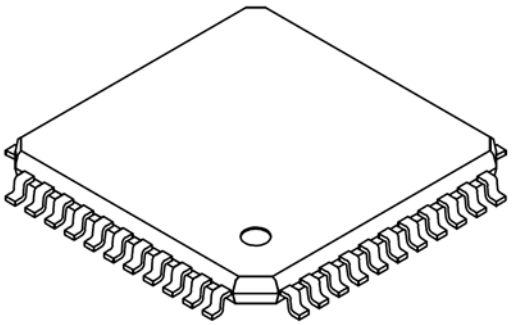
CMOS Digital Integrated Circuit Silicon Monolithic

TC35894FG

Mobile Peripheral Devices

Overview

The TC35894FG is IO Expander LSI which has I^2C bus slave for host interface, PWM/Timer control, GPIO control, and key matrix control. Those features are set and selected at the TC35894FG registers by software.



P-LQFP44-1010-0.80-001

Weight: 0.35g (Typ.)

Features

- I^2C Slave for host interface
- GPIO Functions
  - Maximum 24 general purpose input/output ports
  - Selectable input or output port
  - Selectable pull-up or pull-down / connecting or non-connecting resistor
  - Selectable drive current (3 types)
  - Supporting pseudo open drain output buffer
  - Selectable interrupt detecting by level / edge / both edges and active low / high
  - Automatic escape from sleep mode by signal input
- Key Board
  - Key Matrix (Max 8 × 12 = 96 keys)
  - Support special function keys and dedicated keys
  - Key de-bouncing function
- PWM / Timer
  - 3 timer module for LED back light control
  - LED switching control by PWM sequencer
- Internal oscillator for system clock
- Sleep mode for reducing of power consumption
- Operating voltage 1.62 V to 3.60 V
- 10mm × 10mm LQFP package (44 pins, 0.8mm pin pitch, maximum 1.7mm height)

Table of content

List of Figures

List of Tables

1. Overview

TC35894FG is LSI for IO expansion. The functional block is constituted by PWM/Timer control part, GPIO control part, Key control part, and the I^2C slave interface. Assignment on the terminal of each function can be changed by software.

Features

- I^2C slave for host interface
- GPIO functions
  - Maximal 24 general purpose input/output ports
  - Selected pull-up/pull-down resistor connection or release by register setting
  - Selected three output drive modes by register setting
  - Support pseudo open drain output buffers
  - Selected interrupt sensitivity (positive/negative edge, both edges, high/low level)
  - Escape from sleep mode by input signal
  - Direct keypad function
- Keyboard functions
  - Key matrix **:** maxima 8 × 12 = 96 keys
  - Support special function keys and dedicated keys
  - Key de-bouncing by hardware solution
- PWM/Timer functions
  - Three channel timer modules for LED/back light control
  - LED switching control by PWM sequencer
- Internal oscillator circuit for system clock
- Two operating modes (Sleep & Operation) for low power consumption
- Operating voltage area **:** 1.62 V to 3.60 V
- 44 pins, 0.8mm pin pitch, maximum 1.7mm height, LQFP package

2. FUNCTIONAL OVERVIEW

Each function block and bus architecture of TC35894FG are shown as below figure. Please note that some signals are sharing the same terminals. Further details about the terminal multiplexing can be found in Table 6.1, Table 6.2 and Table 6.3.

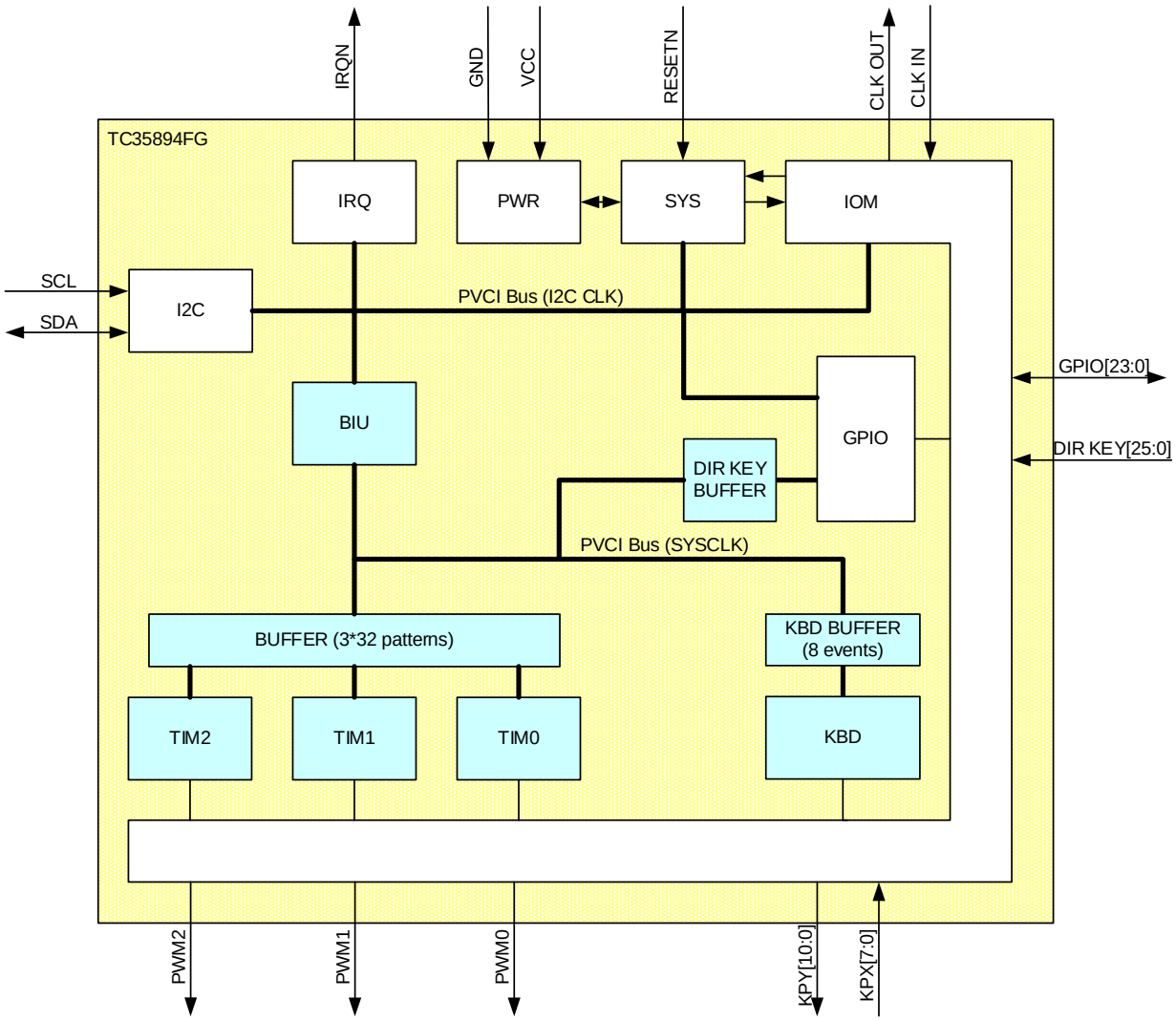


Figure 2.1 Block diagram

The TC35894FG is a simple slave device with limited intelligence. It is controlled by a host via an I^2C interface conforming to the I^2C specification in section 15 [1]. The software protocol run on that interface is a simple register access protocol.

The supply voltage ranges from 1.62 V to 3.60 V for core logic and I/O supply. The power module (PWR) provides a power-on reset for the circuit as well as a watch-dog functionality in case of power failures. The PWR also comprises a linear down converter to generate the internally required 1.5 V core voltage from the supply voltage.

SYS

The system module (SYS) controls the two operating modes (SLEEP, OPERATION) for power saving inside the TC35894FG. An Auto-Sleep feature detects activity at the peripheral and switches dynamically between the power modes, without host CPU intervention.

In addition, the SYS module generates the internal system clock for the TC35894FG based on the DIR24 pin or the internal RC-oscillator. The DIR24 pin can be a direct CMOS input in the frequency range from 32 kHz to 20 MHz.

The clock generator module inside the SYS module can divide a suitable system clock frequency (61.54 kHz to 200 kHz) an external clock input or an internal RC-generated clock of about 2.2 MHz.

I^2C

The I^2C module is a simple slave on the external I^2C bus. It operates from the SCL clock and does not require any other clock. The external I^2C bus is not tied to GND or VCC when the power supply to the TC35894FG is taken away (fail-safe).

All modules in which are shown as a shaded light-blue box, require the system clock to be up and running in order to be fully operational, also for I^2C register control. In contrast, the white module boxes are fully operational also in the absence of a system clock. For instance the operation mode setting in the SYS module can be altered at any time simply via I^2C programming.

In applications where the system clock is not needed, i.e. use for only GPIO functionality, the DIR24 pin shall be connected to VCC or to GND. The internal RC-oscillator should be in inactive state.

The DIR24 pin also controls the programmability of the I^2C slave address of the device. The I^2C address of the device can be reprogrammed when the DIR24 pin is connected to VCC or connected to external clock. When DIR24 pin is connected to GND then re-programming of the default I^2C address of the device is not possible.

IOM

The "I/O multiplexer" module switches different functional configurations onto the package pins.

BIU

The "bus interface" module is simply a bus synchronizer between I^2C bus clock and system clock.

KBD

The "Keyboard module" can be configured to support keyboard layouts from 2 by 2 up to 8 by 12 plus additionally 8 special function keys. Depending on the configuration, also the use of dedicated keys is possible. Those dedicated keys are not embedded into a key matrix.

This method enables safe detection of simultaneous key presses. Keyboard de-bouncing is done in hardware. Up to eight keyboard events can be held in a FIFO to reduce real-time constraints for interrupt servicing.

GPIO

The "general purpose I/O" block provides 24 lines of general purpose in/out functionality. Every GPIO line is capable of triggering an interrupt to the host, also in the absence of a system clock. Regardless, which functionality is mapped to the TC35894FG pin, the GPI input can always be used to scan the corresponding pin.

Within the GPIO module also direct keypad functionality for 26 direct key inputs is realized. Not used GPIO lines can be configured as direct keypad inputs. The pins DIR24 and DIR25 are not allowing general purpose in/out functionality. They can be used as direct keypad inputs. Additionally pin DIR24 can also be used as clock input.

TIM

Three versatile timers are available if they are activated and can generate modulated pulse width (PWM) outputs for LED and vibrator control. They are equipped with a common up-front pattern storage register that is capable to generate PWM patterns without the interference of a host processor. The timers can trigger wake up events and interrupts to the host at a scheduled time.

IRQ

The interrupt control block (IRQ) provides an active low (negative logic) hardware interrupt signal.

An external pull-up resistor on the IRQN pin is required for proper functionality.

3. Terminal

3.1. Pin Layout (44 Pins, LQFP Package)

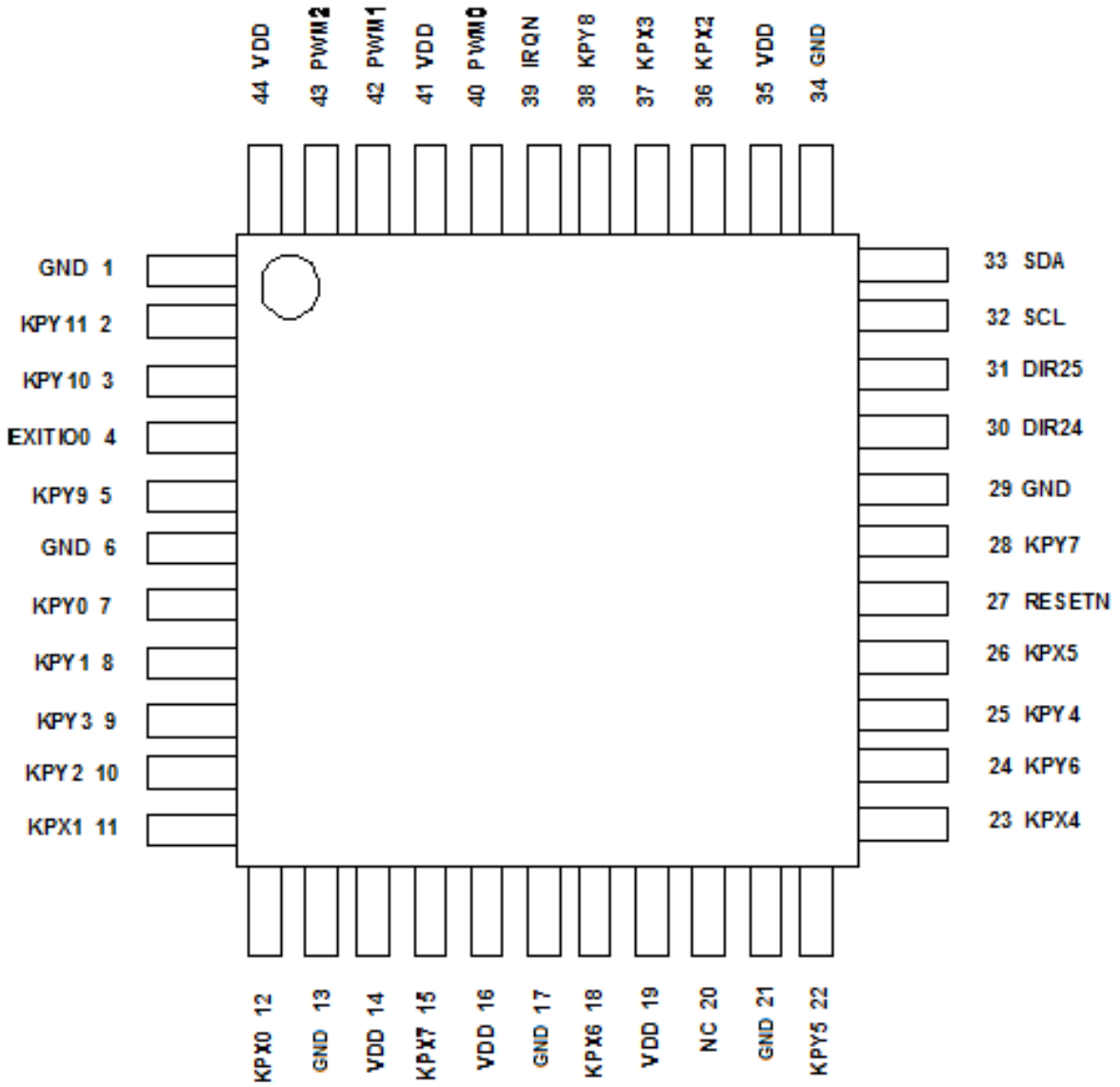


Figure 3.1 Pin Layout (Top view)

3.2. Pin Table

The following table explains the top level pad functionality. All inputs have CMOS Schmitt characteristics. Pins configured as inputs shall never be left floating. They are either driven by an external source or the input is internally terminated by a software configurable pull up/down resistor. The symbol in the table shows "I" for input, "I/O" multiple direction, "OD" for open drain, "P" for selectable setting of pull-up or pull-down, "PU" for pull-up, and "Hi-Z" for high impedance.

Table 3.1 Functional Description of the Pins

Name	I/O Note1	Default Note1	Terminal	Description
VDD	VCC	-	14, 16, 19, 35, 41, 44	LSI supply voltage
GND	GND	-	1, 6, 13, 17, 21, 29, 34	Common ground
SCL	I	I	32	I²C clock, up to 400 kHz (fail-safe Note2)
SDA	I/OD	Hi-Z	33	I²C data (fail-safe Note2)
IRQN	OD	Hi-Z	39	Interrupt to host processor Low active. (fail-safe Note2)
RESETN	I	I	27	reset line, low active (fail-safe Note2)
KPX0 KPX1 KPX2 KPX3 KPX4 KPX5 KPX6 KPX7	I/O, P	I, PU	12 11 36 37 23 26 18 15	General purpose I/O, keyboard or direct key
KPY0 KPY1 KPY2 KPY3 KPY4 KPY5 KPY6 KPY7 KPY8 KPY9 KPY10	I/O, P	I, PU	7 8 10 9 25 22 24 28 38 5 3	General purpose I/O, keyboard or direct Key
KPY11	I/O, P	I, PU	2	General purpose I/O, direct key, keyboard or clock output
PWM0 PWM1 PWM2	I/O, P	I, PU	40 42 43	General purpose I/O, direct key or PWM0 General purpose I/O, direct key or PWM1 General purpose I/O, direct key or PWM2
EXTIO0	I/O, P	I, PU	4	General purpose I/O or direct key
DIR24	I, P	I, PU	30	Clock input, direct key or I²C address control DIR24 can be used as clock input when the internal RC-oscillator is not used. The re-programming of I²C slave address is also controlled by DIR24: ● Pull-down: I²C slave address is fixed ● Pull-up or external clock connected: Re-programming of I²C address is enabled
DIR25	I, P	I, PU	31	Direct key

Table 3.1 Explanation

Note1: I/O can be one of**:** "I" for Input, "I/O" for bidirectional, "OD" for open drain output, "P" for configurable pull-up/pull down, "PU" for pull up, "Hi-Z" for high impedance,.

