Ultra Low Power Microcontroller with 4x32 LCD Driver

Features

- **True Low Power** 1.8 µA active mode, LCD On
  0.4 µA standby mode, LCD Off
  0.2 µA sleep mode
  @ 3 V, 32 KHz, 25 ºC
- **Low Supply Voltage 1.2 V to 3.6 V**
- **Melody, 7 tones + silence inclusive 4-bit timer**
- **Universal 10-bit counter, PWM, event counter**

- LCD 32 segments, 3 or 4 times multiplexed
- Temperature compensated LCD voltage levels
- Built-in LCD voltage multipliers
- **LCD frequency 32 Hz/42.7 Hz/64 Hz**
- **32 KHz or 128 kHz crystal oscillator**
- 72 basic instructions
- 2 clocks per instruction cycle
- Mask ROM 4k x 16 bits
- RAM 128 x 4 bits
- Max. 12 inputs ; port A, port B, port SP
- Max. 8 outputs ; port B, port SP
- Voltage Level Detector (VLD), 8 levels software selectable from 1.2 V up to 4.0 V
- Prescaler down to 1 second (crystal = 32 KHz)
- 1/1000 sec 12 bit binary coded decimal counter with hard or software start/stop function
- 3 wire serial port , 8 bit, master and slave mode
- 5 external interrupts (port A, serial interface)
- 8 internal interrupts (3x prescaler, BCD counter 2x10-bit counter, melody timer, serial interface)
- timer watchdog and oscillation supervisor

Description

The EM6626 is an ultra-low power, low voltage microcontroller with an integrated 3/4 MUX x 32 segments LCD driver and the equivalent of 8kB mask ROM. It features temperature compensated LCD voltage levels, free LCD segment allocation and built-in voltage multipliers. It also has a melody generator, a millisecond counter (BCD) and PWM function. Tools include windows-based simulator and emulator. A flexible MFP version is also available for development stage.
Due to its very low current consumption, the EM6626 is ideal for use in battery-operated and field-powered applications.

Typical Applications

- Household appliance
- Timer / sports timing devices
- Medical devices
- Interactive system with display
- Automotive controls with display
- Measurement equipment
- Bicycle computers
- Safety and security devices

Figure 1. Architecture

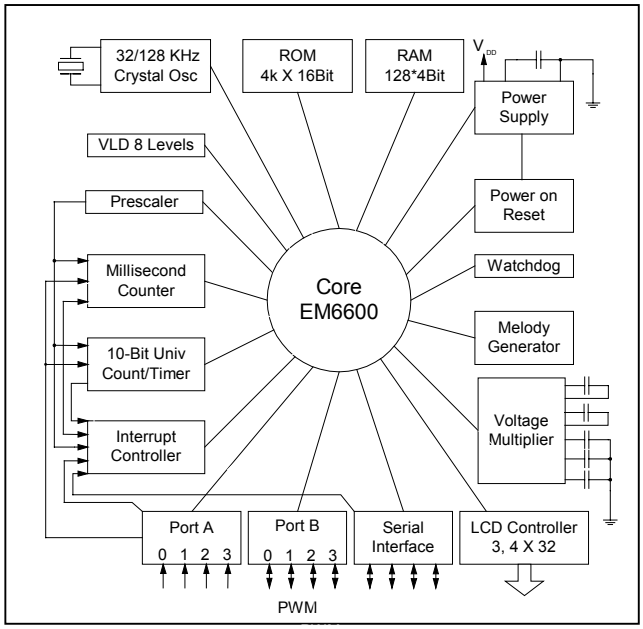
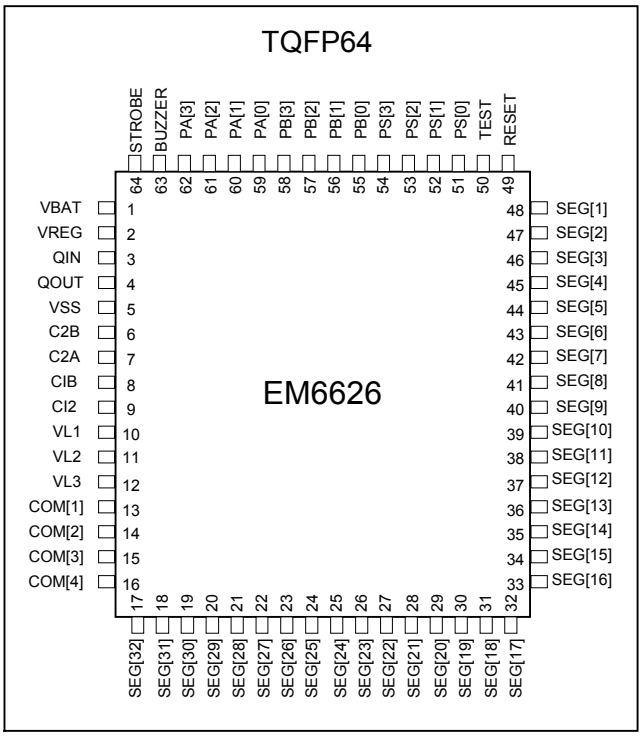


Figure 2. Pin Configuration, TQFP64



EM6626 at a glance

❑ Power Supply
- Low voltage low power architecture including internal voltage regulator
- 1.2 V to 3.6 V battery voltage
- 1.8 µA in active mode (Xtal, LCD on, 25 °C)
- 0.4 µA in standby mode (Xtal, LCD off, 25 °C)
- 0.2 µA in sleep mode (25 °C)
- **32 KHz/128 kHz Oscillator (metal option)**

❑ RAM
- 64 x 4 bit, direct addressable
- 64 x 4 bit, indexed addressable

❑ ROM
- 4k x 16 bit (~8k Byte), metal mask programmable

❑ CPU
- 4-bit RISC architecture
- 2 clock cycles per instruction
- 72 basic instructions

❑ Main Operating Modes and Resets
- Active mode (CPU is running)
- Standby mode (CPU in halt)
- Sleep mode (no clock, reset state)
- Initial reset on power on (POR)
- Watchdog reset (logic and oscillation watchdogs)
- Reset terminal
- Reset with input combination on port A (register selectable)

❑ Prescaler
- 15 stage system clock divider down to 1Hz
- 3 Interrupt requests; 1Hz, 32Hz or 8Hz, Blink
- Prescaler reset (4kHz to 1Hz)

❑ Liquid Crystal Display Driver (LCD)
- 32 Segments 3 or 4 times multiplexed
- Internal or external voltage multiplier
- Free Segment allocation architecture (metal option)
- LCD switch off for power save
- **LCD frequency 32 Hz/42.7 Hz/64 Hz**

❑ 8-Bit Serial Interface
- 3 wire (Clock, DataIn , DataOut) master/slave mode
- READY output during data transfer
- Maximum shift clock is equal to the main system clock
- Interrupt request to the CPU after 8 bits data transfer
- Supports different serial formats
- Can be configured as a parallel 4 bit input/output port
- Direct input read on the port terminals
- All outputs can be put tristate (default)
- Selectable pull-downs in input mode
- CMOS or Nch. open drain outputs
- Weak pull-up selectable in Nch. open drain mode

❑ 4-Bit Input Port A
- Direct input read on the port terminals
- Debouncer function available on all inputs
- Interrupt request on positive or negative edge
- Pull-up or pull-down or none selectable by register
- Test variables (software) for conditional jumps
- PA[0] and PA[3] are inputs for the event counter
- PA[3] is Start/Stop input for the millisecond counter
- Reset with input combination (register selectable)

❑ 4-Bit Bi-directional Port B
- All different functions bit-wise selectable
- Direct input read on the port terminals
- Data output latches
- CMOS or Nch. open drain outputs
- Pull-down or pull-up selectable
- Weak pull-up in Nch. open drain mode
- Selectable PWM, 32kHz, 1kHz and 1Hz output

❑ Melody Generator
- Dedicated Buzzer terminal
- 7 tones plus silence output
- The output can be put tristate (default)
- Internal 4-bit timer, usable also in standalone mode
- 4 different timer input clocks
- Timer with automatic reload or single run
- Timer interrupt request when reaching 0

❑ Voltage Level Detector (SVLD)
- 8 different levels from 1.2 V to 4.0 V.
- Busy flag during measure

❑ 10-Bit Universal Counter
- 10, 8, 6 or 4 bit up/down counting
- Parallel load
- Event counting (PA[0] or PA[3])
- 8 different input clocks-
- Full 10 bit or limited (8, 6, 4 bit) compare function
- 2 interrupt requests (on compare and on 0)
- Hi-frequency input on PA[3] and PA[0]
- Pulse width modulation (**PWM**) output

❑ Millisecond Counter
- 3 digits binary coded decimal counter (12 bits)
- PA[3] input signal pulse width and period measurement
- Internal 1000 Hz clock generation
- Hardware or software controlled start stop mode
- Interrupt request on either 1/10 Sec or 1Sec

❑ Interrupt Controller
- 5 external and 8 internal interrupt request sources
- Each interrupt request can individually be masked
- Each interrupt flag can individually be reset
- Automatic reset of each interrupt request after read
- General interrupt request to CPU can be disabled
- Automatic enabling of general interrupt request flag when going into HALT mode.

Table of Contents

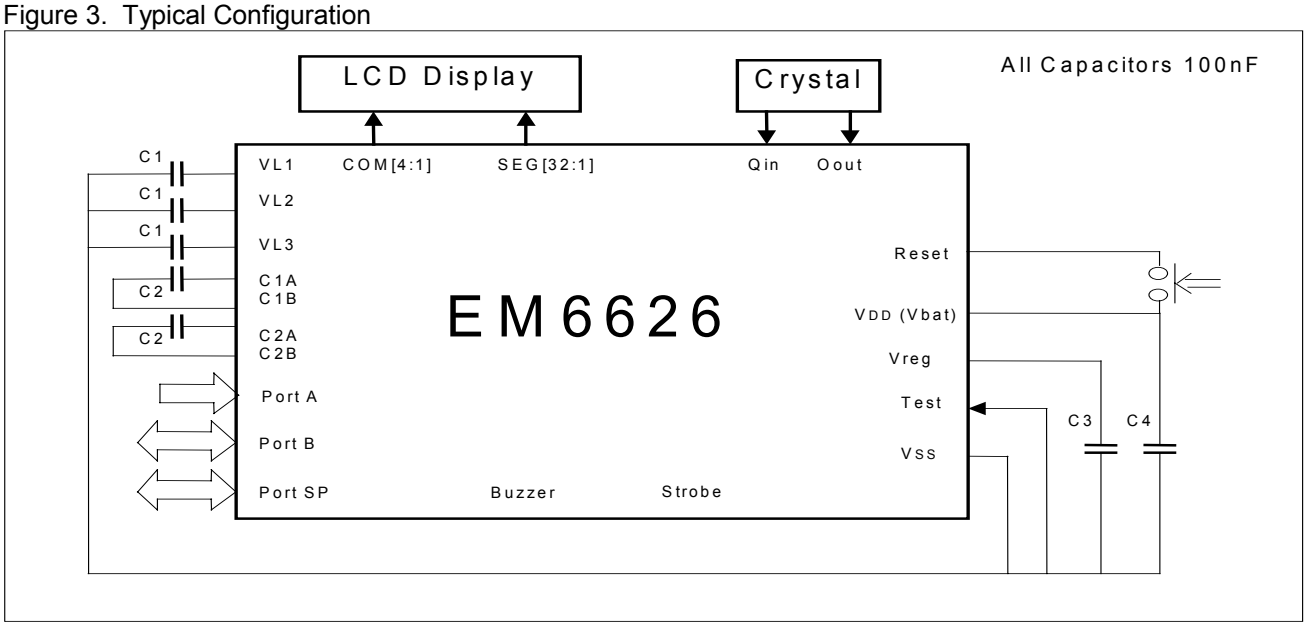
1. Pin Description for EM6626

Chip	TQFP 64	DIL 64	Signal Name	Function	Remarks
1	6	62	C2B	Voltage multiplier	Not needed if ext. supply
2	7	63	C2A	Voltage multiplier	Not needed if ext. supply
3	8	64	C1B	Voltage multiplier	Not needed if ext. supply
4	9	1	C1A	Voltage multiplier	Not needed if ext. supply
5	10	2	VL1	Voltage multiplier level 1	LCD level 1 input, if external supply selected
6	11	3	VL2	Voltage multiplier level 2	LCD level 2 input, if external supply selected
7	12	4	VL3	Voltage multiplier level 3	LCD level 3 input, if external supply selected
8	13	5	COM[1]	LCD back plane 1	
9	14	6	COM[2]	LCD back plane 2	
10	15	7	COM[3]	LCD back plane 3	
11	16	8	COM[4]	LCD back plane 4	Not used if 3 times multiplexed
12	17	9	SEG[32]	LCD Segment 32	
13	18	10	SEG[31]	LCD Segment 31	
14	19	11	SEG[30]	LCD Segment 30	
15	20	12	SEG[29]	LCD Segment 29	
16	21	13	SEG[28]	LCD Segment 28	
17	22	14	SEG[27]	LCD Segment 27	
18	23	15	SEG[26]	LCD Segment 26	
19	24	16	SEG[25]	LCD Segment 25	
20	25	17	SEG[24]	LCD Segment 24	
21	26	18	SEG[23]	LCD Segment 23	
22	27	19	SEG[22]	LCD Segment 22	
23	28	20	SEG[21]	LCD Segment 21	
24	29	21	SEG[20]	LCD Segment 20	
25	30	22	SEG[19]	LCD Segment 19	
26	31	23	SEG[18]	LCD Segment 18	
27	32	24	SEG[17]	LCD Segment 17	
28	33	25	SEG[16]	LCD Segment 16	
29	34	26	SEG[15]	LCD Segment 15	
30	35	27	SEG[14]	LCD Segment 14	
31	36	28	SEG[13]	LCD Segment 13	
32	37	29	SEG[12]	LCD Segment 12	
33	38	30	SEG[11]	LCD Segment 11	
34	39	31	SEG[10]	LCD Segment 10	
35	40	32	SEG[9]	LCD Segment 9	
36	41	33	SEG[8]	LCD Segment 8	
37	42	34	SEG[7]	LCD Segment 7	
38	43	35	SEG[6]	LCD Segment 6	
39	44	36	SEG[5]	LCD Segment 5	
40	45	37	SEG[4]	LCD Segment 4	
41	46	38	SEG[3]	LCD Segment 3	
42	47	39	SEG[2]	LCD Segment 2	
43	48	40	SEG[1]	LCD Segment 1	
44	49	41	Reset	Input reset terminal, internal pull-down 15 KOhm	Main reset
45	50	42	Test	Input test terminal, internal pull-down 15 KOhm	For EM tests only, ground 0 ! Except when needed for MFP programming

Chip	TQFP 64	DIL 64	Signal Name	Function	Remarks
46	51	43	PSP[0]	Input/output , open drain serial port : SIN parallel out terminal 0	Serial interface data in or parallel data[0] in/out
47	52	44	PSP[1]	Output , open drain serial port : Ready/CS parallel out terminal 1	Serial interface Ready CS or parallel data[1] in/out
48	53	45	PSP[2]	Output , open drain serial port : SOUT parallel out terminal 2	Serial interface data out or parallel data[2] in/out
49	54	46	PSP[3]	Input/output , open drain serial port : SCLK parallel out terminal 3	Serial interface clock I/O or parallel data[3] in/out
50	55	47	PB[0]	Input/output, open drain port B terminal 0	Port B data[0] I/O or Ck[1] output
51	56	48	PB[1]	Input/output, open drain port B terminal 1	Port B data[1] I/O or Ck[11] output
52	57	49	PB[2]	Input/output, open drain port B terminal 2	Port B data[2] I/O or Ck[16] output
53	58	50	PB[3]	Input/output, open drain port B terminal 3	Port B data[3] I/O or PWM output
54	59	51	PA[0]	Input port A terminal 0	TestVar 1 ; Event counter
55	60	52	PA[1]	Input port A terminal 1	TestVar 2
56	61	53	PA[2]	Input port A terminal 2	TestVar 3
57	62	54	PA[3]	Input port A terminal 3	Event counter, MSC start/stop
58	63	55	Buzzer	Output Buzzer terminal	
59	64	56	Strobe	Output Strobe terminal	µP reset state or/and port B write or sleep flag out
60	1	57	Vbat = VDD	Positive power supply	MFP Connection
61	2	58	Vreg	Internal voltage regulator	Connect to minimum 100nF, MFP connection
62	3	59	Qin/Osc1	Crystal terminal 1	32 or 128KHz crystal, MFP connection
63	4	60	Qout /Osc2	Crystal terminal 2	32 or 128KHz crystal, MFP connection
64	5	61	VSS	Negative power supply	ref. terminal, MFP connection

Gray shaded areas : Terminals needed for MFP programming connections (VDD, Vreg, Qin, Qout, Test).

Figure 3. Typical Configuration



2. Operating Modes

The EM6626 has two low power dissipation modes, standby and sleep. Figure 4 is a transition diagram for these modes.

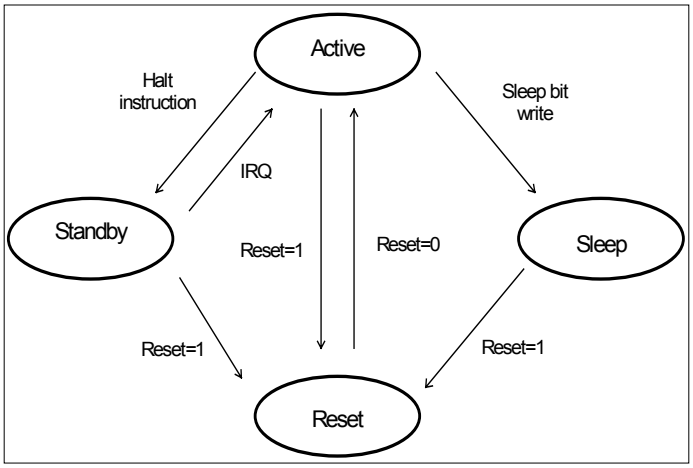
2.1 Active Mode

The active mode is the actual CPU running mode. Instructions are read from the internal ROM and executed by the CPU. Leaving active mode via the halt instruction to go into standby mode, the **Sleep** bit write to go into Sleep mode or a reset from port A to go into reset mode.

2.2 Standby Mode

Executing a halt instruction puts the EM6626 into standby mode. The voltage regulator, oscillator, watchdog timer, LCD, interrupts, timers and counters are operating. However, the CPU stops since the clock related to instruction execution stops. Registers, RAM and I/O pins retain their states prior to standby mode. Standby is canceled by a reset or an interrupt request if enabled.

Figure 4 Mode transition diagram



2.3 Sleep Mode

Writing to the **Sleep** bit in the **RegSysCntl1** register puts the EM6626 in sleep mode. The oscillator stops and most functions of the EM6626 are inactive. To be able to write to the **Sleep** bit, the **SleepEn** bit in **RegSysCntl2** must first be set to "1". In sleep mode only the voltage regulator and the reset input are active. The RAM data integrity is maintained. Sleep mode may be canceled only by a high level of min 10µs at the EM6626 Reset terminal or by the selected port A input reset combination, if option **InpResSleep** is turned on.

Due to the cold-start characteristics of the oscillator, waking up from sleep mode may take some time to guarantee stable oscillation. During sleep mode and the following start up the EM6626 is in reset state. Waking up from sleep clears the **Sleep** flag but not the **SleepEn** bit. Inspecting the **SleepEn** allows to determine if the EM6626 was powered up (**SleepEn** = "0") or woken up from sleep (**SleepEn** = "1").

Table 2.3.1. Internal State in Standby and Sleep Mode

Function	Standby	Sleep
Oscillator	Active	Stopped
Oscillator Watchdog	Active	Stopped
Instruction Execution	Stopped	Stopped
Interrupt Functions	Active	Stopped
Registers and Flags	Retained	Reset
RAM Data	Retained	Retained
Option Registers	Retained	Retained
Timer & Counter	Active	Reset
Logic Watchdog	Active	Reset
I/O Port B and Serial Port	Active	High Impedance, Pull's as defined in option register
Input Port A	Active	No pull-downs and inputs deactivated except if **InpResSleep** = "1"
LCD	Active	Stopped (display off)
Strobe Output	Active	Active
Buzzer Output	Active	High Impedance
Voltage Level Detector	Finishes ongoing measure, then stop	Stopped
Reset Pin	Active	Active

3. Power Supply

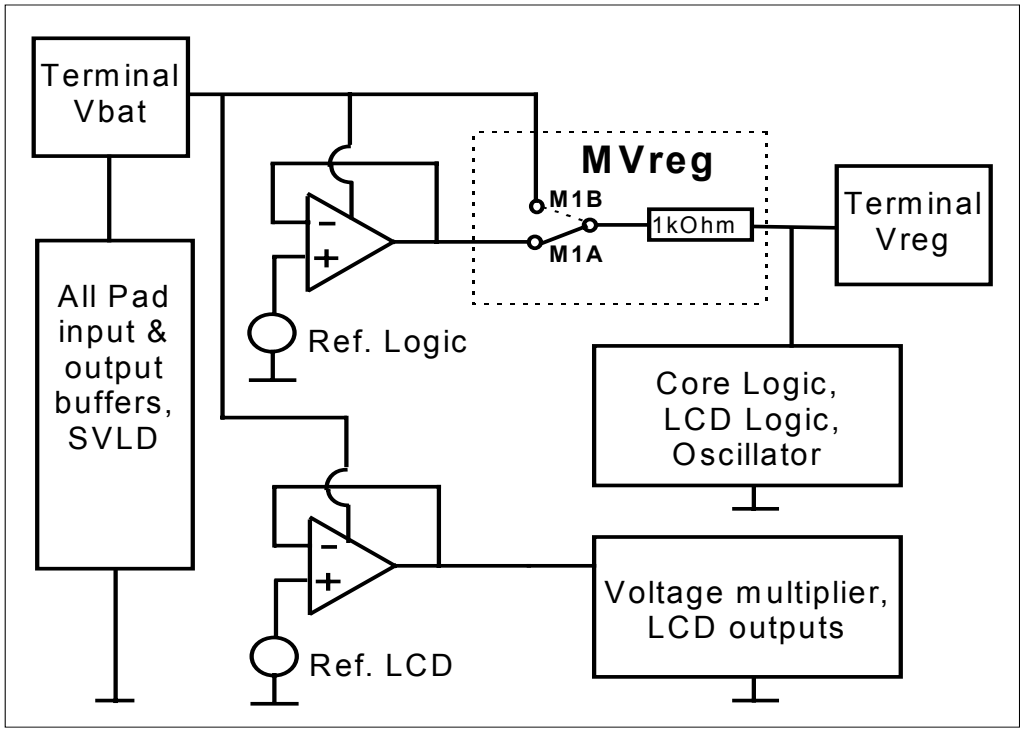
The EM6626 is supplied by a single external power supply between VDD (Vbat) and VSS (Ground). A built-in voltage regulator generates Vreg providing regulated voltage for the oscillator and the internal logic. The output drivers are supplied directly from the external supply VDD. The internal power configuration is shown below in Figure 5.

To supply the internal core logic it is possible to use either the internal voltage regulator (Vreg < VDD) or Vbat directly (Vreg = VDD). The selection is done by metal 1 mask option. By default the voltage regulator is used. Refer to chapter 18.1.5 for the metal mask selection.

The internal voltage regulator is chosen for high voltage systems. It saves power by reducing the internal core logic's power supply to an optimum value. However, due to the inherent voltage drop over the regulator the minimal VDD is restricted to 1.4 V .

A direct Vbat connection can be selected for systems running on a 1.5 V battery. The internal 1 KOhm resistor together with the external capacitor on Vreg is filtering the VDD supply to the internal core. In this case the minimum VDD can be as low as 1.2 V.

Figure 5. Internal Power Supply



4. Reset

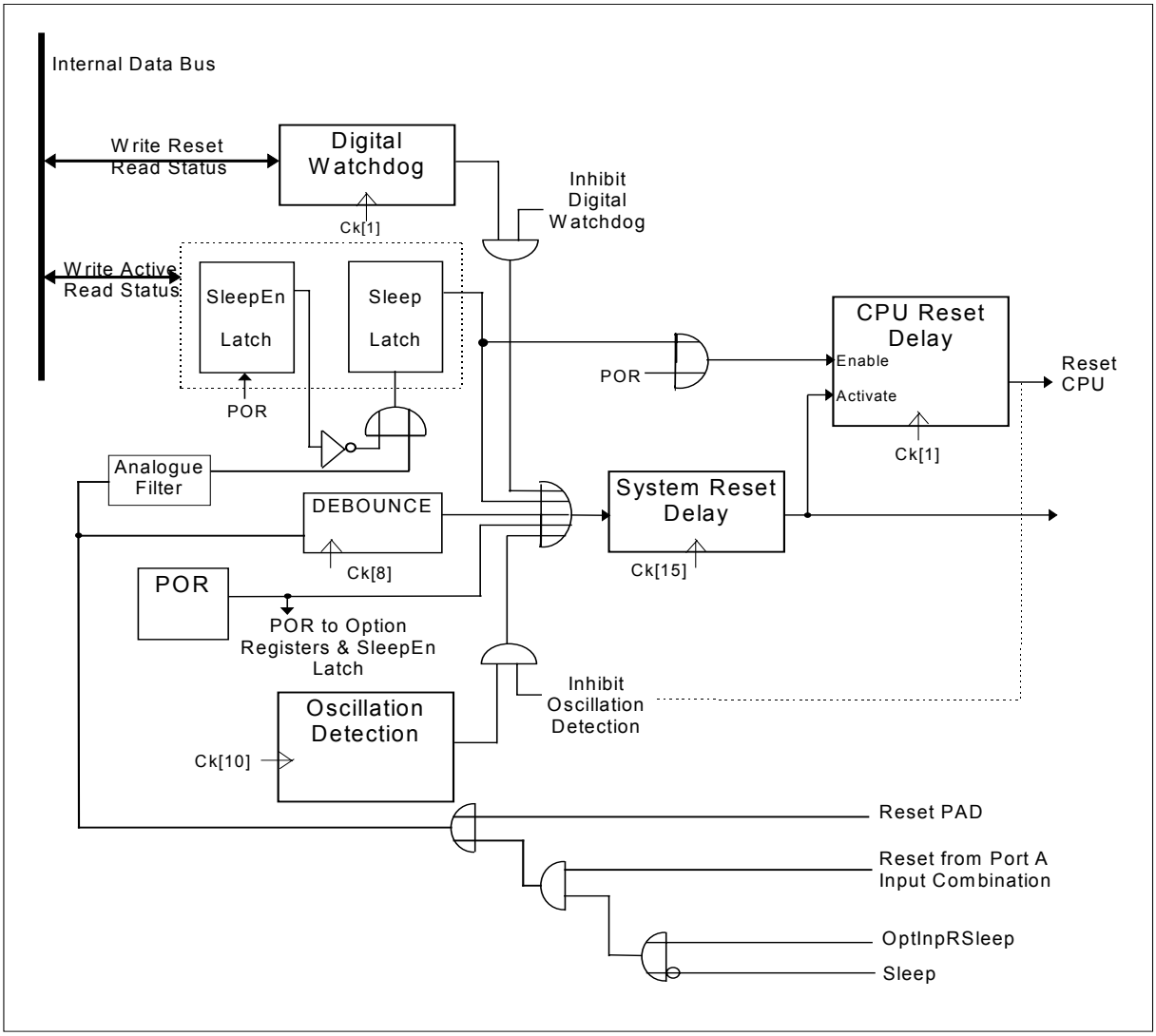
Figure 1 illustrates the reset structure of the EM6626. One can see that there are six possible reset sources :

(1) Internal initial reset from the Power On Reset (POR) circuitry. --> POR
(2) External reset from the Reset terminal. --> System Reset, Reset CPU
(3) External reset by simultaneous high/low inputs to port A. --> System Reset, Reset CPU
(Combinations are defined in the registers **OptInpRSel1** and **OptInpRSel2)**
(4) Internal reset from the Digital Watchdog. --> System Reset, Reset CPU
(5) Internal reset from the Oscillation Detection Circuit. --> System Reset, Reset CPU
(6) Internal reset when sleep mode is activated. --> System Reset, Reset CPU

All reset sources activate the System Reset and the Reset *CPU*. The 'System Reset Delay' ensures that the system reset remains active long enough for all system functions to be reset (active for n system clock cycles). The 'CPU Reset Delay' ensures that the reset CPU remains active until the oscillator is in stable oscillation.

As well as activating the system reset and the reset CPU, the POR also resets all option registers and the sleep enable (**SleepEn)** latch. System reset and reset CPU do not reset the option registers nor the **SleepEn** latch. Reset state can be shown on Strobe terminal by selecting **StrobeOutSel1,0 = 0** in **RegLcdCntl1**.

Figure 1. Reset Structure



4.1 Oscillation Detection Circuit

At power on, the voltage regulator starts to follow the supply voltage and triggers the power on reset circuitry, and thus the system reset. The CPU of the EM6626 remains in the reset state for the 'CPU Reset Delay', to allow the oscillator to stabilize after power up.
The oscillator is disabled during sleep mode. So when waking up from sleep mode, the CPU of the EM6626 remains in the reset state for the CPU Reset Delay, to allow the oscillator to stabilize. During this time, the Oscillation Detection Circuit is inhibited.

