR register value (see Figure 51 for an example with $f_{CPU/2}$ and Figure 52 for an example with $f_{CPU/4}$). This behavior is the same in OPM or PWM mode.
4. The output compare functions can be used both for generating external events on the OCMP*i* pins even if the input capture mode is also used.
5. The value in the 16-bit OC*i*R register and the OLV*i* bit should be changed after each successful comparison in order to control an output waveform or establish a new elapsed timeout.

**Forced Compare Output capability**

When the FOLV*i* bit is set by software, the OLVL*i* bit is copied to the OCMP*i* pin. The OLV*i* bit has to be toggled in order to toggle the OCMP*i* pin when it is enabled (OC*i*E bit=1). The OCF*i* bit is then not set by hardware, and thus no interrupt request is generated.

FOLVL*i* bits have no effect in both one pulse mode and PWM mode.

**Figure 50. Output Compare Block Diagram**



**LINSCI™ SERIAL COMMUNICATION INTERFACE (SCI Mode)** (cont'd)

#### 10.5.5.2 Transmitter

The transmitter can send data words of either 8 or 9 bits depending on the M bit status. When the M bit is set, word length is 9 bits and the 9th bit (the MSB) has to be stored in the T8 bit in the SCICR1 register.

**Character Transmission**

During an SCI transmission, data shifts out least significant bit first on the TDO pin. In this mode, the SCIDR register consists of a buffer (TDR) between the internal bus and the transmit shift register (see Figure 62).

**Procedure**

– Select the M bit to define the word length.

– Select the desired baud rate using the SCIBRR and the SCIETPR registers.

– Set the TE bit to send a preamble of 10 (M = 0) or 11 (M = 1) consecutive ones (Idle Line) as first transmission.

– Access the SCISR register and write the data to send in the SCIDR register (this sequence clears the TDRE bit). Repeat this sequence for each data to be transmitted.

Clearing the TDRE bit is always performed by the following software sequence:
1. An access to the SCISR register
2. A write to the SCIDR register

The TDRE bit is set by hardware and it indicates:

– The TDR register is empty.

– The data transfer is beginning.

– The next data can be written in the SCIDR register without overwriting the previous data.

This flag generates an interrupt if the TIE bit is set and the I[1:0] bits are cleared in the CCR register.

When a transmission is taking place, a write instruction to the SCIDR register stores the data in the TDR register and which is copied in the shift register at the end of the current transmission.

When no transmission is taking place, a write instruction to the SCIDR register places the data directly in the shift register, the data transmission starts, and the TDRE bit is immediately set.

When a character transmission is complete (after the stop bit) the TC bit is set and an interrupt is generated if the TCIE is set and the I[1:0] bits are cleared in the CCR register.

Clearing the TC bit is performed by the following software sequence:
1. An access to the SCISR register
2. A write to the SCIDR register

**Note:** The TDRE and TC bits are cleared by the same software sequence.

**Break Characters**

Setting the SBK bit loads the shift register with a break character. The break character length depends on the M bit (see Figure 63).

As long as the SBK bit is set, the SCI sends break characters to the TDO pin. After clearing this bit by software, the SCI inserts a logic 1 bit at the end of the last break character to guarantee the recognition of the start bit of the next character.

**Idle Line**

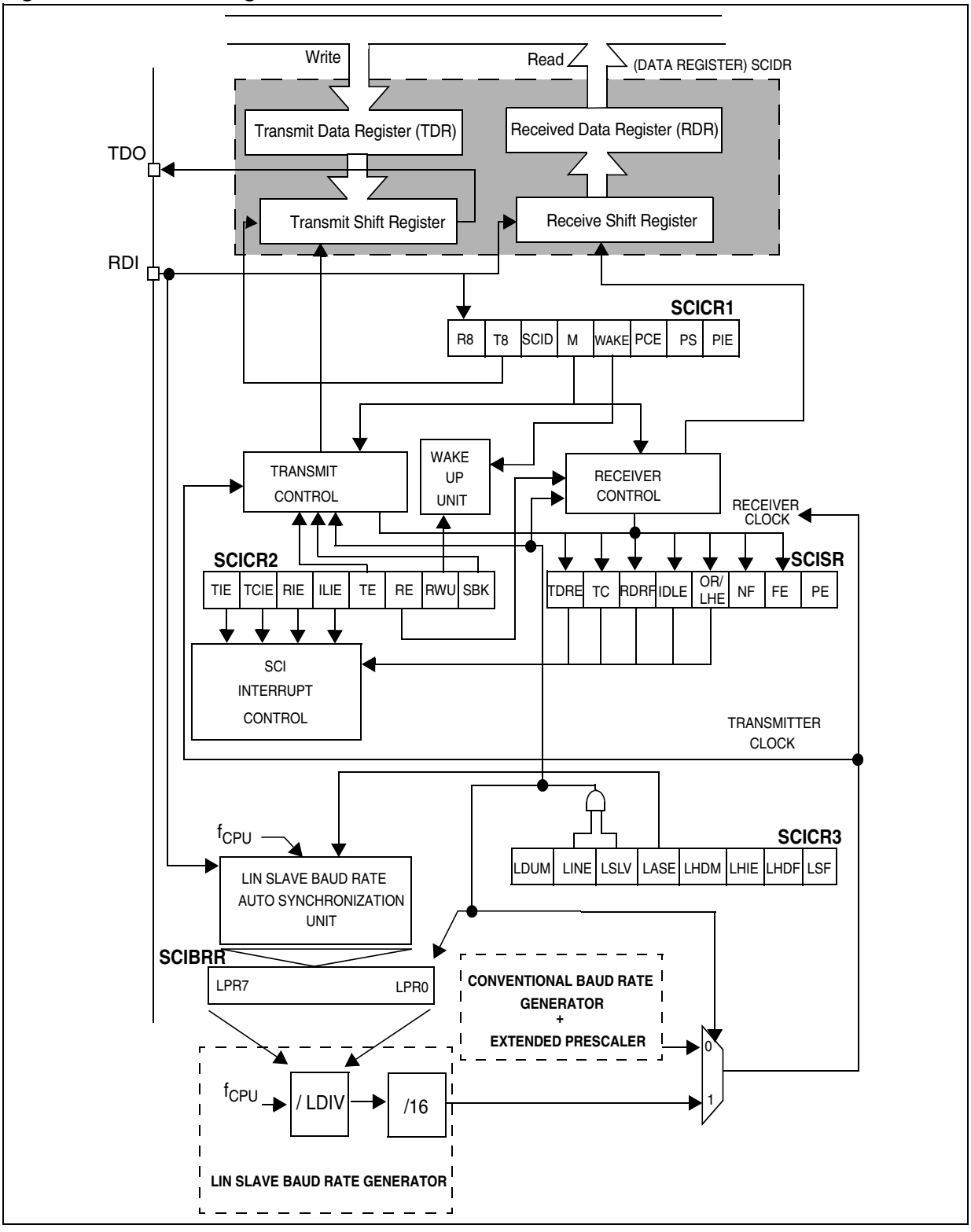
Setting the TE bit drives the SCI to send a preamble of 10 (M = 0) or 11 (M = 1) consecutive '1's (idle line) before the first character.