Note2: The term "fail-safe" means that the TC35894FG can be powered down without harming the functionality of a wire attached to this pad or without causing current flow through the pad.

4. Power-On, Reset and Supply Surveillance

4.1. Power-On Sequence

The TC35894FG requires a single, nominal 1.62 V to 3.60 V power, supplied via pin 14, 16, 19, 35, 41 and 44.

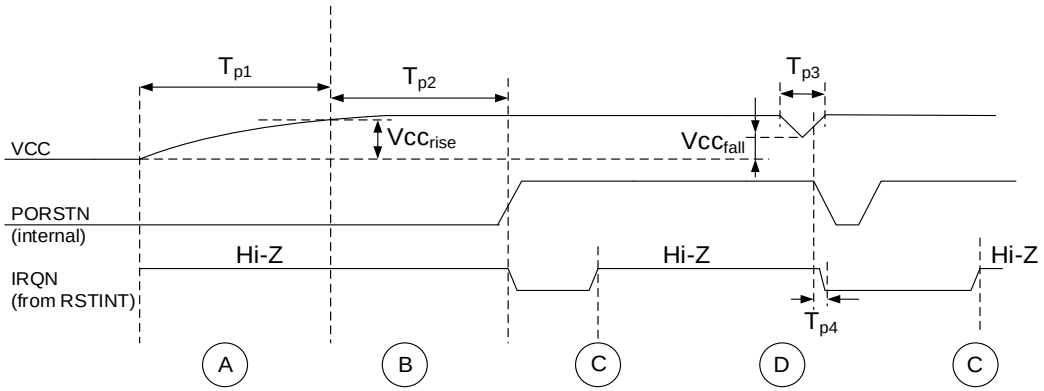


Figure 4.1 Power up and Supply watchdog control

4.2. Power-On Reset

A rising supply at VCC (region A in Figure 4.1) will exceed the power-on reset threshold VCC_{rise} after some time T_{p1}. The period T_{p1} must not exceed 80 µs, during which it is required that VCC rises monotonously. Sometime T_{p2} after VCC has reached $VCC_{rise,}$ the internal power-on-reset (PORSTN) is released. From this point onwards, a stable LSI operation can be granted and the I^2C Slave is ready to take commands from the host.

After Power-on Reset(PORSTN), interrupt output signal IRQN is active-low and interrupt status register IRQST.PORIRG (refer to chapter 11) is set "1." The interrupt for PORSTN needs to be clear by writing of IRQCLR for clear bit of register RSTINTCLR (C area).This clearing of PORIRQ should be done at part of any initialization process.

RSTINTCLR register (0x84)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	-	-	-	-	-	-	-	IRQCLR
	Default	*	*	*	*	*	*	*	0

IRQCLR Clears the RSTINT interrupt.
0 : No impact
1 : Clear PORSTN interrupt (does not need to be re-written to 0)

4.3. Power-On Watchdog

The stability of the VCC supply is continuously monitored. In case VCC drops some time below a threshold voltage VCC_{fall} (region D in Figure 4.1) the internal power-on-reset signal PORSTN will be re-activated on the rising edge of VCC and the entire the TC35894FG is reset. The host needs to reconfigure the TC35894FG. The procedure is the same as at initial power up.

The Power-ON-Reset detect level can be adjusted by the host during operation. In Table 13.4, the values given for VCC_{rise} and VCC_{fall} are defined by factory setting. These values can be overridden by register PORTRIM.

PORTRIM register (0x85)

This register controls Power-on-reset level.

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	POR_SEL	-	-	POR_ TRIM4	POR_ TRIM3	POR_ TRIM2	POR_ TRIM1	POR_ TRIM0
	Default	0	*	*	0	0	0	0	0

POR_SEL Override factory setting for Power-on-reset with POR_TRIMSEL value.
0 : Use factory setting
1 : Use value defined in POR_TRIM

POR_TRIM4:0 Power-on-reset detecting level (VCCfall), two's complement.

4.4. Reset Tree

Resetting the system is possible in four different ways.

- Power-On Reset
- Global reset via a dedicated input pin (asynchronous activation, synchronous release)
- Global software reset via I²C "general call" protocol, general call reset will also reset the I²C slave address (I2CSA).
- Software reset on a per-module basis

The power-on reset (PORSTN) is combined with the dedicated input pin (RESETN) and the I²C "general call" reset (GCRSTN) to a global asynchronous low-active reset signal.

Software reset can be done by programming the RSTCTRL register. This register contains one control bit per modules, so that each module can be independently reset. As the I²C is used for communication with the host, the I²C module cannot be reset by software. The I/O functions multiplexing cannot be reset by software in order to avoid to damage external devices.

This interrupt tells the host that the TC35894FG is ready to be used. The register EXTRSTN, reset by the PORSTN signal will, when set, detach the RESETN ball from the global reset tree. In this case, RESETN can be used as a fail-safe general purpose input. The logic state of the RESETN pin can then be read out at GPIODATA2 [0].

Note: For using the RESETN pin as general purpose input it is mandatory that the level on RESETN goes to high before detaching the RESETN pin from the global reset otherwise a low level on RESETN pin will keep the device always in reset state.

The following diagram shows the reset tree**:**

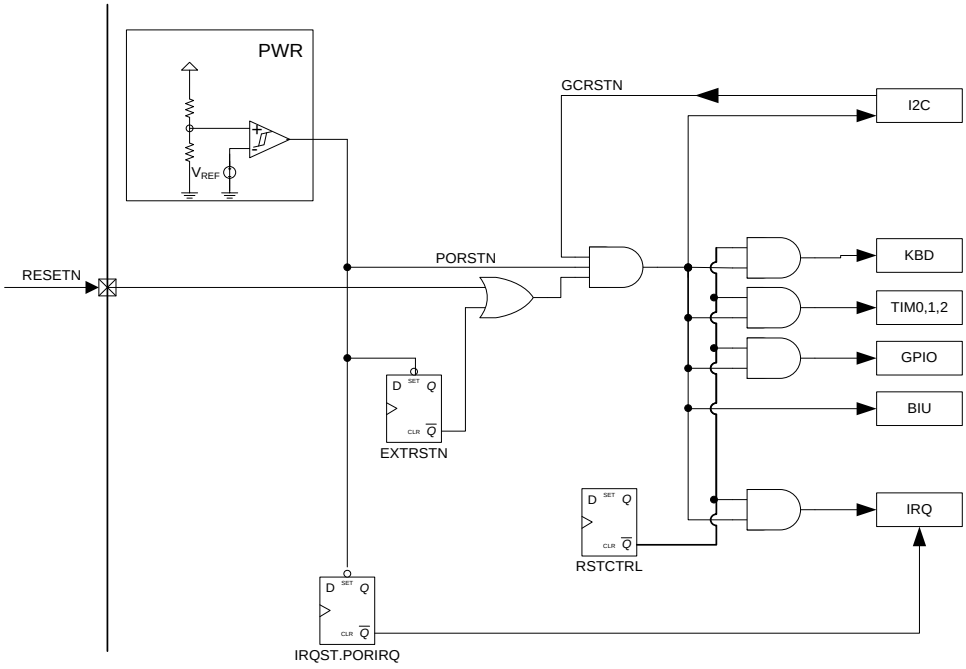


Figure 4.2 Reset Tree

4.5. Reset Registers

Modules that are not required should be set into software reset using the RSTCTRL register. When setting a module into reset, the corresponding clock shall also be switched off. When releasing a module from reset, first the reset shall be released, and then the corresponding module clock shall be switched on. For module clock control, please, refer to CLKEN register (0x8A).

The following table describes the Reset Control Register**:**

RSTCTRL register (0x82)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	Reserved	IRQRST	TIMRST	Reserved	KBDRST	GPIRST
	Default	*	*	0	0	0	0	0	0

IRQRST Interrupt Controller Reset.
Status on pin IRQN remains unaffected. This register bit is only used to control IRQ module register. Interrupt status read out is not possible, when this bit is set. It is recommended to leave this bit always at zero.
0 : Interrupt Controller not reset
1 : Interrupt Controller is reset (need to write back to 0, once reset)

TIMRST Timer Reset for timers 0,1 and 2.
0 : Timer not reset
1 : Timer is reset (need to write back to 0, once reset)

KBDRST Keyboard interface Reset
0 : Keyboard not reset
1 : Keyboard is reset (need to write back to 0, once reset)

GPIRST GPIO Reset
0 : GPIO not reset
1 : GPIO is reset (need to write back to 0, once reset)

EXTRSTN register (0x83)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	reserved	reserved	reserved	reserved	EXTRSTN
	Default	*	*	*	1	1	1	1	1

EXTRSTN External Reset pin (RESETN) Enable.
This register is not on the global reset line, it is reset only by a power-on reset.
0 : RESETN pin is not used as hardware reset.
1 : RESETN is used as hardware reset.

Note: The reserved bits must be set to 1.

4.6. Initial Configuration during Reset

KPY11:0, KPX7:0, PWM2:0, EXTIO0 and DIR25:24

When a global reset is active, the I/O pins KPY11...KPY0, KPX7...KPX0, PWM2...PWM0, EXTIO0, DIR24 and DIR25 are switched back to GPI functionality with the GPI ports set into input direction (DIR24 and DIR25 can be used for direct key inputs but not as general purpose input/output). After a global reset, the I/O pins KPY11...KPY0, KPX7...KPX0, PWM2...PWM0, EXTIO0, DIR24 and DIR25 will see a pull-up resistance attached to them. These initial pull resistor settings are required to avoid damage on the GPI inputs. After release of the global reset, the setting can be overridden by software programming.

IRQN and SDA

During global reset, a pin SDA is set into input mode. There is no pull resistor on pins IRQN and SDA. The output pin IRQN is set to high impedance, when the global reset was triggered by a VCC power up or a power watchdog, IRQN will go low, as soon as the LSI is ready to operate. (Refer to Figure 4.1)

SCL and RESETN

SCL and RESETN pins are always in input mode. There are no pull resistors on pins SCL and RESETN.

When the host tries to access the TC35894FG via I^2C during an active PORSTN, the I^2C slave module will answer that host request by not sending an acknowledge bit and the access attempt will be disregarded.

5. CLOCKING SYSTEM

Figure 5.1 shows the clock distribution network. Programming registers are highlighted in yellow; they can be programmed using only the I²C input clock SCL. Modules requiring an additional internal clock called SYSCLK are highlighted in red. The black bold line indicates the clock line status after a global reset. Additionally SYSCLK input is necessary to the GPIO module when synchronization DBOUNCE.SYNC bit is set or direct key is used.

5.1. Clock Source Selection

The register CLKSRCSEL selects the clock input source. This can be

- Internal RC-oscillator clock (nominal frequency 2 MHz, varying over temperature and process). This is the default after reset.
- External LVCMOS input at DIR24 (32 kHz .. 20 MHz)

The input DIR24 is not fail-safe.

When VCC of the TC35894FG is powered down, a global clock line attached to DIR24 may be disturbed.

To ensure precision on the timer modules, an external clock line should be selected.

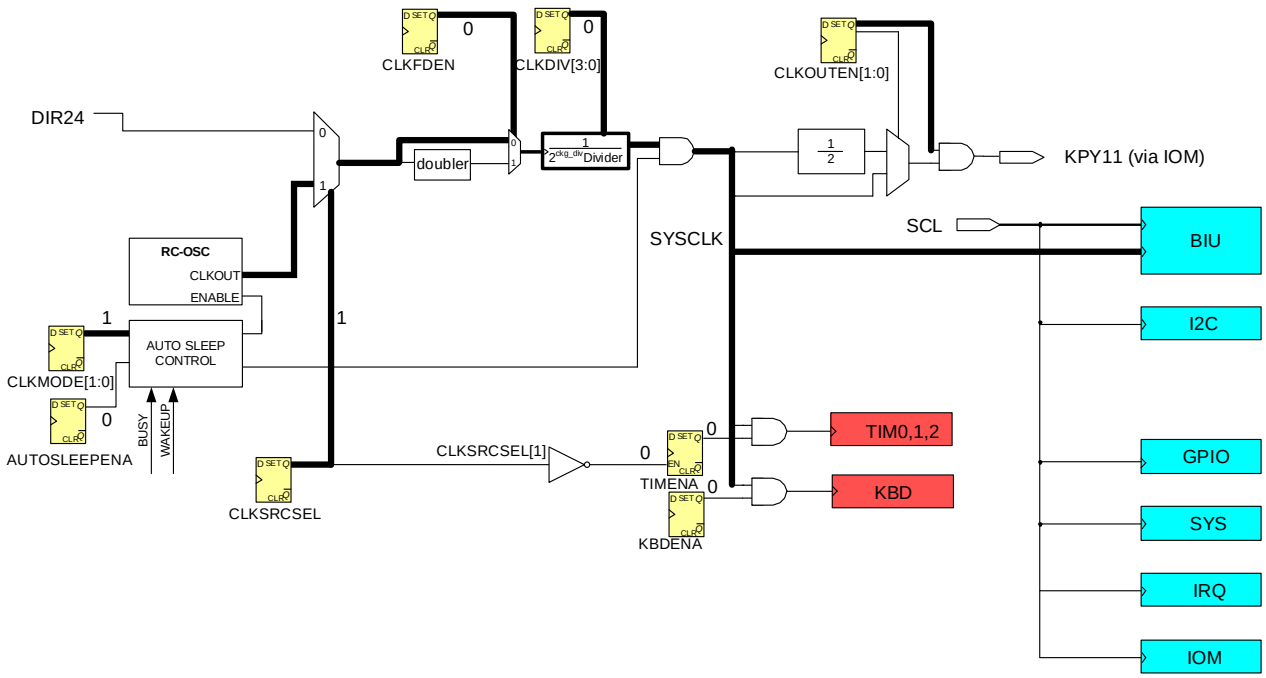


Figure 5.1 Clock distribution and path setup after global reset

5.2. Clock Frequency Setting

Behind the clock source selector multiplexer, the selected clock passes through a clock doubler and a divider to generate the internal SYSCLK.

The SYSCLK frequency must be greater than the SCL frequency divided by 6.5. When running on a 400 kHz I^2C bus clock to fulfill this constraint, the frequency of SYSCLK of using the clock doubler should be set in the range of

$$61.54kHz \leq f_{SYSCLK} \leq 200kHz$$

The SYSCLK is then distributed into the modules. Each module can be individually enabled or disabled. After a global reset, all individual module clocks are disabled.

The SYSCLK can be made available at the output pin KPY11, if the pin is configured accordingly.

The next figure shows a setting for a direct LVCMOS clock source. Again, the black bold lines and numbers stand for the required settings.