In active or standby modes, the oscillator detection circuit monitors the oscillator. If it stops for any reason, a system reset is generated. After clock restart the CPU waits for the CPU Reset Delay before executing the first instructions.
The oscillation detection circuitry can be inhibited with bit **NoOscWD** = 1 in register **RegVldCntl.** At power up, and after any system reset, the function is activated.

The 'CPU Reset Delay' is 32768 system clocks (Ck[16]) long.

4.2 Reset Terminal

During active or standby modes the Reset terminal has a debouncer to reject noise. Reset must therefore be active for at least 16 ms (system clock = 32 KHz).
When canceling sleep mode, the debouncer is not active (no clock), however, reset passes through an analogue filter with a time constant of typical. 5µs. In this case Reset pin must be high for at least 10 µs to generate a system reset.

4.3 Input Port A Reset Function

By writing the **OptInpRSel1** and **OptInpRSel2** registers it is possible to choose any combination of port A input values to execute a system reset. The reset condition must be valid for at least 16ms (system clock = 32kHz) in active and standby mode.

OPTInpRSleep selects the input port A reset function in sleep mode. If set to "1" the occurrence of the selected combination for input port A reset will immediately trigger a system reset (no debouncer) .

Reset combination selection (*InpReset)* is done with registers **OptInpRSel1** and **OptInpRSel2.**
Following formula is applicable :

InpResPA = InpResPA[0] • InpResPA[1] • InpResPA[2] • InpResPA[3]

InpRes1PA[n]	InpRes2PA[n]	InpResPA[n]
0	0	VSS
0	1	PA[n]
1	0	not PA[n]
1	1	VDD

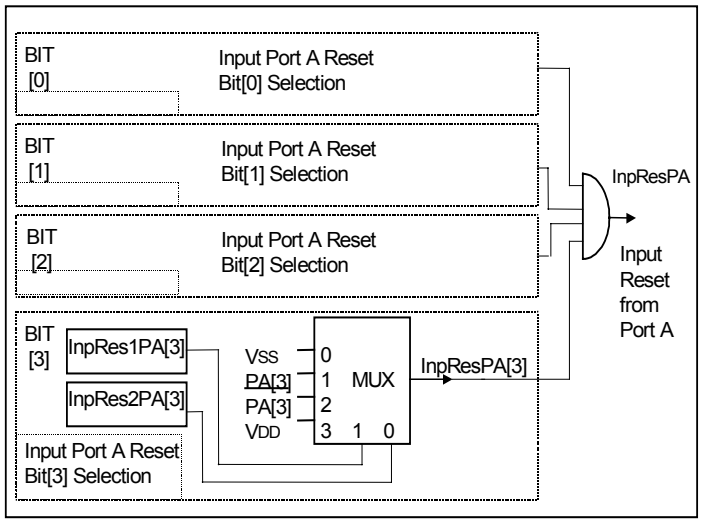
n = 0 to 3

i.e. ; - no reset if InpResPA[n] = VSS.

- Don't care function on a single bit with its InpResPA[n] = VDD.

- Always Reset if InpResPA[3:0] = 'b1111

igure 6. Input Port A Reset Structure



4.4 Digital Watchdog Timer Reset

The digital watchdog is a simple, non-programmable, 2-bit timer, that counts on each rising edge of Ck[1]. It will generate a system reset if it is not periodically cleared. The watchdog timer function can be inhibited by activating an inhibit digital watchdog bit (**NoLogicWD**) located in **RegVldCntl**. At power up, and after any system reset, the watchdog timer is activated.

If for any reason the CPU stops, then the watchdog timer can detect this situation and activate the system reset signal. This function can be used to detect program overrun, endless loops, etc. For normal operation, the watchdog timer must be reset periodically by software at least every 2.5 seconds (system clock = 32 KHz), or a system reset signal is generated.

The watchdog timer is reset by writing a '1' to the **WDReset** bit in the timer. This resets the timer to zero and timer operation restarts immediately. When a '0' is written to **WDReset** there is no effect. The watchdog timer operates also in the standby mode and thus, to avoid a system reset, one should not remain in standby mode for more than 2.5 seconds.

From a system reset state, the watchdog timer will become active after 3.5 seconds. However, if the watchdog timer is influenced from other sources (i.e. prescaler reset), then it could become active after just 2.5 seconds. It is therefore recommended to use the Prescaler **IRQHz1** interrupt to periodically reset the watchdog every second.

It is possible to read the current status of the watchdog timer in **RegSysCntl2**. After watchdog reset, the counting sequence is (on each rising edge of CK[1]) : '00', '01', '10', '11' {**WDVal1 WDVal0**}. When going into the '11' state, the watchdog reset will be active within ½ second. The watchdog reset activates the system reset which in turn resets the watchdog. If the watchdog is inhibited it's timer is reset and therefore always reads '0'.

Table 4.4.1 Watchdog Timer Register RegSysCntl2

Bit	Name	Reset	R/W	Description
3	WDReset	0	R/W	Reset the Watchdog 1 -> Resets the Logic Watchdog 0 -> No action The Read value is always '0'
2	*SleepEn*	*0*	*R/W*	*See Operating modes (sleep)*
1	WDVal1	0	R	Watchdog timer data Ck[1] divided by 4
0	WDVal0	0	R	Watchdog timer data Ck[1] divided by 2

4.5 CPU State after Reset

Reset initializes the CPU as shown in Table 4.5.1 below.

Table 4.5.1 Initial CPU Value after Reset.

Name	Bits	Symbol	Initial Value
Program counter 0	12	PC0	hex 000 (as a result of Jump 0)
Program counter 1	12	PC1	Undefined
Program counter 2	12	PC2	Undefined
Stack pointer	2	SP	PSP[0] selected
Index register	7	IX	Undefined
Carry flag	1	CY	Undefined
Zero flag	1	Z	Undefined
Halt	1	HALT	0
Instruction register	16	IR	Jump 0
Periphery registers	4	Reg.	See peripheral memory map

5. Oscillator and Prescaler

5.1 Oscillator

A built-in crystal oscillator generates the system operating clock for the CPU and peripheral blocks, from an externally connected crystal (typically 32.768kHz or 128kHz: selection done by metal option). The oscillator circuit is supplied by the regulated voltage, Vreg. In sleep mode the oscillator is stopped.
EM's special design techniques guarantee the low current consumption of this oscillator. The external impedance between the oscillator pads must be greater than 10MOhm. Connection of any other components to the two oscillator pads must be confirmed by EM Microelectronic-Marin SA.
If the typical 128kHz option is selected, a special 4 stage frequency divider is added automatically by EM Microelectronics to deliver to the prescaler 32kHz which generates all system clocks except CPU clock frequency to keep the peripheral timing as close as possible to the specification.

5.2 Prescaler

The prescaler consists of fifteen elements divider chain which delivers clock signals for the peripheral circuits such as timer/counter, buzzer, LCD voltage multiplier, debouncer and edge detectors, as well as generating prescaler interrupts. The input to the prescaler is the system clock signal. Power on initializes the prescaler to Hex(0001).

Table 5.2.1 Prescaler Clock Name Definition

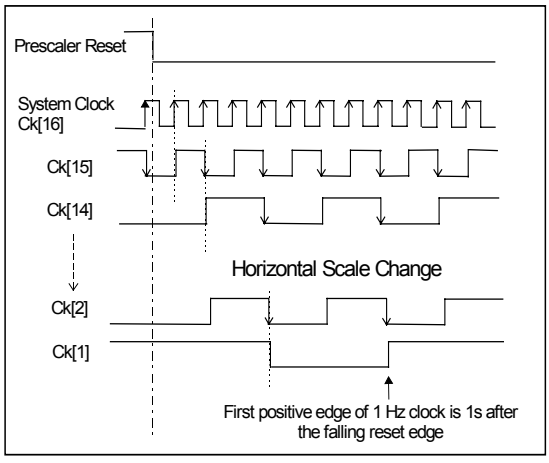
Function	Name	32 KHz Xtal
System clock	Ck[16]	32768 Hz
System clock / 2	Ck[15]	16384 Hz
System clock / 4	Ck[14]	8192 Hz
System clock / 8	Ck[13]	4096 Hz
System clock/ 16	Ck[12]	2048 Hz
System clock / 32	Ck[11]	1024 Hz
System clock / 64	Ck[10]	512 Hz
System clock / 128	ck [9]	256 Hz

Function	Name	32 KHz Xtal
System clock / 256	Ck[8]	128 Hz
System clock / 512	Ck[7]	64 Hz
System clock / 1024	Ck[6]	32 Hz
System clock / 2048	Ck[5]	16 Hz
System clock / 4096	Ck[4]	8 Hz
System clock / 8192	Ck[3]	4 Hz
System clock / 16384	Ck[2]	2 Hz
System clock / 32768	Ck[1]	1 Hz

Table 5.2.2 Control of Prescaler Register RegPresc

Bit	Name	Reset	R/W	Description
3	*PWMOn*	*0*	*R/W*	*see 10 bit counter*
2	ResPresc	0	R/W	Write Reset prescaler 1 -> Resets the divider chain from Ck[14] down to Ck[2], sets Ck[1]. 0 -> No action. The Read value is always '0'
1	PrIntSel	0	R/W	Interrupt select. 0 -> Interrupt from Ck[4] 1 -> Interrupt from Ck[6]
0	DebSel	0	R/W	Debouncer clock select. 0 -> Debouncer with Ck[8] 1 -> Debouncer with Ck[11] or Ck[14]

Figure 7. Prescaler Frequency Timing



With DebSel = 1 one may choose either the Ck[11] or Ck[14] debouncer frequency by selecting the corresponding metal mask option. Relative to 32kHz the corresponding max. debouncer times are then 2 ms or 0.25 ms. For the metal mask selection refer to chapter 18.1.6.

Switching the **PrIntSel** may generate an interrupt request. Avoid it with **MaskIRQ32/8** = 0 selection during the switching operation.

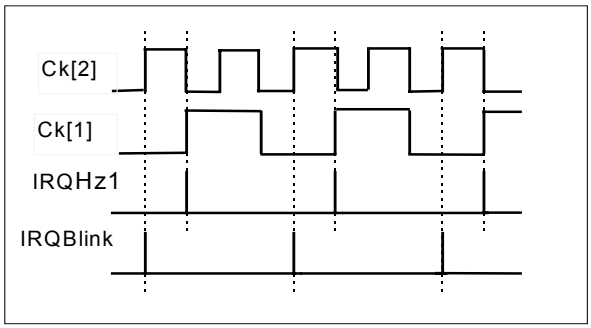
The prescaler contains 3 interrupt sources:
- IRQ32/8 ; this is Ck[6] or Ck[4] positive edge interrupt, the selection is depending on bit **PrIntSel.**
- IRQHz1 ; this is Ck[1] positive edge interrupt
- IRQBlink ; this is 3/4 of Ck[1] period interrupt

There is no interrupt generation on reset.

The first IRQHz1 Interrupt occurs 1 sec (32kHz) after reset.

A possible application for the IRQBlink is LCD-Display blinking control together with IRQHz1.

Figure 8. Prescaler Interrupts



6. Input and Output Ports

The EM6626 has:

- One 4-bit input port (port A)
- One 4-bit input/output port. (port B)
- One serial interface (port SP) also configurable as 4-bit I/O port

Pull-up and pull-down resistors can be added to all this ports with metal and/or register options.

6.1 Ports Overview

Table 6.1.1 Input and Output Ports Overview

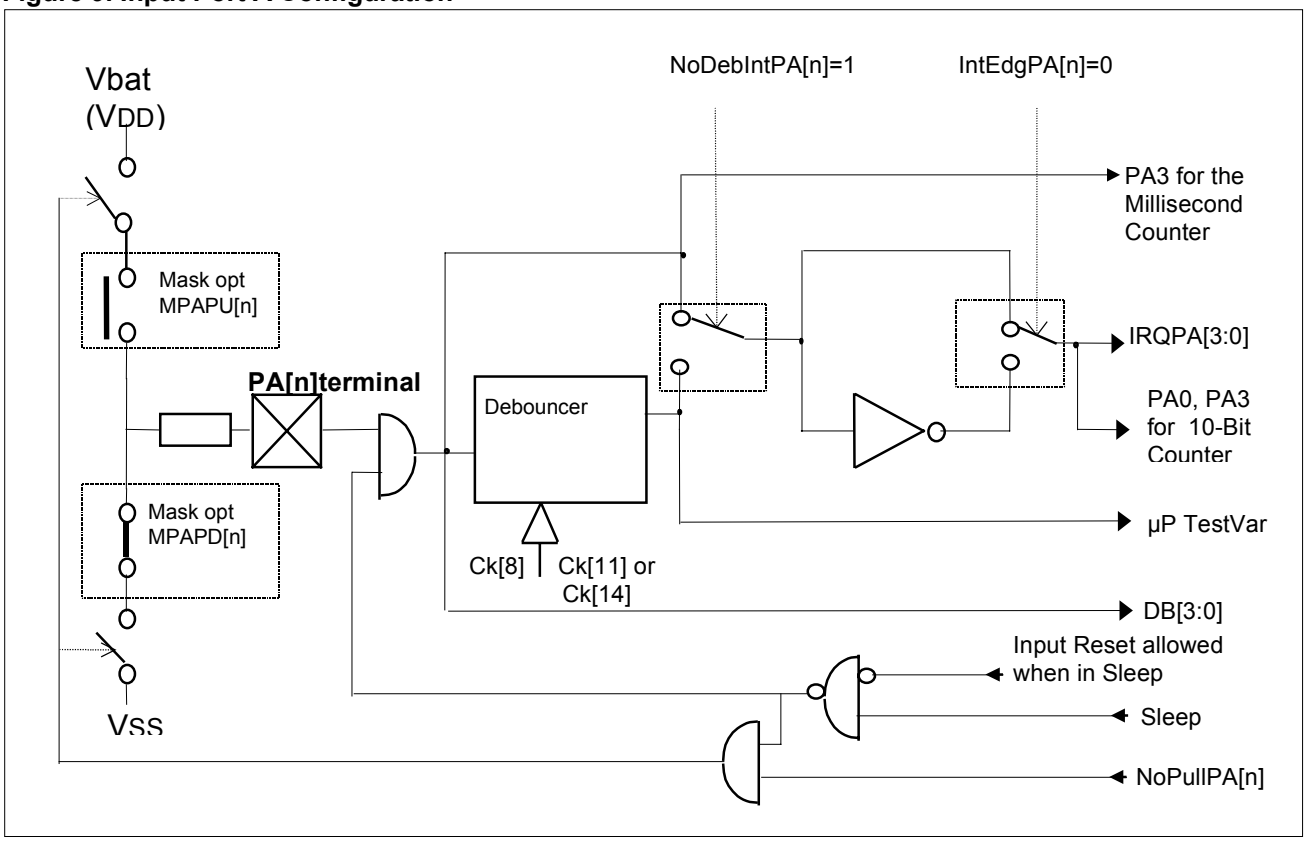
Port	Mode	Mask(M:) or Register(R:) Option	Function	Bit-wise Multifunction on Ports			
PA [3:0]	Input	M: Pull-up M: Pull-down (default) R: Pull(up/down) select R: Debouncer or direct input for IRQ requests and Counter R: + or - for IRQ-edge and counter R: Input reset combination	-Input -Bit-wise interrupt request -Software test variable conditional jump -PA[3],PA[0] input for the event counter -PA[3] input for the millisecond counter -Port A reset inputs	**PA[3]** 10 bit event counter clock start/stop of MSC	**PA[2]** - -	**PA[1]** - -	**PA[0]** 10 bit event counter clock -
PB [3:0]	Individual input or output	R: CMOS or Nch. open drain output R: Pull-down on input R: Pull-up on input	-Input or output -PB[3] for the PWM output -PB[2:0] for the Ck[16,11,1] output -Tristate output	**PB[3]** PWM output	**PB[2]** Ck[16] output	**PB[1]** Ck[11] output	**PB[0]** Ck[1] output
PS [3:0]	Serial I/O or port-wise input / output	R: CMOS or Nch open drain output R: Pull-down on input R: Pull-up on input	-PSP[3], serial clock out -PSP[2], serial data out -PSP[1], serial status out -PSP[0], serial data in -PSP[3:0] 4-bit input/output -Tristate output	**PSP[3]** Serial clock output SCLK	**PSP[2]** Serial data output SOUT	**PSP[1]** Ready or CS Ready/CS	**PSP[0]** Serial data input SIN

6.2 Port A

The EM6626 has one four bit general purpose CMOS input port. The port A input can be read at any time, internal pull-up or pull-down resistors can be chosen. All selections concerning port A are bit-wise executable. I.e. Pull-up on PA[2], pull-down on PA[0], positive IRQ edge on PA[0] but negative on PA[1], etc.

In sleep mode the port A pull-up or pull-down resistors are turned off, and the inputs are deactivated except if the **InpResSleep** bit in the option register **OPTFSel** is set to 1. In this case the port A inputs are continuously monitored to match the input reset condition which will immediately wake the EM6626 from sleep mode (all pull resistors remain).

Figure 9. Input Port A Configuration



6.2.1 IRQ on Port A

For interrupt request generation (IRQ) one can choose direct or debouncer input and positive or negative edge IRQ triggering. With the debouncer selected (**OPTDebIntPA**) the input must be stable for two rising edges of the selected debouncer clock (**RegPresc**). This means a worst case of 16 ms (default) or 2 ms (0.25 ms by metal mask) with a system clock of 32 KHz.
Either a positive or a negative edge on the port A inputs - after debouncer or not - can generate an interrupt request. This selection is done in the option register **OPTIntEdgPA.**

All four bits of port A can provide an IRQ, each pin with its own interrupt mask bit in the **RegIRQMask1** register. When an IRQ occurs, inspection of the **RegIRQ1**, **RegIRQ2** and **RegIRQ3** registers allows the interrupt to be identified and treated.
At power on or after any reset the **RegIRQMask1** is set to 0, thus disabling any input interrupt. A new interrupt is only stored with the next active edge after the corresponding interrupt mask is cleared. See also the interrupt chapter 10.

It is recommended to mask the port A IRQ's while one changes the selected IRQ edge. Else one may generate a IRQ (Software IRQ). I.e. PA[0] on '0' then changing from positive to negative edge selection on PA[0] will immediately trigger an IRQPA[0] if the IRQ was not masked.

6.2.2 Pull-up or Pull-down

Each of the input port terminals PA[3:0] has a resistor integrated which can be used either as pull-up or pull-down resistor, depending on the selected metal mask options. See the port A metal mask chapter for details. The pull resistor can be inhibited using the **NoPullPA[n]** bits in the register **OptNoPullPA.**

Table 6.2.1. Pull-up or Pull-down Resistor on Port A Inputs

Option mask pull-up MPAPU[n]	Option mask pull-down MPAPD[n]	NoPullPA[n] value	Action
no	no	x	no pull-up, no pull-down
no	yes	0	no pull-up, pull-down
no	yes	1	no pull-up, no pull-down
yes	no	0	pull-up, no pull-down
yes	no	1	no pull-up , no pull-down
yes	yes	x	not allowed*

with n=0…3

* only pull-up or pull-down may be chosen on any port A terminal (one choice is excluding the other)

6.2.3 Software Test Variables

The port A terminals PA[2:0] are also used as input conditions for conditional software branches. Independent of the **OPTDebIntPA** and the **OPTIntEdgPA.** These CPU inputs always have a debouncer.

- Debounced PA[0] is connected to CPU TestVar1.
- Debounced PA[1] is connected to CPU TestVar2.
- Debounced PA[2] is connected to CPU TestVar3.

6.2.4 Port A for 10-Bit Counter and MSC

The PA[0] and PA[3] inputs can be used as the clock input terminal for the 10 bit counter in "event count" mode. As for the IRQ generation one can choose debouncer or direct input with the register **OPTDebIntPA** and non-inverted or inverted input with the register **OPTIntEdgPA**. Debouncer input is always recommended.

Pad input PA[3] is also used as start/stop control for the millisecond counter. This control signal is derived from PA[3], it is independent of the port A debouncer and edge selection. Refer also to Figure 9.

6.3 Port A Registers

Table 6.3.1 Register RegPA

Bit	Name	Reset	R/W	Description
3	PAData[3]	-	R*	PA[3] input status
2	PAData[2]	-	R*	PA[2] input status
1	PAData[1]	-	R*	PA[1] input status
0	PAData[0]	-	R*	PA[0] input status

* Direct read on port A terminals

Table 6.3.2 Register RegIRQMask1

Bit	Name	Reset	R/W	Description
3	MaskIRQPA[3]	0	R/W	Interrupt mask for PA[3] input
2	MaskIRQPA[2]	0	R/W	Interrupt mask for PA[2] input
1	MaskIRQPA[1]	0	R/W	Interrupt mask for PA[1] input
0	MaskIRQPA[0]	0	R/W	Interrupt mask for PA[0] input

Default "0" is: interrupt request masked, no new request stored

Table 6.3.3 Register RegIRQ1

Bit	Name	Reset	R/W	Description
3	IRQPA[3]	0	R/W*	Interrupt request on PA[3]
2	IRQPA[2]	0	R/W*	Interrupt request on PA[2]
1	IRQPA[1]	0	R/W*	Interrupt request on PA[1]
0	IRQPA[0]	0	R/W*	Interrupt request on PA[0]

*; Write "1" clears the bit, write "0" has no action, default "0" is: no interrupt request

Table 6.3.4 Register OPTIntEdgPA

Bit	Name	power on value	R/W	Description
3	IntEdgPA[3]	0	R/W	Interrupt edge select for PA[3]
2	IntEdgPA[2]	0	R/W	Interrupt edge select for PA[2]
1	IntEdgPA[1]	0	R/W	Interrupt edge select for PA[1]
0	IntEdgPA[0]	0	R/W	Interrupt edge select for PA[0]

Default "0" is: Positive edge selection

Table 6.3.5 Register OPTDebIntPA

Bit	Name	power on value	R/W	Description
3	NoDebIntPA[3]	0	R/W	Interrupt debounced for PA[3]
2	NoDebIntPA[2]	0	R/W	Interrupt debounced for PA[2]
1	NoDebIntPA[1]	0	R/W	Interrupt debounced for PA[1]
0	NoDebIntPA[0]	0	R/W	Interrupt debounced for PA[0]

Default "0" is: Debounced inputs for interrupt generation

Table 6.3.6 Register OPTNoPullPA

Bit	Name	power on value	R/W	Description
3	NoPullPA[3]	0	R/W	Pull-up/down selection on PA[3]
2	NoPullPA[2]	0	R/W	Pull-up/down selection on PA[2]
1	NoPullPA[1]	0	R/W	Pull-up/down selection on PA[1]
0	NoPullPA[0]	0	R/W	Pull-up/down selection on PA[0]

Default "0" depending on mask selection

6.4 Port B

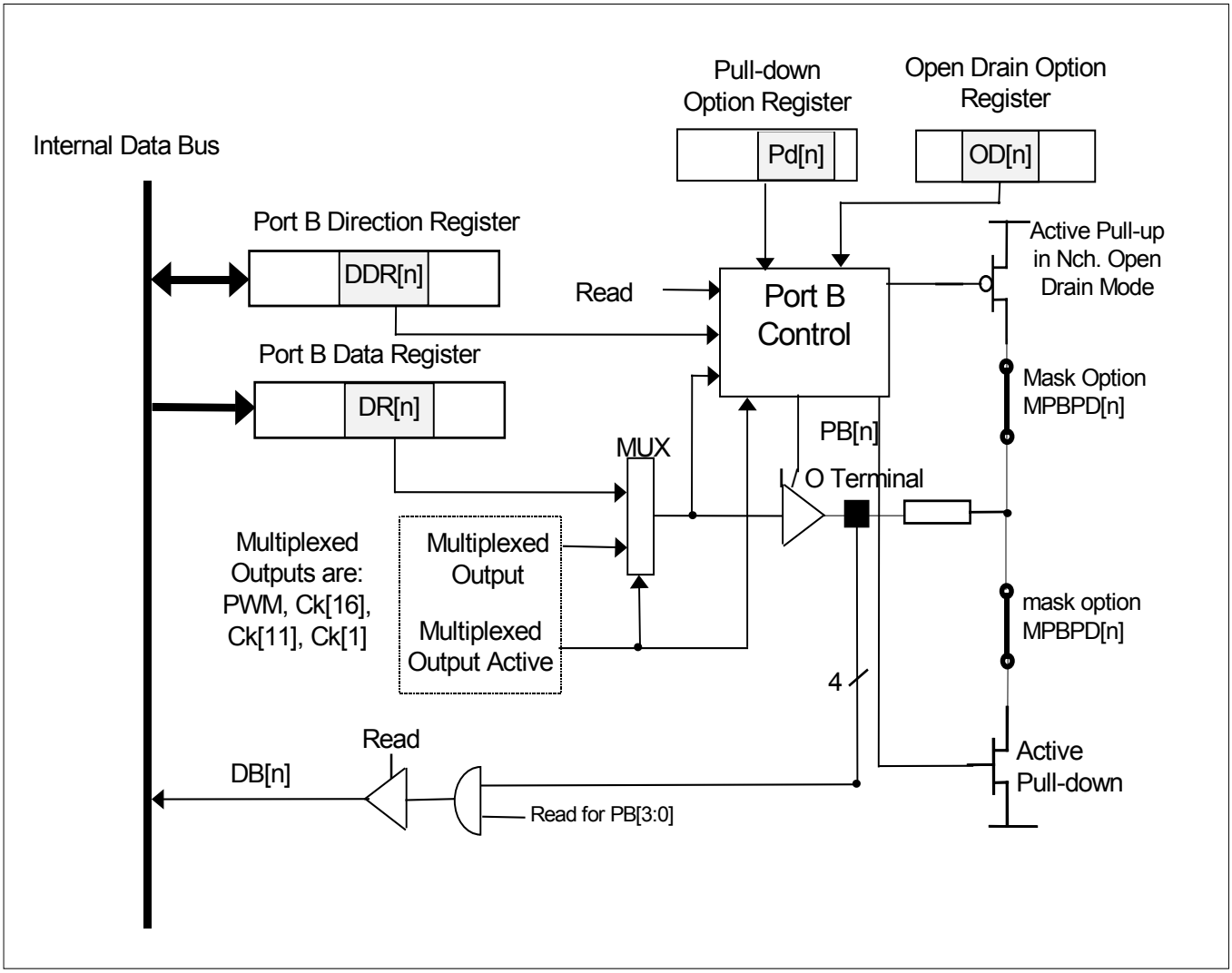
The EM6626 has one four bit general purpose I/O port. Each bit can be configured individually by software for input/output, pull-up, pull-down and CMOS or Nch. open drain output type. The port outputs either data, frequency or PWM signals.

6.4.1 Input / Output Mode

Each port B terminal is bit-wise bi-directional. The input or output mode on each port B terminal is set by writing the corresponding bit in the **RegPBCntl** control register. To set for input (default), 0 is written to the corresponding bit of the **RegPBCntl** register which results in a high impedance state for the output driver. The output mode is set by writing 1 in the control register, and consequently the output terminal follows the status of the bits in the **RegPBData** register.

The port B terminal status can be read on address **RegPBData** even in output mode. Be aware that the data read on port B is not necessary of the same value as the data stored on **RegPBData** register.
See also Figure 10 for details.

Figure 10. Port B Architecture



6.4.2 Pull-up or Pull-down

On each terminal of PB[3:0] an internal input pull-up (metal mask MPBPU[n]) or pull-down (metal mask MPBPD[n]) resistor can be connected per metal mask option. Per default the two resistors are in place. In this case one can chose per software to have either a pull-up, a pull-down or no resistor. See below.

For Metal mask selection and available resistor values refer to chapter 18.1.3.

Pull-down ON : MPBPD[n] must be in place ,
AND bit **NoPdPB**[n] must be '0' .

Pull-down OFF: MPBPD[n] is not in place,
OR if MPBPD[n] is in place **NoPdPB**[n] = '1' cuts off the pull-down.
OR selecting **NchOpDPB**[n] = '1' cuts off the pull-down.

Pull-up ON * : MPBPU[n] must be in place,
AND bit **NchOpDPB**[n] must be '1' ,
AND (bit **PBIOCntl[n]** = '0' (input mode) **OR** if **PBIOCntl[n]** = '1' while **PBData**[n] = 1.)

Pull-up OFF* : MPBPU[n] is not in place,
OR if MPBPU[n] is in place **NchOpDPB**[n] = '0' cuts off the pull-up,
OR if MPBPU[n] is in place and if **NchOpDPB**[n] = '1' then **PBData**[n] = 0 cuts off the pull-up.