In this case, clearing and then setting the TE bit during a transmission sends a preamble (idle line) after the current word. Note that the preamble duration (10 or 11 consecutive '1's depending on the M bit) does not take into account the stop bit of the previous character.

**Note:** Resetting and setting the TE bit causes the data in the TDR register to be lost. Therefore the best time to toggle the TE bit is when the TDRE bit is set, that is, before writing the next byte in the SCIDR.

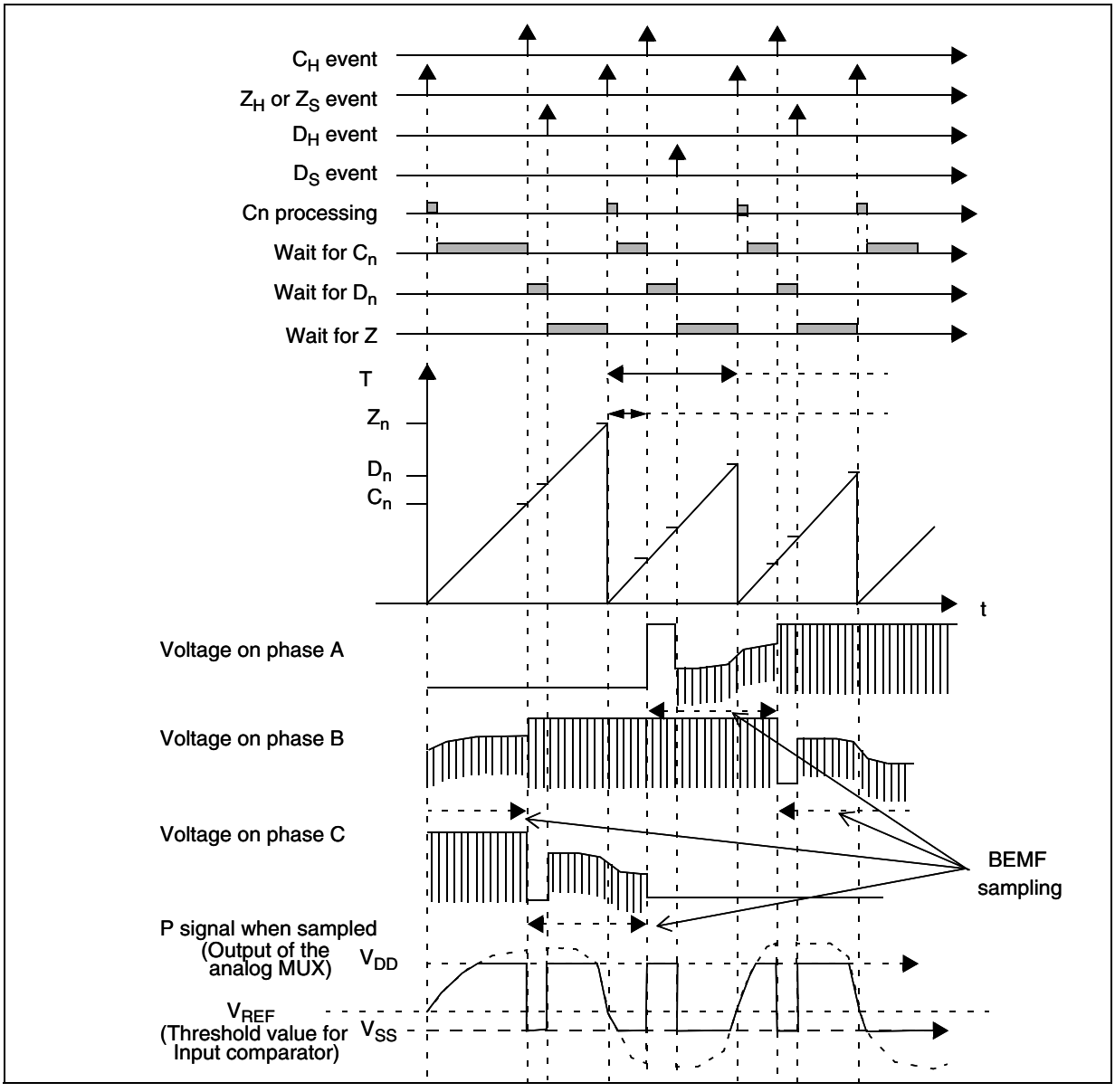
**LINSCI™ SERIAL COMMUNICATION INTERFACE (LIN Mode)** (cont'd)