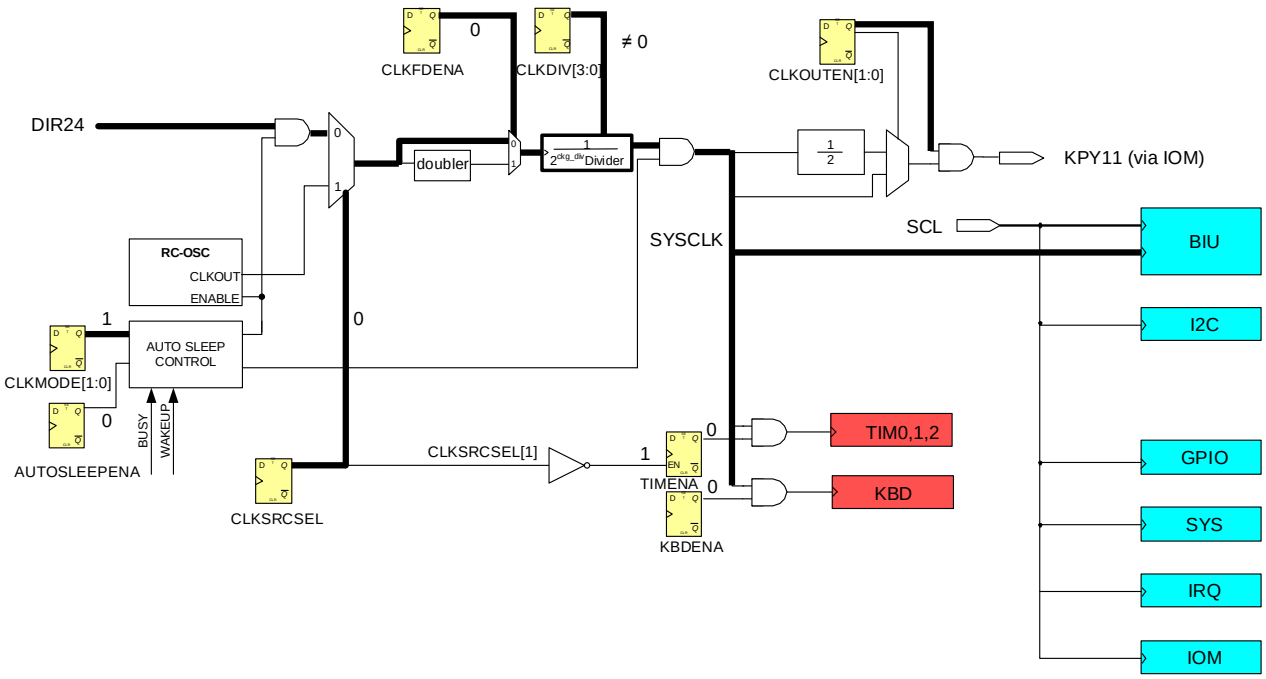


Figure 5.2 Clock system running on an LVCMOS clock feed at DIR24

5.3. Operating Modes

The CLKMODE register controls the operating mode of the TC35894FG.

5.3.1. SLEEP mode

In SLEEP mode, the internal RC-oscillator is disabled and the external clock LVCMOS is gated, so that the internal SYSCLK is not generated, resulting with very low power consumption.

It is recommended to always set I2CWAKEUPEN to 1 when the Auto-Sleep feature is used.

In SLEEP mode, register access via I^2C is not allowed into the Timer and the Keyboard unless Auto-Sleep and I^2C automatic wakeup is enabled.

5.3.2. OPERATION mode

Figure 5.3 shows the transfer figure for each mode. After Power-on, the device becomes automatically operation mode.

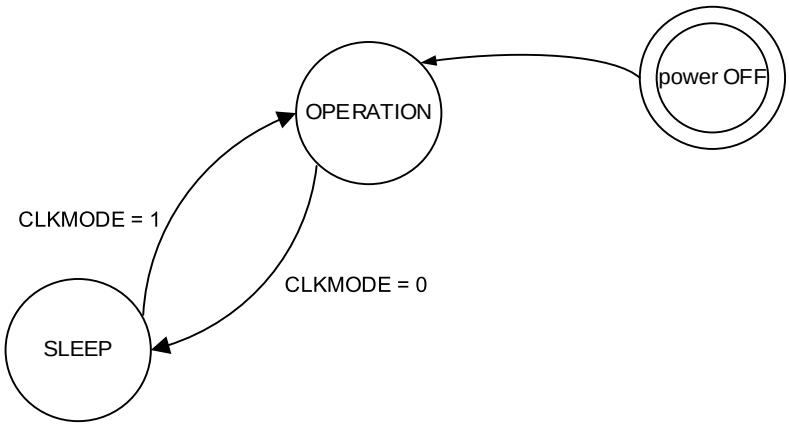


Figure 5.3 Mode state transitions

5.4. Auto Sleep Feature

The purpose of the Auto-Sleep feature is to control dynamically the internal RC clock. The Auto-Sleep function acts by operating directly on the CLKMODE.MODCTL register.

The Auto-Sleep function senses continuously the operation of the two modules TIM and KBD. When all two become inactive, a timer is started. On expiry of this timer, the Auto-Sleep state will be transferred to Sleep mode. Before expiry of this timer, something to operate is sensed by TIM or KBD module and Sleep mode is finished and transferred to OPERATION mode.

The TIM and KBD modules become OPERATION mode when the following any conditions are filled;

- TIM or KBD state machines are busy.
- An interrupt from these modules is pending.
- An event on one of the LSI's pins is sensed and the corresponding GPIOWAKE register bit for this pin is activated.
- The PWM pattern generator stops its activity, if it encounters an END pattern with the RST bit set to 1.
- Optionally an I^2C access to either KBD or TIM can be defined as an event by programming the I2CWAKEUPEN register.

The Auto-Sleep feature is available for only internal RC-oscillator.

Note:

For TC35894FG it is recommended to always set I2CWAKEUPEN to 1 (enabled) when the Auto-Sleep feature is used. Application depending on the time share between times of activity and inactivity, the Auto-Sleep feature can practically reduce operating power of the device down to the quiescent power figure. It is highly recommended to enable this feature.

5.5. Clock System Register settings

The following register tables describe the clock mode settings.

CLKMODE register (0x88)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	-	Reserved	MODCTL
	Default	*	*	*	*	*	*	0	1

MODCTL This register determines the operating mode.
0 : SLEEP mode, no SYSCLK generation
1 : OPERATION mode

AUTOSLPENA register (0x8B)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	-	-	ENABLE
	Default	*	*	*	*	*	*	*	0

ENABLE Auto-Sleep feature enable
When Auto-Sleep is on, the register MODCTL is controlled under a state machine and should not be programmed directly. Also, the register CLKCFG should not be programmed, when Auto-Sleep is enabled.
0 : Auto-Sleep feature is off
1 : Auto-Sleep feature is on

AUTOSLPTIMER register (0x8C)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	UPTIME10	UPTIME9	UPTIME8
		UPTIME7	UPTIME6	UPTIME5	UPTIME4	UPTIME3	UPTIME2	UPTIME1	UPTIME0
	Default	*	*	*	*	*	1	1	1
		1	1	1	1	1	1	1	1

UPTIME10:0 Minimum time the TC35894FG stays in OPERATION mode.
Counts SYSCLK cycles before going into SLEEP mode. The value programmed here is multiplied by 64 and copied into the timer at the time AUTOSLPENA.ENABLE is set to 1.

CLKCFG register (0x89)

The register CLKCFG should only be written once after power up, at a time by where registers in CLKEN are still disabled.

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	reserved	CLKSRCSEL	reserved	CLKFD EN	CLKDIV3	CLKDIV2	CLKDIV1	CLKDIV0
	Default	0	1	0	0	0	0	0	0

CLKSRCSEL Clock source selector
This switch shall not be modified, if CLKMODE. MODCTL is at SLEEP.
0 : LVCMOS clock input
1 : Internal RC-oscillator

CLKFDEN Clock frequency doubler enable (should only be enabled when CLKDIV=0)
0 : Disable clock frequency doubler.
1 : Enable clock frequency doubler.

CLKDIV3:0 Clock divider for SYSCLK
Used to divide the clock source to the SYSCLK frequency. Keep in mind that SYSCLK frequency *6.5 must exceed the maximum allowed SCL frequency.
Clock division ratio is 2^{CLKDIV}
0x0 : Divide by 1
0x1 : Divide by 2
0x2 : Divide by 4
|
0x9 : Divide by 512 (maximum legal division factor)
0xA : reserved
| reserved
0xF : reserved

CLKEN register (0x8A)

The CLKEN register is used for individual clock enabling for each module. The I^2C module doesn't have a clock enable as it runs only using the I^2C SCL clock line.

Setting any clock enable bit inputs the SYSCLK into the selected module, but only when clock mode is not in SLEEP mode.

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	CLKOUTEN1	CLKOUTEN0	TIMOSCEN2	TIMOSCEN1	TIMOSCEN0	TIMEN	reserved	KBDEN
	Default	0	0	Write Only	Write Only	Write Only	0	0	0

CLKOUTEN1:0 Clock output enable Output clock is selected from KPY11 ball. (Refer to chapter 5.2)
00 : CLKOUT clock disabled. Fixed to Low level.
01 : CLKOUT frequency = SYSCLK frequency
11 : CLKOUT frequency = ½ SYSCLK frequency
10 : Reserved

TIMOSCEN[2:0] Timer Clock Enable for using Internal RC-OSC
101 : Timer clock enable (Note 1)
Other : Timer clock disable

TIMEN Timer 0,1,2 clock enable
0 : Timer 0, 1 and 2 clock disabled
1 : Timer 0, 1 and 2 clock enabled. (Note 1)

KBDEN Keyboard clock enable
0 : Keyboard clock disabled.
1 : Keyboard clock enabled.

Note 1:

In case of using the external clock input, the timer clock is valid in TIMEN bit = 1.

In case of using internal RC-oscillator, the following continuous twice setting is necessary to be valid the timer clock.

Set TIMEN = 1 and TIMOSCEN[2:0] = 3'b101.

Please consider the oscillator frequency tolerance when using the timer with the internal RC-oscillator!

I2CWAKEUPEN register (0x8E)

It is recommended to always set I2CWAKEUPEN.I2CWEN to 1 if Auto-Sleep is enabled by AUTOSLPENA.ENABLE. The device will wake-up automatically if there is any I^2C access to either KBD or TIM module. However if I^2C access is taking place to any other module like GPIO, the device will stay in SLEEP mode. Please be careful that Auto-Sleep needs to be enabled for this feature to function.

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	-	-	I2CWEN
	Default	*	*	*	*	*	*	*	1

I2CWEN I^2C wake-up enable
0 : Device does not wake-up by I^2C access to KBD/TIM module when in SLEEP.
1 : Device wakes up by I^2C access to KBD/TIM module when in SLEEP.

6. IOM (INPUT/OUTPUT CONFIGURATION)

The TC35894FG provides an I/O multiplexer module. In addition inside this module, the pull-up or pull-down resistor programming onto the functional input pads and the I/O drive strength of the output pads can be programmed. The IOCFG should be written only once after reset during the initialization sequence. It should be written, before the SYSCLK is enabled.

6.1. Functional I/O Multiplex

After reset, the TC35894FG comes up having all pins configured in direct key functionality.

The pins KPX [7:0], KPY [11:0], PWM [2:0] and EXTIO0 can be configured as general purpose input/output or as direct key by DIRECT3…DIRECT0 register within the GPIO module.

The pins KPX [7:0] and KPY [11:0] can be configured as part of a keyboard matrix or as dedicated keyboard depending on KBDSIZE register and KBDDEDCFG register setting in the keyboard module.

In addition, the pins KPY [11], PWM [2:0], EXTIO0 and DIR24 can be configured to other functionality by the BALLCFG register bit. Please refer to Table 6.1.

IOCFG register (0xA7)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	GPIOSEL3	GPIOSEL2	GPIOSEL1	GPIOSEL0	IG	Reserved	BALLCFG1	BALLCFG0
	Default	1	1	1	1	1	0	0	0

GPIOSEL3:0 Control allotted functions for DIR24 and PWM [2:0] pins.
0 : Use functionality defined in BALLCFG for DIR24 and PWM[2:0] pins.
1 : Allotted GPIO functions to DIR24 and PWM [2:0].

IG Global input gate
All of inputs can be gating. In this case, external input signals are not connected with internal circuit.
0 : Disable all inputs
1 : Enable all inputs

BALLCFG1:0 Pin configuration setting according to Table 6.1, Table 6.2 and Table 6.3.

Table 6.1 KPX [7:0] and KPY [11:0] pin setting

PIN	MODULE CONNECTIVITY		
	BALLCFG [1:0]		
	00	01	10
KPX0 (DIR0)	GPIO0 or KPX0 Row 0		
KPX1 (DIR1)	GPIO1 or KPX1 Row 1		
KPX2 (DIR2)	GPIO2 or KBD Row 2		
KPX3 (DIR3)	GPIO3 or KBD Row 3		
KPX4 (DIR4)	GPIO4 or KBD Row 4		
KPX5 (DIR5)	GPIO5 or KBD Row 5		
KPX6 (DIR6)	GPIO6 or KBD Row 6		
KPX7 (DIR7)	GPIO7 or KBD Row 7		
KPY0 (DIR8)	GPIO8 or KBD Col 0		
KPY1 (DIR9)	GPIO9 or KBD Col 1		
KPY2 (DIR10)	GPIO10 or KBD Col 2		
KPY3 (DIR11)	GPIO11 or KBD Col 3		
KPY4 (DIR12)	GPIO12 or KBD Col 4		
KPY5 (DIR13)	GPIO13 or KBD Col 5		
KPY6 (DIR14)	GPIO14 or KBD Col 6		
KPY7 (DIR15)	GPIO15 or KBD Col 7		
KPY8 (DIR16)	GPIO16 or KBD Col 8		
KPY9 (DIR17)	GPIO17 or KBD Col 9		
KPY10 (DIR18)	GPIO18 or KBD Col 10		
KPY11 (DIR19)	GPIO19 or KBD Col 11	SYS OUTCLK	GPIO19 or KBD Col 11

Table 6.2 PWM [2:0] and DIR24 pin setting

PIN	MODULE CONNECTIVITY			
	GPIOSEL	BALLCFG [1:0]		
		00	01	10
PWM0 (DIR20)	GPIOSEL0 = 1	GPIO20		
	GPIOSEL0 = 0	GPIO20	PWM0	PWM0
PWM1 (DIR21)	GPIOSEL1 = 1	GPIO21		
	GPIOSEL1 = 0	GPIO21	PWM1	PWM1
PWM2 (DIR22)	GPIOSEL2 = 1	GPIO22		
	GPIOSEL2 = 0	GPIO22	PWM2	PWM2
DIR24	GPIOSEL3 = 1	GPIO24 (direct Key only)		
	GPIOSEL3 = 0	GPIO24 (direct Key only)		Clock In

Table 6.3 EXTIO0 and DIR25 pin setting

PIN	MODULE CONNECTIVITY		
	BALLCFG [1:0]		
	00	01	10
EXTIO0	GPIO23		
DIR25	GPIO25 (direct Key only)		

Note:

In Table 6.1, the GPIO and keyboard functionality is depending on KBDSIZE and KBDDEDCFG register setting in the keyboard module. When GPIO functionality is used (KEBDSIZE = 0 and KBDDEDCFG = 0) then depending on DIRECT3 - DIRECT0 register it can be selected general purpose or direct key functionality.

6.1.1. I/O Multiplexing for KPX [7:0]

The I/O multiplexing to the KPX [7:0] pins is depicted in the following figure.