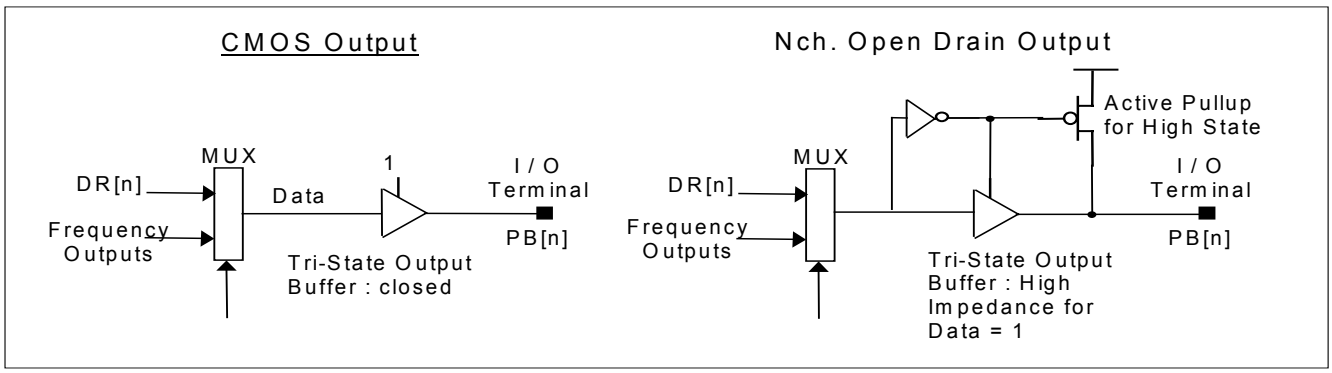
Never pull-up and pull-down can be active at the same time.

For **POWER SAVING** one can switch off the port B pull resistors between two read phases. No cross current flows in the input amplifier while the port B is not read. The recommended order is :

- Switch on the pull resistor.
- Allow sufficient time - RC constant - for the pull resistor to drive the line to either V_{SS} or V_{DD}.
- Read the port B
- Switch off the pull resistor

Minimum time with current on the pull resistor is 4 system clock periods, if the RC time constant is lower than 1 system clock period. Adding a NOP instruction before reading moves the number of periods with current in the pull resistor to 6 and the maximum RC delay to 3 clock periods.
The port B outputs can be configured as either CMOS or Nch. open drain outputs. In CMOS both logic '1' and '0' are driven out on the terminal. In Nch. Open Drain only the logic '0' is driven on the terminal, the logic '1' value is defined by the internal pull-up resistor (if implemented), or high impedance.

Figure 11. CMOS or Nch. Open Drain Outputs



6.4.3 PWM and Frequency Output

PB[3] can also be used to output the PWM (Pulse Width Modulation) signal from the 10-Bit Counter, the Ck[16], Ck[11] as well as the Ck[1] prescaler frequencies.

-Selecting PWM output on PB[3] with bit **PWMOn** in register **RegPresc** and running the counter.
-Selecting Ck[16] output on PB[2] with bit **PB32kHzOut** in register **OPTFSelPB**
-Selecting Ck[11] output on PB[1] with bit **PB1kHzOut** in register **OPTFSelPB**
-Selecting Ck[1] output on PB[0] with bit **PB1HzOut** in register **OPTFSelPB**

6.5 Port B Registers

Table 6.5.1 Register RegPBData

Bit	Name	Reset	R/W	Description
3	PBData[3]	-	R*/W	PB[3] input and output
2	PBData[2]	-	R*/W	PB[2] input and output
1	PBData[1]	-	R*/W	PB[1] input and output
0	PBData[0]	-	R*/W	PB[0] input and output

* : Direct read on the port B terminal (not the internal register)

Table 6.5.2 Register RegPBCntl

Bit	Name	Reset	R/W	Description
3	PBIOCntl[3]	0	R/W	I/O control for PB[3]
2	PBIOCntl[2]	0	R/W	I/O control for PB[2]
1	PBIOCntl[1]	0	R/W	I/O control for PB[1]
0	PBIOCntl[0]	0	R/W	I/O control for PB[0]

Default "0" is: port B in input mode

Table 6.5.3 Option Register OPTFSelPB

Bit	Name	power on value	R/W	Description
3	*InpResSleep*	*0*	*R/W*	*Reset from sleep with port A*
2	PB32kHzOut	0	R/W	Ck[16] output on PB[2]
1	PB1kHzOut	0	R/W	Ck[11] output on PB[1]
0	PB1HzOut	0	R/W	Ck[1] output on PB[0]

Default "0" is: No frequency output, port A Input Reset can not reset the SLEEP mode.

Table 6.5.4 Option Register OPTNoPdPB

Bit	Name	power on value	R/W	Description
3	NoPdPB[3]	0	R/W	No pull-down on PB[3]
2	NoPdPB[2]	0	R/W	No pull-down on PB[2]
1	NoPdPB[1]	0	R/W	No pull-down on PB[1]
0	NoPdPB[0]	0	R/W	No pull-down on PB[0]

Default "0" is: Pull-down on

Table 6.5.5 Option Register OPTNchOpDPB

Bit	Name	power on value	R/W	Description
3	NchOpDPB[3]	0	R/W	Nch. Open Drain on PB[3]
2	NchOpDPB[2]	0	R/W	Nch. Open Drain on PB[2]
1	NchOpDPB[1]	0	R/W	Nch. Open Drain on PB[1]
0	NchOpDPB[0]	0	R/W	Nch. Open Drain on PB[0]

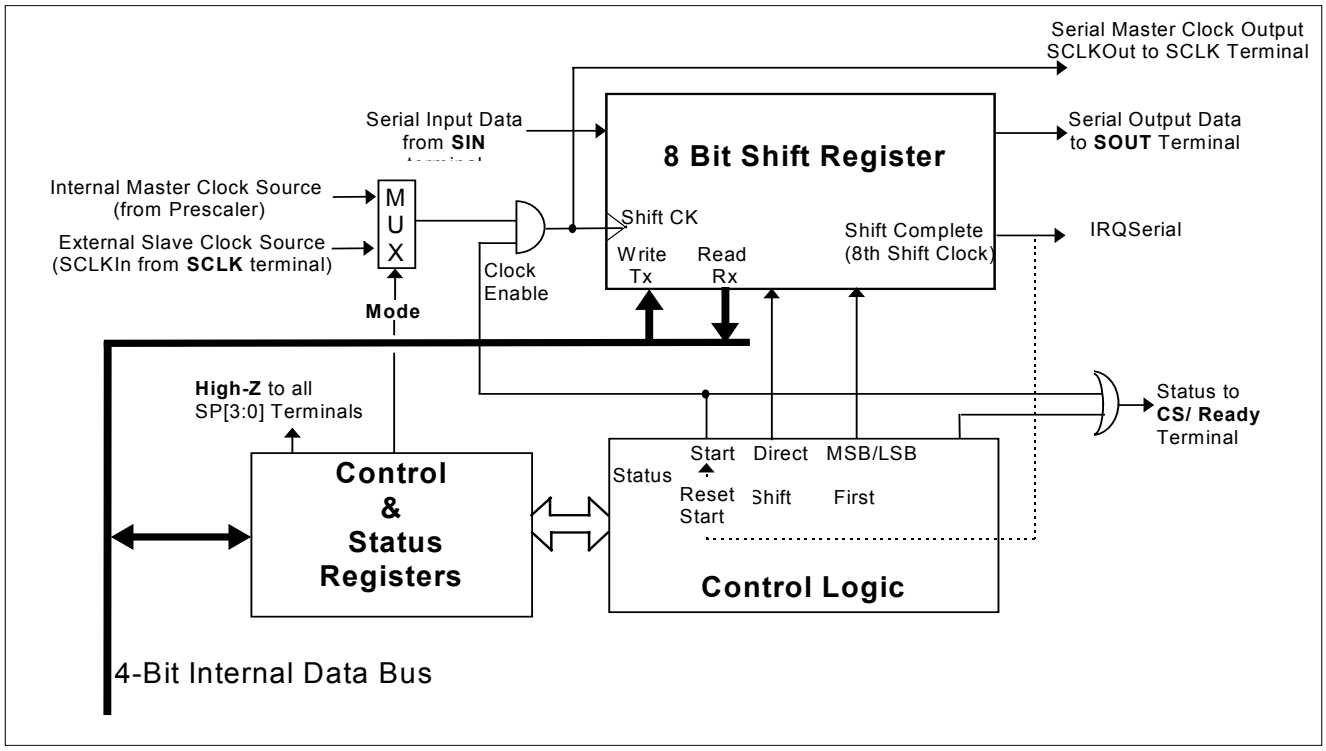
Default "0" is: CMOS output

6.6 Port Serial

The EM6626 contains a simple, half duplex three wire synchronous type serial interface., which can be used to program or read an external EEPROM, ADC, ... etc.

For data reception, a shift-register converts the serial input data on the SIN(PSP[0]) terminal to a parallel format, which is subsequently read by the CPU in registers **RegSDataL** and **RegSDataH** for low and high nibble. To transmit data, the CPU loads data into the shift register, which then serializes it on the SOUT(PSP[2]) terminal. It is possible for the shift register to simultaneously shift data out on the SOUT terminal and shift data on the SIN terminal. In Master mode, the shifting clock is supplied internally by the Prescaler : one of three prescaler frequencies are available, Ck[16], Ck[15] or Ck[14]. In Slave mode, the shifting clock is supplied externally on the SCLKIn(PSP[3]) terminal. In either mode, it is possible to program : the shifting edge, shift MSB first or LSB first and direct shift output. All these selection are done in register **RegSCntl1** and **RegSCntl2**.

Figure 12. Serial Interface Architecture



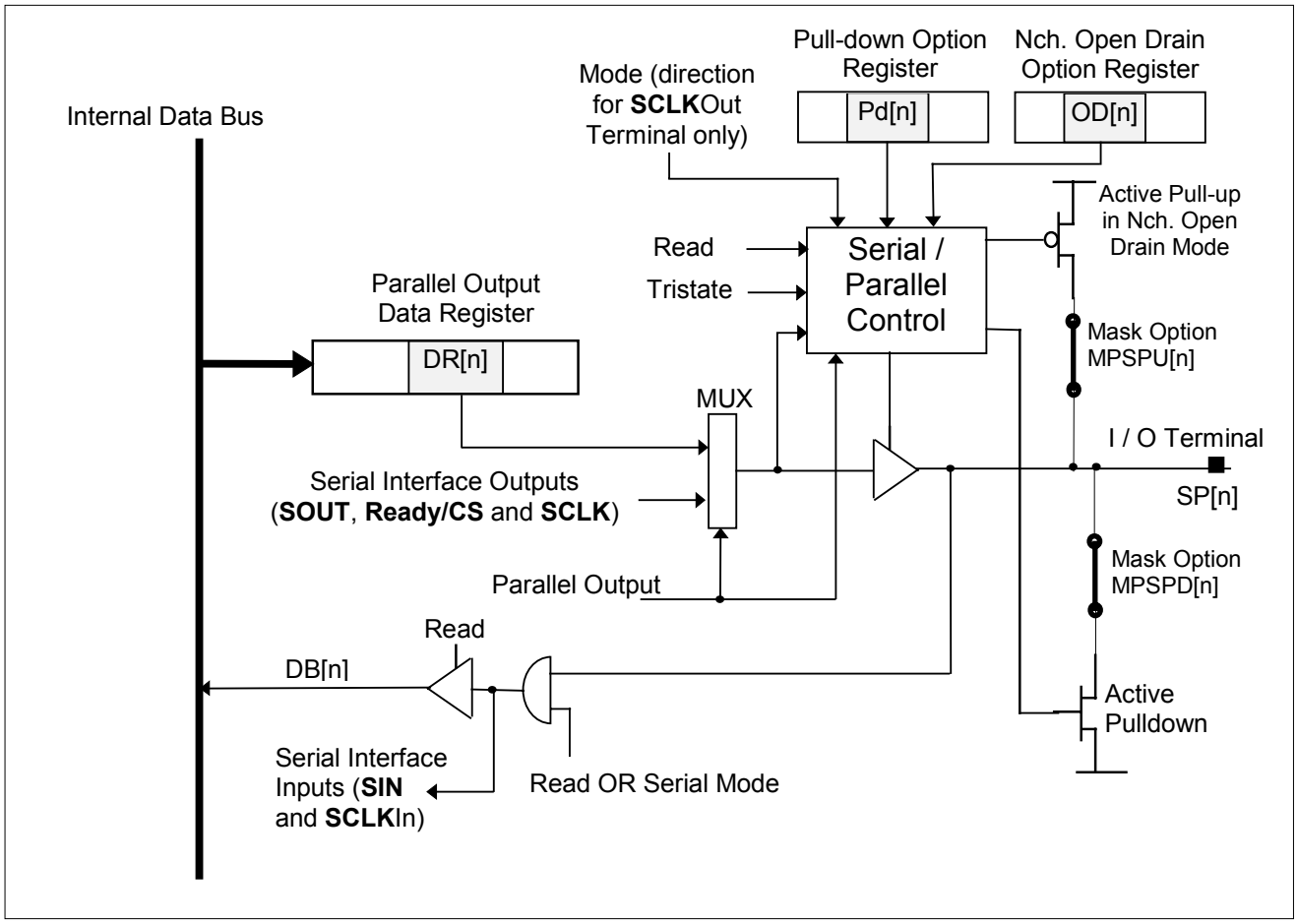
The PSP[3..0] terminal configuration is shown in Figure 13. When the Serial Interface is active then :

- PSP[1] {Ready / CS) is outputting the ready (slave mode) or the CS signal (master mode).
- PSP[2] {SOUT} is always an output.
- PSP[0] {SIN} is always an input.
- PSP[3] {SCLK} is an output for Master mode {SCLKOut} and an input for Slave mode {SCLKIn}

6.6.1 4-bit Parallel I/O

Selecting **OM[1],OM[0]** = '1' in register **RegSCntl2** the PSP[3:0] terminals are configured as a 4-bit Output. Output data is stored in the register **RegSPData** .
The **RegSPData** is defined as a read/write register, but what is read is not the register output, but the port PSP[3:0] terminal values
Selecting **OM[1],OM[0]** = '0' in register **RegSCntl2** the PSP[3:0] outputs are cut off (tristate). The terminals can be used as inputs with individual (bit-wise) pull-up or pull-down settings.
Independent of the selected configuration, the PSP[3:0] terminal levels are always readable.

Figure 13. Port SP Terminal Configuration



6.6.2 Pull-up or Pull-down

For each terminal of PSP[3:0] an input pull-up (metal mask MPSPU[n]) or pull-down (metal mask MPSPD[n]) resistor can be implemented per metal mask option. Per default the two metal masks are in place, so one can chose per software to have either a pull-up, a pull-down or no resistor. For Metal mask selection and available resistor values refer to chapter 18.1.4

Pull-down ON : MPSPD[n] must be in place ,
AND the bit **NoPdPS**[n] must be '0' .

Pull-down OFF: MPSPD[n] is not in place,
OR if MPSPD[n] is in place **NoPdPS**[n] = '1' cuts off the pull-down.
OR selecting **NchOpDPS**[n] = '1' cuts off the pull-down.

Pull-up ON * : MPSPU[n] must be in place,
AND the bit **NchOpDPS**[n] must be '1' ,
AND (** the bits **OM[1,0] in **RegSysCntl2** = '00' (input mode) **OR** any of the port SP terminals is in output mode with a logic '1' to be driven) .

Pull-up OFF* : MPSPU[n] is not in place,
OR if MPSPU[n] is in place **NchOpDPS**[n] = '0' cuts off the pull-up,
OR if MPSPU[n] is in place and **NchOpDPS**[n] = '1' then **SerPData**[n] = 0 cuts off the pull-up.

For **POWER SAVING** one can switch off the port SP pull resistors between two read phases. No cross current flows in the input amplifier while the port SP is not read.
This power saving feature must only be used in tristate mode (OM[0,1]=0). The recommended order is :

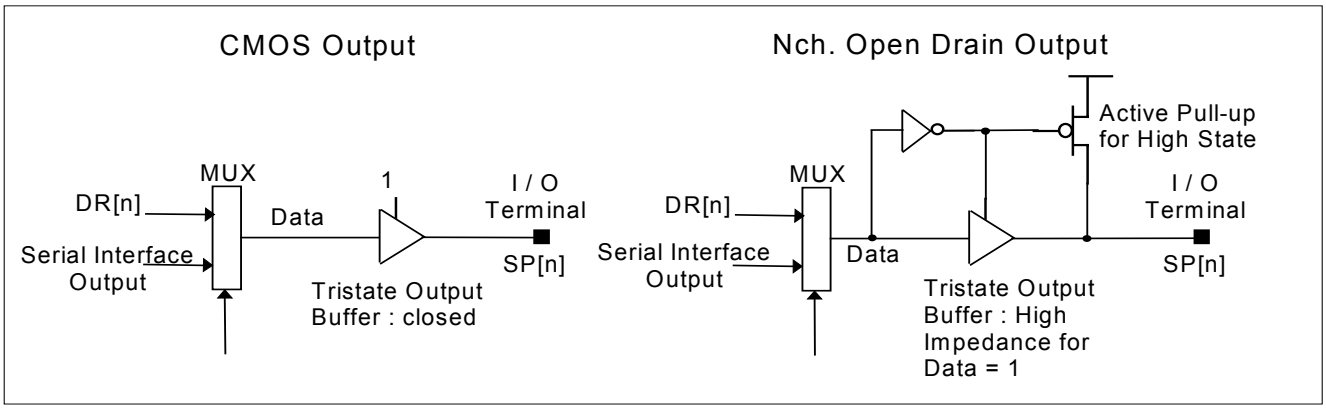
- switch on the pull resistor.
- allow sufficient time - RC constant - for the pull resistor to drive the line to either V_{SS} or V_{DD}.
- Read the port SP
- Switch off the pull resistor

Minimum time with current on the pull resistor is 4 periods of the system clock, if the RC constant is lower than 1 system clock period. Adding a NOP before reading moves the number of periods with current in the pull resistor to 6 and the maximum RC delay to 3 clock periods.

6.6.3 Nch. Open Drain Outputs

The port SP outputs can be configured as either CMOS or Nch. open drain outputs.
In CMOS both logic '1' and '0' are driven out on the terminal.
In Nch. open drain only the logic '0' is driven out on the terminal, the logic '1' value is high impedance or defined by the internal pull-up resistor (if existing).

Figure 14. CMOS or Nch. Open Drain outputs



6.6.4 General Functional Description

After power on or after any reset the serial interface is in serial slave mode with **Start** and **Status** set to 0, LSB first, negative shift edge and all outputs are in high impedance state.
When the **Start** bit is set, the shift operation is enabled and the serial interface is ready to transmit or receive data, eight shift operations are performed: 8 serial data values are read from the data input terminal into the shift register and the previous loaded 8-bits are send out via the data output terminal. After the eight shift operation, an interrupt is generated, and the **Start** bit is reset.

Parallel to serial conversion procedure (master mode example).
Write to **RegSCntl1** serial control (clock freq. in master mode, edge and MSB/LSB select).
Write to **RegSDataL** and **RegSDataH** (shift out data values).
Write to **RegSCntl2** (Start=1, mode select, status). ---> Starts the shift out
After the eighth clock an interrupt is generated, **Start** becomes low. Then, interrupt handling

Serial to parallel conversion procedure (slave mode example).
Write to **RegSCntl1** (slave mode, edge and MSB/LSB select).
Write to **RegSCntl2** (Start=1, mode select, status).
After eight serial clocks an interrupt is generated, **Start** becomes low.
Interrupt handling.
Shift register **RegSDataL** and **RegSDataH** read.
A new shift operation can be authorized.

6.6.5 Detailed Functional Description

Master or Slave mode is selected in the control register **RegSCntl1.**
In Slave mode, the serial clock comes from an external device and is input via the PSP[3] terminal as a synchronous clock (SCLKIn) to the serial interface. The serial clock is ignored as long as the **Start** bit is not set. After setting **Start,** only the eight following active edges of the serial clock input PSP[3] are used to shift the serial data in and out. After eight serial clock edges the **Start** bit is reset. The PSP[1] terminal is a copy of the (**Start OR Status)** bit values, it can be used to indicate to the external master, that the interface is ready to operate or it can be used as a chip select signal in case of an external slave.
In Master mode, the synchronous serial clock is generated internally from the system clock. The frequency is selected from one out of three sources (**MS0** and **MS1** bits in **RegSCntl1**) **.** The serial shifting clock is only generated during **Start** = high and is output to the SCLK terminal as the Master Clock (SCLKOut). When **Start** is low, the serial clock output on PSP[3] is 0.
An interrupt request **IRQSerial** is generated after the eight shift operations are done. This signal is set by the last negative edge of the serial interface clock on PSP[3] (master or slave mode) and is reset to 0 by the next write of **Start** or by any reset. This interrupt can be masked with register **RegIRQMask3.** For more details about the interrupt handling see chapter 10.
Serial data input on PSP[0] is sampled by the positive or negative serial shifting clock edge, as selected by the Control Register **POSnNeg** bit**.** Serial data input is shifted in LSB first or MSB first, as selected by the Control Register **MSBnLSB** bit**.**

6.6.6 Output Modes

Serial data output is given out in two different ways (Refer also to Figure 15 and Figure 16).

- **OM[1]** = 1, **OM[0]** = 0 :
The serial output data is generated with the selected shift register clock (**POSnNeg**). The first data bit is available directly after the **Start** bit is set.

-**OM[1]** = 0, **OM[0]** = 1 :
The serial output data is re-synchronized by the positive serial interface clock edge, independent of the selected clock shifting edge. The first data bit is available on the first positive serial interface clock edge after Start='1'.

Figure 15. Direct or Re-Synchronized Output

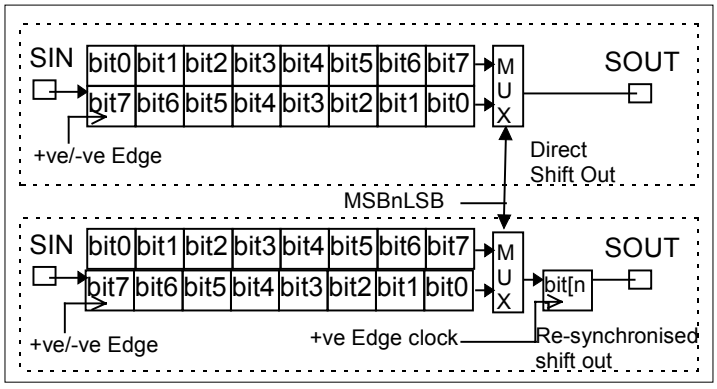
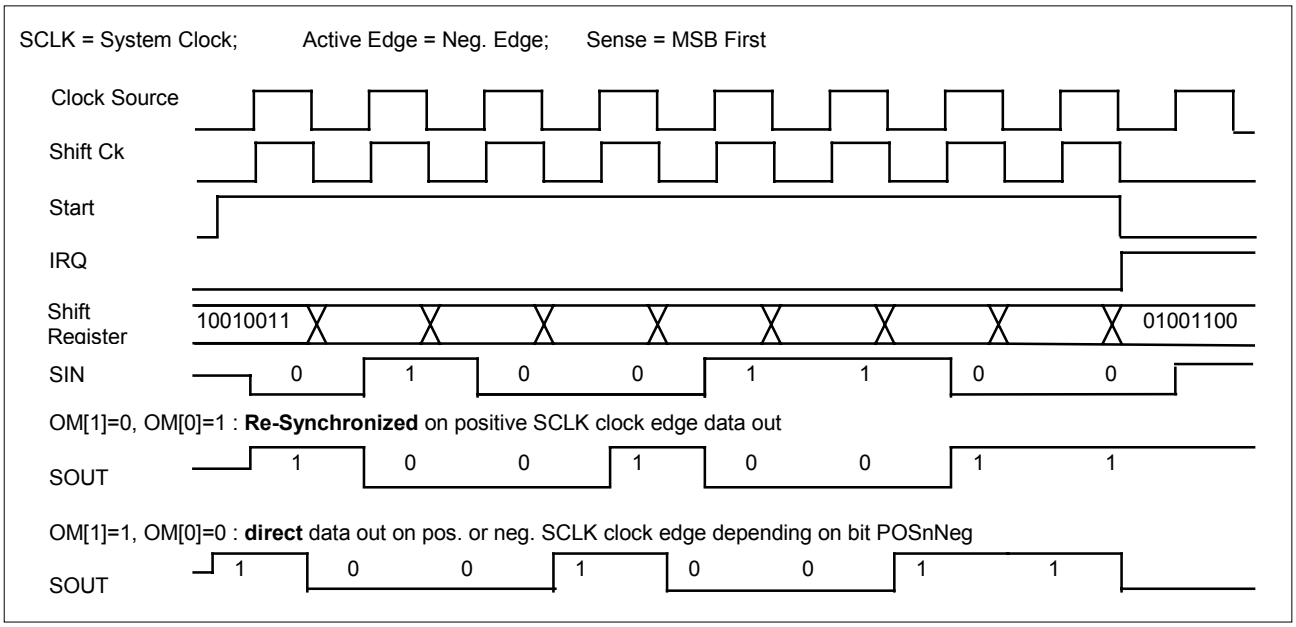


Table 6.6.6 Output Mode Selection in RegSCntl2

OM[1]	OM[0]	Output mode	Description
0	0	Tristate	Output disable (tristate on PSP[3:0])
0	1	Serial-Synchronized	Re-synchronized positive edge data shift out
1	0	Serial-Direct	Direct shift pos. or neg. edge data out
1	1	Parallel	Parallel port SP output

Tristate output is selected by default.

Figure 16 Shift Operation and IRQ Generation

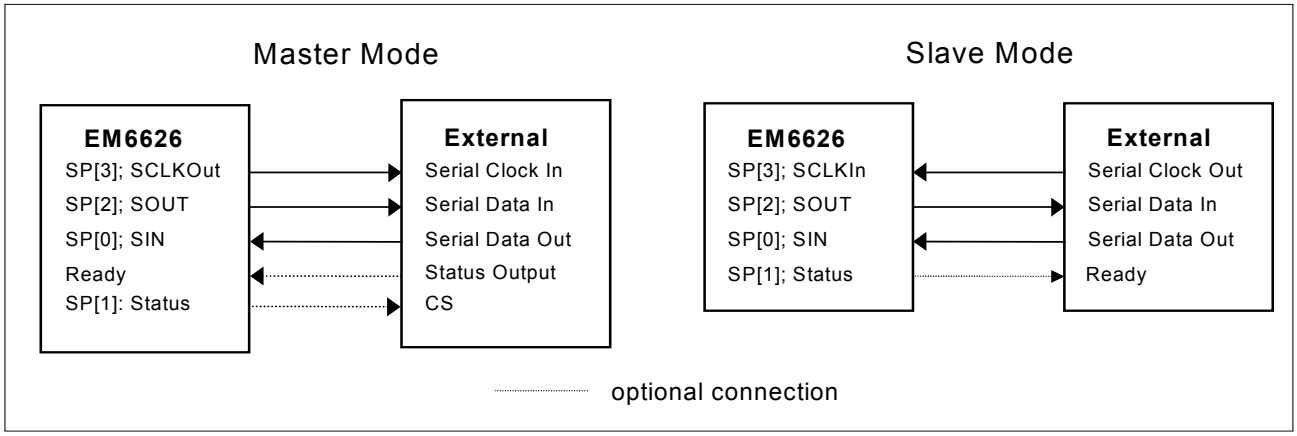


6.6.7 Reset and Sleep on Port SP

During circuit initialization, all option registers are reset by Power On Reset and therefore all pull-ups are off and all pull-downs are on. During Sleep mode, Port SP inputs are cut-off , the circuit is in Reset State. However the Reset State does not reset the option registers and pull-downs, if previously turned on, remain on even during Sleep mode. After any reset the serial interface parameters are reset to : Slave mode, Start and Status = 0, LSB first, negative edge shift , PSP[3:0] tristate.

Note : A write operation in the control registers or in the data registers while **Start** is high will change internal values and may cause an error condition. The user must take care of the serial interface status before writing internal registers. In order to read the correct values on the data registers, the shift operation must be halted during the read accesses.

Figure 17. Sample Basic Serial Port Connections



6.7 Serial Interface Registers

Table 6.7.1 Register RegSCntl1

Bit	Name	Reset	R/W	Description
3	MS1	0	R/W	Frequency selection
2	MS0	0	R/W	Frequency selection
1	POSnNeg	0	R/W	Positive or negative clock edge selection for shift operation
0	MSBnLSB	0	R/W	Shift MSB or LSB value first

Default "0" is: Slave mode external clock, negative edge, LSB first

Table 6.7.2 Frequency and Master Slave Mode Selection

MS1	MS0	Description
0	0	Slave mode: Clock from external
0	1	Master mode: System clock / 4
1	0	Master mode: System clock / 2
1	1	Master mode: System clock

Table 6.7.3 Register RegSCntl2

Bit	Name	Reset	R/W	Description
3	Start	0	R/W	Enabling the interface,
2	Status	0	R/W	Ready or Chip Select output on PSP[1]
1	OM[1]	0	R/W	Output mode select 1
0	OM[0]	0	R/W	Output mode select 0

Default "0" is: Interface disabled, status 0, serial mode, output tristate.