**Figure 66. SCI Block Diagram in LIN Slave Mode**



**MOTOR CONTROLLER** (Cont'd)

**Figure 73. Chronogram of Events (in Autoswitched Mode)**



**MOTOR CONTROLLER** (Cont'd)

#### 10.6.6.2 Sensorless Mode

This mode is used to detect BEMF zero crossing and end of demagnetization events.

The analog phase multiplexer connects the non-excited motor winding to an analog 100mV hysteresis comparator referred to a selectable reference voltage.

IS[1:0] bits in MPHST register allow to select the input which will be drive to the comparator (either MCIA, B or C). Be careful that the comparator is OFF until CKE and/or DAC bit are set in MCRA register.

The VR[2:0] bits in the MCRC register select the reference voltage from seven internal values depending on the noise level and the application voltage supply. The reference voltage can also be set externally through the MCVREF pin when the VR[2:0] bits are set.

**Table 25. Threshold voltage setting**

| VR2 | VR1 | VR0 | Vref voltage threshold |
|---|---|---|---|
| 1 | 1 | 1 | Threshold voltage set by external MCVREF pin |
| 1 | 1 | 0 | 3.5V* |
| 1 | 0 | 1 | 2.5V* |
| 1 | 0 | 0 | 2V* |
| 0 | 1 | 1 | 1.5V* |
| 0 | 1 | 0 | 1V* |
| 0 | 0 | 1 | 0.6V* |
| 0 | 0 | 0 | 0.2V* |

*Typical value for $V_{DD}$=5V.

BEMF detections are performed during the measurement window, when the excited windings are free-wheeling through the low side switches and diodes. At this stage the common star connection voltage is near to ground voltage (instead of $V_{DD}/2$ when the excited windings are powered) and the complete BEMF voltage is present on the non-excited winding terminal, referred to the ground terminal.

The zero crossing sampling frequency is then defined, in current mode, by the measurement window generator frequency (SA[3:0] bits in the MPRSR register) or, in voltage mode, by the PWM generator frequency and phase U duty cycle.

During a short period after a phase commutation (C event), the winding where the back-emf will be read is no longer excited but needs a demagnetisation phase during which the BEMF cannot be read. A demagnetization current goes through the free-wheeling diodes and the winding voltage is stuck at the high voltage or to the ground terminal. For this reason an "end of demagnetization event" D must be detected on the winding before the detector can sense a BEMF zero crossing.

For the end-of-demagnetization detection, no special PWM configuration is needed, the comparator sensing is done at a selectable frequency ($f_{SCF}$), see Table 82.

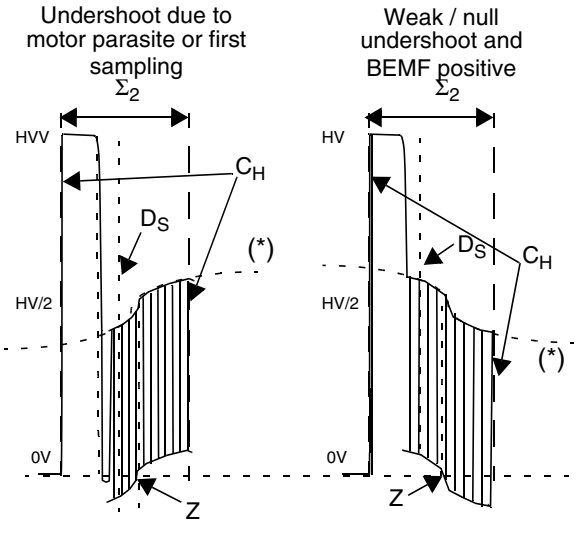
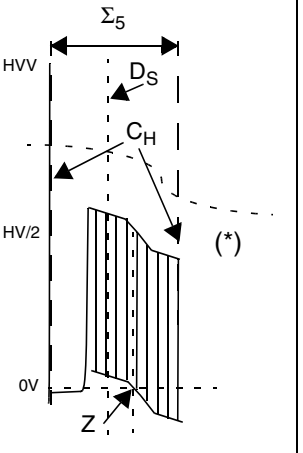
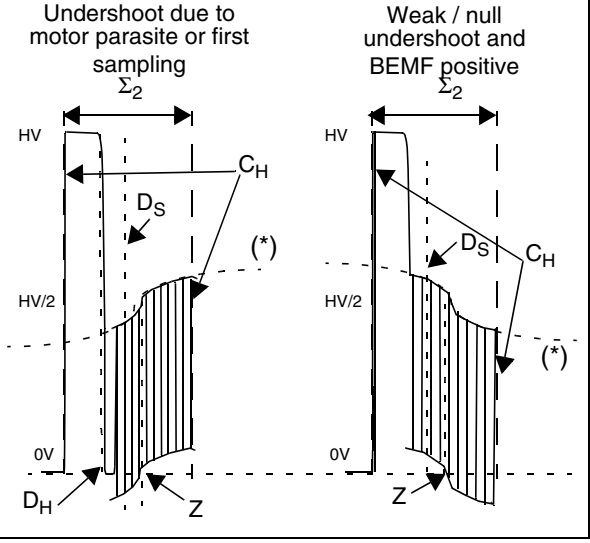
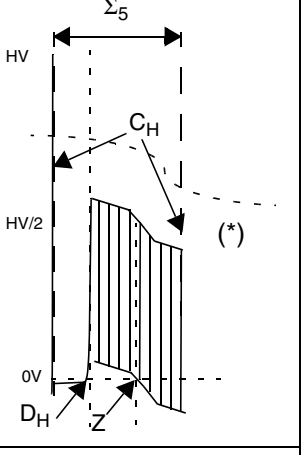
So, the three events: C (commutation), D (demagnetization) and Z (BEMF zero crossing) must always occur in this order in autoswitched mode when hard commutation is selected.