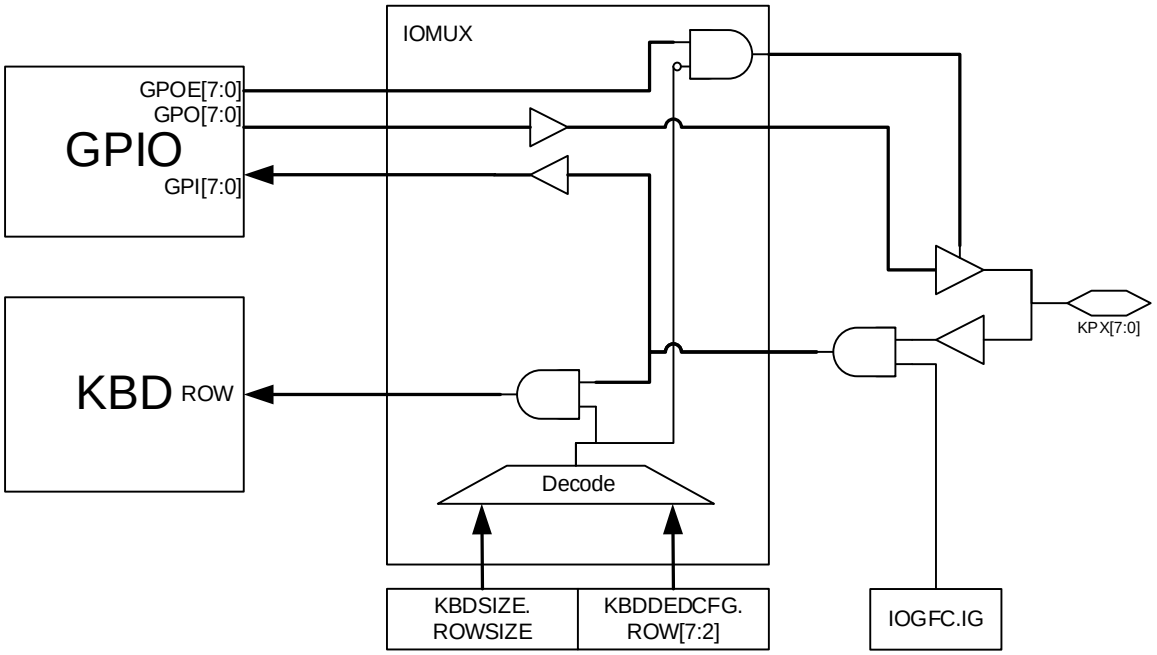


Figure 6.1 I/O Multiplexing for KPX [7:0]

Multiplexing of KPX [7:0] pins is controlled directly by the keyboard configuration registers KBDSIZE.ROWSIZE (Keyboard matrix configuration) and KBDDEDCFG.ROW [7:2] (Keyboard dedicated keys configuration) setting.

Signal condition can be monitored in GPIO module as KPX[7:0] input is connected with GPI[7:0] of GPIO module depended not on configuration of keyboard layout.

KPX [7:0] pin outputs are connected to GPIO module outputs. Output is enabled only for those pins that are not used by the keyboard interface.

6.1.2. I/O Multiplexing for KPY [10:0]

The I/O multiplexing to the KPY [10:0] pins is depicted in the following figure.

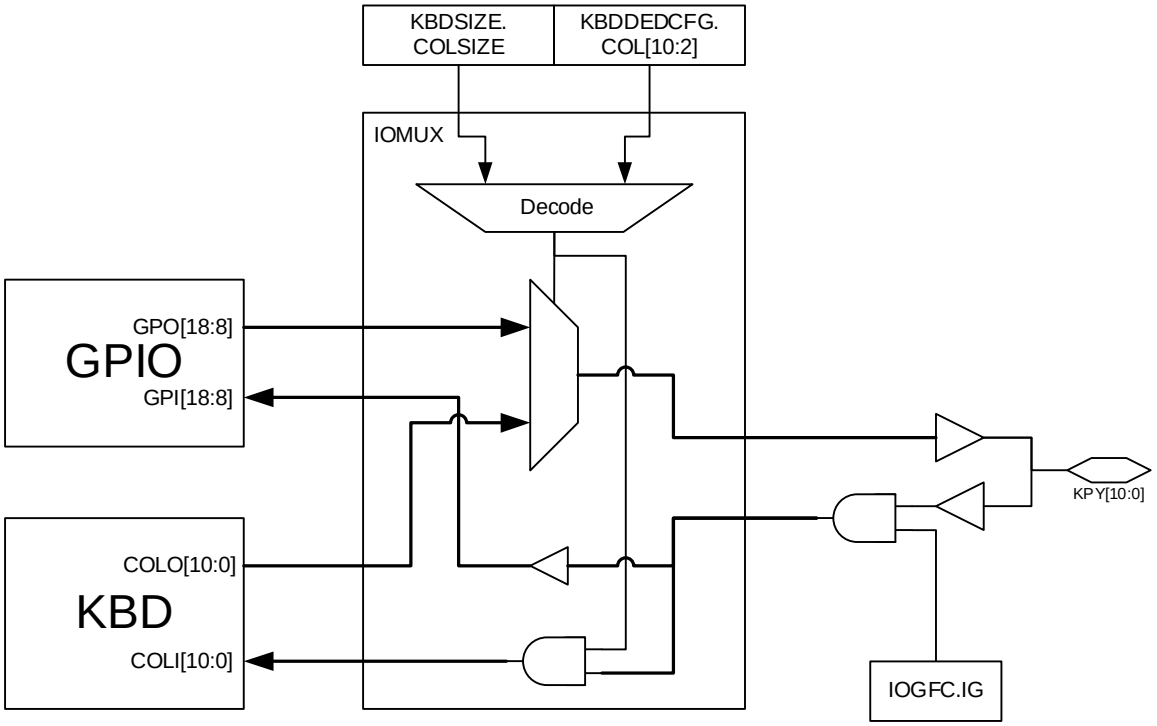


Figure 6.2 I/O Multiplexing for KPY [10:0]

KPY [10:0] are connected as KBD columns 10-0 for output.

GPIO signal condition can be monitored in GPIO module as input or output GPIO[18:8] is always connected with GPIO module.

6.1.3. I/O Multiplexing for KPY [11]

The I/O multiplexing to the KPY [11] pins is depicted in Figure 6.3. In case of no using KBD function, we recommend to keep the KBD clock shut down and the KBD module itself reset

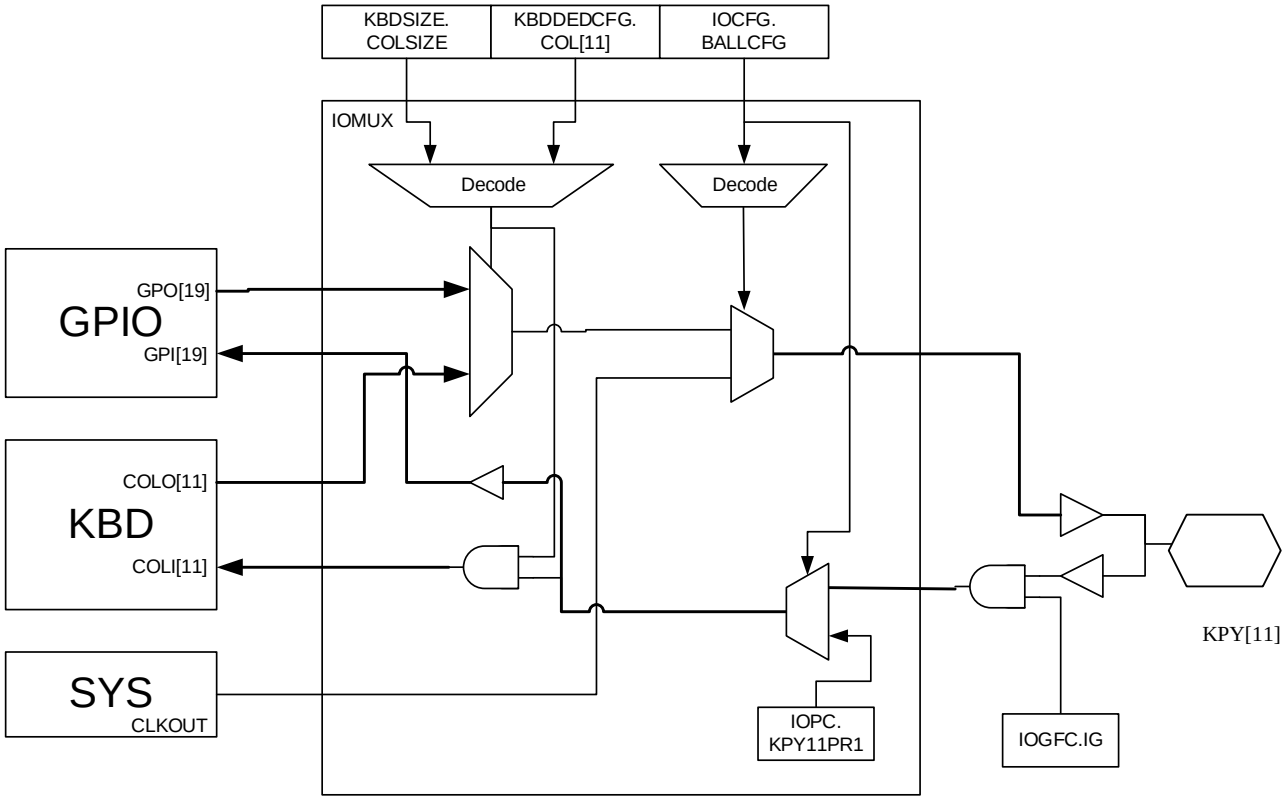


Figure 6.3 I/O Multiplexing for KPY [11]

When IOCFG.BALLCFG = 00 or 10, then depending on the keyboard matrix configuration KBDSIZE.COLSIZE and dedicated key configuration KBDDEDCFG.COL [11], the KPY [11] are connected either to keyboard columns 11 or to GPIO19.

When IOCFG.BALLCFG is 1, then KPY [11] is connected to the SYS module to output the clock.

6.1.4. I/O Multiplexing for PWM [2:0]

The I/O multiplexing to the PWM [2:0] pins are depicted in Figure 6.4.

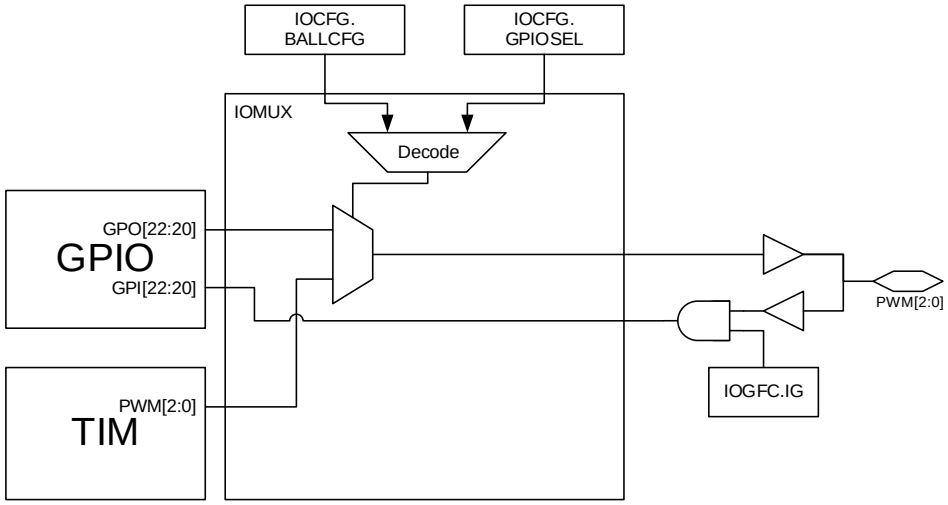


Figure 6.4 I/O Multiplexing for PWM [2:0]

Depending on IOCFG.GPIOSEL [2:0], it is available as PWM [2:0] or GPIO [22:20].

6.1.5. I/O Multiplexing for DIR24 (clock input)

The I/O multiplexing to the DIR24 pin is depicted in Figure 6.5.

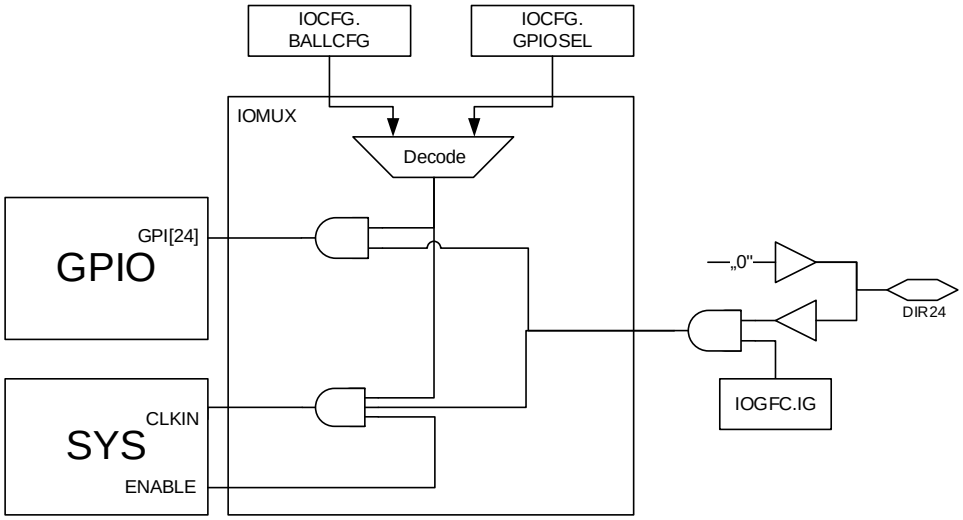


Figure 6.5 I/O Multiplexing for DIR24

Depending on IOCFG.GPIOSEL [3] the DIR24 pins is connected with GPI [24] as direct key input or CLKIN (external clock input) selected by IOCFG.BALLCFG.

6.2. Pull Resistor Programming

The following registers allow pull up/down resistor programming on each functional output of the TC35894FG. Floating pin inputs shall be avoided, setting the PR1:0 for any pin to a value different to 00 is strongly recommended on unconnected or undriven pins. The pull resistors are switched dynamically inactive, when the corresponding pin operates in output mode.

IOPCEXT register (0xA8)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	DIR25R1	DIR25R0	DIR24R1	DIR24R0
	Default	0	0	0	0	1	0	1	0

DIR[25:24]PR 1:0 Resistor enable for DIR[25:24] pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

IOPC0 register (0xAA)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	KPX7PR1	KPX7PR0	KPX6PR1	KPX6PR0	KPX5PR1	KPX5PR0	KPX4PR1	KPX4PR0
		KPX3PR1	KPX3PR0	KPX2PR1	KPX2PR0	KPX1PR1	KPX1PR0	KPX0PR1	KPX0PR0
	Default	1	0	1	0	1	0	1	0
		1	0	1	0	1	0	1	0

KPX [7:0] PR1:0 Resistor enable for KPX [7:0] pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

IOPC1 register (0xAC)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	KPY7PR1	KPY7PR0	KPY6PR1	KPY6PR0	KPY5PR1	KPY5PR0	KPY4PR1	KPY4PR0
		KPY3PR1	KPY3PR0	KPY2PR1	KPY2PR0	KPY1PR1	KPY1PR0	KPY0PR1	KPY0PR0
	Default	1	0	1	0	1	0	1	0
		1	0	1	0	1	0	1	0

KPY [7:0] PR1:0 Resistor enable for KPY [7:0] pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

IOPC2 register (0xAE)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	EXTIO0PR1	EXTIO0PR0	PWM2PR1	PWM2PR0	PWM1PR1	PWM1PR0	PWM0PR1	PWM0PR0
		KPY11PR1	KPY11PR0	KPY10PR1	KPY10PR0	KPY9PR1	KPY9PR0	KPY8PR1	KPY8PR0
	Default	1	0	1	0	1	0	1	0
		1	0	1	0	1	0	1	0

EXTIO0PR 1:0 Resistor enable for EXTIO0 pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

PWM[2:0]PR 1:0 Resistor enable for PWM[2:0] pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

KPY[11:8]PR 1:0 Resistor enable for KPY [11:8] pin
- 00 : No pull resistor
- 01 : Pull down resistor
- 10 : Pull up resistor (default)
- 11 : Pull up resistor

The RESETN, IRQN, the SDA and the SCL pad have no internal resistors attached, because they are fail-safe.
Eternal pull-up resistors must be connected on those pins.

6.3. Output Drive Strength Programming

All the TC35894FG outputs can be programmed to 3 different drive strengths. For each output two bits exist in the drive strength registers. Programming these two bits to "00" means weakest drive strength, programming it to "01" or "10" means medium drive strength and "11" means highest drive strength. The static current that can be output is a function of the drive setting and the V_{CC} supply voltage.