Table 6.7.4 Register RegSDataL

Bit	Name	Reset	R/W	Description
3	SerDataL[3]	0	R/W	Serial data low nibble
2	SerDataL[2]	0	R/W	Serial data low nibble
1	SerDataL[1]	0	R/W	Serial data low nibble
0	SerDataL[0]	0	R/W	Serial data low nibble

Default "0" is: Data equal 0.

Table 6.7.5 Register RegSDataH

Bit	Name	Reset	R/W	Description
3	SerDataH[3]	0	R/W	Serial data high nibble
2	SerDataH[2]	0	R/W	Serial data high nibble
1	SerDataH[1]	0	R/W	Serial data high nibble
0	SerDataH[0]	0	R/W	Serial data high nibble

Default "0" is: Data equal 0.

Table 6.7.6 Register RegSPData

Bit	Name	Reset	R/W	Description
3	SerPData[3]	0	R* /W	Parallel output data
2	SerPData[2]	0	R* /W	Parallel output data
1	SerPData[1]	0	R* /W	Parallel output data
0	SerPData[0]	0	R* /W	Parallel output data

R* : The input terminal value is read, not the register

Table 6.7.7 Option Register OPTNoPdPS

Bit	Name		R/W	Description
3	NoPdPS[3]	0	R/W	No pull-down on PSP[3]
2	NoPdPS[2]	0	R/W	No pull-down on PSP[2]
1	NoPdPS[1]	0	R/W	No pull-down on PSP[1]
0	NoPdPS[0]	0	R/W	No pull-down on PSP[0]

Default "0" is: Pull-down on

Table 6.7.8 Option Register OPTNchOpDPS

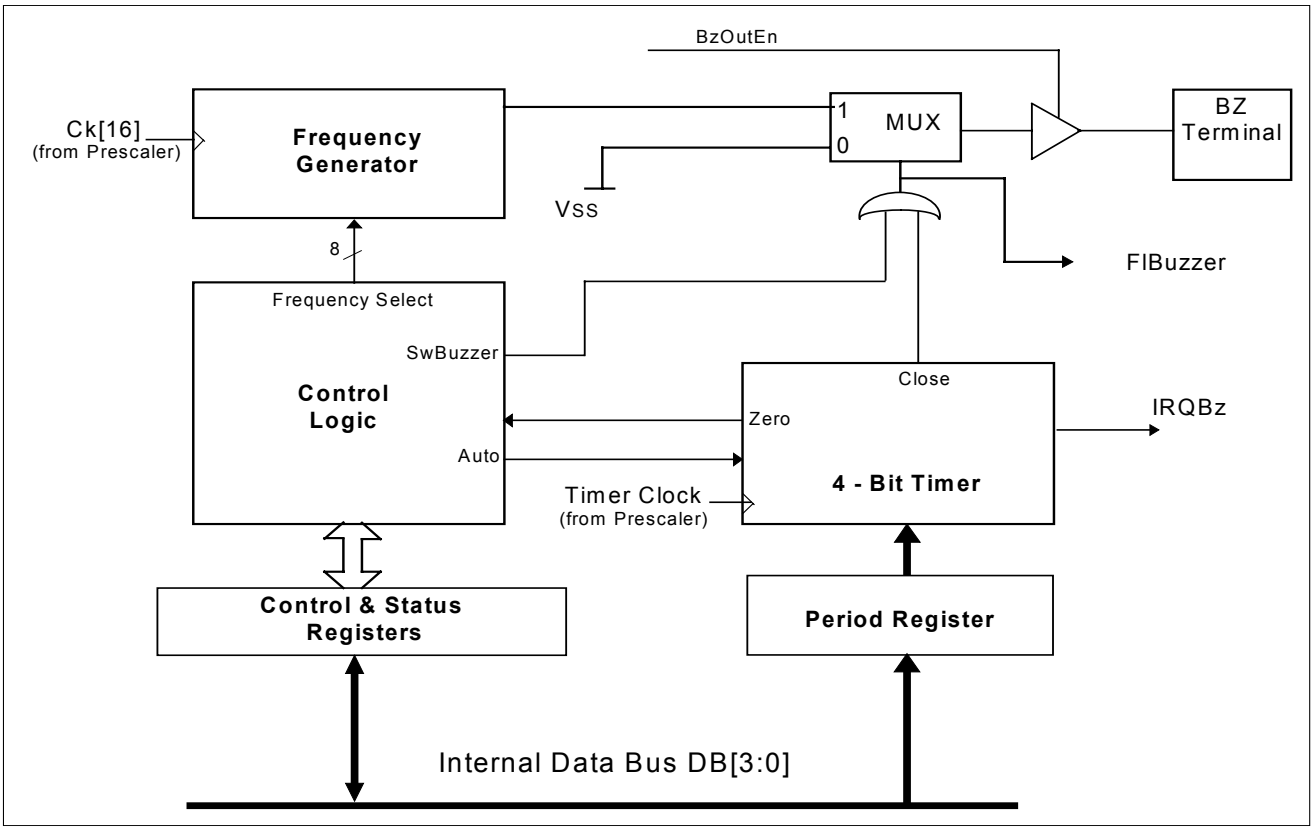
Bit	Name		R/W	Description
3	NchOpDPS[3]	0	R/W	Nch. Open Drain on PSP[3]
2	NchOpDPS[2]	0	R/W	Nch. Open Drain on PSP[2]
1	NchOpDPS[1]	0	R/W	Nch. Open Drain on PSP[1]
0	NchOpDPS[0]	0	R/W	Nch. Open Drain on PSP[0]

Default "0" is: CMOS output

7. Melody, Buzzer

A normal application is to drive a buzzer connected onto the terminal Buzzer.
This peripheral cell is a combination of a 7 frequency tone generator and a 4-bit timer, used to provide a 50% duty cycle signal on the Buzzer terminal of a pre-selected length and frequency. The Buzzer terminal is active as long as the timer is not 0 or the **SwBuzzer** is set to '1'. The 4-bit timer can be used for another application independent of the Buzzer terminal by selecting "silence" instead of another frequency on the Buzzer output. "Silence" can also be used as part of a melody, or to switch off the buzzer.
To use the buzzer independent of the 4-bit timer one has to set the switch **SwBuzzer.** This bit is in register **RegMelTim** and selects the signal duration on the buzzer output. If **SwBuzzer**=1 then the signal is output until the bit is set back to 0 . With **SwBuzzer**=0 the output signal duration is controlled by the 4bit timer. If neither the **SwBuzzer** or the timer are active, the Buzzer terminal is on 0.
The high impedance state setting with **BzOutEn** is independent of the **SwBuzzer** and Timer settings. As soon as the bit is set to 1 the Buzzer terminal is set tristate. See also Figure 18.

Figure 18. Melody Generator Block Diagram



7.1 4-Bit Timer

The timer has 2 modes:

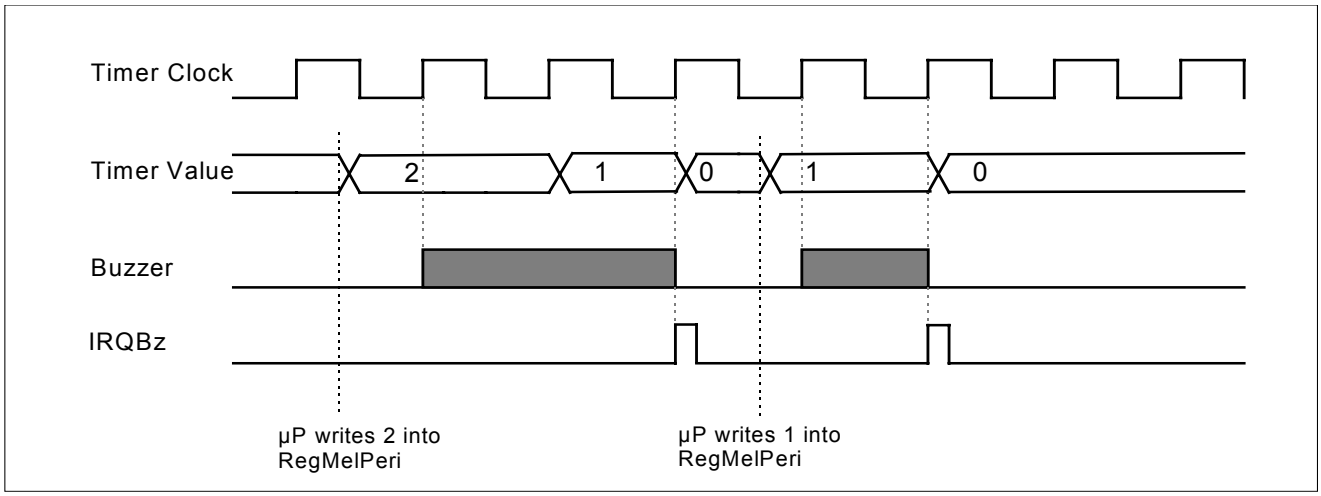
- Single run mode (**Auto**=0)
- Continuos run mode (**Auto**=1)

Mode selection and timer count down frequency is done in register **RegMelTim.** All timer frequencies are coming from the prescaler. The 4-bit timer can be used independent of the melody buzzer application.
Whenever the timer reaches 0 it generates an interrupt request **IRQBz** in the register **RegIRQ2** . This interrupt can be masked with the bit **MaskIRQBz** in register **RegIRQMask2.** By writing 0 into the timer period register the timer stops immediately and does not generate an interrupt.

7.1.1 Single Run Mode

The timer duration is controlled by the **RegMelPeri** value and the selected timer frequency in **RegMelTim.** The timer is counting down from its previously charged value until it reaches 0. On 0 the timer stops and generates an interrupt request. The buzzer frequency output is enabled after the next positive timer clock edge and remains enabled until the timer reaches 0.

Figure 19. Single Run Mode

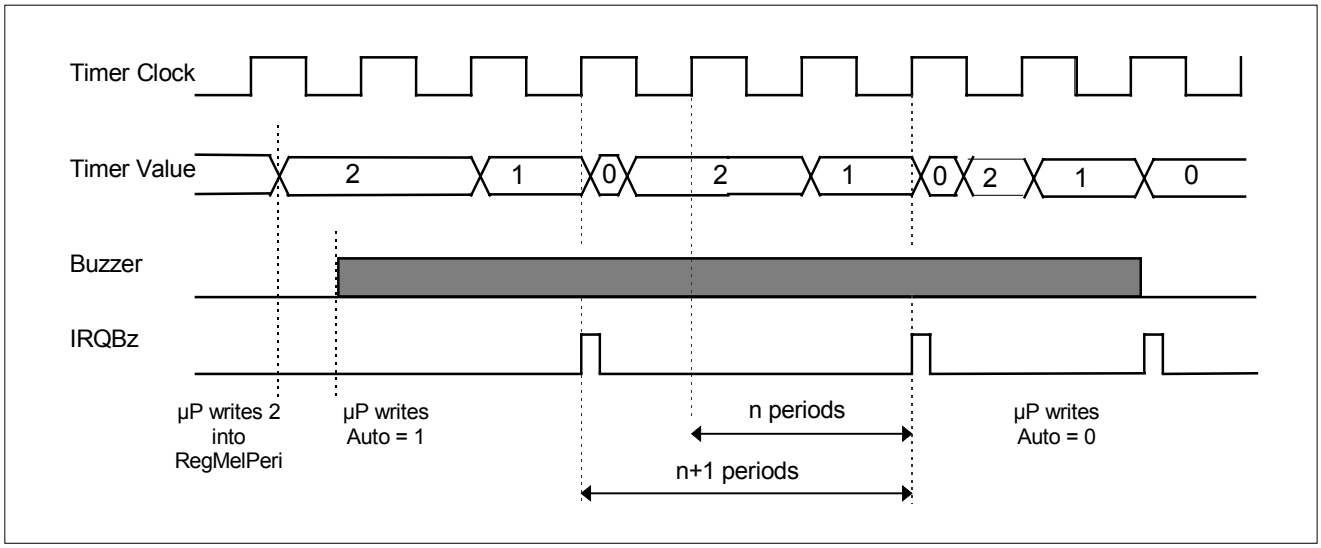


7.1.2 Continuos Run Mode

This is almost the same as the single run mode only that in this case the timer after reaching 0 reloads itself automatically with the register **RegMelPeri** value. Every time the timer reaches 0 an interrupt request is send. There are 2 ways to stop the continuos mode.

- First, changing the mode to single run mode. As the timer reaches 0 it stops. The last period after **Auto**=0 is of length **RegMelPeri + 1**.
- Second, loading 0 into the timer period register **RegMelPeri** stops the timer immediately, no interrupt is generated and the **Auto** flag is reset. The buzzer frequency output is enabled directly by writing **Auto**=1.

Figure 20. Continuos Run Mode



7.2 Programming Order

Single run mode usage

1st, selecting the buzzer frequency into **RegMelFSel**.
2nd, selecting the timer clock frequency in **RegMelTim**.
3rd, selecting the timer period in **RegMelPeri**.
--> On the next positive clock edge the buzzer output is enabled.

Continuos run mode usage

1st, selecting the buzzer frequency into **RegMelFSel**.
2nd, selecting the timer clock frequency in **RegMelTim** (**Auto**=0).
3rd, selecting the timer period in **RegMelPeri**.
4th, set bit **Auto** in **RegMelTim**.
--> Immediately the buzzer output is active.

Avoid timer clock frequency switch during buzzer operation.

7.3 Melody Registers

Table 7.3.1 Register RegMelFSel

Bit	Name	Reset	R/W	Description
3	BzOutEn	0	R/W	Buzzer Output tristate
2	MelFSel[2]	0	R/W	Buzzer frequency select
1	MelFSel[1]	0	R/W	Buzzer frequency select
0	MelFSel[0]	0	R/W	Buzzer frequency select

Default : Buzzer tristate, silence

Table 7.3.2 Buzzer Output Frequency Selection with MelFSel[2..0]

MelFSel[2]	MelFSel[1]	MelFSel[0]	Frequency	
0	0	0	Vss (silence)	
0	0	1	SysClock/8	DO8
0	1	0	SysClock/10	SOL7#
0	1	1	SysClock/12	FA7
1	0	0	SysClock/14	RE7
1	0	1	SysClock/16	DO7
1	1	0	SysClock/20	SOL6#
1	1	1	SysClock/24	FA6

Table 7.3.3 Register RegMelTim

Bit	Name	Reset	R/W	Description
3	SwBuzzer FlBuzzer	0 0	W R	Write: switch buzzer Read: flag buzzer
2	Auto	0	R/W	Single or continuos run mode
1	FTimSel1	0	R/W	Timer clock frequency select
0	FTimSel0	0	R/W	Timer clock frequency select

Default : Single run mode, Ck[3] from prescaler as timer clock

Table 7.3.4 Timer Clock Frequency Select

FTimSel0	FTimSel1	Timer Clock	On 32 KHz operation
0	0	Ck[3]	4 Hz
1	0	Ck[5]	16 Hz
0	1	Ck[7]	64 Hz
1	1	Ck[1]	1 Hz

Table 7.3.5 Register RegMelPeri

Bit	Name	Reset	R/W	Description
3	Per[3]	0	W	Melody timer period MSB
2	Per[2]	0	W	Melody timer period
1	Per[1]	0	W	Melody timer period
0	Per[0]	0	W	Melody timer period LSB

The total timer period duration is calculated as following:

Duration = Value(RegMelPeri) x 1/Ck[n]

Where, Ck[n] is the timer clock frequency and Value(RegMelPeri) is the value of the register **RegMelPeri.**

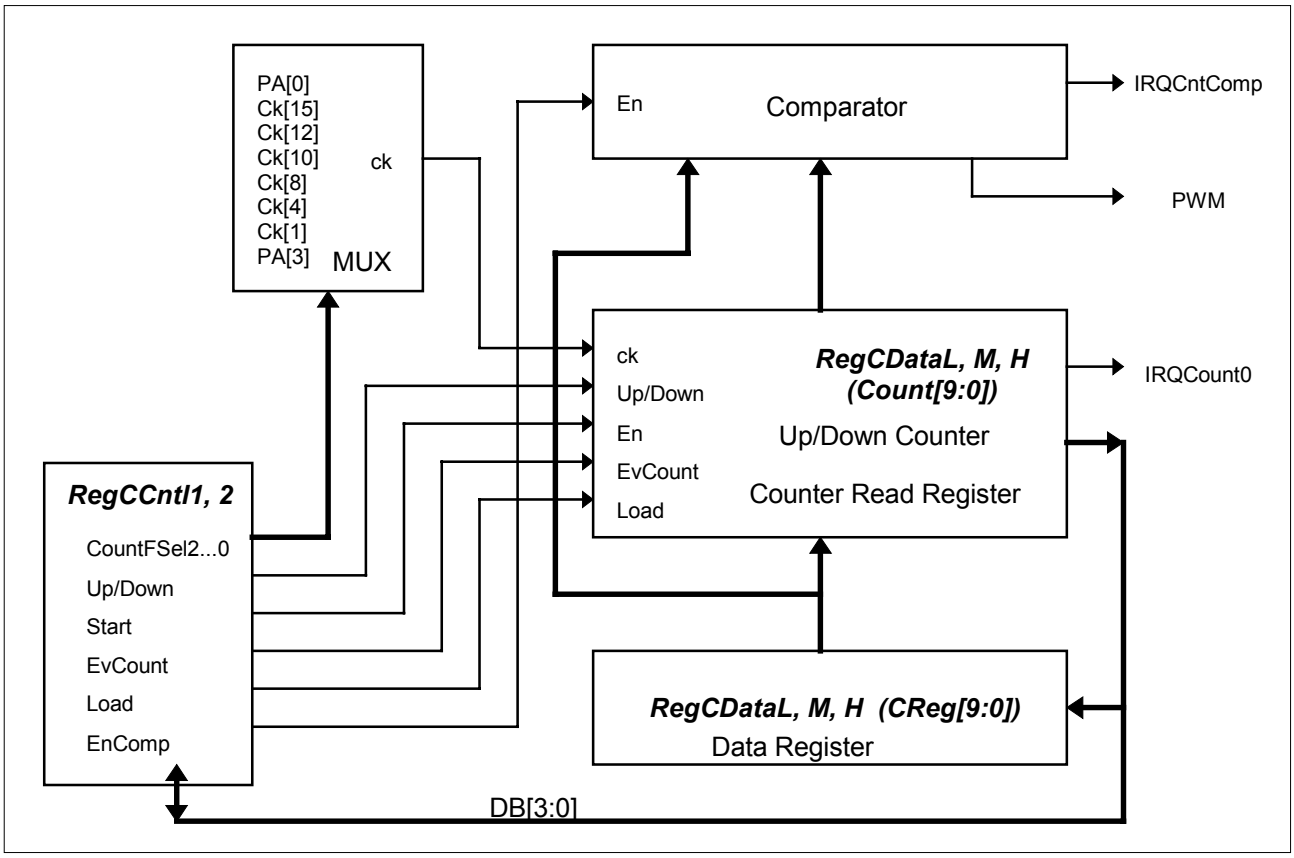
8. 10-bit Counter

The EM6626 has a built-in universal cyclic counter. It can be configured as 10, 8, 6 or 4-bit counter. If 10-bits are selected we call that full bit counting, if 8, 6 or 4-bits are selected we call that limited bit counting.
The counter works in up- or down count mode. Eight clocks can be used as the input clock source, six of them are prescaler frequencies and two are coming from the input pads PA[0] and PA[3]. In this case the counter can be used as an event counter.
The counter generates an interrupt request **IRQCount0** every time it reaches 0 in down count mode or 3FF in up count mode. Another interrupt request **IRQCntComp** is generated in compare mode whenever the counter value matches the compare data register value. Each of this interrupt requests can be masked (default). See section 10 for more information about the interrupt handling.

A 10-bit data register **CReg[9:0]** is used to initialize the counter at a specific value (load into **Count[9:0]**). This data register (**CReg[9:0]**) is also used to compare its value against **Count[9:0]** for equivalence.

A Pulse-Width-Modulation signal (PWM) can be generated and output on port B terminal PB[3].

Figure 21. 10-bit Counter Block Diagram



8.1 Full and Limited Bit Counting

In Full Bit Counting mode the counter uses its maximum of 10-bits length (default). With the **BitSel[1,0]** bits in register **RegCDataH** one can lower the counter length, for IRQ generation, to 8, 6 or 4 bits. This means that actually the counter always uses all the 10-bits, but IRQCount0 generation is only performed on the number of selected bits. The unused counter bits may or may not be taken into account for the **IRQComp** generation depending on bit **SelIntFull**. Refer to chapter 8.4.

Table 7.3.1. Counter length selection

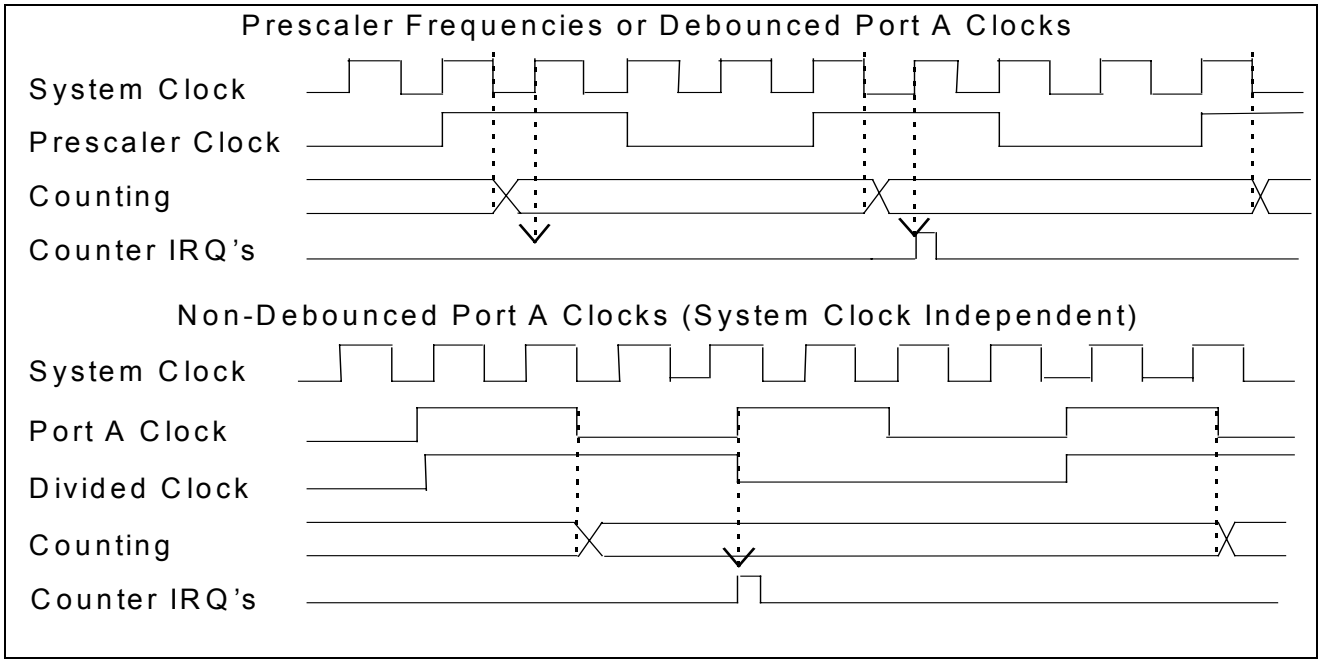
BitSel[1]	BitSel[0]	counter length
0	0	10-Bit
0	1	8-Bit
1	0	6-Bit
1	1	4-Bit

8.2 Frequency Select and Up/Down Counting

8 different input clocks can be selected to drive the Counter. The selection is done with bits **CountFSel2…0** in register **RegCCntl1**. 6 of this input clocks are coming from the prescaler. The maximum prescaler clock frequency for the counter is half the system clock and the lowest is 1Hz. Therefore a complete counter roll over can take as much as 17.07 minutes (1Hz clock, 10 bit length) or as little as 977 µs (Ck[15], 4 bit length). The **IRQCount0**, generated at each roll over, can be used for time bases, measurements length definitions, input polling, wake up from Halt mode, etc. The **IRQCount0** and **IRQComp** are generated with the system clock Ck[16] rising edge. IRQCount0 condition in up count mode is : reaching 3FF if 10-bit counter length (or FF, 3F, F in 8, 6, 4-bit counter length). In down count mode the condition is reaching '0'. The non-selected bits are 'don't care'. For IRQComp refer to section 8.4.

Note: The Prescaler and the Microprocessor clock's are usually non-synchronous, therefore time bases generated are max. n, min. n-1 clock cycles long (n being the selected counter start value in count down mode). However the prescaler clock can be synchronized with µP commands using for instance the prescaler reset function.

Figure 22. Counter Clock Timing



The two remaining clock sources are coming from the PA[0] or PA[3] terminals. Refer to the Figure 9 on page 13 for details. Both sources can be either debounced (Ck[11] or Ck[8]) or direct inputs, the input polarity can also be chosen. The output after the debouncer polarity selector is named PA3 , PA0 respectively. For the debouncer and input polarity selection refer to chapter 6.3.

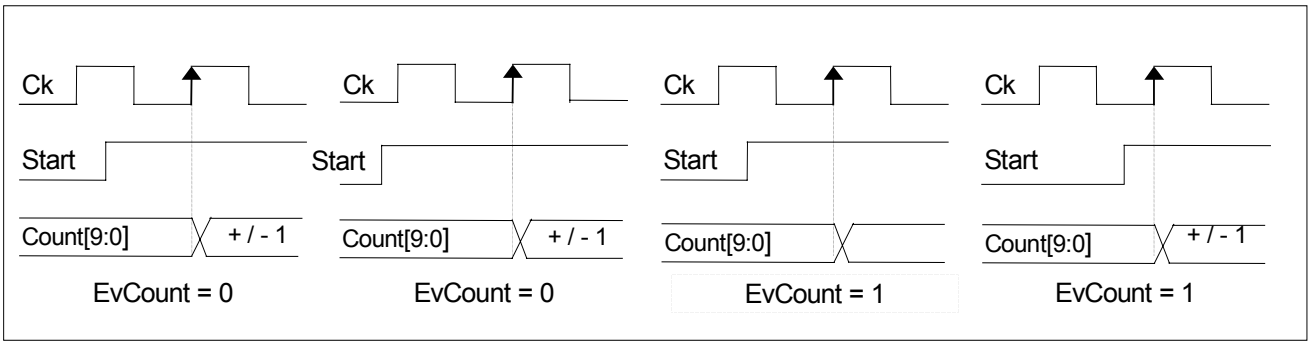
In the case of port A input clock without debouncer, the counting clock frequency will be half the input clock on port A. The counter advances on every odd numbered port A negative edge (divided clock is high level). IRQCount0 and IRQComp will be generated on the rising PA3 or PA0 input clock edge. In this condition the EM6626 is able to count with a higher clock rate as the internal system clock (Hi-Frequency Input). Maximum port A input frequency is limited to 200kHz (@$V_{DD} \geq 1.5$ V). If higher frequencies are needed, please contact EM-Marin.

In both, up or down count (default) mode, the counter is cyclic. The counting direction is chosen in register **RegCCntl1** bit **Up/Down** (default '0' is down count). The counter increases or decreases its value with each positive clock edge of the selected input clock source. Start up synchronization is necessary because one can not always know the clock status when enabling the counter. With EvCount=0, the counter will only start on the next positive clock edge after a previously latched negative edge, while the **Start** bit was already set to '1'. This synchronization is done differently if event count mode (bit **EvCount**) is chosen. Refer also to Figure 23. Internal Clock Synchronization.

8.3 Event Counting

The counter can be used in a special event count mode where a certain number of events (clocks) on the PA[0] or PA[3] input are counted. In this mode the counting will start directly on the next active clock edge on the selected port A input.
The Event Count mode is switched on by setting bit **EvCount** in the register **RegCCntl2** to '1'.PA[3] and PA[0] inputs can be inverted depending on register **OPTIntEdgPA** and should be debounced. The debouncer is switched on in register **OPTDebIntPA** bits NoDebIntPA[3,0]=0. Its frequency depends on the bit **DebSel** from register **RegPresc** setting. The inversion of the internal clock signal derived from PA[3] or PA[0] is active with **IntEdgPA[3]** respectively **IntEdgPA[0]** equal to 1. Refer also to Figure 9 for internal clock signal generation.

Figure 23. Internal Clock Synchronization



8.4 Compare Function

A previously loaded register value (**CReg[9:0]**) can be compared against the actual counter value (**Count[9:0]**). If the two are matching (equality) then an interrupt (**IRQComp**) is generated. The compare function is switched on with the bit **EnComp** in the register **RegCCntl2**. With **EnComp** = 0 no **IRQComp** is generated. Starting the counter with the same value as the compare register is possible, no IRQ is generated on start. Full or Limited bit compare are possible, defined by bit **SelIntFull** in register **RegSysCntl1**.
EnComp must be written after a load operation (**Load** = 1). Every load operation resets the bit EnComp.