The comparator output is processed by a detector that automatically recognizes the D or Z event, depending on the CPB or ZVD edge and level configuration bits as described in Table 30.

To avoid wrong detection of D and Z events, a blanking window filter is implemented for spike filtering. In addition, by means of an event counter, software can filter several consecutive events up to a programmed limit before generating the D or Z event internally. This is shown in Figure 79 and Figure 80.

**MOTOR CONTROLLER** (Cont'd)

**Table 31. Demagnetisation (D) Event Generation (example for ZVD=0)**

<table>
<tr><th>HDM bit</th><th>Meaning</th><th>CPB bit = 1</th><th>CPB bit = 0</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Simulated Mode<br>(SDM bit =1 and HDM bit = 0)</td><td colspan="2">D = $D_S$ = Output Compare [MDREG, MTIM registers]</td></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td>1</td><td>Hardware/Simulated Mode<br>(SDM bit = 1 and HDM bit = 1)</td><td>D = $D_H$ + $D_S$<br>(Hardware detection or Output compare true)<br> 
</td><td>D = $D_H$<br>(Hardware detection only)<br> 
</td></tr>
<tr><td colspan="4">(*) Note: This is a zoom to the additional voltage induced by the rotor (Back EMF)</td></tr>
</table>

**MOTOR CONTROLLER** (Cont'd)

### 10.6.6.7 Protection for $Z_H$ event detection

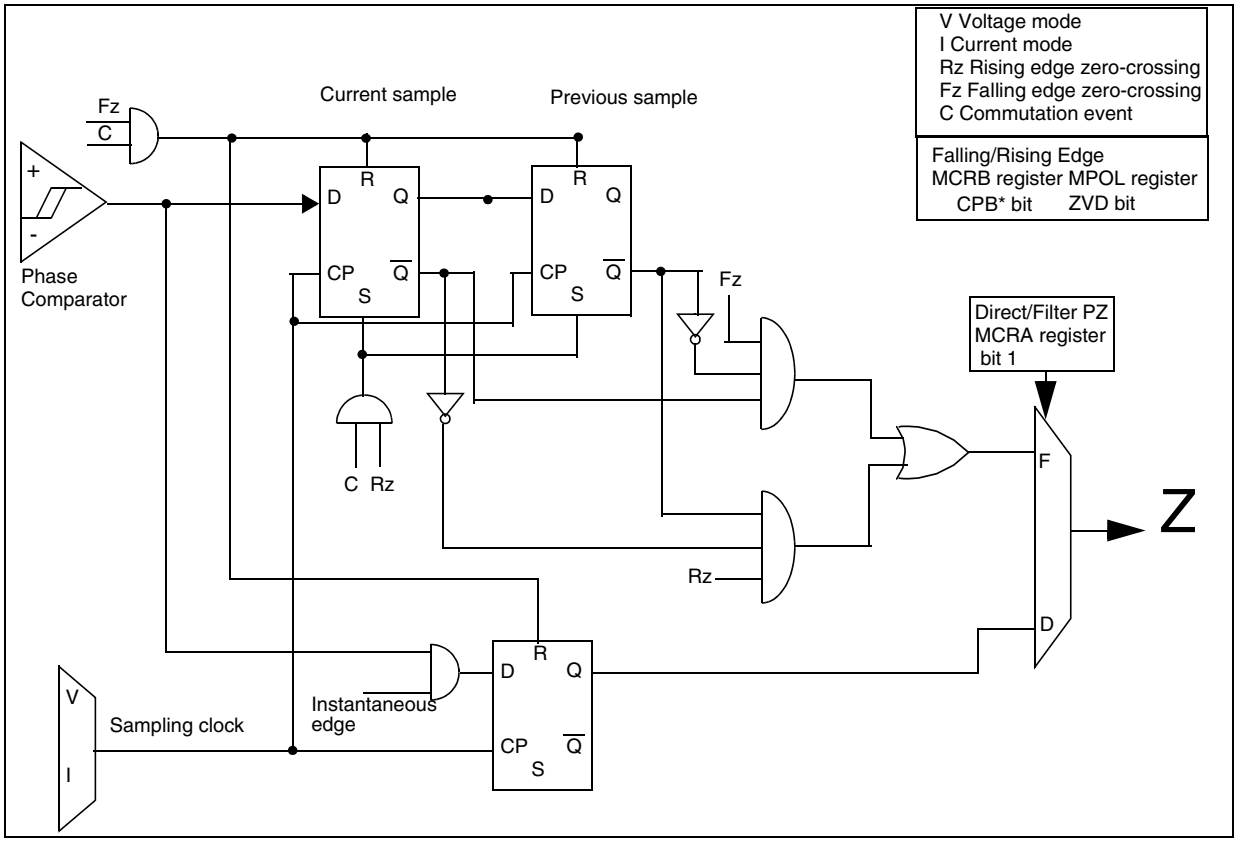
To avoid an erroneous detection of a hardware zero-crossing event, a filter can be enabled by setting the PZ bit in the MCRA register. This filter will ensure the detection of a $Z_H$ event on an edge transition between D event and $Z_H$ event.