Table 6.4 Output Drive Programming

V_{CC}	DR [1:0] (Drive control in registers DRIVE0, 1, 2)		
	00	01/10	11
1.8 V	Low current	Medium current	High current
2.5 V	Medium current	High current	Don't use

DRIVE0 register (0xA0)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	KPX7DRV1	KPX7DRV0	KPX6DRV1	KPX6DRV0	KPX5DRV1	KPX5DRV0	KPX4DRV1	KPX4DRV0
		KPX3DRV1	KPX3DRV0	KPX2DRV1	KPX2DRV0	KPX1DRV1	KPX1DRV0	KPX0DRV1	KPX0DRV0
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

KPX [7:0] DRV1:0 Output drive strength for KPX [7:0] pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

DRIVE1 register (0xA2)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	KPY7DRV1	KPY7DRV0	KPY6DRV1	KPY6DRV0	KPY5DRV1	KPY5DRV0	KPY4DRV1	KPY4DRV0
		KPY3DRV1	KPY3DRV0	KPY2DRV1	KPY2DRV0	KPY1DRV1	KPY1DRV0	KPY0DRV1	KPY0DRV0
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

KPY [7:0] DRV1:0 Output drive strength for KPY [7:0] pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

DRIVE2 register (0xA4)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	EXTIO0 DRV1	EXTIO0 DRV0	PWM2 DRV1	PWM2 DRV0	PWM1 DRV1	PWM1 DRV0	PWM0 DRV1	PWM0 DRV0
		KPY11 DRV1	KPY11 DRV0	KPY10 DRV1	KPY10 DRV0	KPY9 DRV1	KPY9 DRV0	KPY8 DRV1	KPY8 DRV0
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

EXTIO0DRV1:0 Output drive strength for EXTIO0 pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

PWM [2:0] DRV1:0 Output drive strength for PWM2, PWM1 and PWM0 pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

KPY [11:8] DRV1:0 Output drive strength for KPY[11:8] pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

DRIVE3 register (0xA6)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	IRQNDRV1	IRQNDRV0	SDADRV1	SDADRV0
	Default	*	*	*	*	0	0	0	0

IRQNDRV1:0 Output drive strength for IRQNDRV pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

SDADRV1:0 Output drive strength for SDADRV pin
- 00 : Lowest strength (default)
- 01 : Medium strength
- 10 : Medium strength
- 11 : Highest strength

7. I²C

The following features are supported by the I²C slave module.

- 1.8 V fail-safe I²C pad operation
- 400 kHz fast mode operation
- 7 bit slave address recognition
- Default slave address is "1000101."
- Full reprogramming feature for 7-bit slave address
- General call issues global reset onto the TC35894FG (byte 0x06 support) without resetting the SA
- Auto increment on register address to allow read and write bursts access for consecutive register addresses.

The I²C slave interface handles all internal LSI communication with the host. The TC35894FG uses an 8 bit address index scheme to access one from it's up to internal registers.

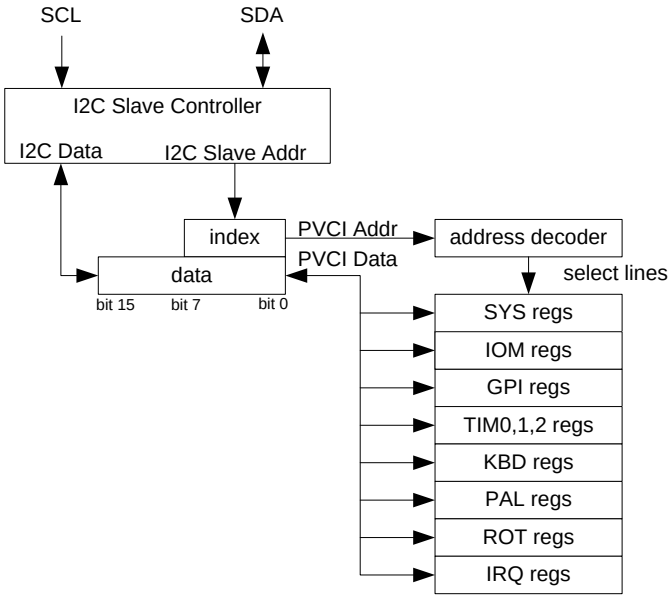


Figure 7.1 Programmer's model and I²C decode

The I²C slave is compatible to I²C "normal" (up to 100 kHz) and "fast" mode (up to 400 kHz). The I²C responds to a general call (SA [7:0] = "00000000") and to a 7-bit device address SA [7:1]. The RESTART detection is supported.

7.1. Re-Programming of the I²C Address

The default slave address can be changed to any 7-bit I²C address by writing the new address into register I2CSA. This feature is only available when the DIR24 pin is configured as clock input, or when DIR24 is either connected to an external clock input or pulled up to VCC. In case of disabled the slave address changing, DIR24 pin should be connected with GND or external pull-down resistor.

A write access into the register I2CSA is accepted if after the power-on reset at least eight cycles of SYSCLK were counted inside the TC35894FG or immediately accepted if the DIR24 input is pulled up to VCC after power-on reset.

7.2. I²C Transfer

The I²C Slave supports 8-bit data transfer and bursts of 8 bit data transfers. Some internal registers are defined as two byte registers requiring a 2 byte data access on the I²C. For those registers, if write access is aborted after the transmission of the first byte, the register content is not changed.

The I²C master sends a START condition (S-bit) onto the I²C bus, which initiates the slave interface to match its own device address with the address sent over the I²C bus. Upon successful device address matching, the slave replies with an ACK = 0.

7.2.1. I²C Write Operation

The host, as a master puts an 8 bit register address (RA7..RA0) onto the I²C, the TC35894FG I²C slave puts this register address into the index register. If SA [0] equals "0", the TC35894FG I²C slave interface can write one or more data bytes from the I²C bus acknowledging each with an acknowledge bit ACK = 0. The index register is automatically increment, and can send the optional byte data as burst transfer. After that, the master stops the transfer by sending a STOP condition (P-bit). When a STOP condition is received, the index register returns to the register address at starting for burst access.

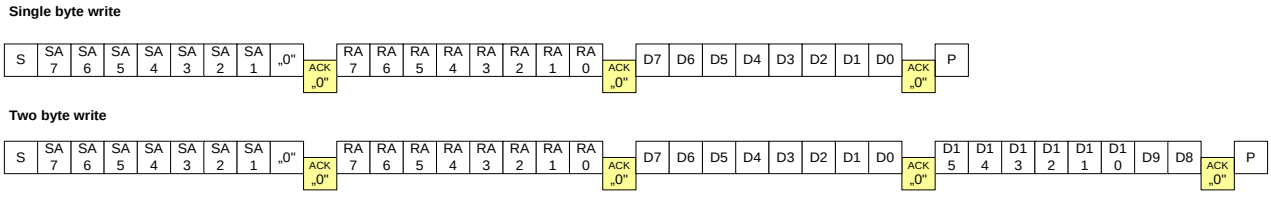


Figure 7.2 Host write access from I²C address

7.2.2. I²C Read Operation

When the TC35894FG index register is read, host should be set the index register by the following process. (Figure 7.3 upper side)

The slave address and register address are set after START of RESTART condition. In this case, due to write access SA[0] = 0 should be set.

After index register is set, read access should be achieved by the following process. (Figure 7.3 lower side)

The host should set slave address again. In this case, read access is enable by SA[0] = 1. TC35894FG outputs designate register value by index address after returning ACK = 0.

After host receives first byte, read can interrupt by sending STOP condition. Index address is automatically increment every one byte read.

In Figure 7.4, the yellow shaded bits are controlled by the I²C slave of the TC35894FG, whereas the non-shaded bits are controlled by the I²C master.

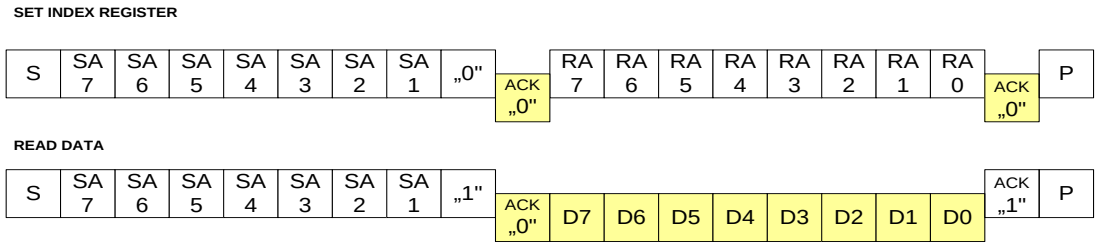


Figure 7.3 Single byte host read access from I²C slave

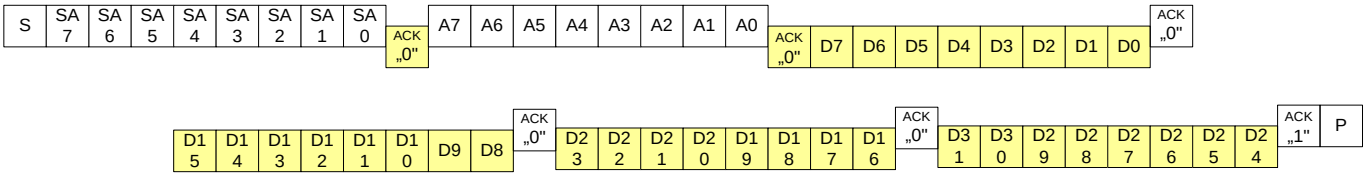


Figure 7.4 2-byte host read access from I²C slave

7.2.3. I²C General Call

The TC35894FG reacts to a general call command from the host in two different ways, depending on bit B**:**

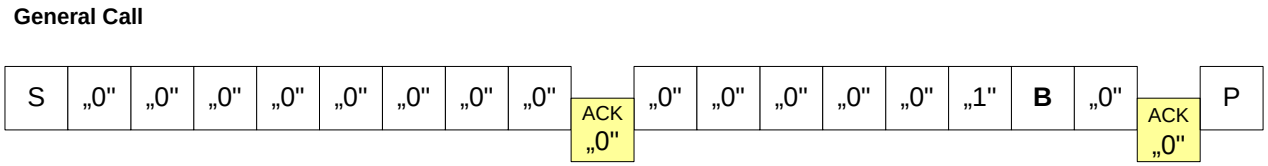


Figure 7.5 General Call command

When B is 1, the TC35894FG is globally reset (during the ACK cycle before the STOP condition) without resetting the I²C Slave Address (SA).

When B is 0, then the I²C Slave Address (SA) will be reset back to 8AH in case it was changed before.

Note:

After power-on-reset, the general call functionality can be used only after having read out the manufacturer code (0x80) and the software version number (0x81) of the TC35894FG.

7.3. I²C Register Map

The complete TC35894FG register map can be found in chapter 16.

8. TIM (Timer Module)

The Timer module provides precisely timed events in a large timing range with a high resolution. The design is optimized to suit applications, such as LCD controller back light steering or vibrator control.

8.1. Timer Features

- Easy timing adjustable by programmable prescaler
- System clock counter function
- Continuous operating mode (perpetual repetition)
- One shot timer mode
- PWM duty cycle programmable function
- PWM duty cycle modulation
- Timing event generating function independent from host controller in the pattern storage register
- Timer triggered interrupt to host
- Timer cascading possible

8.2. Timer Architecture

The Figure 8.1 below shows the basic architecture of one timer channel. The timer consists of three parts. The blue shaded boxes belong to the basic timer functionality, the yellow shaded boxes are the modules that control PWM duty cycle modulation and the green shaded boxes correspond to a pattern generator.

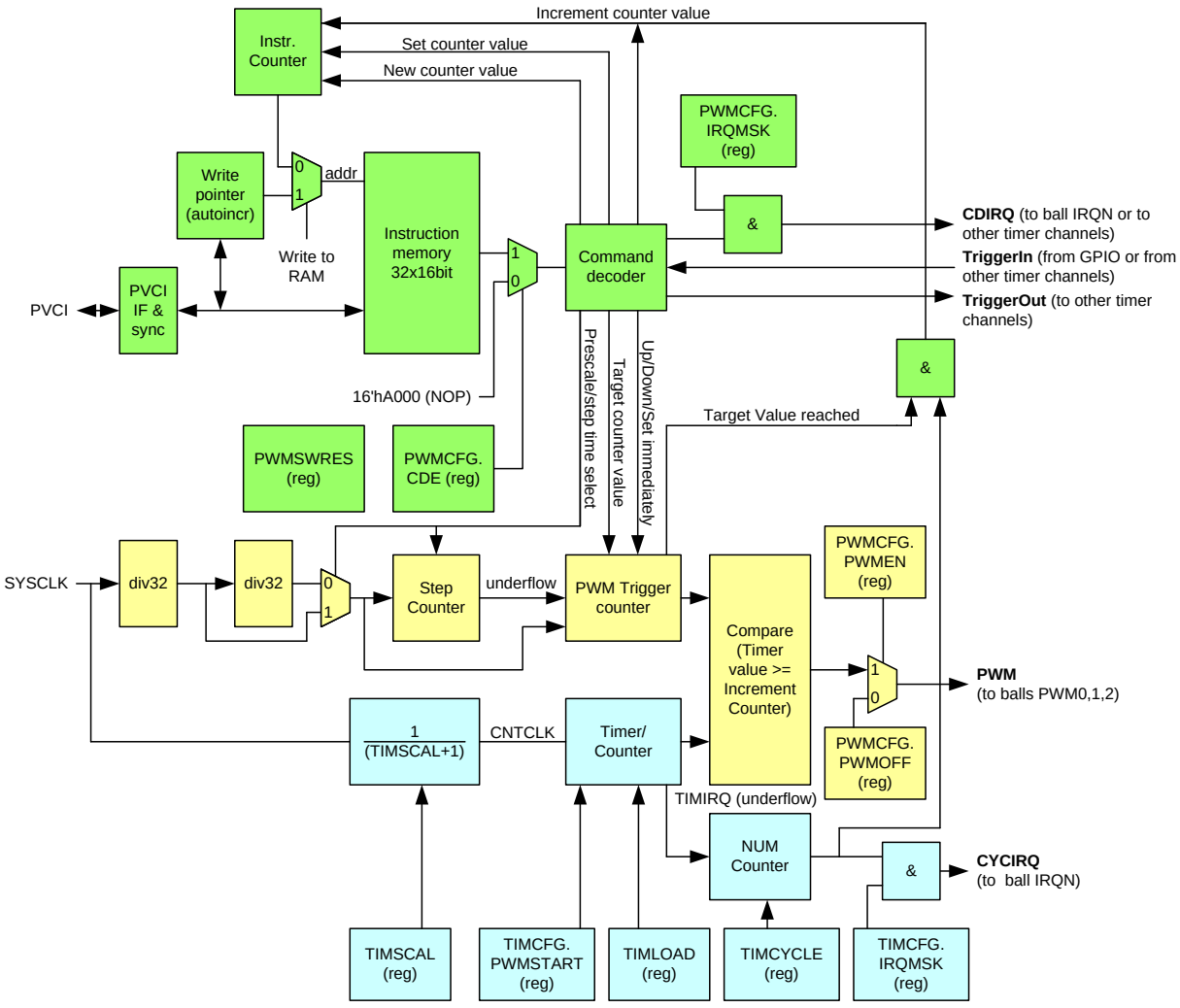


Figure 8.1 Timer Architecture

8.3. Simple Timer Control

The Timer/Counter counts pre-scaled system clocks SYSCLK until one timer period defined by register TIMLOAD is reached. Register TIMSCAL defines the 8 bit pre-scaler value. Once the timer has expired and the NUM Counter is incremented and is reached the value which is set in register TIMCYCLE, an interrupt CYCIRQ is triggered. The final frequencies at CYCIRQ is defined by Equation (2).

Equation (1): $$f_{PWM} = f_{SYSCLK} \cdot \frac{1}{(TIMSCAL + 1)} \cdot \frac{1}{(TIMLOAD + 1)}$$

Equation (2): $$f_{CYCIRQ} = f_{PWM} \cdot \frac{1}{(TIMCYCLE + 1)}$$

The SYSCLK is an internal clock. Regarding the detail information, refer to Chapter 5.

The followings are timer setting registers. Each register is set at every timer channel.

TIMSCAL 0, 1 and 2 register (0x62, 0x6A, 0x72)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	SCAL7	SCAL6	SCAL5	SCAL4	SCAL3	SCAL2	SCAL1	SCAL0
	Default	0	0	0	0	0	0	0	0

SCAL7:0 Load value for timer pre-scaler.
The system clock is divided by (SCAL+1) and makes CNTCLK. The resulting CNTCLK is the reference clock for timer related operations.