Full bit compare function.
Bit **SelIntFull** is set to '1'. The function behaves as described above independent of the selected counter length. Limited bit counting together with full bit compare can be used to generate a certain amount of IRQCount0 interrupts until the counter generates the IRQComp interrupt. With **PWMOn**='1' the counter would have automatically stopped after the IRQComp, with **PWMOn**='0' it will continue until the software stops it. **EnComp** must be cleared before setting SelIntFull and before starting the counter again. Be careful, PWMOn also redefines the port B PB[3] output data.(refer to section 8.5).

Limited bit compare
With the bit **SelIntFull** set to '0' (default) the compare function will only take as many bits into account as defined by the counter length selection **BitSel[1:0]** (see chapter 8.1).

8.5 Pulse Width Modulation (PWM)

The PWM generator uses the behavior of the Compare function (see above) so **EnComp** must be set to activate the PWM function.. At each Roll Over or Compare Match the PWM state - which is output on port B PB[3] - will toggle. The start value on PB[3] is forced while **EnComp** is 0 the value is depending on the up or down count mode. Every counter value load operation resets the bit **EnComp** and therefore the PWM start value is reinstalled.
Setting **PWMOn** to '1' in register **RegPresc** routes the counter PWM output to port B terminal PB[3]. Insure that PB[3] is set to output mode . Refer to section 6.4 for the port B setup.
The PWM signal generation is independent of the limited or full bit compare selection bit **SelIntFull.** However if **SelIntFull** = 1 (FULL) and the counter compare function is limited to lower than 10 bits one can generate a predefined number of output pulses. In this case, the number of output pulses is defined by the value of the unused counter bits. It will count from the start value until the IRQComp match.
One must not use a compare value of hex 0 in up count mode nor a value of hex 3FF (or FF,3F, F if limited bit compare) in down count mode.

For instance, loading the counter in up count mode with hex 000 and the comparator with hex C52 which will be identified as :

- bits[11:10] are limiting the counter to limits to 4 bits length, =03 (BitSel[1,0])
- bits [9:4] are the unused counter bits = hex 05 (bin 000101), (number of PWM pulses)
- bits [3:0] (comparator value = 2). (length of PWM pulse)

Thus after 5 PWM-pulses of 2 clocks cycles length the Counter generates an **IRQComp** and stops.
The same example with SelIntFull=0 (limited bit compare) will produce an unlimited number of PWM at a length of 2 clock cycles.

8.5.1 How the PWM Generator works.

For Up Count Mode; Setting the counter in up count and PWM mode the PB[3] PWM output is defined to be 0 (**EnComp**=0 forces the PWM output to 0 in upcount mode, 1 in downcount). Each Roll Over will set the output to '1' and each Compare Match will set it back to '0'. The Compare Match for PWM always only works on the defined counter length. This, independent of the SelIntFull setting which is valid only for the IRQ generation. Refer also to the compare setup in chapter 8.4.

In above example the PWM starts counting up on hex 0,
2 cycles later compare match -> PWM to '0',
14 cycles later roll over -> PWM to '1'
2 cycles later compare match -> PWM to '0' , etc. until the completion of the 5 pulses.

The normal IRQ generation remains on during PWM output. If no IRQ's are wanted, the corresponding masks need to be set.

Figure 24. PWM Output in Up Count Mode

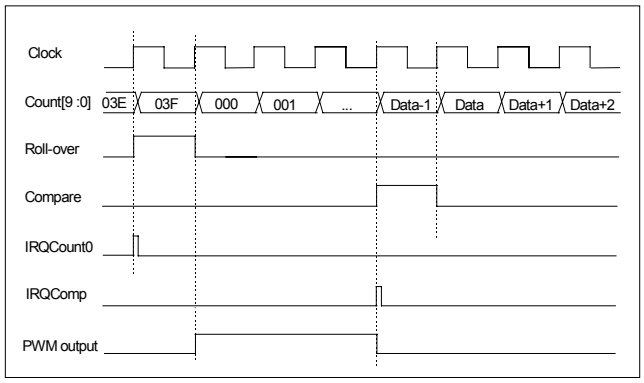
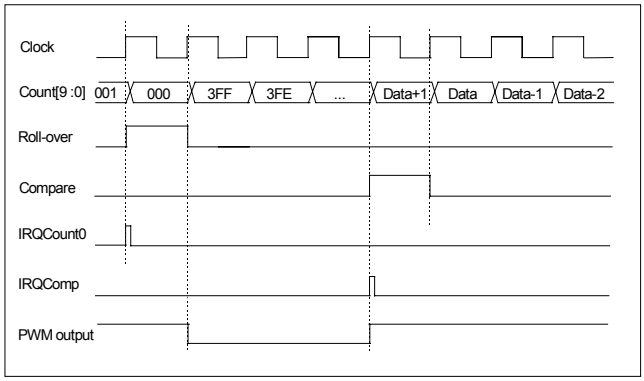


Figure 25. PWM Output in Down Count Mode



In Down Count Mode everything is inverted. The PWM output starts with the '1' value. Each Roll Over will set the output to '0' and each Compare Match will set it back to '1'. For limited pulse generation one must load the complementary pulse number value. I.e. for 5 pulses counting on 4 bits load bits[9 :4] with hex 3A (bin 111010).

8.5.2 PWM Characteristics

PWM resolution is	: 10bits (1024 steps), 8bits (256 steps), 6bits (64 steps) or 4 bits (16 steps)			
the minimal signal period is	: 16 (4-bit) x Fmax*	-> 16 x 1/Ck[15]	-> 977 µs	(32 KHz)
the maximum signal period is	: 1024 x Fmin*	-> 1024 x 1/Ck[1]	-> 1024 s	(32 KHz)
the minimal pulse width is	: 1 bit	-> 1 x 1/Ck[15]	-> 61 µs	(32 KHz)

* This values are for Fmax or Fmin derived from the internal system clock (32kHz). Much shorter (and longer) PWM pulses can be achieved by using the port A as frequency input.

One must not use a compare value of hex 0 in up count mode nor a value of hex 3FF (or FF,3F, F if limited bit compare) in downcount mode.

8.6 Counter Setup

RegCDataL[3:0], RegCDataM[3:0], RegCDataH[1:0] are used to store the initial count value called **CReg[9:0]** which is written into the count register bits **Count[9:0]** when writing the bit **Load** to '1' in **RegCCntl2**. This bit is automatically reset thereafter. The counter value **Count[9:0]** can be read out at any time, except when using non-debounced high frequency port A input clock. To maintain data integrity the lower nibble **Count[3:0]** must always be read first. The **ShCount[9:4]** values are shadow registers to the counter. To keep the data integrity during a counter read operation (3 reads), the counter values [9:4] are copied into these registers with the read of the count[3:0] register. If using non-debounced high frequency port A input the counter must be stopped while reading the **Count[3:0]** value to maintain the data integrity.
In down count mode an interrupt request **IRQCount0** is generated when the counter reaches 0. In up count mode, an interrupt request is generated when the counter reaches 3FF (or FF,3F,F if limited bit counting).

Never an interrupt request is generated by loading a value into the counter register.

When the counter is programmed from up into down mode or vice versa, the counter value **Count[9:0]** gets inverted. As a consequence, the initial value of the counter must be programmed after the **Up/Down** selection.

Loading the counter with hex 000 is equivalent to writing stop mode, the **Start** bit is reset, no interrupt request is generated.

How to use the counter;

If PWM output is required one has to put the port B[3] in output mode and set PWMOn=1 in step 5.

1st, set the counter into stop mode (**Start**=0).
2nd, select the frequency and up- or down count mode in **RegCCntl1.**
3rd, write the data registers **RegCDataL, RegCDataM, RegCDataH** (counter start value and length)
4th, load the counter, **Load**=1, and choose the mode. (**EvCount**, **EnComp**=0)
5th, select bits **PWMOn** in **RegPresc** and **SelIntFull** in **RegSysCntl1**
6th, if compare mode desired , then write **RegCDataL, RegCDataM, RegCDataH** (compare value)
7th, set bit **Start** and select **EnComp** in **RegCCntl2**

8.7 10-bit Counter Registers

Table 8.7.1 Register RegCCntl1

Bit	Name	Reset	R/W	Description
3	Up/Down	0	R/W	Up or down counting
2	CountFSel2	0	R/W	Input clock selection
1	CountFSel1	0	R/W	Input clock selection
0	CountFsel0	0	R/W	Input clock selection

Default : PA0 ,selected as input clock, Down counting

Table 8.7.2 Counter Input Frequency Selection with CountFSel[2..0]

CountFSel2	CountFSel1	CountFSel0	clock source selection
0	0	0	Port A PA[0]
0	0	1	Prescaler Ck[15]
0	1	0	Prescaler Ck[12]
0	1	1	Prescaler Ck[10]
1	0	0	Prescaler Ck[8]
1	0	1	Prescaler Ck[4]
1	1	0	Prescaler Ck[1]
1	1	1	Port A PA[3]

Table 8.7.3 Register RegCCntl2

Bit	Name	Reset	R/W	Description
3	Start	0	R/W	Start/Stop control
2	EvCount	0	R/W	Event counter enable
1	EnComp	0	R/W	Enable comparator
0	Load	0	R/W	Write: load counter register; Read: always 0

Default : Stop, no event count, no comparator, no load

Table 8.7.4 Register RegSysCntl1

Bit	Name	Reset	R/W	Description
3	*IntEn*	*0*	*R/W*	*General interrupt enable*
2	*SLEEP*	*0*	*R/W*	*Sleep mode*
1	SelIntFull	0	R/W	Compare Interrupt select
0	*ChTmDis*	*0*	*R/W*	*For EM test only*

Default : Interrupt on limited bit compare

Table 8.7.5 Register RegCDataL, Counter/Compare Low Data Nibble

Bit	Name	Reset	R/W	Description
3	CReg[3]	0	W	Counter data bit 3
2	CReg[2]	0	W	Counter data bit 2
1	CReg[1]	0	W	Counter data bit 1
0	CReg[0]	0	W	Counter data bit 0
3	Count[3]	0	R	Data register bit 3
2	Count[2]	0	R	Data register bit 2
1	Count[1]	0	R	Data register bit 1
0	Count[0]	0	R	Data register bit 0

Table 8.7.6 Register RegCDataM, Counter/Compare Middle Data Nibble

Bit	Name	Reset	R/W	Description
3	CReg[7]	0	W	Counter data bit 7
2	CReg[6]	0	W	Counter data bit 6
1	CReg[5]	0	W	Counter data bit 5
0	CReg[4]	0	W	Counter data bit 4
3	ShCount[7]	0	R	Data register bit 7
2	ShCount[6]	0	R	Data register bit 6
1	ShCount[5]	0	R	Data register bit 5
0	ShCount[4]	0	R	Data register bit 4

Table 8.7.7 Register RegCDataH, Counter/Compare High Data Nibble

Bit	Name	Reset	R/W	Description
3	BitSel[1]	0	R/W	Bit select for limited bit count/compare
2	BitSel[0]	0	R/W	Bit select for limited bit count/compare
1	CReg[9]	0	W	Counter data bit 9
0	CReg[8]	0	W	Counter data bit 8
1	ShCount[9]	0	R	Data register bit 9
0	ShCount[8]	0	R	Data register bit 8

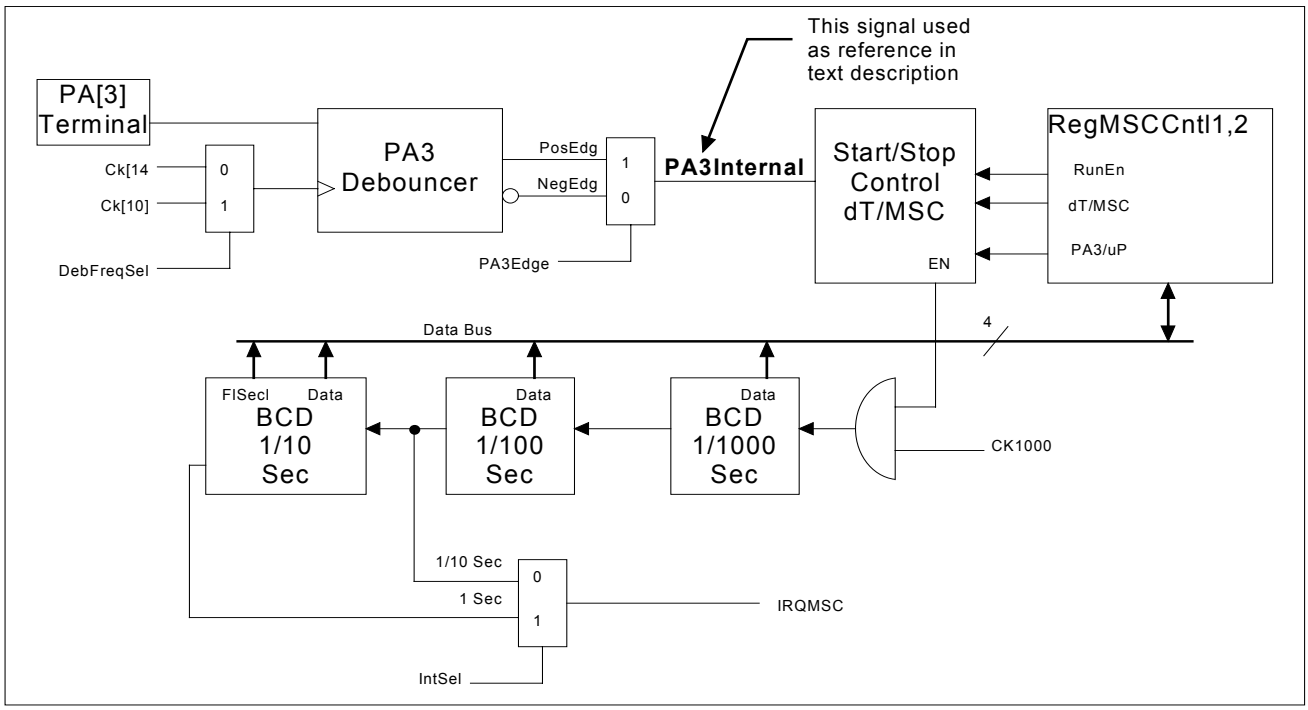
Table 8.7.8 Counter Length Selection

BitSel[1]	BitSel[0]	counter length
0	0	10-Bit
0	1	8-Bit
1	0	6-Bit
1	1	4-Bit

9. Millisecond Counter

The EM6626 has a built-in millisecond binary coded decimal counter. It can be used to measure the time elapsed between two events (hardware or software events). With a system clock of 32kHz, the counter generates every 1/10 second or every second an interrupt request.
The counter value read on registers **RegMSCDataL**, **RegMSCDataM** and **RegMSCDataH** is in binary coded decimal format (000 to 999). To maintain the data integrity for the 3 decimal digits inside **BCD[11:0]** one must stop the counter while reading the full 3 digit value.
An overflow flag **FISec** is set whenever the counter reached 999. This flag is helpful when the counter is used in polling mode and twice the same value is read. In this case, if the flag is set to 1, it indicates that the two readings were 1 second apart, in the case the flag is not set, the two readings must have been very short one after the other. After every read of **RegMSCCntl2** the **FISec** gets automatically reset.
The millisecond counter is reset with every system reset. Setting the **ResMSC** flag located in register **RegMSCCntl1** resets the counter value only. This flag is automatically reset after the write operation. For good resolution in Pa3-mode use the Ck[14] debouncer clock (250us). Or if the 1/1000 sec is not relevant then choose Ck[10] (4ms) as debouncer clock. Doing so will save power. The debouncer selection is made in register **RegMSCCntl2** bit **DebFreqSel.**

Figure 26. MSC Block Diagram



Changing **PA3Edge** while **RunEn**=1 or **PA3/up**=1 may generate a MSC event (start or stop). This behavior is useful for the - CPU controlled start and PA3 controlled stop - mode, But in general one does all the setup before starting the counter.

9.1 PA[3] Input for MSC

In hardware Start/Stop mode the counter is triggered with the port A terminal PA[3] input. In this case PA[3] is debounced with the prescaler Ck[14] (or Ck[10]) clock. The triggering edge selection is made with bit **PA3Edge** in register **RegMSCCntl2** (default negative edge). The PA[3] input for the millisecond counter is totally independent of the PA[3] interrupt edge selection and the PA[3] polarity selection for the 10 bit counter. However the pull-up or pull-down selection is common to all peripheries sharing the port A.

9.2 IRQ from MSC

An Interrupt request **IRQMSC** is send on either every 1/10 seconds or every second, depending on the bit **IntSel** in register **RegMSCCntl2**. For interrupt handling please refer to the interrupt control section.

9.3 MSC-Modes

The millisecond counter can have many different modes of operation. The most common are :

- CPU controlled start and stop.
- CPU controlled start and PA[3] controlled stop.
- Port A terminal PA[3] controlled start and stop mode.
- Pulse width measurement of port A terminal PA[3] input signals.

All these different modes are controlled with the bits in the registers **RegMSCCntl1** and **RegMSCCntl2.** The main bits are :

- **dT/MSC** ; Pulse-width or start stop measure. This bit only has a action if PA[3] input is chosen. If pulse-width measure is selected, the counter starts with the first active edge on PA[3] and stops with the next inverse edge (sets **RunEn** = 0). If MSC measure selected, the counter starts with the first active PA[3] edge, stops on the next, restarts on the following etc. It does not reset **RunEn**.

- **PA3/µP** ; Direct port A terminal PA[3] or CPU (µP) controlled start and stop function. If direct PA[3] controlled start stop mode is chosen the counter, once enabled by setting **RunEn/Stop** = 1, starts counting on the first active edge seen on PA[3]. It stops counting depending on the **dT/MSC** bit either on the next inverse edge or on the next active edge. If **µP** is chosen, the counter starts and stops depending on bit **RunEn/Stop**.

- **RunEn/Stop**; In CPU mode this bit starts or stops the counter. In PA3 mode it enables the counter which will start with the next event on port A terminal PA[3]. If **dT** and PA3 mode, the **RunEn** gets reset with the second active PA[3] edge.

- **PA3Edge** ; This bit selects the active PA[3] edge which will trigger the **dT/MSC** selected measurement mode. It has no effect if **PA3/µP**=0. Default 0 is negative edge.

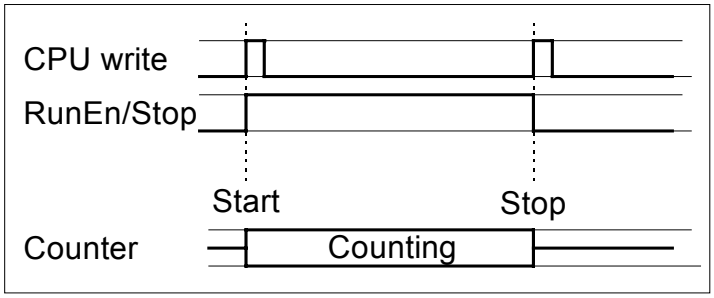
9.4 Mode selection

Before using, the MSC counter needs to be reset by setting bit **ResMSC** to '1'. This bit is automatically reset thereafter. Then select the IRQ frequency and the counting mode. Now the **RunEn** can be set to '1' . To display the counter value during run you may only want to read the MSB (1/10 sec) digit ,driven by IRQ or with polling, and fully read the MSC value only once the counter is stopped. The counter data registers are read only. Any Reset (system reset, POR, watchdog) is setting the MSC into stop mode and clears the counter registers.

- **CPU controlled Start and Stop**

As soon as the CPU writes the start bit **RunEn/Stop**=1 the counter starts up counting until the CPU clears the start bit. The bit **PA3/uP** is '0' for this mode.

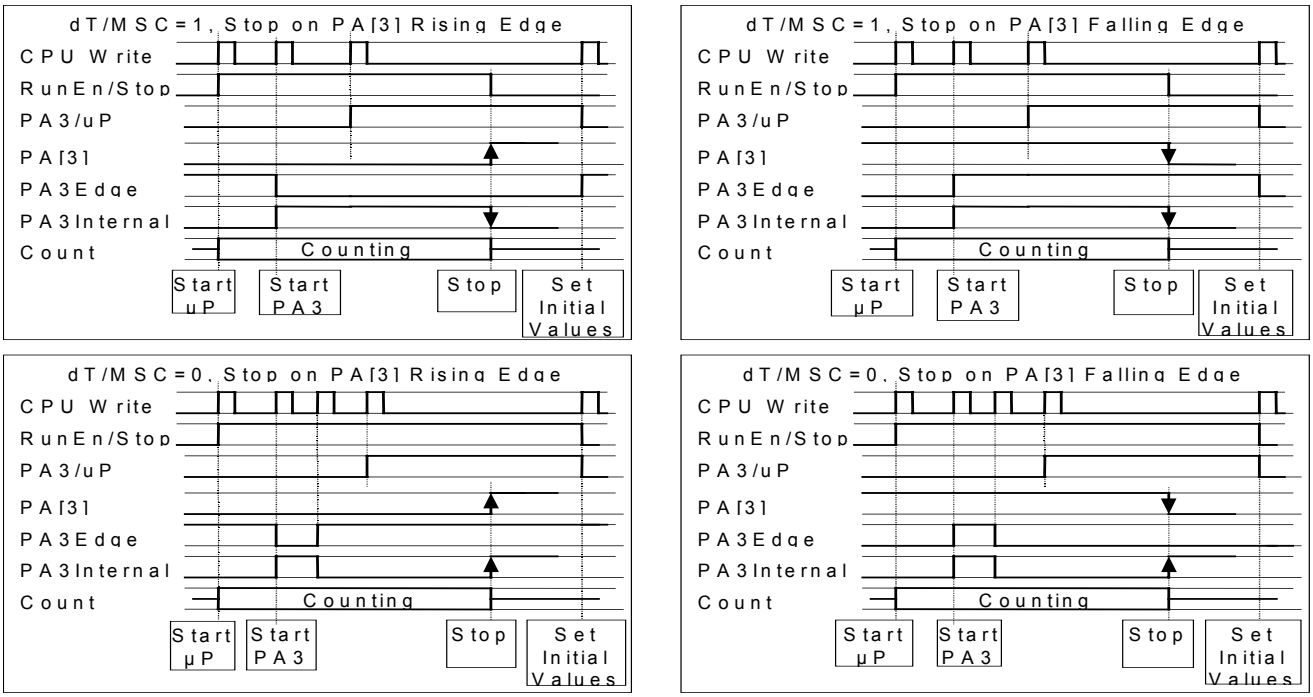
Figure 27. CPU controlled Start Stop



- **CPU controlled Start and PA[3] controlled Stop.**

In this mode setting the bit **RunEn**=1 while **PA3/uP**=0 while immediately start the counting action. Afterwards one needs to prepare for the stop by PA[3]. Therefore the PA[3] start condition must first be fulfilled. This is in **dT** mode a rising edge on the PA3internal signal (PA3internal, refer to Figure 26). In **MSC** mode the start condition is a positive pulse on PA3internal signal. The creation of this edge or pulse is done per software by manipulating the **PA3Edge** selection. See Figure 28 for details. Afterwards one can change to PA3 controlled stop mode (**PA3/uP**=1) where the next positive edge on PA3internal will stop the Counter. In **dT** mode the **RunEn/stop** bit will be cleared with the PA3 stop condition where as in MSC mode MSC mode the **RunEn** is not cleared.

Figure 28. CPU controlled Start PA[3] controlled Stop



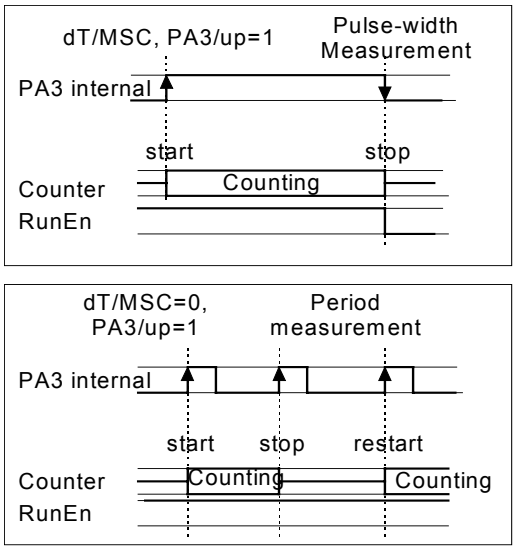
- **Pulse-width measurement of PA[3] Input Signals.**

In this mode the bit **dT/MSC**=1 and **PA3/uP**=1. Setting RunEn/stop=1 enables the operation. The first positive edge on PA3Internal signal will start the counter, the following negative edge will stop the counter end set bit **RunEn/Stop** to 0 . PA3internal signal is a copy of the PA[3] terminal status if **PA3Edge**=1. with **PA3Edge**=0 PA3Internal has the inverted PA[3] value. See also Figure 26 and Figure 29.

- **Port A PA[3] controlled Start and Stop Mode.**

In this mode the bit **dT/MSC**=0 and **PA3/uP**=1. Setting RunEn/stop=1 enables the operation. The first positive edge on PA3Internal signal will start the counter , the second edge will stop the counter, the third one will restart, etc, . PA3internal signal is a copy of the PA[3] terminal status if **PA3Edge**=1. With **PA3Edge**=0 PA3Internal has the inverted PA[3] value. See also Figure 26 and Figure 29.

Figure 29. dT/MSC behavior



9.5 Millisecond Counter Registers

Table 9.5.1 Register RegMSCCntl1

Bit	Name	Reset	R/W	Description
3	RunEn/Stop	0	R/W	Enable counter
2	PA3/µP	0	R/W	Port A or CPU start stop control
1	dT/MSC	0	R/W	Pulse-width measurement
0	ResMSC	0	R/W	Reset if write of 1 Read value is always 0

Default: Stop, CPU controlled.

Table 9.5.2 Register RegMSCCntl2

Bit	Name	Reset	R/W	Description
3	DebFreqSel	0	R/W	Debouncer frequency select
2	PA3Edge	0	R/W	PA[3] edge selection
1	IntSel	0	R/W	Interrupt source selection
0	FlSec	0	R	Seconds flag

Default: Ck[14] is debouncer clock, negative edge, 1/10 Sec Interrupt requests

Table 9.5.3 Register RegMSCDataL

Bit	Name	Reset	R/W	Description
3	BCD[3]	0	R	1/1000 Seconds BCD value 3
2	BCD[2]	0	R	1/1000 Seconds BCD value 2
1	BCD[1]	0	R	1/1000 Seconds BCD value 1
0	BCD[0]	0	R	1/1000 Seconds BCD value 0

Table 9.5.4 Register RegMSCDataM

Bit	Name	Reset	R/W	Description
3	BCD[7]	0	R	1/100 Seconds BCD value 3
2	BCD[6]	0	R	1/100 Seconds BCD value 2
1	BCD[5]	0	R	1/100 Seconds BCD value 1
0	BCD[4]	0	R	1/100 Seconds BCD value 0

Table 9.5.5 Register RegMSCDataH

Bit	Name	Reset	R/W	Description
3	BCD[11]	0	R	1/10 Seconds BCD value 3
2	BCD[10]	0	R	1/10 Seconds BCD value 2
1	BCD[9]	0	R	1/10 Seconds BCD value 1
0	BCD[8]	0	R	1/10 Seconds BCD value 0

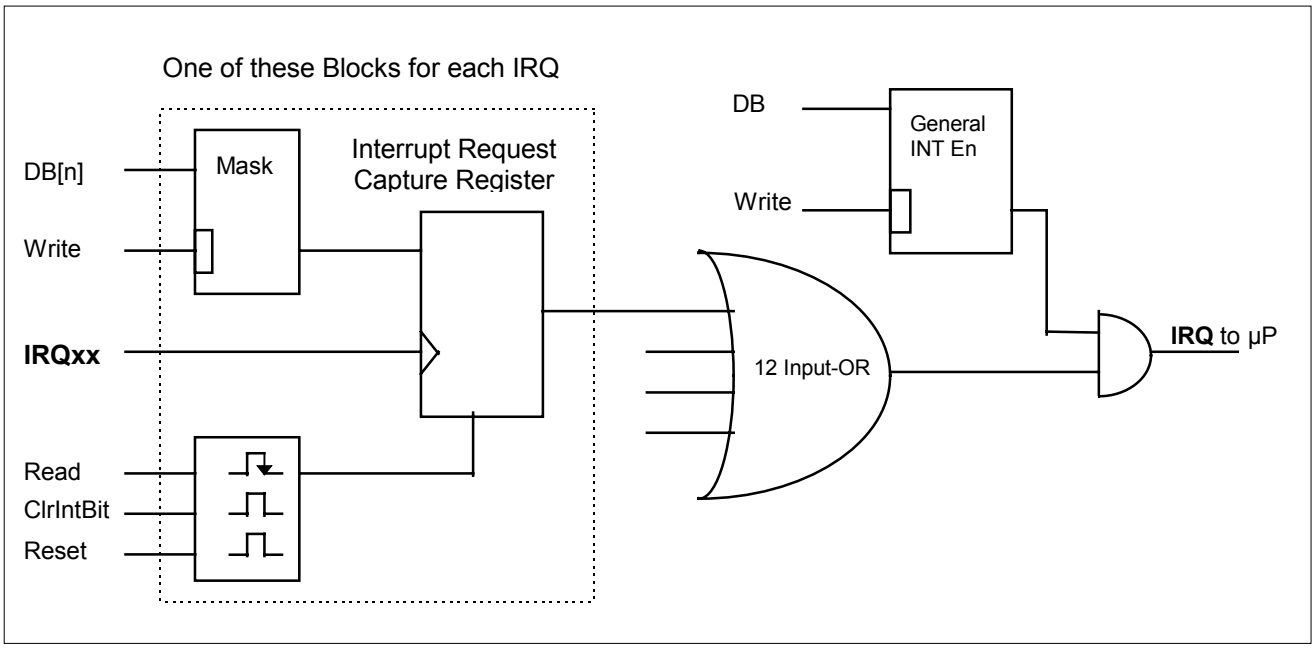
10. Interrupt Controller

The EM6626 has 12 different interrupt request sources, each of which is maskable. Five of them come from external sources and seven from internal sources.