Without this protection, $Z_H$ event detection is done directly on the current sample in comparison with the expected state at the output of the phase comparator. For example, if a falling edge transition (meaning a transition from 1 to 0 at the output of the phase comparator) is configured for $Z_H$ event through the CPB bit in MCRB register, then, the state 0 is expected at the comparator output and once this state is detected, the $Z_H$ event is generated without any verification that the state at the comparator output of the previous sample was 1. The purpose of this protection filter is to be sure that the state of the comparator output at the sample before was really the opposite of the current state which is generating the $Z_H$ event. With this filter, the $Z_H$ event generation is done on edge transition level comparison.

This filter is not needed in sensor mode (SR=1) and for simulated zero-crossing event ($Z_S$) generation.

When the PZ bit is set, the Z event filter ZEF[3:0] in the MZFR register is ignored.

**Figure 83. Protection of $Z_H$ event detection**



**MOTOR CONTROLLER** (Cont'd)

#### 10.6.8.2 Over Current Handling in Voltage mode

When the current limitation interrupt is enabled by setting the CLIM bit in the MIMR register (available only in Voltage mode), the OCV bit in MCRB register will determine the effect of this interrupt on the MCOx outputs as shown in Table 43.

**Table 43. OCV bit effect**

| CLIM bit | CLI bit | OCV bit | Output effect | Interrupt |
|---|---|---|---|---|
| 0 | 0 | x | Normal running mode | No |
| 0 | 1 | x | PWM is put OFF on Current loop effect | No |
| 1 | 0 | x | Normal running mode | No |
| 1 | 1 | 0 | PWM is put OFF on Current loop effect | Yes |
| 1 | 1 | 1 | All MCOx outputs are put in reset state (MOE reset)[1)] | Yes |

For safety purposes, it can be necessary to put all MCOx outputs in reset state (high impedance, high state or low state depending on the setting made by the option byte) on a current limitation interrupt. This is the purpose of the OCV bit. When a current limitation interrupt occurs, if the OCV bit is reset, the effect on the MCOx outputs is only to put the PWM signal OFF on the concerned outputs. If the OCV bit is set, when the current limitation interrupt occurs, all the MCOx outputs are put in reset state.

**Note 1:** Only this functionality (CLIM = CLI = OCV = 1) is valid when the 3 PWM channels are enabled (PCN bit =1 in the MDTG register). It can also be used as an over-current protection for three-phase PWM application (only if voltage mode is selected)

#### 10.6.8.3 Current Mode

In current mode, the PWM output signal is generated by a combination of the output of the measurement window generator (see Figure 107) and the output of the current comparator, and is directed to the output channel manager as well (Figure 108).

The current reference is provided to the comparator by Phase U, V or W of the PWM Generator (up to 12-bit accuracy) the signal from the three compare registers U, V or W can be output by setting the PWMU, PWMV or PWMW bits in the MPWME register. The PWM signal is filtered through an external RC filter on pin MCCREF.

The detected current input must be present on the MCCFI pin.

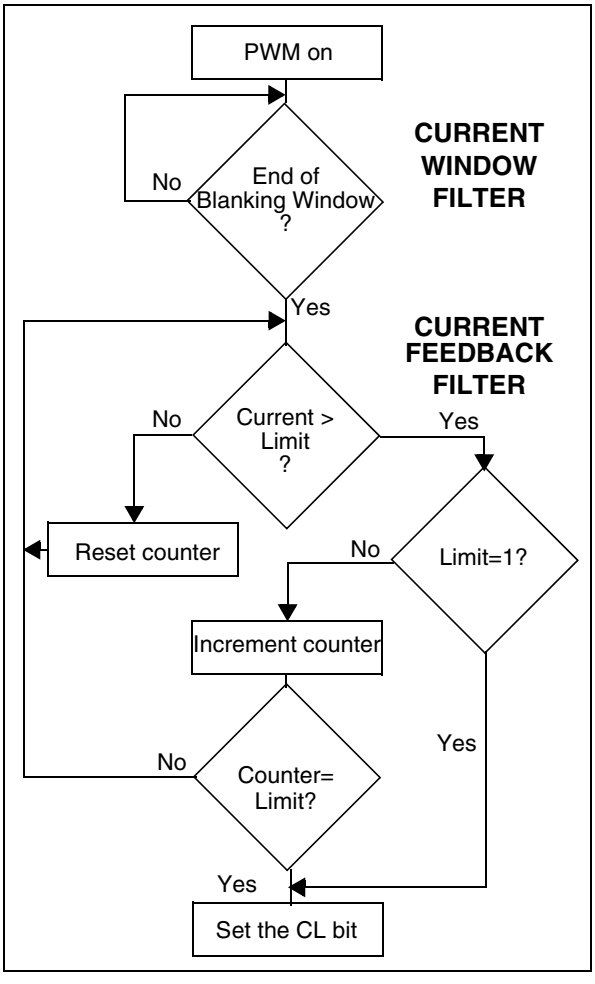
#### 10.6.8.4 Current Feedback Comparator

Two programmable filters are implemented:

– A blanking window ( Current Window Filter) after PWM has been switched ON to avoid spurious PWM OFF states caused by parasitic noise

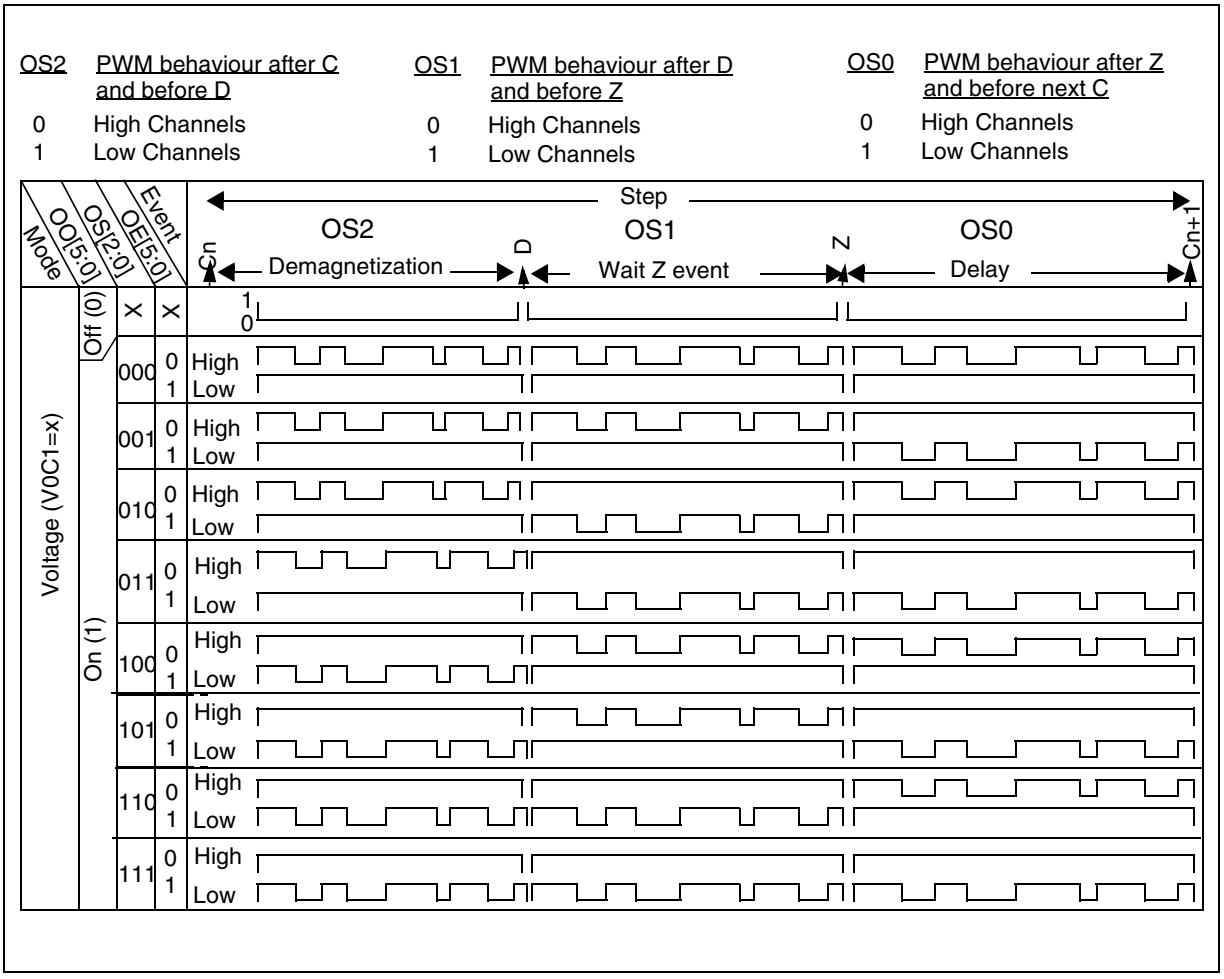
– An event counter (Current Feedback Filter) to prevent PWM being turned OFF when the first comparator edge is detected.

**Figure 105. Current Window and Feedback Filters**



**MOTOR CONTROLLER** (Cont'd)

**Figure 109. PWM application in Voltage or Current sensorless mode (see Table 61)**



**MOTOR CONTROLLER** (Cont'd)

If the PCN bit is reset, one of the three PWM signals (the one set by the compare U register pair) or the output of the measurement window generator (depending on if the driving mode is voltage or current) is used to provide six-step signals through the PWM manager (to drive a PM BLDC motor for instance).

In that case, DTE behaves like a standard bit (with multiple write capability). When the deadtime generator is enabled (bit DTE=1), some restrictions are applied, summarized in Table 53:

- Channels are now grouped by pairs: Channel[0:1], Channel[2:3], Channel[4:5]; a deadtime generator is allocated to each of these pairs (see cautions below);
- The input signal of the deadtime generator is the active output of the PWM manager for the corresponding channel. For instance, if we consider the Channel[0:1] pair, it may be either Channel0 or Channel1.
- When both channels of a pair are inactive, the corresponding outputs will also stay inactive (this is mandatory to allow BEMF zero-crossing detection).

Table 53 summarizes the functionality of the deadtime generator when the PCN bit is reset. 1(pwm*) means that the corresponding channel is active (1 in the corresponding bit in the MPHST register), and a PWM signal is applied on it (using the MPAR register and the OS[2:0] bits in MCRB register). $\overline{PWM}$ represents the complementary signals (although the duty cycle is slightly different due to deadtime insertion). 0 means that the channel is inactive and 1 means that the channel is active and a logic level 1 is applied on it (no PWM signal).