TIMLOAD 0, 1 and 2 register (0x64, 0x6C, 0x74)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	LOAD7	LOAD6	LOAD5	LOAD4	LOAD3	LOAD2	LOAD1	LOAD0
	Default	1	1	1	1	1	1	1	1

LOAD7:0 The timer/counter counts down from TIMLOAD value to 0 in (LOAD+1) steps. The value programmed into this register is transferred into the timer/counter synchronously to the pre-scaled timer clock CNTCLK.

TIMCYCLE 0, 1 and 2 register (0x63, 0x6B, 0x73)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	CYCLE7	CYCLE6	CYCLE5	CYCLE4	CYCLE3	CYCLE2	CYCLE1	CYCLE0
	Default	0	0	0	0	0	0	0	0

CYCLE7:0 Generated timing setting for timer interrupt (CYCIRQ)
0 : Interrupt generated immediately, when timer/counter expired.
N : Interrupt generated after N+1 expiries of timer/counter.

TIMCFG 0, 1 and 2 register (0x60, 0x68, 0x70)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	IRQMASK	CYC CTRL	FREE	SYNC	START
	Default	*	*	*	0	0	0	0	0

IRQMASK interrupt mask for CYCIRQ
0 : Interrupt enabled
1 : Interrupt masked

CYCCTRL CYCLE counter control register
0 : Timer/counter stops after TIMLOAD cycles of CNTCLK. The interrupt is issued when the NUM Counter (TIMCYCLE controller) is only at 0.
1 : Timer/counter counts TIMLOAD cycle as many times as specified in the TIMCYCLE register. Then, the timer stops and an interrupt is generated.

FREE Switches between free-running timer and one time count. In both operating modes, the register TIMCYCLE influences the behavior of the interrupt generation.
0 : One Time Count Timer mode. The interrupt depends on TIMCYCLE bit.
1 : Free Running mode. After timer/counter counts down from TIMLOAD to 0, the value of TIMLOAD is re-loaded and count down is restarted again.

SYNC Synchronization of pattern generator and timer
0 : Pattern generator is started and stopped by the PWMCFG.PGE bit
1 : Pattern generator and Timer are enabled simultaneously by setting bit TIMCFG.START, pattern generator is stopped by PWMCFG.PGE=0, timer is stopped by TIMCFG.START=0

START Timer start/stop control. **WRITE ONLY!**
0 : Timer is stopped (can also be stopped from internal state machine)
1 : Timer is started.

In the next diagrams, the different settings of TIMCFG are explained.

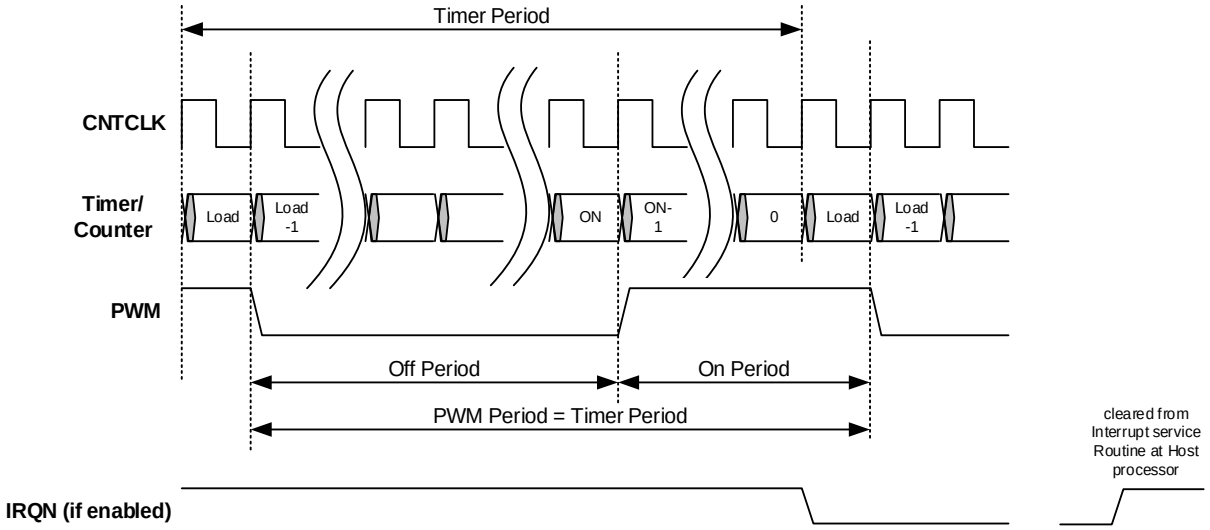


Figure 8.2 Timer in free-running mode (TIMCFG.FREE = 1)

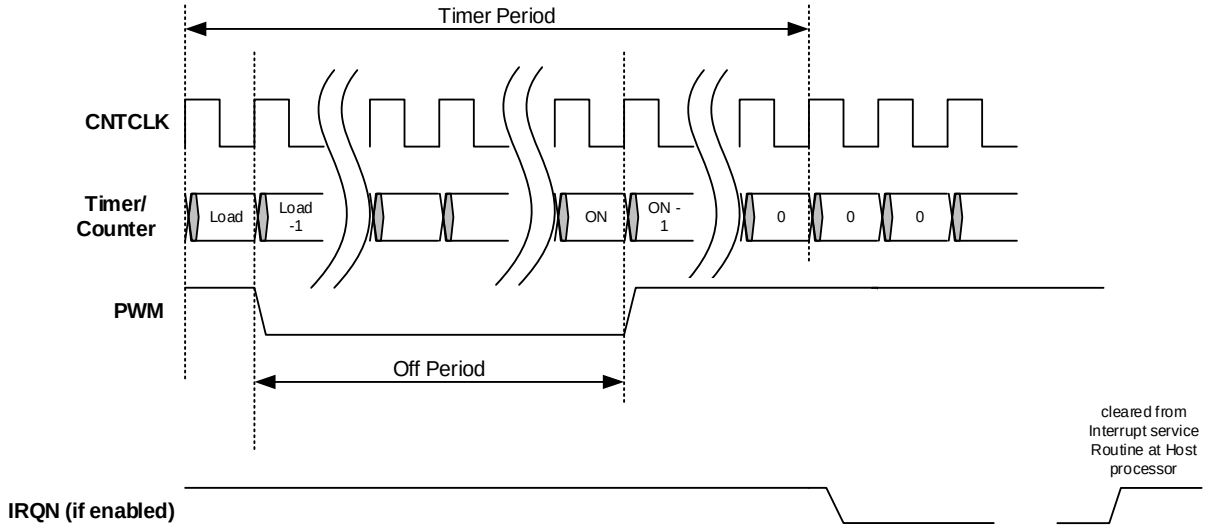


Figure 8.3 Timer in one-shot mode (TIMCFG.FREE = 0, TIMCFG.CYCLE = 0)

8.4. PWM Generation

PWM modulated timing is adjusted by the values of TIMLOAD and PWM trigger counter. The value of PWM trigger counter can be able to set by pattern generator. (Refer to Chapter 8.8. Pattern Set.)

After reset, the PWM trigger counter is set to 0 and PWMCFG.PWMEN is also reset to 0 so that the ball PWM is set low.

Operating in case of set RAMP pattern in pattern generator is shown in Figure 8.4. It can be increased at step by step duty ratio of the PWM output.

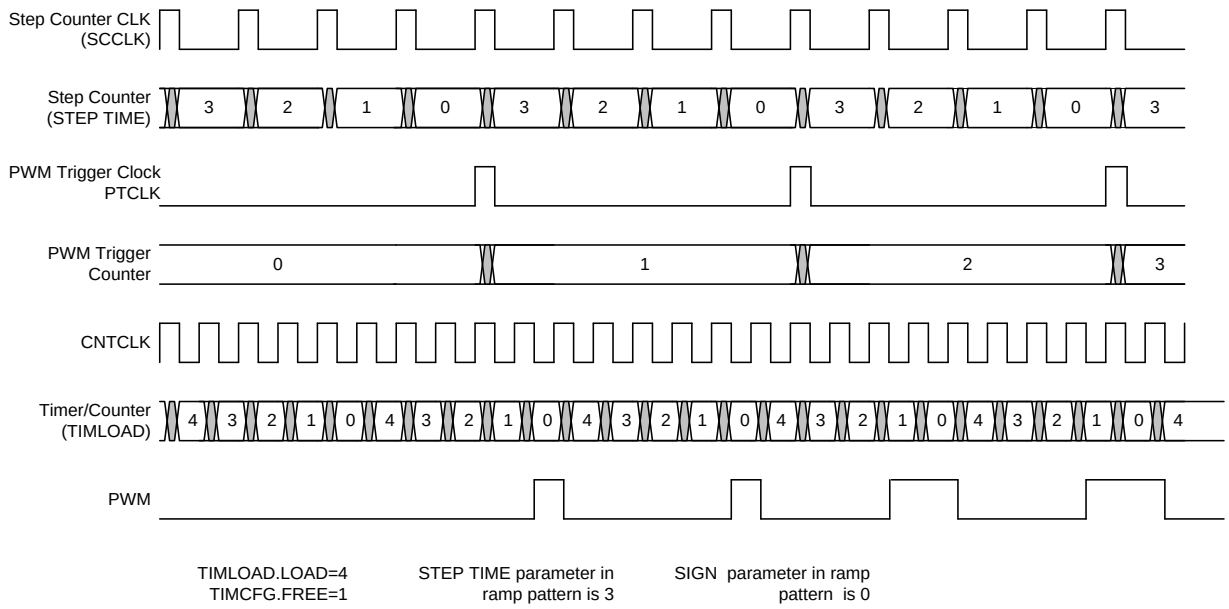


Figure 8.4 PWM modulation timing showing slowly increasing duty cycle modulation

8.5. Pattern Storage Register Access

The pattern storage register is access from the host via I^2C. Due to using Pattern Generator, the complex PWM duty cycle modulated patterns is enable so easily.

8.6. Pattern Storage Register Control

All three timers share one common software register TIMSWRES.

TIMSWRES register (WRITE ONLY) (0x78)

These bits reset the pattern generator and stops it (stops all state-machines and timer). Patterns stored in the pattern configuration register remain unaffected. Since interrupts from TIMERs are not cleared, they need to be cleared by writing into register TIMIC.

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	-	-	-	-	-	SWRES2	SWRES1	SWRES0
	Default	*	*	*	*	*	0	0	0

SWRES2 Software reset of TIMER2
0 : no action
1 : Software reset on timer 2, needs not to be written back to 0.

SWRES1 Software reset of TIMER1
0 : no action
1 : Software reset on timer 1, needs not to be written back to 0.

SWRES0 Software reset of TIMER0
0 : no action
1 : Software reset on timer 0, needs not to be written back to 0.

PWMCFG 0, 1 and 2 register (0x61, 0x69, and 0x71)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	IRQMASK	PGE	PWMEN	PWMPOL
	Default	*	*	*	*	0	0	0	0

IRQMASK Mask for CDIRQ
0 : CDIRQ enabled
1 : CDIRQ disabled/masked

PGE Pattern Generator Enable
This bit is ignored, if the SYNC bit of the corresponding TIMCFG register is set
0 : Pattern generator disabled
1 : Pattern generator enabled

PWMEN PWM Enable
0 : PWM disabled.
PWM timer output assumes value programmed in PWMPOL
1 : PWM enabled

PWMPOL OFF-state of PWM output, when PWMEN=0.
0 : PWM off-state is low
1 : PWM off-state is high

TIMRIS register (READ ONLY) (0x7A)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	-	-	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
	Default	*	*	0	0	0	0	0	0

CDIRQ2:0 Raw interrupt status for CDIRQ timer 2, 1 and 0
0 : No interrupt pending
1 : Unmasked interrupt generated

CYCIRQ2:0 Raw interrupt status for CYCIRQ timer 2, 1 and 0
0 : No interrupt pending
1 : Unmasked interrupt generated

TIMMIS register (READ ONLY) (0x7B)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	-	-	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
	Default	*	*	0	0	0	0	0	0

CDIRQ2:0 Interrupt masking status for CDIRQ timer2, 1 and 0.
0 : No interrupt pending
1 : Interrupt generated

CYCIRQ2:0 Interrupt masking status for CYCIRQ timer2, 1 and 0.
0 : No interrupt pending
1 : Interrupt generated

TIMIC register (0x7C)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	-	-	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
	Default	*	*	0	0	0	0	0	0

CDIRQ2:0 Clears interrupt CDIRQ timer 2, 1 and 0
0 : No effect
1 : Interrupt is cleared. Does not need to be written back to 0.

CYCIRQ2:0 Clears interrupt CYCIRQ timer 2, 1 and 0
0 : No effect
1 : Interrupt is cleared. Does not need to be written back to 0.

8.7. Storing Patterns into the Pattern Storage Register

For writing a pattern sequence into the pattern storage register, the following steps are needed.

1. Setup of the timer channel
2. Writing timer pattern at the patterns storage register via PWMPAT register
3. Starting the timer and the pattern generator
4. Detection for interrupt (for example, on an interrupt flag on CDIRQ or on an interrupt at pin IRQN)

An index register (PWMWP) shows address for the pattern storage register. Pattern Storage Register is automatically increment in case of write accessing to PWMPAT register.

When the pattern generator has finished or the pattern generation requires modification, the pattern storage registers must be overwritten in a defined way. There are two different options to do this**:**

1. Issue a software reset by means of RSTCTRL register
2. Stop the pattern generation with the PWMCFG.PGE register bit.

PWMWP register (0x7D)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	POINTER6	POINTER5	POINTER4	POINTER3	POINTER2	POINTER1	POINTER0
	Default	*	0	0	0	0	0	0	0

POINTER6**:**0 Write Pointer Address for Pattern Storage Register. This register is automatically increment by write access to PWMPAT register.

0 ≤ POINTER < 32 **:** Write Pointer for Timer0 patterns
32 ≤ POINTER < 64 **:** Write Pointer for Timer1 patterns
64 ≤ POINTER < 96 **:** Write Pointer for Timer2 patterns
96 ≤ POINTER < 128 **:** Not Valid

PWMPAT register (0x7E)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	PAT15	PAT14	PAT13	PAT12	PAT11	PAT10	PAT9	PAT8
		PAT7	PAT6	PAT5	PAT4	PAT3	PAT2	PAT1	PAT0
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

PAT15:0 Input port to the pattern storage register. After writing by I²C-bus. PWMWP is automatically increment. Writing into this register must be written in two bytes burst accesses.

When writing into PWMPAT in an I²C burst write command, the I²C address is not increment beyond 0x7F but cycles back to 0x7E. The PWMWP is increment every two transmitted bytes

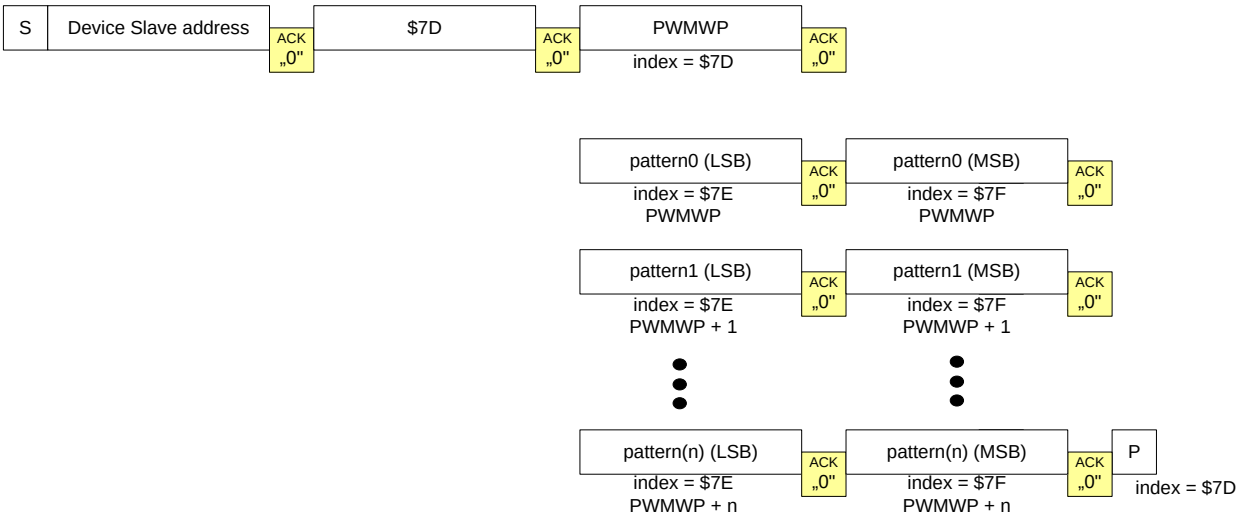


Figure 8.5 Burst Write Access to pattern storage register

8.8. Pattern Set

TC35894FG supplies pattern sets which shows in the following table.