External(4)	**- Port A,**	**PA[3] .. PA[0] inputs**
	- Serial Interface	
Internal(8)	**- Prescaler**	**Ck[1], Blink, 32Hz/8Hz**
	- Melody timer	
	- Serial Interface	
	- Millisecond-Counter	**1/10Sec or 1Sec**
	- 10-bit Counter	**Count0, CountComp**

To be able to send an interrupt to the CPU, at least one of the interrupt request flags must '1' (**IRQxx**) and the general interrupt enable bit **IntEn** located in the register **RegSysCntl1** must be set to 1. The interrupt request flags can only be set high by a positive edge on the **IRQxx** data flip-flop while the corresponding mask register bit (**MaskIRQxx**) is set to 1.

Figure 30. Interrupt Controller Block Diagram



At power on or after any reset all interrupt request mask registers are cleared and therefore do not allow any interrupt request to be stored. Also the general interrupt enable **IntEn** is set to 0 (No IRQ to CPU) by reset.

After each read operation on the interrupt request registers **RegIRQ1**, **RegIRQ2** or **RegIRQ3** the contents of the addressed register are reset. Therefore one has to make a copy of the interrupt request register if there was more than one interrupt to treat. Each interrupt request flag may also be reset individually by writing 1 into it .

Interrupt handling priority must be resolved through software by deciding which register and which flag inside the register need to be serviced first.

Since the CPU has only one interrupt subroutine and the **IRQxx** registers are cleared after reading, the CPU does not miss any interrupt request which comes during the interrupt service routine. If any occurs during this time a new interrupt will be generated as soon as the software comes out of the current interrupt subroutine.

Any interrupt request sent by a periphery cell while the corresponding mask is not set will not be stored in the interrupt request register. All interrupt requests are stored in their **IRQxx** registers depending only on their mask setting and not on the general interrupt enable status.
Whenever the EM6626 goes into halt mode the **IntEn** bit is automatically set to 1, thus allowing to resume from halt mode with an interrupt.

10.1 Interrupt Control Registers

Table 10.1.6 Register RegIRQ1

Bit	Name	Reset	R/W	Description
3	IRQPA[3]	0	R *	Port A PA[3] interrupt request
2	IRQPA[2]	0	R *	Port A PA[2] interrupt request
1	IRQPA[1]	0	R *	Port A PA[1] interrupt request
0	IRQPA[0]	0	R *	Port A PA[0] interrupt request

*; Writing of 1 clears the corresponding bit.

Table 10.1.7 Register RegIRQ2

Bit	Name	Reset	R/W	Description
3	IRQHz1	0	R *	Prescaler interrupt request
2	IRQHz32/8	0	R *	Prescaler interrupt request
1	IRQBlink	0	R *	Prescaler interrupt request
0	IRQBz	0	R *	Melody timer interrupt request

*; Writing of 1 clears the corresponding bit.

Table 10.1.8 Register RegIRQ3

Bit	Name	Reset	R/W	Description
3	IRQSerial	0	R *	Serial interrupt request
2	IRQMSC	0	R *	Millisecond counter int. request
1	IRQCount0	0	R *	Counter interrupt request
0	IRQCntComp	0	R *	Counter interrupt request

*; Writing of 1 clears the corresponding bit.

Table 10.1.9 Register RegIRQMask1

Bit	Name	Reset	R/W	Description
3	MaskIRQPA[3]	0	R/W	Port A PA[3] interrupt mask
2	MaskIRQPA[2]	0	R/W	Port A PA[2] interrupt mask
1	MaskIRQPA[1]	0	R/W	Port A PA[1] interrupt mask
0	MaskIRQPA[0]	0	R/W	Port A PA[0] interrupt mask

Interrupt is not stored if the mask bit is 0.

Table 10.1.10 Register RegIRQMask2

Bit	Name	Reset	R/W	Description
3	MaskIRQHz1	0	R/W	Prescaler interrupt mask
2	MaskIRQHz32/8	0	R/W	Prescaler interrupt mask
1	MaskIRQBlink	0	R/W	Prescaler interrupt mask
0	MaskIRQBz	0	R/W	Melody timer interrupt mask

Interrupt is not stored if the mask bit is 0.

Table 10.1.11 Register RegIRQMask3

Bit	Name	Reset	R/W	Description
3	MaskIRQSerial	0	R/W	Serial interrupt mask
2	MaskIRQMSC	0	R/W	Millisecond counter int. mask
1	MaskIRQCount0	0	R/W	Counter interrupt mask
0	MaskIRQCntComp	0	R/W	Counter interrupt mask

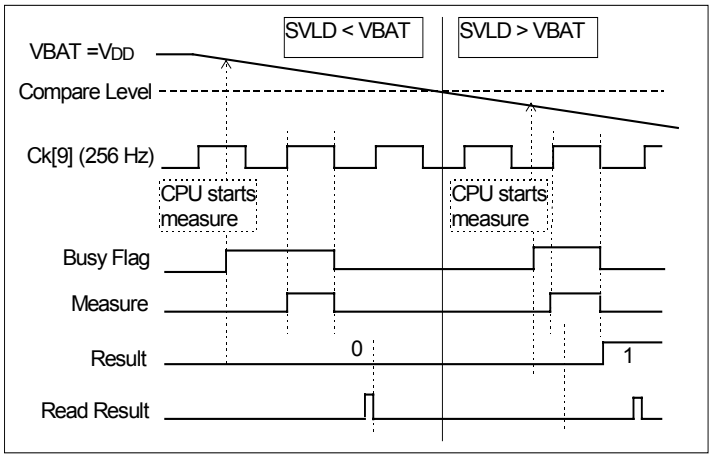
Interrupt is not stored if the mask bit is 0

11. Supply Voltage Level Detector

The EM6626 has a built-in Supply Voltage Level Detector (SVLD) circuitry, such that the CPU can compare the supply voltage against a pre-selected value. During sleep mode this function is inhibited.

The CPU activates the supply voltage level detector by writing **VldStart** = 1 in the register **RegVldCntl**. The actual measurement starts on the next Ck[9] rising edge and lasts during the Ck[9] high period (2 ms at 32 KHz). The busy flag **VldBusy** stays high from **VldStart** set until the measurement is finished. The worst case time until the result is available is 1.5 Ck[9] prescaler clock periods (32 KHz -> 6 ms). The detection level must be defined in register **RegVldLevel** before the **VldStart** bit is set.
During the actual measurement (2 ms) the device will draw an additional 5 µA of I_{VDD} current. After the end of the measure the result is available by inspection of the bit **VldResult**. If the result is read 0, then the power supply voltage was greater than the detection level value. If read 1, the power supply voltage was lower than the detection level value. During each read while **Busy=1** the **VldResult** is not guaranteed.

Figure 31. SVLD Timing Diagram



11.1 SVLD Register

Table 11.1.1 Register RegVldCntl

Bit	Name	Reset	R/W	Description
3	VldResult	0	R*	Vld result flag
2	VldStart	0	W	Vld start
2	VldBusy	0	R	Vld busy flag
1	*NoOscWD*	*0*	*R/W*	*No Oscillator watchdog*
0	*NoLogicWD*	*0*	*R/W*	*No logic watchdog*

R*; Read value while VLDBusy=1 is not guaranteed.

Table 11.1.2 Register RegVldLevel (Detection Level Value)

Bit	Name	Reset	R/W	Description
3	--	x	--	not active
2	VldLevel2	0	R/W	Vld level selection
1	VldLevel1	0	R/W	Vld level selection
0	VldLevel0	0	R/W	Vld level selection

Table 11.1.3 Voltage Level Detector Value Selecting

	VldLevel2	VldLevel1	VldLevel0	Typical voltage level
Level1	0	0	0	4.00
Level2	0	0	1	2.95
Level3	0	1	0	2.35
Level4	0	1	1	1.95
Level5	1	0	0	1.70
Level6	1	0	1	1.45
Level7	1	1	0	1.30
Level8	1	1	1	1,20

12. Strobe Output

The Strobe output is used to indicate either the EM6626 reset condition, a write operation on port B (WritePB) or the sleep mode. The selection is done in register **RegLcdCntl1**. Per default, the reset condition is output on the Strobe terminal.
For a port B write operation the strobe signal goes high for half a system clock period. Data can be latched on the falling edge of the strobe signal. This function is used to indicate when data on port B output terminals is changing.
The reset signal on the Strobe output is a copy of the internal CPU reset signal. The Strobe pin remains active high as long as the CPU gets the reset.
Both the reset condition and the port B write operation can be output simultaneously on the Strobe pin.

The strobe output select latches are reset by initial power on reset only.

Figure 32 . Strobe Output

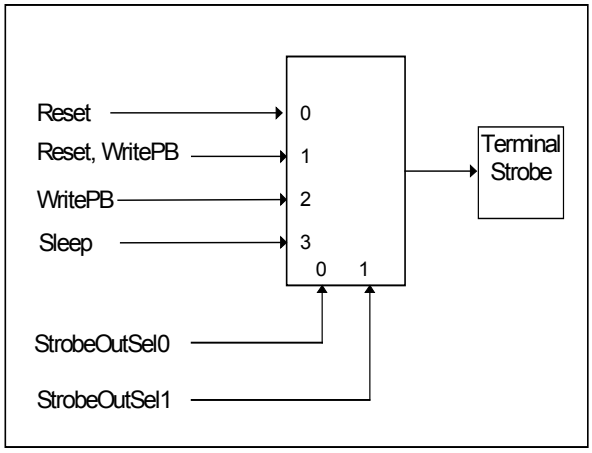


Table 11.1.1. Strobe Output Selection

StrobeOutSel1	StrobeOutSel0	Strobe Terminal Output
0	0	System Reset
0	1	System Reset and WritePB
1	0	WritePB
1	1	Sleep

12.1 Strobe Register

Table 12.1.1 Register RegLCDCntl1

Bit	Name	power on value	R/W	Description
3	StrobeOutSel1	0	R/W	Strobe output select
2	StrobeOutSel0	0	R/W	Strobe output select
1	*CkTripSel1*	*0*	*R/W*	*LCD multiplier clock select*
0	*CkTripSel0*	*0*	*R/W*	*LCD multiplier clock select*

The CKTripSel1, CKTripSel0 values are reset with every system reset.

13. RAM

The EM6626 has two 64x4 bit RAM's built-in.

The main RAM (RAM1) is direct addressable on addresses decimal(0 to 63). A second RAM (RAM2) is indirect addressable on addresses 64,65, 66 and 67 together with the index from RegIndexAdr.

Figure 33. Ram Architecture

64 x 4 direct addressable RAM1	
RAM1_63	4 bit R/W
RAM1_62	4 bit R/W
RAM1_61	4 bit R/W
RAM1_60	4 bit R/W
. . .	. . .
RAM1_3	4 bit R/W
RAM1_2	4 bit R/W
RAM1_1	4 bit R/W
RAM1_0	4 bit R/W

64 x 4 indexed addressable RAM2		
RAM2_3	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W
RAM2_2	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W
RAM2_1	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W
RAM2_0	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W

The RAM2 addressing is indirect using the **RegIndexAdr** value as an offset to the directly addressed base **RAM2_0**, **RAM2_1** , **RAM2_2 or RAM2_3** registers.

To write or read the RAM2 the user has first to set the offset value in the **RegIndexAdr** register. The actual access then is made on the RAM2 base addresses **RAM2_0 , RAM2_1, RAM2_2** or **RAM2_3.** Refer to Figure 33. Ram Architecture, for the address mapping.

i.e. Writing hex(5) to Ram2 add location 30: First write hex(E) to RegIndexAdr, then write hex(5) to RAM2_1

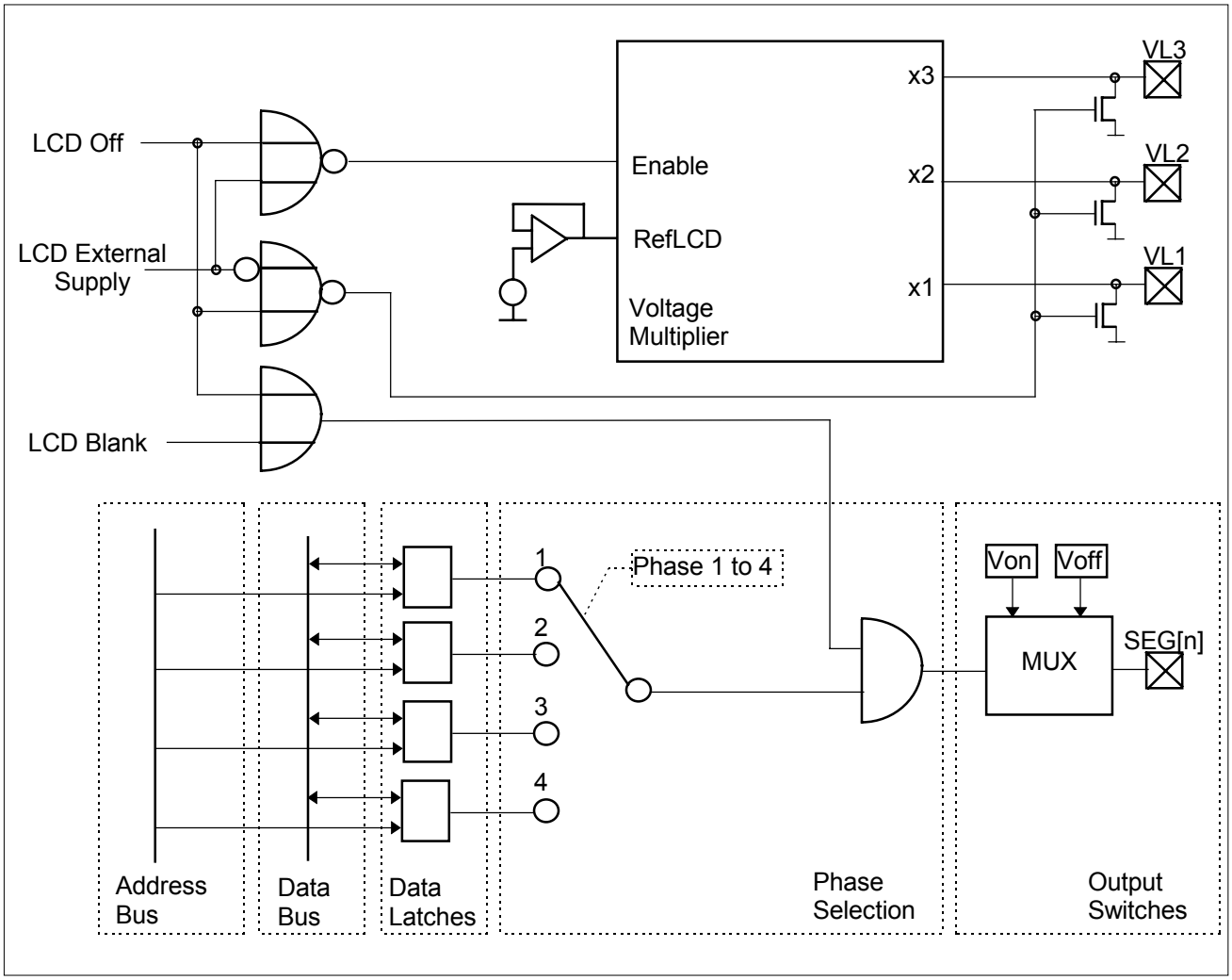
RAM Extension : Unused R/W Registers can often be used as possible RAM extension. Be careful not to use register which start, stop, or reset some functions. Unused LCD register latches can also be used as RAM memory.
In case of 3 times multiplex and using all the 20 Segment outputs you may have five additional 4 bit registers.. Also for each unused Segment output you may have one additional 4 bit register.

14. LCD Driver

The EM6626 has a built-in Liquid Crystal Display (LCD) driver. A maximum of 128 Segments can be displayed using the 32 Segment driver outputs (SEG[32:1) in 4:1 multiplex ,96 Segments in the case of 3:1 multiplex, and the 4 back-planes (COM[4:1]).
The LCD driver has its own voltage regulator (1.05 Volt) and voltage multiplier to generate the driver bias voltages VL1, VL2 and VL3 (VLCD). Using the metal1 mask the user can choose higher LCD reference voltages. Please check with EM Microelectronic the possible values and their impact on power consumption.

The special architecture of this LCD driver allows the user to freely specify the data and address for each individual Segment using the interconnect metal2 mask . It therefore adapts to every possible LCD display with a maximum of 128 independent segments. The LCD clock frequency is by default 256Hz. Other two possible options are 341Hz or 512Hz (refer to table 18.1.8). Thus the frame frequency is for instance 256/8 Hz if 4:1 multiplex, or 256/6 if 3:1 multiplex.

Figure 34. LCD Architecture



14.1 LCD Control

The LCD driver has two control registers **RegLCDCntl1**, **RegLCDCntl2** to optimize for display contrast, power consumption, operation mode and bias voltage source.

LCDExtSupply: Choosing external supply (**LCDExtSupply** ='1') disables the internal LCD voltage regulator and voltage multiplier, it also puts the bias voltage terminals VL1, VL2 and VL3 into high impedance state. External bias levels can now be connected to VL1, VL2 and VL3 terminals. (Resistor divider chain or others).

Another way to adapt the VL1, VL2 and VL3 levels to specific user needs is to overdrive the VL1 output **(LCDExtSupply** =0) with the desired value. The internal multiplier will multiply this new VL1 level to generate the corresponding levels VL2 and VL3. The bit **LCDExtSupply** is only reset by initial POR.

LCD4Mux: With this switch one selects either 3:1 or 4:1 (default) times multiplexing of the 32 Segment driver outputs. In the case of 3:1 multiplexing the COM[4] is off.

LCDOff: Disables the LCD. The voltage multiplier and regulator are switched off (0 current).The Segment latch information is maintained. The VL1,VL2 and VL3 outputs are pulled to VSS.

LCDBlank: All Segment outputs are turned off. The voltage multiplier and regulator remain switched on. **LCDBlank** can be used with the 1Hz and Blink interrupt to let the whole display blink (software controlled).

CkTripSel1,0: Selecting the appropriate voltage multiplier frequency to optimize display contrast and power consumption. This value to use is also depending on the selected multiplier booster capacitors (typically 100nF).

14.2 LCD Addressing

The LCD driver addressing is indirect using the **RegIndexAdr** value as an offset to the directly addressed base **LCD_1**, **LCD_2** or **LCD_3** registers. All LCD Segment registers are R/W.
At address LCD_3 only the first 8 Index locations are usable. The Index locations hex(8 to F) are non implemented.
A total of 40 addresses are available to the user to freely define the addressing of the LCD Segment latches. For each of these latches the user may choose the address and data to be connected. See also section 14.3. However only 32x4 LCD Segment latches are implemented. The unused address locations are empty and can not be used as RAM.

Figure 35. LCD Address Mapping

40 x 4 Indexed Addressable LCD Latches but Maximum 32x4 Bits are R/W		
LCD_3	RegIndexAdr[8]	4 bit R/W
	RegIndexAdr[7]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W
LCD_2	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W
LCD_1	RegIndexAdr[F]	4 bit R/W
	RegIndexAdr[E]	4 bit R/W
	...	...
	RegIndexAdr[1]	4 bit R/W
	RegIndexAdr[0]	4 bit R/W

14.3 Free Segment Allocation

Each Segment (SEG[32:1]) terminal outputs the time multiplexed information from its 4 Segment data latches. Information stored in latch 1 is output during phase1, latch 2 during phase 2, latch 3 during phase 3 and latch 4 during phase 4. In the case of 3 to 1 multiplexing the phase 4 and the latch 4 are not used. This phase information on the segment outputs together with the common outputs (COM[4:1]) - also called back-planes - defines if a given LCD segment is light or not. COM[1] is on during phase 1 and off during phase 2,3,4 , COM[2] is on during phase 2 and off during phase 1,3,4 , etc.

For each segment data latch the address location within the LCD address spacing (**LCD_3 + Index(8), LCD_2 + Index(16), LCD_1 + Index(16) --> LCDAdr[39:0]**) can be user defined.
For each segment data latch the data bus connection (DB[3:0]) can be user defined.

Table 14.3.1 Default LCD Configuration

Segment outputs	COM[1] = phase1	COM[2] = phase2	COM[3] = phase3	COM[4] = phase4
SEG[1]	DB[0], LCDAdr[0]	DB[1], LCDAdr[0]	DB[2], LCDAdr[0]	DB[3], LCDAdr[0]
SEG[2]	DB[0], LCDAdr[1]	DB[1], LCDAdr[1]	DB[2], LCDAdr[1]	DB[3], LCDAdr[1]
SEG[3]	DB[0], LCDAdr[2]	DB[1], LCDAdr[2]	DB[2], LCDAdr[2]	DB[3], LCDAdr[2]
...	...	...	...	...
SEG[30]	DB[0], LCDAdr[30]	DB[1], LCDAdr[30]	DB[2], LCDAdr[30]	DB[3], LCDAdr[30]
SEG[31]	DB[0], LCDAdr[31]	DB[1], LCDAdr[31]	DB[2], LCDAdr[31]	DB[3], LCDAdr[31]
SEG[32]	DB[0], LCDAdr[32]	DB[1], LCDAdr[32]	DB[2], LCDAdr[32]	DB[3], LCDAdr[32]

14.4 LCD Registers

Table 14.4.1 Register RegLcdCntl1

Bit	Name	Reset	R/W	Description
3	*StrobeOutSel1*	*POR to '0'*	*R/W*	*Strobe output select*
2	*StrobeOutSel0*	*POR to '0'*	*R/W*	*Strobe output select*
1	CkTripSel1	0	R/W	LCD multiplier clock select
0	CkTripSel0	0	R/W	LCD multiplier clock select

StrobeOutSel1,0 is reset by initial power on only.

Table 14.4.2 Multiplier Clock Frequency Select

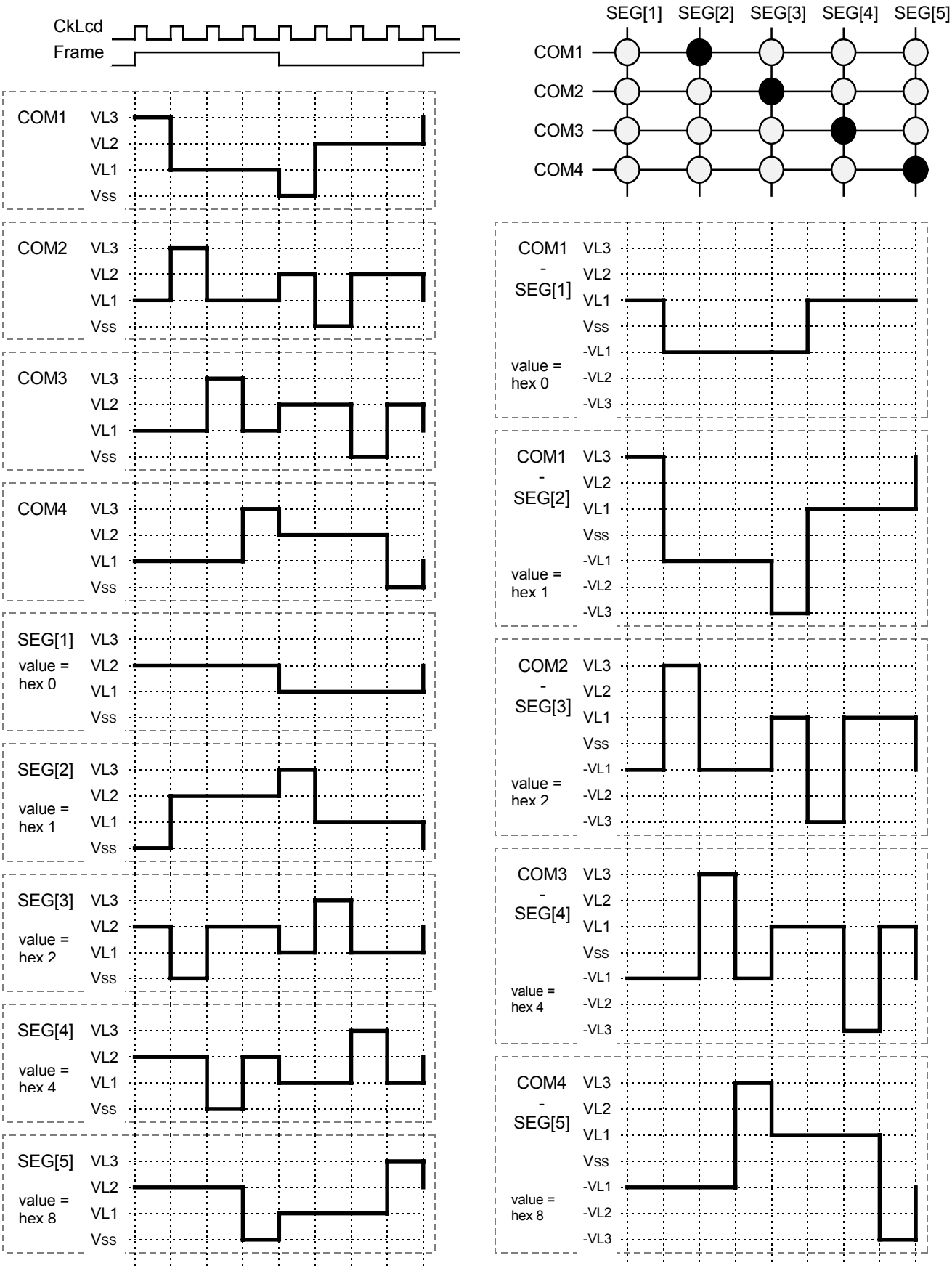
CkTripSel0	CkTripSel1	Multiplier Clock	on 32 KHz operation
0	0	Ck[10]	512 Hz
1	0	Ck[9]	256 Hz
0	1	Ck[8]	128 Hz
1	1	Ck[7]	64 Hz

Table 14.4.3 Register LcdCntl2

Bit	Name	Reset	R/W	Description
3	LCDBlank	1	R/W	LCD Segment outputs off
2	LCDOff	1	R/W	LCD off (multiplier off)
1	LCD4Mux	1	R/W	4 : 1 multiplexed
0	LCDExtSupply	POR to '0'	R/W	External supply for VL1, VL2 and VL3

LCDExtSupply is reset to '0' by POR only.

Figure 36 LCD Multiplexing Waveform



15. Peripheral Memory Map

Reset values are valid after power up or after every system reset.