**Table 53. Dead Time generator outputs**

| PCN = 0; DTE =1; x= 0, 2, 4 | | | |
|---|---|---|---|
| **On/Off x (OOx bit)** | **On/Off x+1 (OOx+1 bit)** | **MCOx output** | **MCOx+1 output** |
| 0 | 1 (pwm*) | $\overline{PWM}$ | PWM |
| 1 (pwm*) | 0 | PWM | $\overline{PWM}$ |
| 1 | 1 (pwm*) | 0 | 0 |
| 1 (pwm*) | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 0 |

* PWM generation enabled

**Warning:** Grouping channels by pairs imposes the external connections between the MCO outputs and power devices; the user must therefore pay attention to respect the "recommended schematics" described in Figure 123. on page 228 and Figure 124

**Note:** As soon as the channels are grouped in pairs, special care has to be taken in configuring the MPAR register for a PM BLDC drive. If both channels of the same pair are both labelled "high" for example and if the PWM is applied on high channels, the active MCO output x (OOx=1 bit in the MPHST register) outputs PWM and the paired MCO output x+1 (OOx+1bit in the MPHST register) outputs $\overline{PWM}$ and vice versa.

**Caution:** When PCN=0 and a complementary PWM is applied (DTE=1) on one channel of a pair, if both channels are active, this corresponds in output to both channels OFF. This is for security purpose to avoid cross-conduction.

**Caution:** To clear the DTE bit from reset state of MDTG register (FFh), the PCN bit must be cleared before.

**MOTOR CONTROLLER** (Cont'd)

**Effect on PWM generator**: the PWM generator 12-bit counter is reset as soon as CKE = 0; this ensures that the PWM signals start properly in all cases. When these bits are set, all registers with preload on Update event are transferred to active registers.

Bit 5 = **SR**: *Sensor ON/OFF.*
0: Sensorless mode
1: Position Sensor mode

**Table 57. Sensor Mode Selection**

| SR bit | Mode | OS[2:0] bits | Behaviour of the output PWM |
|---|---|---|---|
| 0 | Sensors not used | OS[2:0] bits enabled | "Between $C_n$&D" behaviour, "between D&Z" behaviour and "between Z&$C_{n+1}$" behaviour |
| 1 | Sensors used | OS1 disabled | "Between $C_n$&Z" behaviour and "between Z&$C_{n+1}$" behaviour |

See also Table 61 and Table 62

Bit 4 = **DAC**: *Direct Access to phase state register.*
0: No Direct Access (reset value). In this mode the preload value of the MPHST and MCRB registers is taken into account at the C event.
1: Direct Access enabled. In this mode, write a value in the MPHST register to access the outputs directly.

**Note:** In Direct Access Mode (DAC bit is set in MCRA register), a C event is generated as soon as there is a write access to the OO[5:0] bits in MPHST register. In this case, the PWM low/high selection is done by the OS0 bit in the MCRB register.

**Table 58. DAC Bit Meaning**

| MOE bit | DAC bit | Effect on Output |
|---|---|---|
| 0 | x | Reset state depending on the option bit |
| 1 | 0 | Standard running mode. |
| 1 | 1 | MPHST register value (depending on MPOL, MPAR register values and PWM setting) see Table 74 |

Bit 3 = **V0C1**: *Voltage/Current Mode*
0: Voltage Mode
1: Current Mode

Bit 2 = **SWA**: *Switched/Autoswitched Mode*
0: Switched Mode
1: Autoswitched Mode

**Note 1** : after reset, in autoswitched mode (SWA =1) , the motor control peripheral is waiting for a C commutation event.

**Note 2:** After reset, a C event is immediately generated when CKE and SWA are simultaneaously set due to a nil value of MCOMP.

Bit 1 = **PZ**: *Protection from parasitic Zero-crossing event detection*
0: Protection disabled
1: Protection enabled

**Note:** If the PZ bit is set, the Z event filter (ZEF[3:0] in the MZFR register is ignored.

Bit 0 = **DCB**: *Data Capture bit*
0: Use MZPRV ($Z_N$-1) for multiplication
1: Use MZREG ($Z_N$) for multiplication

**Table 59. Multiplier Result**

| DCB bit | Commutation Delay |
|---|---|
| 0 | MCOMP = MWGHT x MZPRV / 256 |
| 1 | MCOMP = MWGHT x MZREG / 256 |

**MOTOR CONTROLLER** (Cont'd)

**MOTOR CURRENT FEEDBACK REGISTER (MCFR)**
Read/Write
Reset Value: 0000 0000 (00h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RPGS | RST | CFF2 | CFF1 | CFF0 | CFW2 | CFW1 | CFW0 |

Bit 7= **RPGS:** *Register Page Selection:*
0: Access to registers mapped in page 0
1: Access to registers mapped in page 1

Bit 6= **RST**: *Reset MTC registers.*
Software can set this bit to reset all MTC registers without resetting the ST7.
0: No MTC register reset
1: Reset all MTC registers

Bits 5:3 = **CFF[2:0]**: *Current Feedback Filter bits*
These bits select the number of consecutive valid samples (when the current is above the limit) needed to generate the active event. Sampling is done at $f_{PERIPH}/4$.