Table 8.1 Pattern Overview

Pattern	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RAMP	0	PS	Step time						Sign	Counter increment (0-127)						
WAIT	0	PS	Step time						0	0	0	0	0	0	0	0
SET_PWM	0	1	0	0	0	0	0	0	Value							
RESTART	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NEW	1	0	1	0	0	0	0	0	0	0	"x"	New pattern index				
LOOP	1	0	1	Loop count (1-63)						1	"x"	Next pattern index				
END	1	1	0	IRQ	RST	"x"										
TRIGER	1	1	1	Input trigger channel						Output trigger channel						"x"

Note**:** "x" means "don't care".

8.8.1. RAMP Pattern

Table 8.2 RAMP Pattern

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RAMP	0	PS	Step time						Sign	Counter increment (0-127)						

FUNCTION

Steadily increases/decreases the PWM duty cycle of the PWM output pins.

PARAMETER

PS	0	Disable the pre-scaler
	1	Enable the pre-scaler (Refer to Table 8.1.)
Step time		Load value for step counter, defines update speed for PWM trigger. The PWM trigger is updated any time step counter expires.
Sign	0	Increment the PWM trigger on expiry of the step counter
	1	Decrement the PWM trigger on expiry of the step counter
Counter increment		Defines how many times the step counter expires for incrementing or decrementing the PWM trigger before the RAMP pattern finishes.

DESCRIPTION

With this pattern it is possible to be increased or decreased the duty cycle of the PWM output step by step. The PS bit (bit 14) influences the pre-scaler, set to 0 it divides the SYSCLK by 32, set to 1 it divides the SYSCLK by 1024. The parameter "step time" defines number of counts for the step counter. Every time the step counter expires, the PWM Trigger is incremented or decremented, depending on the sign flag in bit 7 and the step counter is re-loaded with parameter "step time." The value "counter increment" specifies, how many times the PWM Trigger increments (SIGN=0) or decrements (SIGN=1) before the RAMP pattern comes to an end.

The time until Finished for RAMP pattern (Ramp up time) is in the following Equation.

(Equation 3) PS = 0**:** RAMP up time = 32 × "Step time" × "Counter increment" × T_{SYSCLK}

(Equation 4) PS = 1**:** RAMP up time = 1024 × "Step time" × "Counter increment" × T_{SYSCLK}

After reset, all counters receive the value 0.

The following is actual example.

- External 32.768 kHz clock input
- PWM frequency for 1024 Hz at a SYSCLK of 65.536 kHz
- Setting in 32 steps between ON (100% intensity) and OFF (0% intensity) for LED

A RAMP pattern shall ramp up the LED light in 5 seconds from 25% intensity to 75% intensity.

- Register CLKCFG.CLKFDEN (clock doubler) is set to 1 in order to create 65.536 kHz SYSCLK from external clock source.
- TIMLOAD is set to 31, and LED is set for 32 intensity steps between ON (100% intensity) and OFF (0% intensity) state.
- TIMSCAL is calculated according to Equation (1)**:** TIMSCAL = 65536 / (32*1024) -1 = 1.
- Ramping from 25% to 75% requires exactly 16 ramping steps, meaning 16 times incrementing the PWM Trigger. This means that the RAMP pattern "counter increment" value must be 16.
- Ramping up requires the Sign bit in the RAMP pattern to be 0.
- It takes five seconds for RAMP up, and in the periods16 times increments for PWM trigger counter are necessary. Due to equation (4), the following Equation is concluded.

RAMP up time [s] = 1024 × "step time" × "Counter increment" × T_{SYSCLK}

Ramp up time = 5 [s], Counter increment = 16, T_{SYSCLK} = 1/65536 [s],
Due to these values, "Step time" parameter is 20.

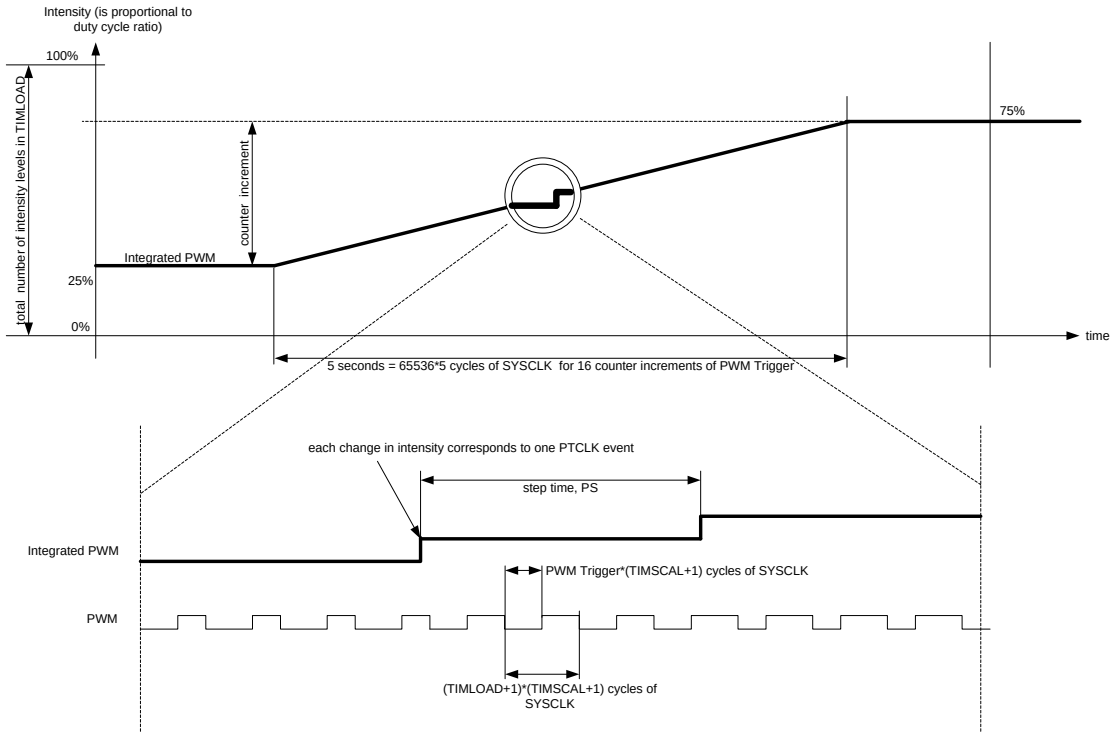


Figure 8.6 Ramping up a Light intensity from 25% to 75%

8.8.2. WAIT Pattern

Table 8.3 WAIT Pattern

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WAIT	0	PS	Step-time						0	0	0	0	0	0	0	0

FUNCTION

Pauses pattern generation for a defined number of cycles.

PARAMETER

PS Pre-scaler setting
0 : Disable the pre-scaler
1 : Enable the pre-scaler

Step-time Load value for step counter, defines the wait period.

DESCRIPTION

Pattern generation will pause for 32 × Step-time cycles of SYSCLK when PS is not set, and for 1024 × Step-time cycles when PS is set.

8.8.3. SET_PWM Pattern

Table 8.4 SET_PWM Pattern

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SET_PWM	0	1	0	0	0	0	0	0	Value							

FUNCTION

Set PWM trigger and thus duty cycle of the PWM pin to an absolute value.

PARAMETER

VALUE PWM trigger is set to this value. The parameter value should be chosen between 0 and the value programmed into the TIMLOAD register.

8.8.4. RESTART Pattern

Table 8.5 RESTART Pattern

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

FUNCTION

Restarts pattern storage register from base index. The position of the base index differs, depending on the timer being used.

PWM0**:** The base index is at position 0 step in pattern storage register

PWM1**:** The base index is at position 32 step in pattern storage register

PWM2**:** The base index is at position 64 step in pattern storage register

8.8.5. NEW Pattern

Table 8.6 NEW Pattern

Pattern	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NEW	1	0	1	0	0	0	0	0	0	0	"x"	New pattern index				

Note**:** "x" means "don't care".

FUNCTION

It is continued pattern generation from pattern index for setting in NEW pattern index.

PARAMETER

New pattern index "New pattern index"**:** defines the new pattern index from which the pattern generation continues. The pattern index relates to the pattern position inside the pattern storage register depending on the timer instance being used.

PWM0 **:** pattern position = New pattern index
PWM1 **:** pattern position = New pattern index + 32
PWM2 **:** pattern position = New pattern index + 64

8.8.6. LOOP Pattern

Table 8.7 LOOP Pattern

Pattern	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOOP	1	0	1	Loop count (1-63)						1	"x"	Next pattern index				

Note: "x" means "don't care".

FUNCTION

Number of set "Loop Count", enforcement of location pattern which is set by "Next pattern index."

PARAMETER

Loop count number of branches to be taken
Next pattern index Address set value for destination to be branched
Address for destination to be branched = Current pattern register executing address - Next pattern index setting value

DESCRIPTION

When "Loop Count" value is 0, the pattern is repeated eternally.

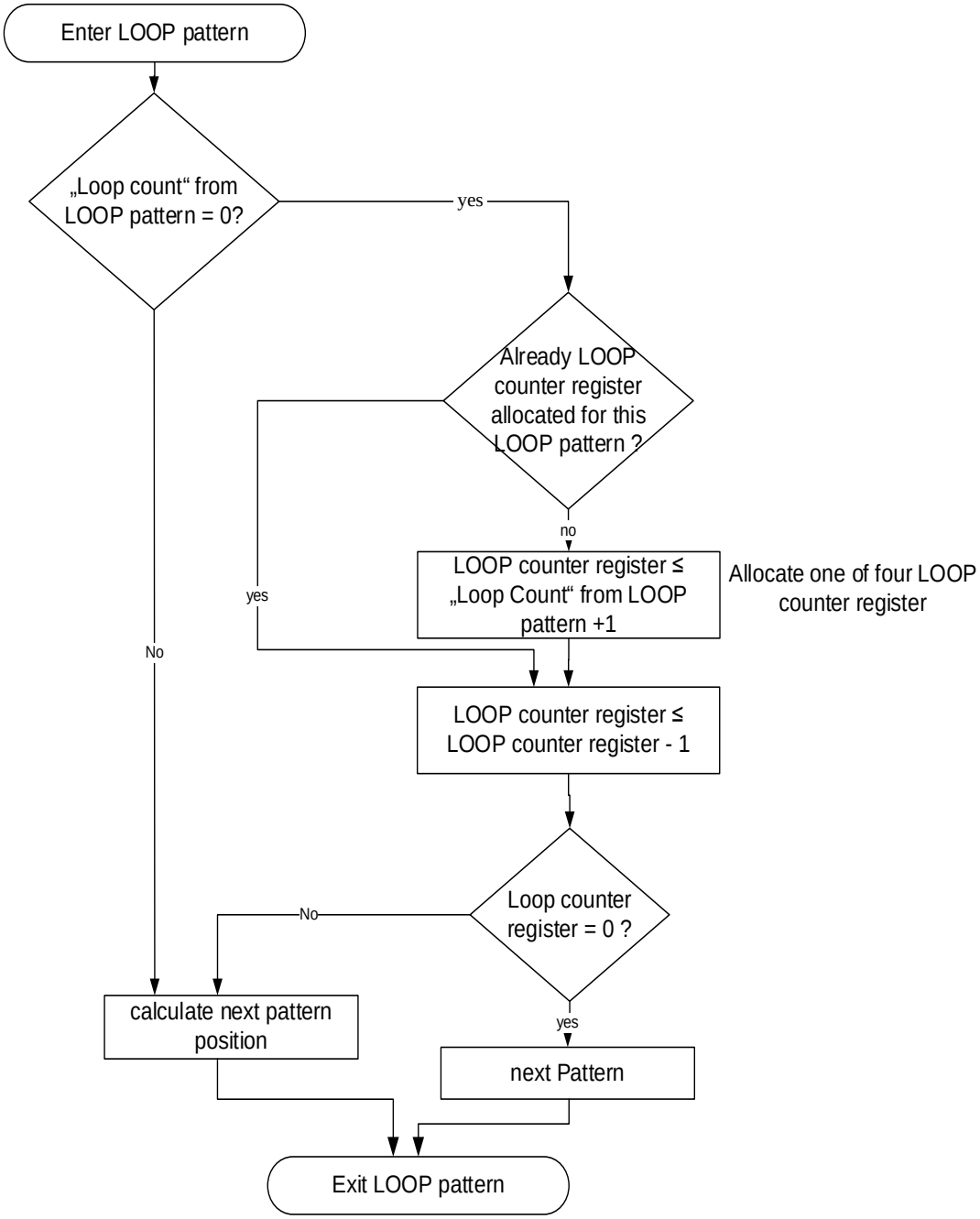


Figure 8.7 LOOP Pattern

8.8.7. END Pattern

Table 8.8 END Pattern

Pattern	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
END	1	1	0	IRQ	RST	"x"										

Note: "x" means "don't care".

FUNCTION

Stop pattern generation in a defined way.

PARAMETER

RST Software Reset
- 0 : No software reset issued.
- 1 : Issues a software reset similar to the effect of writing the PWMSWRES bit. When running in Auto-Sleep mode, the RC oscillator is shut down with END pattern executive.

IRQ Interrupt enable
- 0 : No interrupt issued.
- 1 : Issues an interrupt on IRQN via CDIRQ, when CDIRQ interrupt is enabled.

8.8.8. TRIGGER Pattern

Table 8.9 TRIGGER Pattern

Pattern	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGER	1	1	1	TriggerIn[5:0]						TriggerOut[5:0]						"x"

Note: "x" means "don't care".

FUNCTION

Pauses a pattern generation until a number of trigger events is sensed. Issues output triggers to the other two pattern generators or to an output pin.

PARAMETER

TriggerIn[5:0] Enables input trigger sources
- 0 : Disable input trigger from corresponding channel
- 1 : Enable input trigger from corresponding channel

TriggerOut[5:0] Enables output triggers
- 0 : Disable output trigger from corresponding channel
- 1 : Enable output trigger from corresponding channel

DESCRIPTION

By using the trigger pattern, it is possible to make pattern generation synchronize to external trigger events. "Input trigger channel" controls valid/invalid by setting TriggerIn[5:0] bit (bit12 corresponds to TriggerIn5, and bit7 corresponds to TriggerIn0), When "Input trigger channel" is only all of high, the pattern generation is continued.

Output trigger channel controls valid/invalid by setting for TriggerOut[5:0](bit6 corresponds to TriggerOut5, and bit1 corresponds to TriggerOut0.)

The following figure shows trigger contribution for three timers. GPIOWAKEIN signal is connected with TriggerIn5 for each timer.