Register Name	Add Hex	Add Dec.	Reset Value	Read Bits	Write Bits	Remarks
			b'3210	Read / Write Bits		
Ram1_0	00	0	xxxx	0: Data0 1: Data1 2: Data2 3: Data3		Normal addressable Ram 64x4 bit
...	...	...	...			...
Ram1_63	3F	63	xxxx	0: Data0 1: Data1 2: Data2 3: Data3		Normal addressable Ram 64x4 bit
Ram2_0	40	64	xxxx	0: Data0 1: Data1 2: Data2 3: Data3		16 nibbles addressable over index register on add 'H70
...	...	...	...			
Ram2_3	43	67	xxxx	0: Data0 1: Data1 2: Data2 3: Data3		16 nibbles addressable over index register on add 'H70
LCD_1	44	68	xxxx	Connections are user definable. See LCD section		16 nibbles addressable over index register on add 'H70
LCD_2	45	69	xxxx	Connections are user definable. See LCD section		16 nibbles addressable over index register on add 'H70
LCD_3	46	70	xxxx	Connections are user definable. See LCD section		The 8 lower nibbles are addressable over the index register on add 'H70. The 8 higher Nibbles are not used and not implemented
---	47	71				Reserved, not implemented
...	...	...				...
---	4F	79				Reserved, not implemented
RegPA	50	80	xxxx	0: PAData[0] 1: PAData[1] 2: PAData[2] 3: PAData[3]	----	Read port A directly
RegPBCntl	51	81	0000	0: PBIOCntl[0] 1: PBIOCntl[1] 2: PBIOCntl[2] 3: PBIOCntl[3]		Port B control Default: input mode
RegMSCCntl1	60	96	0000	0: '0' 1: dT/MSC 2: PA3/µP 3:RunEn/Stop	0: ResMSC 1: dT/MSC 2: PA3/µP 3:RunEn/Stop	millisecond counter control register 1; reset, delta time, control source

Register Name	Add Hex	Add Dec.	Reset Value	Read Bits	Write Bits	Remarks

			b'3210	Read / Write Bits		
RegMSCCntl1	60	96	0000	0: '0' 1: dT/MSC 2: PA3/µP 3:RunEn/Stop	0: ResMSC 1: dT/MSC 2: PA3/µP 3:RunEn/Stop	millisecond counter control register 1; reset, delta time, control source
RegMSCCntl2	61	97	0000	0: FlSec 1: IntSel 2: PA3Edge 3: DebFreqSel	0: -- 1: IntSel 2: PA3Edge 3: DebFreqSel	Millisecond counter control register 2; 1 sec flag, Interrupt and PA3 edge select
RegMSCDataL	62	98	0000	0: BCD[0] 1: BCD[1] 2: BCD[2] 3: BCD[3]	0: - 1: - 2: - 3: -	Millisecond counter; binary coded decimal value, low nibble
RegMSCDataM	63	99	0000	0: BCD[4] 1: BCD[5] 2: BCD[6] 3: BCD[7]	0: - 1: - 2: - 3: -	Millisecond counter; binary coded decimal value, middle nibble
RegMSCDataH	64	100	0000	0: BCD[8] 1: BCD[9] 2: BCD[10] 3: BCD[11]	0: - 1: - 2: - 3: -	Millisecond counter; binary coded decimal value, high nibble
RegIRQMask1	65	101	0000	0: MaskIRQPA[0] 1: MaskIRQPA[1] 2: MaskIRQPA[2] 3: MaskIRQPA[3]		Port A interrupt mask; masking active 0
RegIRQMask2	66	102	0000	0: MaskIRQBz 1: MaskIRQBlink 2: MaskIRQHz32/8 3: MaskIRQHz1		Buzzer and prescaler interrupt mask; masking active low
RegIRQMask3	67	103	0000	0: MaskIRQCntComp 1: MaskIRQCount0 2: MaskIRQMSC 3: MaskIRQSerial		10-bit counter, millisecond counter, serial interrupt mask masking active low
RegIRQ1	68	104	0000	0: IRQPA[0] 1: IRQPA[1] 2: IRQPA[2] 3:IRQPA[3]	0:RIRQPA[0] 1:RIRQPA[1] 2:RIRQPA[2] 3:RIRQPA[3]	Read: port A interrupt Write: Reset IRQ if data bit = 1.
REgIRQ2	69	105	0000	0: IRQBz 1: IRQBlink 2: IRQHz32/8 3: IRQHz1	0:RIRQBz 1:RIRQBlink 2:RIRQHz32/8 3:RIRQHz1	Read: buzzer and prescaler IRQ ; Write: Reset IRQ id data bit = 1
RegIRQ3	6A	106	0000	0:IRQCntComp 1: IRQCount0 2: IRQMSC 3: IRQSerial	0:RIRQCntComp 1:RIRQCount0 2:RIRQMSC 3:RIRQSerial	Read: 10-bit counter, millisecond counter, serial interrupt Write: Reset IRQ if data bit =1.
RegSysCntl1	6B	107	0000	*0: ChTmDis* 1: SelIntFull 2: '0' 3: IntEn	*0: ChTmDis* 1: SelIntFull 2: Sleep 3: IntEn	System control 1 *ChTmDis only usable only for EM test modes with Test=1*
RegSysCntl2	6C	108	0p00 p = POR	0: WDVal0 1: WDVal1 2: SleepEn 3: '0'	0: -- 1: -- 2: SleepEn 3: WDReset	System control 2; watchdog value and periodical reset, enable sleep mode
RegPresc	6D	109	0000	0: DebSel 1: PrIntSel 2: '0' 3: PWMOn	0: DebSel 1: PrIntSel 2: ResPresc 3: PWMOn	Prescaler control; debouncer and prescaler interrupt select

Register Name	Add Hex	Add Dec.	Reset Value	Read Bits	Write Bits	Remarks
			b'3210	Read / Write Bits		

IXLow	6E	110	xxxx	0: IXLow[0] 1: IXLow[1] 2: IXLow[2] 3: IXLow[3]	Internal µP index register low nibble; for µP indexed addressing	
IXHigh	6F	111	xxxx	0: IXHigh[4] 1: IXHigh[5] 2: IXHigh[6] 3: '0'	0: IXHigh[4] 1: IXHigh[5] 2: IXHigh[6] 3: --	Internal µP index register high nibble; for µP indexed addressing
RegIndexAdr	70	112	0000	0: IndexAdr[0] 1: IndexAdr[1] 2: IndexAdr[2] 3: IndexAdr[3]		Indexed addressing register for 4x16 nibble RAM2 and 3x16 + 8 nibble LCD
RegLCDCntl1	71	113	PP00	0: CkTripSel0 1: CkTripSel1 2: StrobeOutSel0 3: StrobeOutSel1		LCD control 0; multiplier clock and strobe output select
RegLCDCntl2	72	114	111P	0: LCDExtSupply 1: Lcd4xMux 2: LCDOff 3: LCDBlank		LCD control 1; main selects
RegVldCntl	73	115	0000	0: NoLogicWD 1: NoOscWD 2: VldBusy 3: VldResult	0: NoLogicWD 1: NoOscWD 2: VldStart 3: --	Voltage level detector control
RegVldLevel	74	116	x000	0: VldLevel0 1: VldLevel1 2: VldLevel2 3: --		Voltage level detector; detection level selection

P = defined by POR (power on reset)

16. Option Register Memory Map

The values of the option registers are set by initial reset on power up and through write operations only. Other resets ; as reset from watchdog, reset from input port A, reset from pin RESET, etc. do not change the options register value.

Register Name	Add Hex	Add Dec.	Reset Value b'3210	Read Bits	Write Bits	Remarks
				Read / Write Bits		
OPTDebIntPA OPT[3:0]	75	117	0000	0: NoDebIntPA[0] 1: NoDebIntPA[1] 2: NoDebIntPA[2] 3: NoDebIntPA[3]		Debouncer on port A for interrupt gen. Default: debouncer on
OPTIntEdgPA OPT[7:4]	76	118	0000	0: IntEdgPA[0] 1: IntEdgPA[1] 2: IntEdgPA[2] 3: IntEdgPA[3]		Interrupt edge select on port A. Default: pos. edge
OPTNoPullPA OPT[11:8]	77	119	0000	0: NoPullPA[0] 1: NoPullPA[1] 2: NoPullPA[2] 3: NoPullPA[3]		Pull-down selection on port A Default: pull-down
OPTNoPdPB OPT[15:12]	78	120	0000	0: NoPdPB[0] 1: NoPdPB[1] 2: NoPdPB[2] 3: NoPdPB[3]		Pull-down selection on port B Default: pull-down
OPTNchOpDPB OPT[19:16]	79	121	0000	0: NchOpDPB[0] 1: NchOpDPB[1] 2: NchOpDPB[2] 3: NchOpDPB[3]		Nch. open drain output on port B Default: CMOS output
OPTNchOpDPS OPT[23:20]	7A	122	0000	0: NchOpDPS[0] 1: NchOpDPS[1] 2: NchOpDPS[2] 3: NchOpDPS[3]		Nch. open drain output on port serial Default: CMOS output
OPTFSelPB OPT[31:28]	7B	123	0000	0: PB1HzOut 1: PB1kHzOut 2: PB32kHzOut 3: InpResSleep		Frequency output on port B, reset from sleep mode with port A
OPTInpRSel1	7C	124	0000	0: InpRes1PA[0] 1: InpRes1PA[1] 2: InpRes1PA[2] 3: InpRes1PA[3]		Reset through port A inputs selection. Refer to reset part
OPTInpRSel2	7D	125	0000	0: InpRes2PA[0] 1: InpRes2PA[1] 2: InpRes2PA[2] 3: InpRes2PA[3]		Reset through port A inputs selection. Refer to reset part
OPTNoPdPS OPT[35:32]	7E	126	0000	0: NoPdPS[0] 1: NoPdPS[1] 2: NoPdPS[2] 3: NoPdPS[3]		No Pull-down on port SP Default: pull-down
RegTestEM	*7F*	*127*	*----*	*----*	*----*	*for EM test only;*

17. Active Supply Current Test

For this purpose, five instructions at the end of the ROM will be added. This will be done at EM Microelectronic. So the user must keep must only use up to 4091 Instructions.

```
Testloop:  STI      00H, 0AH
           LDR      1BH
           NXORX
           JPZ      Testloop
           JMP      00H
```

To stay in the testloop, these values must be written in the corresponding addresses before jumping in the loop:

```
1BH:    0101b
32H:    1010b
6EH:    0010b
6FH:    0011b
```

Free space after last instruction: JMP 00H (0000)

Remark: empty space within the program are filled with NOP (FOFF).

18. Mask Options

Most options which in many µControllers are realized as metal mask options are directly user selectable with the option registers, therefore allowing a maximum freedom of choice .See chapter: Option Register Memory Map.
The following options can be selected at the time of programming the metal mask ROM, except the LCD Segment allocation which is defined using the interconnect metal2 mask.

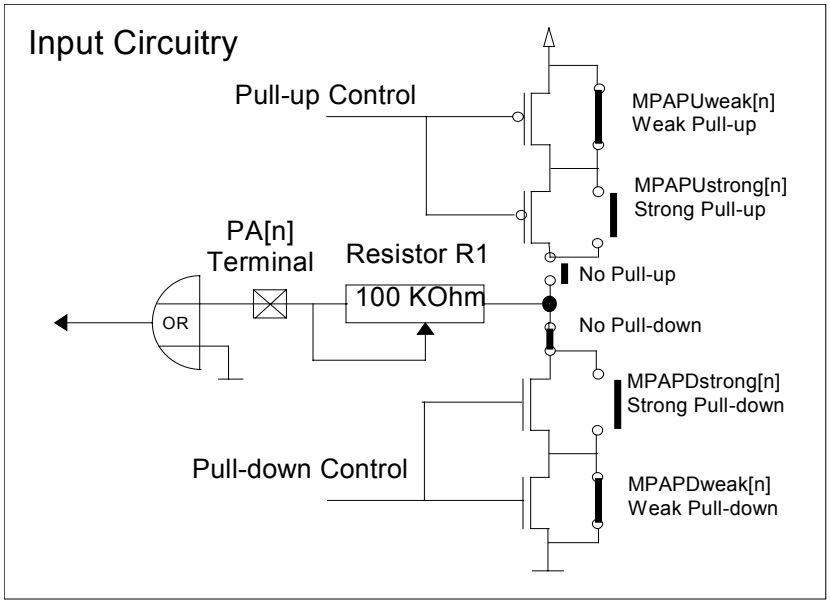
18.1 Input / Output Ports

18.1.1 Port A Metal Options

18.1.2 Port A Metal Options

Pull-up or no pull-up can be selected for each port A input. A pull-up selection is excluding a pull-down on the same input.
Pull-down (default) or no pull-down can be selected for each port A input. A pull-down selection is excluding a pull-up on the same input.
The total pull value (pull-up or pull-down) is a series resistance out of the resistance R1 and the switching transistor. As a switching transistor the user can choose between a high impedance (weak) or a low impedance (strong) switch. Weak, strong or none must be chosen. The default is strong. The default resistor R1 value is 100 KOhm. The user may choose a different value from 150 KOhm down to 0 Ohm. However the value must first be checked and agreed by EM Microelectronic Marin SA. Refer also to chapter 19.2 and 19.4 for the pull values.

Figure 37. Port A Pull Options



Option Name		Strong Pull-down	W Pull-down	R1 Value Typ.100 k	No Pull-down
		1	**2**	**3**	**4**
MPAPD[3]	PA3 input pull-down				
MPAPD[2]	PA2 input pull-down				
MPAPD[1]	PA1 input pull-down				
MPAPD[0]	PA0 input pull-down				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

The default value is : Strong pull-down with R1=100 KOhm
--> Total value of typ. 102 KOhm at V_{DD}=3.0V

Option Name		Strong Pull-up	Weak Pull-up	R1 Value typ.100k	No Pull-up
		1	**2**	**3**	**4**
MPAPU[3]	PA3 input pull-up				
MPAPU[2]	PA2 input pull-up				
MPAPU[1]	PA1 input pull-up				
MPAPU[0]	PA0 input pull-up				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

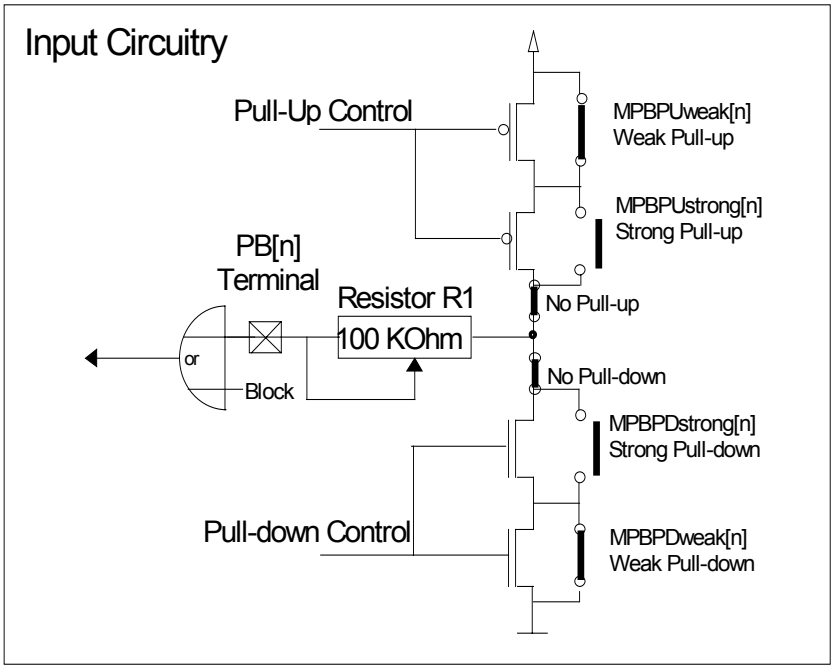
The default value is : Strong pull-up with R1=100 KOhm
--> Total value of typ. 103 KOhm at V_{DD}=3.0

18.1.3 Port B Metal Options

Pull-up or no pull-up can be selected for each port B input. The pull-up is only active in Nch. open drain mode.
Pull-down or no pull-down can be selected for each port B input.

The total pull value (pull-up or pull-down) is a series resistance out of the resistance R1 and the switching transistor. As a switching transistor the user can choose between a high impedance (weak) or a low impedance (strong) switch. Weak , strong or none must be chosen. The default is strong. The default resistor R1 value is 100 KOhm. The user may choose a different value from 150 KOhm down to 0 Ohm. However the value must first be checked and agreed by EM Microelectronic Marin SA. Refer also to chapter 19.2 and 19.4 for the pull values.

Figure 38. Port B Pull Options



Option Name		Strong Pull-down	Weak Pull-down	R1 Value Typ.100k	No Pull-down
		1	**2**	**3**	**4**
MPBPD[3]	PB3 input pull-down				
MPBPD[2]	PB2 input pull-down				
MPBPD[1]	PB1 input pull-down				
MPBPD[0]	PB0 input pull-down				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

The default value is : Strong pull-down with R1=100 KOhm
--> Total value of typ. 102 KOhm at V_{DD}=3.0V

Option Name		Strong Pull-up	Weak Pull-up	R1 value Typ. 100k	NO Pull-up
		1	**2**	**3**	**4**
MPBPU[3]	PB3 input pull-up				
MPBPU[2]	PB2 input pull-up				
MPBPU[1]	PB1 input pull-up				
MPBPU[0]	PB0 input pull-up				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

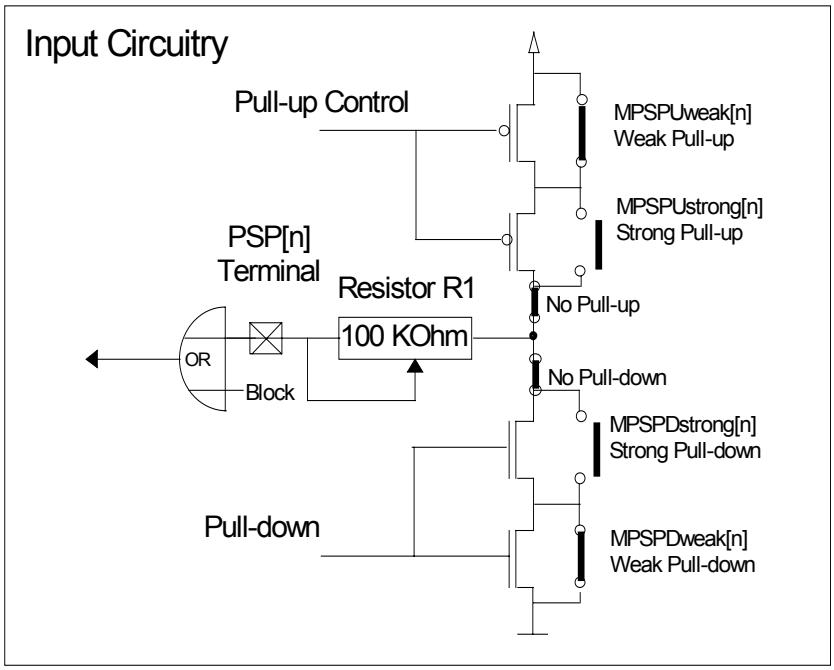
The default value is : Strong pull-up with R1=100 KOhm
--> Total value of typ. 103kOhm at V_{DD}=3.0V

18.1.4 Port SP Metal Options

Pull-up or no pull-up can be selected for each port SP input. The pull-up is only active in Nch. open drain mode. Pull-down or no pull-down can be selected for each port SP input.

The total pull value (pull-up or pull-down) is a series resistance out of the resistance R1 and the switching transistor. As a switching transistor the user can choose between a high impedance (weak) or a low impedance (strong) switch. Weak , strong or none must be chosen. The default is strong. The default resistor R1 value is 100 KOhm. The user may choose a different value from 150 KOhm down to 0 Ohm. However the value must first be checked and agreed by EM Microelectronic Marin SA. Refer also to chapter 19.2 and 19.4 for the pull values.

Figure 39. Port SP Pull Options



Option Name		Strong Pull-down	Weak Pull-down	R1 Value Typ.100k	NO Pull-Down
		1	**2**	**3**	**4**
MPSPD[3]	PS3 input pull-down				
MPSPD[2]	PS2 input pull-down				
MPSPD[1]	PS1 input pull-down				
MPSPD[0]	PS0 input pull-down				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

The default value is : strong pull-down with R1=100 KOhm
--> Total value of typ. 102 KOhm at V_{DD}=3.0V

Option Name		Strong Pull-up	weak Pull-up	R1 Value Typ. 100k	NO Pull-up
		1	**2**	**3**	**4**
MPSPU[3]	PS3 input pull-up				
MPSPU[2]	PS2 input pull-up				
MPSPU[1]	PS1 input pull-up				
MPSPU[0]	PS0 input pull-up				

To select an option put an **X** in column 1,2 and 4 and reconfirm the R1 value in column 3.

The default value is : strong pull-up with R1=100 KOhm
--> Total value of typ. 103 KOhm at V_{DD}=3.0V

18.1.5 Voltage Regulator Option

Option Name		Default Value	User Value
		A	**B**
MVreg	Voltage Regulator	**YES**	

By default the internal voltage regulator supplies the core logic the RAM and the ROM. With option **MVreg(B)** the regulator is cut and Vbat is supplying the core logic the ROM and the RAM.

18.1.6 Debouncer Frequency Option

Option Name		Default Value	User Value
		A	**B**
MDeb	Debouncer freq.	**Ck[11]**	

By default the debouncer frequency is Ck[11]. The user may choose Ck[14] instead of Ck[11].
Ck[14]corresponds to maximum 0.25ms debouncer time in case of a 32kHz oscillator.

18.1.7 Frequency Selection Option

Option Name		Default Value	User Value
		A	**B**
Mfreq	Freq. selection	**32kHz**	

By default the 32kHz option is selected. The user may chose 128kHz instead of 32kHz.Refer to chapter 5.1.

18.1.8 LCD Frame Frequency Option

Option Name		Default Value	User Value
		A	**B**
MLCDFreq	Freq. selection	**256/8Hz**	

By default the 256/8 Hz (LCD frame frequency=32hz) option is selected. Other two possible options are: 341/8 Hz or 512/8 Hz. Refer to chapter 14.

18.1.9 User defined LCD Segment Allocation

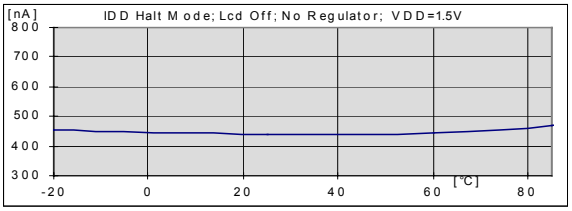
If using a different Segment allocation from the one described in chapter 14.3 , one needs to fill in following table. The Segment allocation connection are realized with the interconnect Metal2 mask.

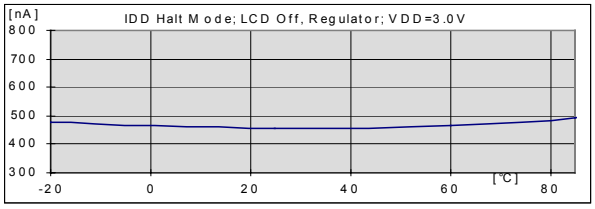
4 times MUX	COM[1]	COM[2]	COM[3]	COM[4]
3 times MUX	COM[1]	COM[2]	COM[3]	--
SEG[1]				
SEG[2]				
SEG[3]				
SEG[4]				
SEG[5]				
SEG[6]				
SEG[7]				
SEG[8]				
SEG[9]				
SEG[10]				
SEG[11]				
SEG[12]				
SEG[13]				
SEG[14]				
SEG[15]				
SEG[16]				
SEG[17]				
SEG[18]				
SEG[19]				
SEG[20]				
SEG[21]				
SEG[22]				
SEG[23]				
SEG[24]				
SEG[25]				
SEG[26]				
SEG[27]				
SEG[28]				
SEG[29]				
SEG[30]				
SEG[31]				
SEG[32]				

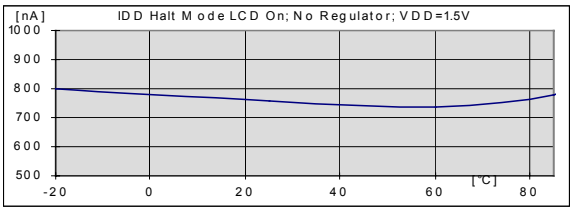
The customer should specify the required options at the time of ordering. A copy of the pages 54 to 58, as well as the « Software ROM characteristic file » generated by the assembler (*.STA) should be attached to the order.

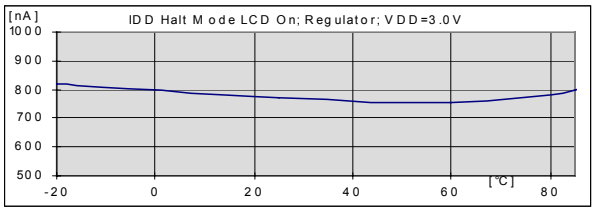
19. Temp. and Voltage Behaviors

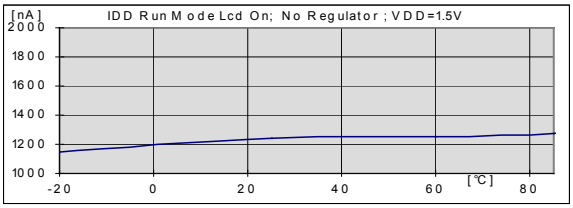
19.1 IDD Current (Typical)

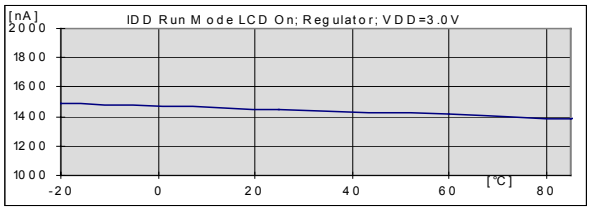




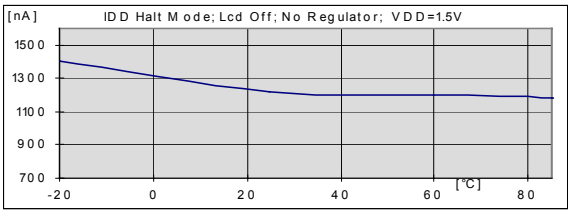


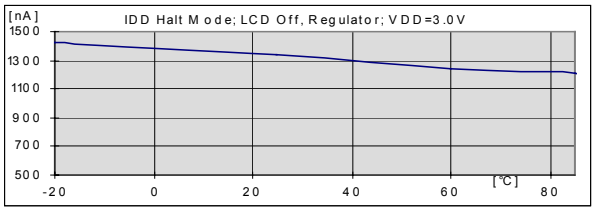


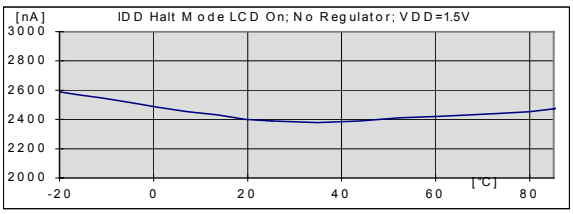


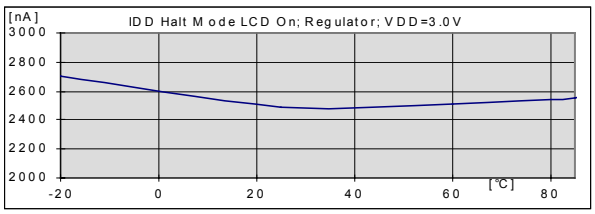


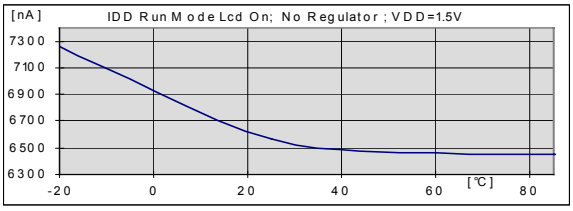
19.2 IDD Current @ 128kHZ

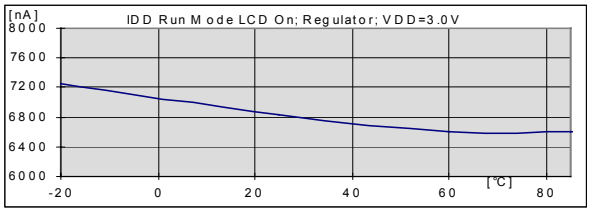




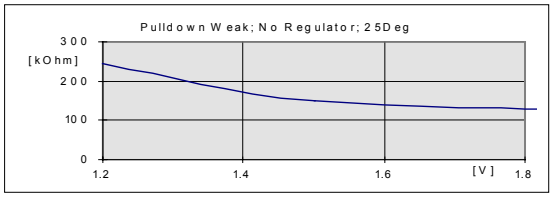


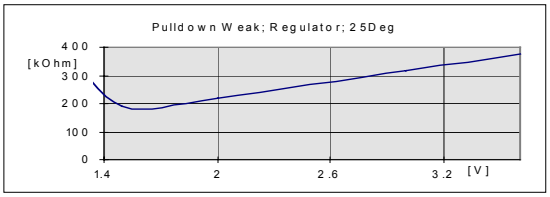


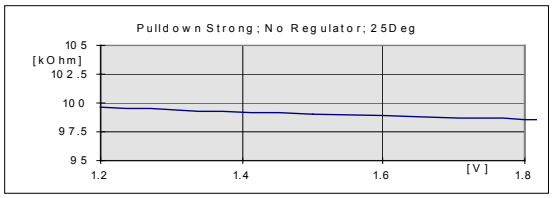


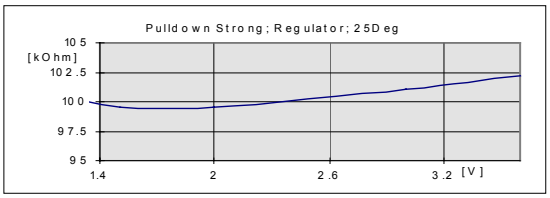


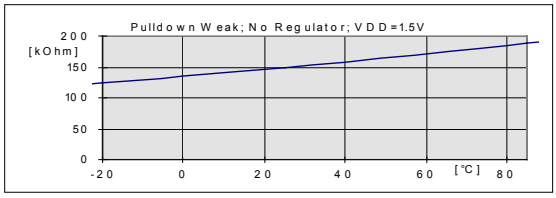
19.3 Pull-down Resistance (Typical)