**Table 67. Current Feedback Filter Setting**

| CFF2 | CFF1 | CFF0 | Current Feedback Samples |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 2 |
| 0 | 1 | 0 | 3 |
| 0 | 1 | 1 | 4 |
| 1 | 0 | 0 | 5 |
| 1 | 0 | 1 | 6 |
| 1 | 1 | 0 | 7 |
| 1 | 1 | 1 | 8 |

Bits 2:0 = **CFW[2:0]**: *Current Window Filter bits:*

These bits select the length of the blanking window activated each time PWM is turned ON. The filter blanks the output of the current comparator.

**Table 68. Current Feedback Window Setting**

| CFW2 | CFW1 | CFW0 | Blanking Window |
|---|---|---|---|
| 0 | 0 | 0 | Blanking window off |
| 0 | 0 | 1 | 0.5µs |
| 0 | 1 | 0 | 1µs |
| 0 | 1 | 1 | 1.5µs |
| 1 | 0 | 0 | 2µs |
| 1 | 0 | 1 | 2.5µs |
| 1 | 1 | 0 | 3µs |
| 1 | 1 | 1 | 3.5µs |

**Note:** Times are indicated for 4 MHz $f_{PERIPH}$

**MOTOR CONTROLLER** (Cont'd)

**PARITY REGISTER (MPAR)**
Read/Write
Reset Value: 0000 0000 (00h)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| TES1 | TES0 | OE5 | OE4 | OE3 | OE2 | OE1 | OE0 |

Bits 7:6 = **TES[1:0]** *: Tacho Edge Selection bits*
The primary function of these bits is to select the edge sensitivity of the tachogenerator capture logic; clearing both TES[1:0] bits specifies that the Input Detection block does not operate in Speed Sensor Mode but either in Position Sensor or Sensorless Mode for a six-step motor drive).

**Table 79. Tacho edges and input mode selection**

| TES 1 | TES 0 | Edge sensitivity | Operating Mode |
|---|---|---|---|
| 0 | 0 | Not applicable | Position Sensor or Sensorless |
| 0 | 1 | Rising edge | Speed Sensor |
| 1 | 0 | Falling edge | Speed Sensor |
| 1 | 1 | Rising and falling edges | Speed Sensor |

Bits 5:0 = **OE[5:0]**: *Output Parity Mode.*
0: Output channel is High
1: Output channel Low

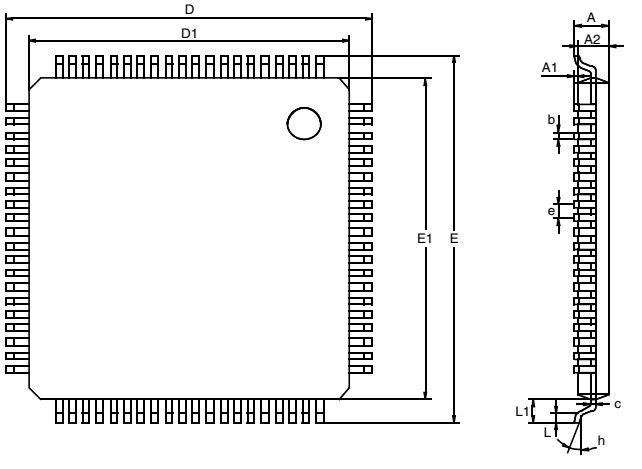
**Note**: These bits are not significant when PCN=1 (configuration with three independent phases).

# 13 PACKAGE CHARACTERISTICS

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK® packages, depending on their level of environmental compliance. ECOPACK® specifications, grade definitions and product status are available at: www.st.com. ECOPACK® is an ST trademark.

## 13.1 PACKAGE MECHANICAL DATA

**Figure 162. 80-Pin Low Profile Quad Flat Package**




| Dim. | mm | | | inches[1] | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| A | | | 1.60 | | | 0.0630 |
| A1 | 0.05 | | 0.15 | 0.0020 | | 0.0059 |
| A2 | 1.35 | 1.40 | 1.45 | 0.0531 | 0.0551 | 0.0571 |
| b | 0.22 | 0.32 | 0.38 | 0.0087 | 0.0126 | 0.0150 |
| C | 0.09 | | 0.20 | 0.0035 | | 0.0079 |
| D | | 16.00 | | | 0.6299 | |
| D1 | | 14.00 | | | 0.5512 | |
| E | | 16.00 | | | 0.6299 | |
| E1 | | 14.00 | | | 0.5512 | |
| e | | 0.65 | | | 0.0256 | |
| θ | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| L | 0.45 | 0.60 | 0.75 | 0.0177 | 0.0236 | 0.0295 |
| L1 | | 1.00 | | | 0.0394 | |
| | Number of Pins | | | | | |
| N | 80 | | | | | |

**ST7MC DEVICE CONFIGURATION AND ORDERING INFORMATION** (Cont'd)

### 14.3.5 PACKAGE/SOCKET FOOTPRINT PROPOSAL

**Table 92. Suggested List of Socket Types**

| Package / Probe | Socket Reference | | Emulator Adapter | |
|---|---|---|---|---|
| LQFP64 14x14 | CAB | 3303262 | CAB | 3303351 |
| LQFP80 14x14 | YAMAICHI | IC149-080-*51-*5 | YAMAICHI | ICP-080-7 |
| LQFP32 7x7 | IRONWOOD | SF-QFE32SA-L-01 | IRONWOOD | SK-UGA06/32A-01 |
| LQFP44 10x10 | YAMAICHI | IC149-044-*52-*5 | YAMAICHI | ICP-044-5 |
| SDIP32 | Standard | | Standard | |
| SDIP56 | Standard | | Standard | |