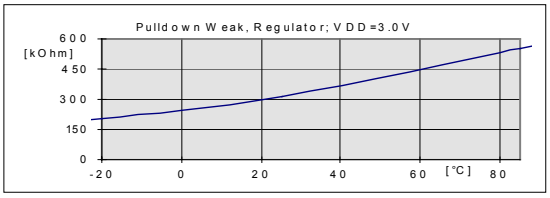


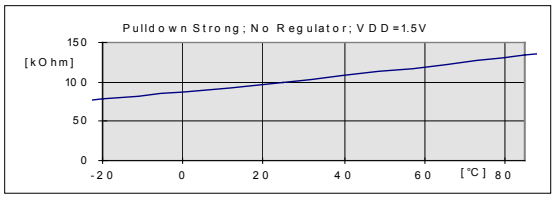


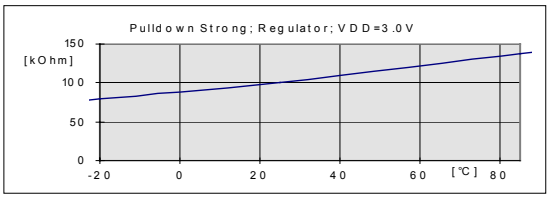




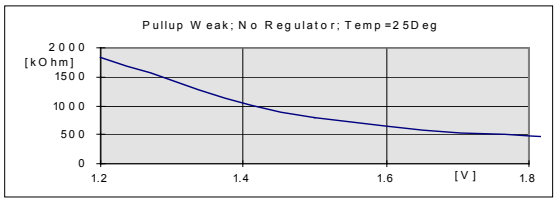


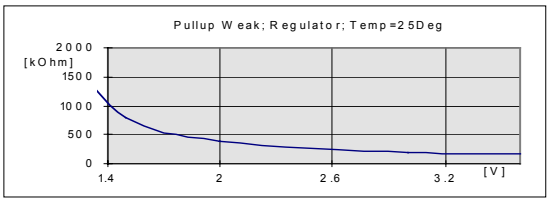


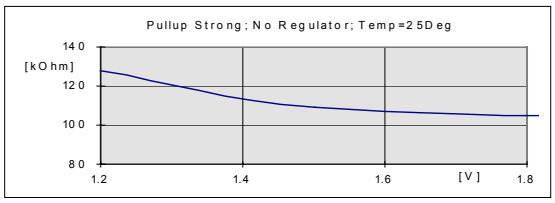


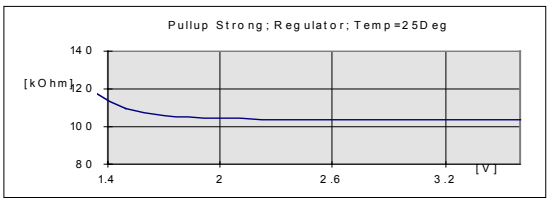


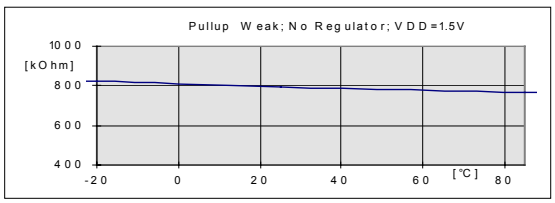
19.4 Pull-up Resistance (Typical)

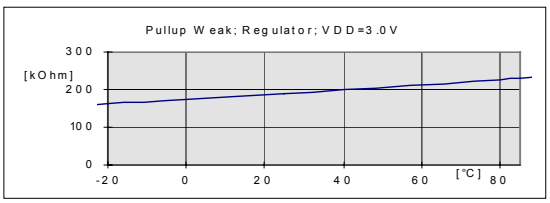


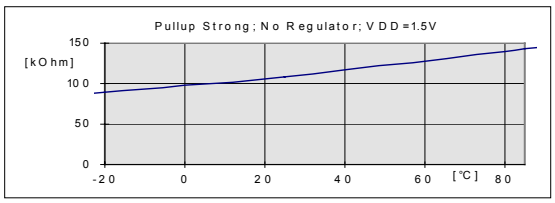


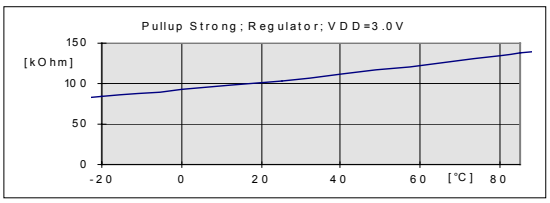




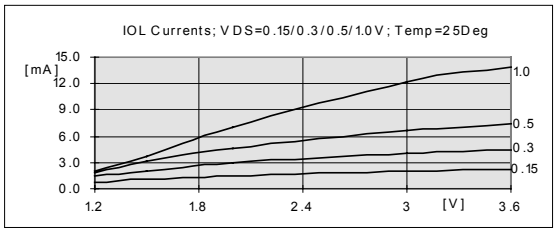


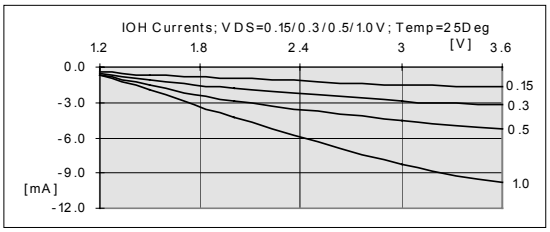


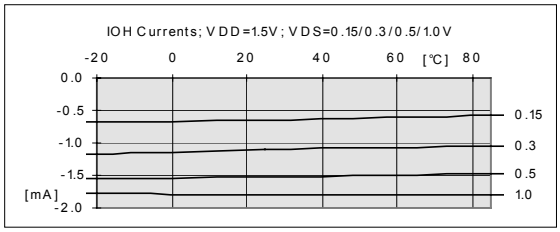


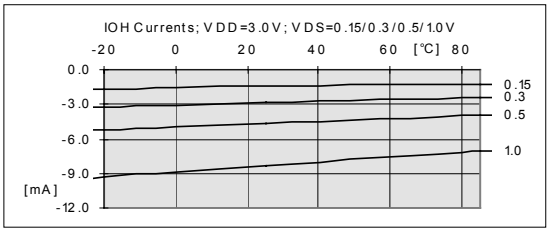


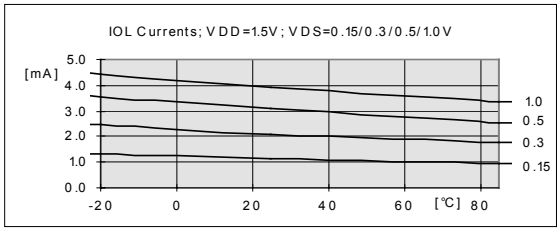
19.5 Output Currents (Typical)

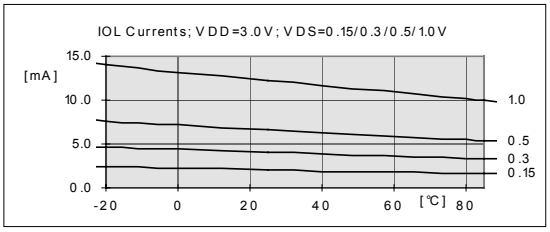












20. Electrical Specification

20.1 Absolute Maximum Ratings

	Min.	Max.	Units
Power supply V_{DD}-V_{SS}	- 0.2	+ 3.6	V
Input voltage	V_{SS} – 0.2	V_{DD}+0.2	V
Storage temperature	- 40	+ 125	°C
Electrostatic discharge to Mil-Std-883C method 3015.7 with ref. to V_{SS}	-2000	+2000	V
Maximum soldering conditions		10s x 250°C	

Stresses above these listed maximum ratings may cause permanent damage to the device.
Exposure beyond specified electrical characteristics may affect device reliability or cause malfunction.

20.2 Handling Procedures

This device has built-in protection against high static voltages or electric fields; however, anti-static precautions should be taken as for any other CMOS component.
Unless otherwise specified, proper operation can only occur when all terminal voltages are kept within the supply voltage range.

20.3 Standard Operating Conditions

Parameter	MIN	TYP	MAX	Unit	Description
Temperature	-20	25	85	°C	
V_{DD}_Range1	1.4	3.0	3.6	V	with internal voltage regulator
V_{DD}_Range2	1.2	1.5	1.8		without internal voltage regulator
V_{SS}		0		V	reference terminal
$C_{VDD}CA$ (note 1)	100			nF	regulated voltage capacitor
Fq		32768		Hz	nominal frequency
Rqs		35		KOhm	typical quartz serial resistance
CL		8.2		pF	typical quartz load capacitance
df/f		± 30		ppm	quartz frequency tolerance

Note 1: This capacitor filters switching noise from V_{DD} to keep it away from the internal logic cells. In noisy systems the capacitor should be chosen bigger than minimum value.

20.4 DC Characteristics - Power Supply

Conditions: V_{DD}=1.5V, T=25°C, RC=32kHz without internal voltage regulator (unless otherwise specified)

Parameter	Conditions	Symbol	Min.	Typ.	Max.	Unit
ACTIVE Supply Current	(note2,3)	IV_{DDa1}		1.5	2.5	µA
(in active mode with LCD on)	-20 ... 85°C (note2,3)	IV_{DDa1}			3.5	µA
STANDBY Supply Current		IV_{DDh1}		0.4	0.8	µA
(in Halt mode, LCDOff)	-20 ... 85°C	IV_{DDh1}			1.0	µA
SLEEP Supply Current		IV_{DDs1}		0.2	0.4	µA
	-20 ... 85°C	IV_{DDs1}			0.5	µA
POR static level	-20 ... 85°C	V_{POR1}		0.90	1.2	V
RAM data retention		V_{rd1}	1.0			V

Note 2: LCD Display not connected.
Note 3: The instruction loop described in chapter 17 is used for these tests. The oscillator is externally driven on a frequency of 32.768 KHz and an amplitude of Vreg voltage.

Conditions: V_{DD}=3.0V, T=25°C, RC=32kHz, with internal voltage regulator (unless otherwise specified)

Parameter	Conditions	Symbol	Min.	Typ.	Max.	Unit
ACTIVE Supply Current (in active mode with LCD on)	(note2,3)	I_{VDDa2}		1.8	2.5	µA
	-20 ... 85°C (note2,3)	I_{VDDa2}			3.5	µA
STANDBY Supply Current (in Halt mode, LCDOff)		I_{VDDh2}		0.4	0.8	µA
	-20 ... 85°C	I_{VDDh2}			1.0	µA
SLEEP Supply Current		I_{VDDs2}		0.2	0.4	µA
	-20 ... 85°C	I_{VDDs2}			0.55	µA
POR static level	-20 ... 85°C, No Load on Vreg	V_{POR2}		0.95	1.25	V
RAM data retention	-20 ... 85°C	V_{rd2}	1.2			V
Regulated voltage	Halt mode, No Load	V_{reg}	1.2	1.4	1.7	V

Note 2: LCD Display not connected.

Note 3: The instruction loop described in chapter 17 is used for these tests. The oscillator is externally driven on a frequency of 32.768 KHz and an amplitude of Vreg voltage.

20.5 Supply Voltage Level Detector

Parameter	Conditions	Symbol	Min.	Typ.	Max.	Unit
SVLD voltage Level1	-10 to 60°C	V_{SVLD1}	3.65	3.96	4.35	V
SVLD voltage Level2	-10 to 60°C	V_{SVLD2}	2.70	3.0	3.27	V
SVLD voltage Level3	-10 to 60°C	V_{SVLD3}	2.20	2.44	2.65	V
SVLD voltage Level4	-10 to 60°C	V_{SVLD4}	1.82	2.03	2.23	V
SVLD voltage Level5	-10 to 60°C	V_{SVLD5}	1.62	1.79	1.95	V
SVLD voltage Level6	-10 to 60°C	V_{SVLD6}	1.39	1.56	1.70	V
SVLD voltage Level7	-10 to 60°C	V_{SVLD7}	1.25	1.39	1.53	V
SVLD voltage Level8	-10 to 60°C	V_{SVLD8}	1.11	1.24	1.38	V
Temperature coefficient	0 to 50°C			< ±0.1		mV/°C

20.6 Oscillator

Conditions: T=25°C (unless otherwise specified)

Parameter	Conditions	Symbol	Min.	Typ.	Max.	Unit
Temperature stability	+15 ... +35 °C	df/f x dT			0.3	ppm /°C
Voltage stability (note 5)	V_{DD}=1,4 - 1,6 V	df/f x dU			5	ppm /V
Input capacitor	Ref. on V_{SS}	Cin	5.6	7	8.4	pF
Output capacitor	Ref. on V_{SS}	Cout	12.1	14	15.9	pF
Transconductance	50mVpp,V_{DD}min	Gm	2.5		15.0	µA/V
Oscillator start voltage	Tstart < 10 s	Ustart	V_{DD}min			V
Oscillator start time	V_{DD} > V_{DD}Min	tdosc		0.5	3	s
System start time (oscillator + cold-start + reset)		tdsys		1.5	4	s
Oscillation detector frequency	V_{DD} > V_{DD}min	$t_{DetFreq}$			12	kHz

Note 5: Applicable only for the versions without the internal voltage regulator

20.7 DC characteristics - I/O Pins

Conditions: T= -20 ... 85°C (unless otherwise specified)
V_{DD}=1.5V means; measures without voltage regulator
V_{DD}=3.0V means; measures with voltage regulator

Parameter	Conditions	Symb.	Min.	Typ.	Max.	Unit
Input Low voltage						
Ports A,B,SP Test	V_{DD} < 1.5V	V_{IL}	Vss		0.2V_{DD}	V
Ports A,B,SP Test	V_{DD} > 1.5V	V_{IL}	Vss		0.3V_{DD}	V
QIN with Regulator		V_{IL}	Vss		0.1Vreg	V
QIN without Regulator		V_{IL}	Vss		0.1V_{DD}	V
QOUT (note 7)						
Input High voltage						
Ports A,B,SP Test		V_{IH}	0.7V_{DD}		V_{DD}	V
QIN with Regulator		V_{IH}	0.9Vreg		Vreg	V
QIN without Regulator		V_{IH}	0.9V_{DD}		V_{DD}	V
QOUT (note 7)						
Output Low Current	V_{DD}=1.5V , VOL=0.15V	I_{OL}		1.1		mA
all logic outputs	V_{DD}=1.5V , VOL=0.30V	I_{OL}		2.1		mA
	V_{DD}=1.5V , VOL=0.50V	I_{OL}	1.55	3.1		mA
	V_{DD}=3.0V , VOL=0.15V	I_{OL}		1.9		mA
	V_{DD}=3.0V , VOL=0.30V	I_{OL}		3.9		mA
	V_{DD}=3.0V , VOL=0.50V	I_{OL}		6.4		mA
	V_{DD}=3.0V , VOL=1.00V	I_{OL}	5.5	12.0		mA
Output High Current	V_{DD}=1.5V, VOH= V_{DD}-0.15V	I_{OH}		-0.6		mA
all logic outputs	V_{DD}=1.5V, VOH= V_{DD}-0.30V	I_{OH}		-1.1		mA
	V_{DD}=1.5V, VOH= V_{DD}-0.50V	I_{OH}		-1.5	-0.75	mA
	V_{DD}=3.0V, VOH= V_{DD}-0.15V	I_{OH}		-1.5		mA
	V_{DD}=3.0V , VOH= V_{DD}-0.30V	I_{OH}		-3.0		mA
	V_{DD}=3.0V , VOH= V_{DD}-0.50V	I_{OH}		-5.0		mA
	V_{DD}=3.0V , VOH= V_{DD}-1.00V	I_{OH}		-9.5	-4.2	mA
Input Pull-down	V_{DD}=1.5V, Pin at 1.5V, 25°C	R_{PD}		20k		Ohm
Test	V_{DD}=3.0V, Pin at 3.0V, 25°C	R_{PD}		20k		Ohm
Input Pull-down	V_{DD}=1.5V, Pin at 1.5V, 25°C	R_{PD}	100k	180k	350k	Ohm
Port A,B,SP (note 8) weak	V_{DD}=3.0V, Pin at 3.0V, 25°C	R_{PD}	200k	375k	750k	Ohm
Input Pull-up	V_{DD}=1.5V, Pin at 0.0V, 25°C	R_{PU}	500k	800k	2000k	Ohm
Port A,B,SP (note 8) weak	V_{DD}=3.0V, Pin at 0.0V, 25°C	R_{PU}	115k	160k	475k	Ohm
Input Pull-down	V_{DD}=1.5V, Pin at 1.5V, 25°C	R_{PD}	72k	102k	132k	Ohm
Port A,B,SP (note 8) strong	V_{DD}=3.0V, Pin at 3.0V, 25°C	R_{PD}	70k	100k	130k	Ohm
Input Pull-up	V_{DD}=1.5V, Pin at 0.0V, 25°C	R_{PU}	76k	109k	142k	Ohm
Port A,B,SP (note 8) strong	V_{DD}=3.0V, Pin at 0.0V, 25°C	R_{PU}	72k	103k	134k	Ohm

Note 7: QOUT (OSC2) is used only with Quartz.
Note 8: Weak or strong are standing for weak pull or strong pull transistor. Values are for R1=100kOhm

20.8 LCD SEG[20:1] Outputs

Conditions: T=25°C (unless otherwise specified)

Parameter	Conditions	Symb.	Min.	Typ.	Max.	Unit
Driver Impedance Level 0	Iout = ±5µA, Ext. Supply	R_{SEGVL0}			20	KOhm
Driver Impedance Level 1	Iout = ±5µA, Ext Supply	R_{SEGVL1}			20	KOhm
Driver Impedance Level 2	Iout = ±5µA, Ext Supply	R_{SEGVL2}			20	KOhm
Driver Impedance Level 3	Iout = ±5µA, Ext Supply	R_{SEGVL3}			20	KOhm

20.9 LCD Com[4:1] Outputs

Conditions: T=25°C (unless otherwise specified)

Parameter	Conditions	Symb.	Min.	Typ.	Max.	Unit
Driver Impedance Level 0	Iout = ±5µA, Ext. Supply	R_{comVL0}			10	KOhm
Driver Impedance Level 1	Iout = ±5µA, Ext. Supply	R_{comVL1}			10	KOhm
Driver Impedance Level 2	Iout = ±5µA, Ext Supply	R_{comVL2}			10	KOhm
Driver Impedance Level 3	Iout = ±5µA, Ext Supply	R_{comVL3}			10	KOhm

20.10 DC Output Component

Conditions: T=25°C (unless otherwise specified)

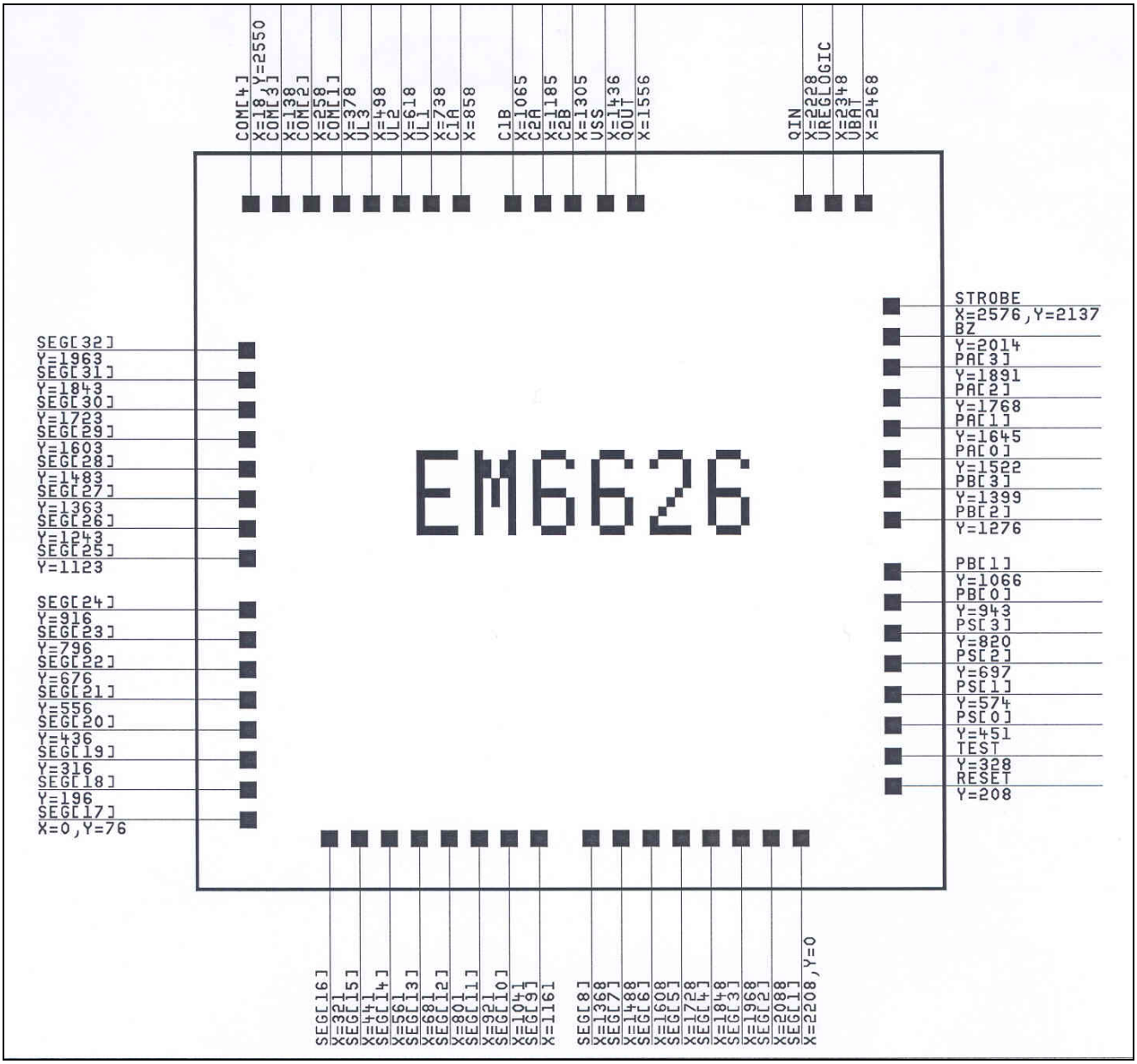
Parameter	Conditions	Symb.	Min.	Typ.	Max.	Unit
DC Output component	No Load	$\pm V_{DC\_com}$			20	mV

20.11 LCD Voltage Multiplier

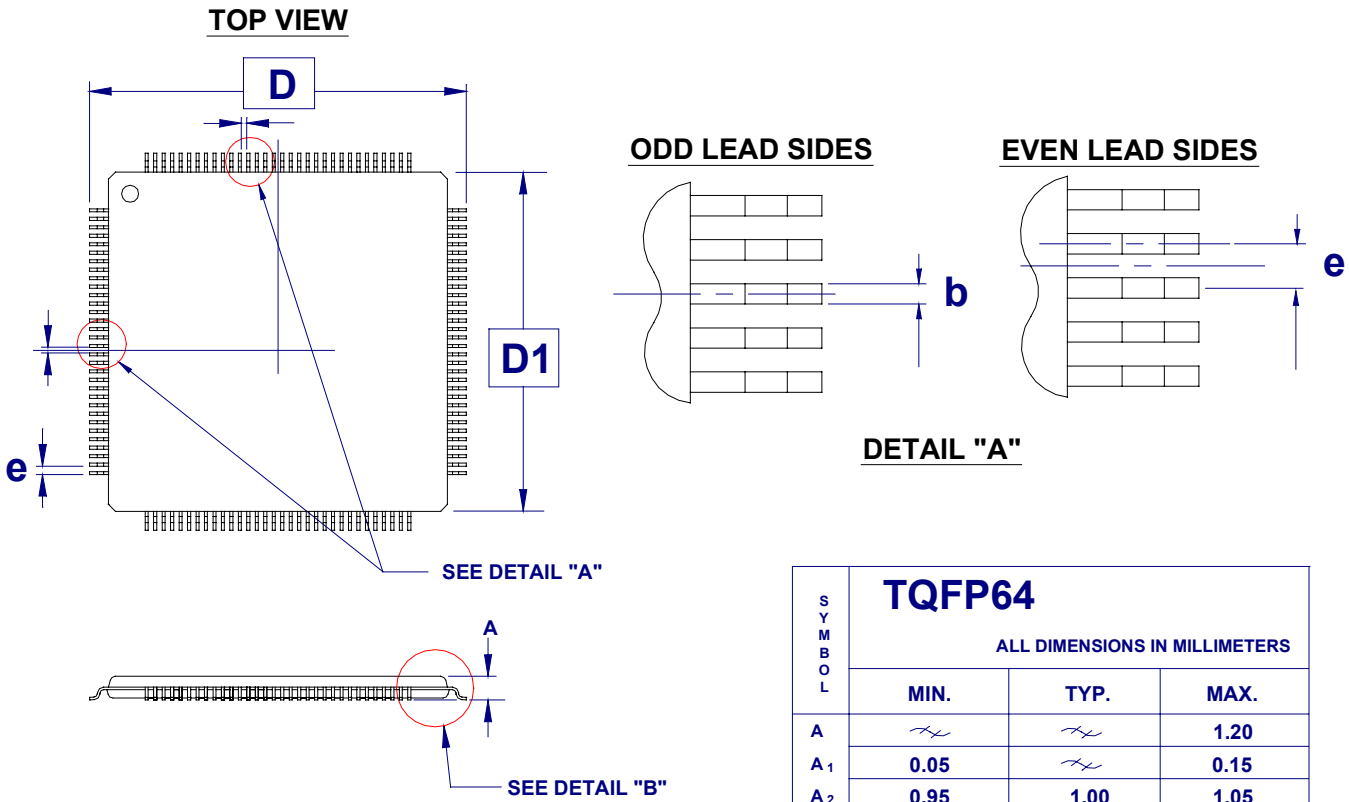
Conditions: T=25°C, All Multiplier Capacitors 100nF, freq=512Hz. (unless otherwise specified)

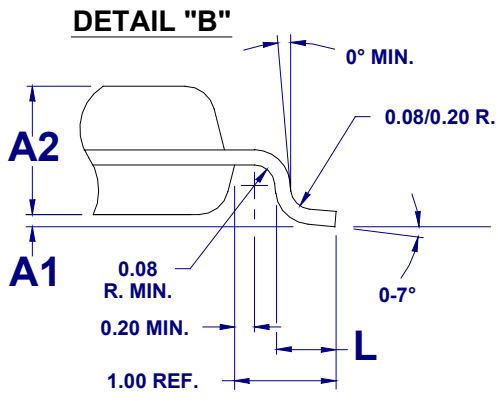
Parameter	Conditions	Symb.	Min.	Typ.	Max.	Unit
Voltage Bias Level 1	1 A load	V_{VL1}	0.95	1.06	1.15	V
Voltage Bias Level 2	1 A load	V_{VL2}		2.12		V
Voltage Bias Level 3	1 A load	V_{VL3}		3.18		V
Temp dependency V_{VL1}	1 A load, -10...60°C	dV_{VL1}/dT		-4.9		mV/°C

21. Pad Location Diagram



X=2.99mm
Y=2.94mm





SYMBOL	TQFP64 ALL DIMENSIONS IN MILLIMETERS		
	MIN.	TYP.	MAX.
A			1.20
A_1	0.05		0.15
A_2	0.95	1.00	1.05
D		12.00 BSC.	
D_1		10.00 BSC.	
L	0.45	0.60	0.75
N		64	
e		0.50 BSC	
b	0.17	0.22	0.23

1.00/0.10 MM FORM, 1.00 MM THICK

PACKAGE OUTLINE, TQFP, 10X10 MM BODY,

21.1 Ordering Information

Packaged Device:

EM6626 TQ64 B - %%%

Package:
TQ64 = TQFP 64 pin

Delivery Form:
B = Tape & Reel
D = Trays (Plate)

Customer Version:
customer-specific number
given by EM Microelectronic

Device in DIE Form:

EM6626 WS 11 - %%%

Die form:
WW = Wafer
WS = Sawn Wafer/Frame
WP = Waffle Pack

Thickness:
11 = 11 mils (280um), by default
27 = 27 mils (686um), not backlapped
(for other thickness, contact EM)

Customer Version:
customer-specific number
given by EM Microelectronic

Ordering Part Number (selected examples)

Part Number	Package / Die Form	Delivery Form / Thickness
EM6626TQ64B-%%%	TQFP 64	Tape & Reel
EM6626TQ64D-%%%	TQFP 64	Trays (Plate)
EM6626WS11-%%%	Sawn Wafer	11 mils
EM6626WP11-%%%	Die in waffle pack	11 mils

Please make sure to give the complete Part Number when ordering, including the 3-digit version. The version is made of 3 numbers %%% (e.g. 005 , 012, 131, etc.).

21.2 Package Marking

TQFP64 marking:

First line:	E	M	6	6	2	6		%	%	%	Y
Second line:	P	P	P	P	P	P	P	P	P	P	P
Third line:	C	C	C	C	C	C	C	C	C	C	C

Where: %%% = customer version, specific number given by EM (e.g. 005, 012, 131, etc.)
Y = Year of assembly
PP…P = Production identification (date & lot number) of EM Microelectronic
CC…C = Customer specific package marking on third line, selected by customer

21.3 Customer Marking

There are **11** digits available for customer marking on **TQFP64**

Please specify below the desired customer marking.

Please contact EM headquarters or your local EM office for any other detail. Also refer to page 58 for additional ordering information.

Updates since Rev B/218 (Sept 98)

Date of Update Name	Chapter concerned	New Version	Changes