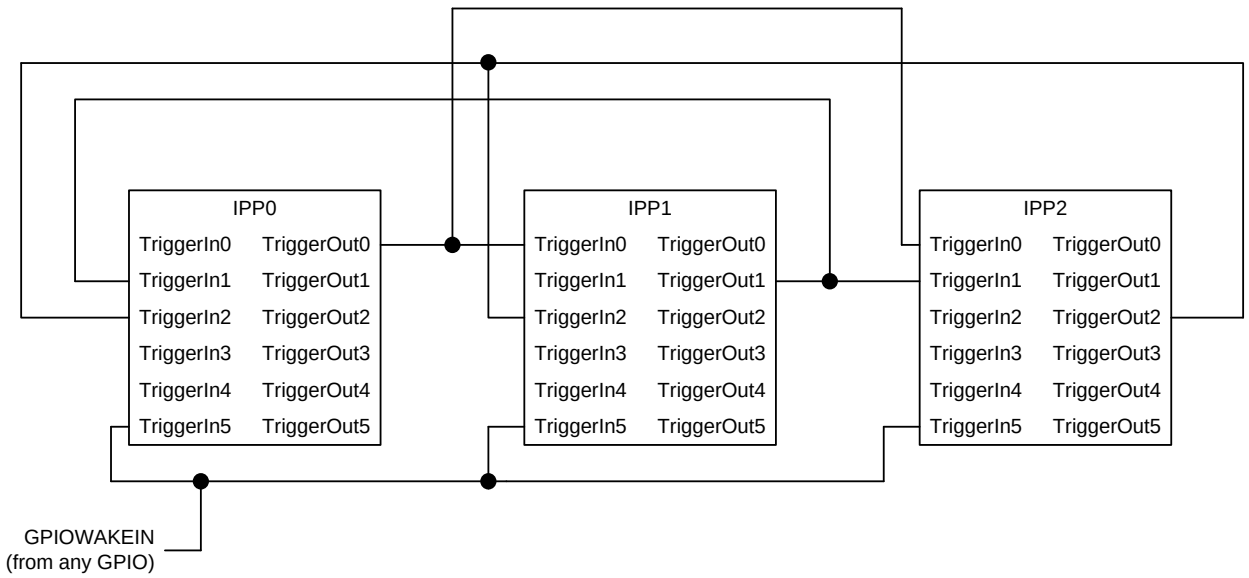


Figure 8.8 Trigger Routing for the three Timers

9. GPIO

The TC35894FG provides maximum 24 general purpose input/output lines. Not depending on multiple configurations for I/O, all of general purpose input lines are connected with internally GPIO module. The GPIO module supports beside the general purpose input/output functionality also the connection of up to 26 direct key inputs.

9.1. GPIO Features

- 24 general purpose input/output ports
- Each port can be configured either as input or output port
- Edge detecting interrupt is supported and selectable between positive, negative or both edges
- Level detecting interrupt is supported and selectable between high level and low level active
- GPIO interrupts can be masked individually
- In case of register write access, bit masking can be supported
- Drives pseudo open drain output buffers with NMOS or PMOS transistor
- By register setting, Pull-Up/Pull-Down resistors can be allotted for each port individually
- Gated clock function in order to reduce dynamic power consumption
- Protects external components by switching all I/Os to input mode after reset
- Supports up to 26 direct keypad connection
- 8 step event buffer for direct key events protecting
- Create interrupt on direct key event generation
- Programmable key de-bouncing set function

9.2. GPIO Operation

All registers mentioned in this chapter are accessible through the I^2C interface.

9.2.1. GPIO DATA register

In GPIO output mode, data that is written into this register is applied to the respective output lines. Each data bit is necessary for setting to 1 with a corresponding mask bit (refer to Figure 9.1). In case of setting 0 at mask bit, writing to data bit is invalid.

A read access to the DATA register returns the vale for each GPIO input line in GPIO input mode. **(Read for GPIODATA register is unsettled, when GPIODIR is in output mode.)** Read access to MASK registers returns zero. The mask bit setting is only applied on write access. Write access to GPIODATA register needs 2byte burst forwarding of DATA byte and MASK byte. In that case, firstly forward DATA byte.

GPIODATA2 register (0xC4)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W R/W	MNEMONIC	MASK23	MASK22	MASK21	MASK20	MASK19	MASK18	MASK17	MASK16
		DATA23	DATA22	DATA21	DATA20	DATA19	DATA18	DATA17	DATA16
	Default	0	0	0	0	0	0	0	0
		X	X	X	X	X	X	X	X
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIODATA1 register (0xC2)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W R/W	MNEMONIC	MASK15	MASK14	MASK13	MASK12	MASK11	MASK10	MASK9	MASK8
		DATA15	DATA14	DATA13	DATA12	DATA11	DATA10	DATA9	DATA8
	Default	0	0	0	0	0	0	0	0
		X	X	X	X	X	X	X	X
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIODATA0 register (0xC0)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W R/W	MNEMONIC	MASK7	MASK6	MASK5	MASK4	MASK3	MASK2	MASK1	MASK0
		DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
	Default	0	0	0	0	0	0	0	0
		X	X	X	X	X	X	X	X
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

MASK23:0 Mask bit for MASK23:0. (**WRITE ONLY**)
0 : Disable MASK23:0 bit setting
1 : Enable MASK23:0 bit setting

DATA23:0 Data23:0 (on pin EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0] when GPIO selected)
0 : Output "0" when corresponding MASK bit is set to "0"
1 : Output "1" when corresponding MASK bit is set to "1"

Note: The default value for the read DATA (input direction) is depending on the signal levels at the pins.

9.2.2. GPIO DIR Registers

The DIR registers controls direction of the GPIO ports. All ports are in input mode after reset in order to protect any external device connected to the GPIO ports.

GPIODIR2 register (0xC8)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIR23	DIR22	DIR21	DIR20	DIR19	DIR18	DIR17	DIR16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIODIR1 register (0xC7)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIR15	DIR14	DIR13	DIR12	DIR11	DIR10	DIR9	DIR8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIODIR0 register (0xC6)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIR7	DIR6	DIR5	DIR4	DIR3	DIR2	DIR1	DIR0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

DIR23:0 Direction bits for DIR23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Input mode
1 : Output mode

9.2.3. GPIO IS Register

IS register controls GPIO interrupt detecting mode. When writing 1 to IS bit, corresponding GPIO input port is changed to level detecting mode. When writing 0 to IS bit, input port for edge detecting is composed.

GPIOIS2 register (0xCB)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IS23	IS22	IS21	IS20	IS19	IS18	IS17	IS16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOIS1 register (0xCA)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IS15	IS14	IS13	IS12	IS11	IS10	IS9	IS8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOIS0 register (0XC9)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

IS23:0 Interrupt detecting bits for IS23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Edge detecting interrupt
1 : Level detecting

9.2.4. GPIO IBE register

IBE register controls GPIO interrupt detecting mode. When IBE bit is set to 1, both edge for corresponding input port is detected as interrupt. When IBE bit is set to 0, interrupt detecting mode depends on IEV register setting.

GPIOIBE2 register (0xCE)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IBE23	IBE22	IBE21	IBE20	IBE19	IBE18	IBE17	IBE16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOIBE1 register (0xCD)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IBE15	IBE14	IBE13	IBE12	IBE11	IBE10	IBE9	IBE8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOIBE0 register (0xCC)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IBE7	IBE6	IBE5	IBE4	IBE3	IBE2	IBE1	IBE0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

IBE23:0 Interrupt detecting mode setting for IBE23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Interrupt generated depending on IEV register setting
1 : Interrupt generated at both edges

9.2.5. GPIO IEV register

IEV register controls GPIO Interrupt detecting mode. When setting 1 to IEV bit, interrupt is generated at detecting rising edge or high level of corresponding GPIO input. When setting 0 to IEV bit, interrupt is generated at detecting falling edge or low level of corresponding GPIO input. The initial value of this register is 0x0000.

GPIOIEV2 register (0xD1)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IEV23	IEV22	IEV21	IEV20	IEV19	IEV18	IEV17	IEV16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOIEV1 register (0xD0)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IEV15	IEV14	IEV13	IEV12	IEV11	IEV10	IEV9	IEV8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOIEV0 register (0xCF)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IEV7	IEV6	IEV5	IEV4	IEV3	IEV2	IEV1	IEV0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

IEV23:0 Interrupt detecting mode setting for IEV23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Interrupt generating at falling edge/low level detecting
1 : Interrupt generating at rising edge/high level detecting

9.2.6. GPIO IE register

IE register is interrupt enable register. When setting 1 to IE bit, interrupt for corresponding GPIO input line and wake-up event generating are enable. When setting 0 to IE bit, the interrupt and wake-up event generating are disable. After reset, all of interrupts are masked.

GPIOIE2 register (0xD4)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IE23	IE22	IE21	IE20	IE19	IE18	IE17	IE16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOIE1 register (0xD3)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IE15	IE14	IE13	IE12	IE11	IE10	IE9	IE8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOIE0 register (0xD2)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

IE23:0 Interrupt enable for IE23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Disable interrupt
1 : Enable interrupt

9.2.7. GPIO RIS register (READ ONLY)

RIS is interrupt status register. When interrupt is generated at corresponding GPIO line, RIS bit is set to 1 not depending on IE register (Interrupt enable register) setting. In case of notified interrupt to external, suitable IE register setting is necessary. RIS register can be cleared by reset or by writing 1 to the IC register.

GPIORIS2 register (0xD8)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	RIS23	RIS22	RIS21	RIS20	RIS19	RIS18	RIS17	RIS16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIORIS1 register (0xD7)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	RIS15	RIS14	RIS13	RIS12	RIS11	RIS10	RIS9	RIS8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIORIS0 register (0xD6)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	RIS7	RIS6	RIS5	RIS4	RIS3	RIS2	RIS1	RIS0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

RIS23:0 Raw interrupt status for RIS23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
Not depend on interrupt enable (IE) register setting.
0 : No interrupt condition at GPIO
1 : Interrupt condition at GPIO

9.2.8. GPIO MIS register

MIS register is masked interrupt status register. MIS bit is set to 1, when corresponding bit of RIS register is set to 1, and interrupt for IE register setting is enable. When RIS bit is 0, it means interrupt is not generated to this line or interrupt is masked by IE register. MIS register can be clear interrupt by writing 1 to IC register.

GPIOMIS2 register (0xDB)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	MIS23	MIS22	MIS21	MIS20	MIS19	MIS18	MIS17	MIS16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOMIS1 register (0xDA)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	MIS15	MIS14	MIS13	MIS12	MIS11	MIS10	MIS9	MIS8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOMIS0 register (0xD9)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	MIS7	MIS6	MIS5	MIS4	MIS3	MIS2	MIS1	MIS0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

MIS23:0 Masked interrupt status for MIS23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : No interrupt condition at GPIO
1 : Interrupt condition at GPIO

9.2.9. GPIO IC register

IC is interrupt clear register. When writing 1 to IC bit, the corresponding interrupt is clear. In case of writing 0, no effect. IC register is for only write.

GPIOIC2 register (0xDE)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	IC23	IC22	IC21	IC20	IC19	IC18	IC17	IC16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOIC1 register (0xDD)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	IC15	IC14	IC13	IC12	IC11	IC10	IC9	IC8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOIC0 register (0xDC)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	IC7	IC6	IC5	IC4	IC3	IC2	IC1	IC0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

IC23:0 Clear interrupt of IC23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : No effect
1 : Clear corresponding interrupt

9.2.10. GPIO OMS register

Open drain mode register uses to enable a pseudo open drain output buffer. When ODE bit is set to 0, ODM bit is ignored and selected a standard CMOS output buffer. When ODE bit is set to 1, a kind of pseudo open drain buffer is selected by ODM bit. Standard CMOS output buffers are active after reset.

GPIOOMS2 register (0xE4, 0xE5)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	ODM23	ODM22	ODM21	ODM20	ODM19	ODM18	ODM17	ODM16
		ODE23	ODE22	ODE21	ODE20	ODE19	ODE18	ODE17	ODE16
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOOMS1 register (0xE2, 0xE3)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	ODM15	ODM14	ODM13	ODM12	ODM11	ODM10	ODM9	ODM8
		ODE15	ODE14	ODE13	ODE12	ODE11	ODE10	ODE9	ODE8
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOOMS0 register (0xE0, 0xE1)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	ODM7	ODM6	ODM5	ODM4	ODM3	ODM2	ODM1	ODM0
		ODE7	ODE6	ODE5	ODE4	ODE3	ODE2	ODE1	ODE0
	Default	0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

ODM23:0 Open Drain Mode Select for ODM23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 : Output can be driven to GND or Hi-Z
1 : Output can be driven to VCC or Hi-Z

ODE23:0 Open Drain Mode Enable for ODE23:0 (EXTIO0, PWM[2:0], KPY[11:0] and KPX[7:0])
0 Full buffer
1 Open drain functionality

9.2.11. GPIO WAKE register

It is possible to use GPIO input as wake-up signal from auto-sleep by GPIO WAKE register setting. And it is also possible to use as trigger signal to Timer module. (Refer to chapter 8.8.8.)

GPIOWAKE2 register (0xEB)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	WAKE23	WAKE22	WAKE21	WAKE20	WAKE19	WAKE18	WAKE17	WAKE16
	Default	0	0	0	0	0	0	0	0
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

GPIOWAKE1 register (0xEA)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	WAKE15	WAKE14	WAKE13	WAKE12	WAKE11	WAKE10	WAKE9	WAKE8
	Default	0	0	0	0	0	0	0	0
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

GPIOWAKE0 register (0xE9)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	WAKE7	WAKE6	WAKE5	WAKE4	WAKE3	WAKE2	WAKE1	WAKE0
	Default	0	0	0	0	0	0	0	0
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

WAKE23:0 Control for wake up signal from Auto-Sleep and for trigger signal to Timer module. Each bit corresponds to ball except WAKE23. WAKE23 is "or" setting for corresponding not only EXTIO0 bit but also DIR24 bit or DIR25 bit.

0 : Wake up signal generating disable
1 : Wake up signal generating enable

9.2.12. Direct Keypad Register

This register activates the direct key functionality within the GPIO module. When the bit is set to "1", the corresponding pin is enable to use as direct key input.

DIRECT3 register (0xEF)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	reserved	reserved	reserved	reserved	reserved	reserved	DIRECT25	DIRECT24
	Default	-	-	-	-	-	-	1	1
	Pin	-	-	-	-	-	-	DIR25	DIR24

DIRECT2 register (0xEE)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIRECT23	DIRECT22	DIRECT21	DIRECT20	DIRECT19	DIRECT18	DIRECT17	DIRECT16
	Default	1	1	1	1	1	1	1	1
	Pin	EXTIO0	PWM2	PWM1	PWM0	KPY11	KPY10	KPY9	KPY8

DIRECT1 register (0xED)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIRECT15	DIRECT14	DIRECT13	DIRECT12	DIRECT11	DIRECT10	DIRECT9	DIRECT8
	Default	1	1	1	1	1	1	1	1
	Pin	KPY7	KPY6	KPY5	KPY4	KPY3	KPY2	KPY1	KPY0

DIRECT0 register (0xEC)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	DIRECT7	DIRECT6	DIRECT5	DIRECT4	DIRECT3	DIRECT2	DIRECT1	DIRECT0
	Default	1	1	1	1	1	1	1	1
	Pin	KPX7	KPX6	KPX5	KPX4	KPX3	KPX2	KPX1	KPX0

DIRECT23:0 Direct keypad bits take priority over anything else. These bits must be cleared to '0' before IOCFG is accessed to set other functions for the pins.
0 : General purpose input/output functionality is active
1 : Direct keypad functionality is active

9.2.13. Direct Key Event Code register

Event code detected in the direct key input is stored into an 8-byte internal event buffer. The Event buffer is organized as a FIFO; the FIFO can be read out at register DEVTCODE.

Read value of this register is 0x3F when FIFO is empty and all of keys are released, or 0x1F when FIFO is empty and any key is pressed.

DEVTCODE register (0xE6)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	Reserved	Reserved	DKEYSTAT	DKEY CODE4	DKEY CODE3	DKEY CODE2	DKEY CODE1	DKEY CODE0
	Default	0	0	1	1	1	1	1	1

DKEYSTAT Indicates, whether keyboard event is a key press or a key release.
0 : Key is pressed
1 : Key is released

DKEYCODE4:0 Direct key event code
0x01: event on KPX0 pin
0x02: event on KPX1 pin
…
0x19: event on DIR24 pin
0x1A: event on DIR25 pin
0x1F: event buffer empty

9.2.14. Input De-Bounce register

The de-bouncing feature is automatically activated when the GPIO input is configured as direct key input. The de-bounce feature is also activated for the pure general purpose inputs when the SYNC bit is set.

DBOUNCE register (0xE8)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	Reserved	Reserved	SYNC	DBOUNCE4	DBOUNCE3	DBOUNCE2	DBOUNCE1	DBOUNCE0
	Default	0	0	0	0	0	1	1	0

DBOUNCE4:0 De-bounce time for the inputs.
De-bounce Time = 128*DBOUNCE[4:0]/fsysclk

SYNC Enables de-bouncing feature on general purpose input lines.
0 : De-bounce function is disable, in case of using general purpose input.
1 De-bounce function is enable, in case of using general purpose input.

9.2.15. Direct Key Raw Interrupt register

DKBDRIS is Raw Interrupt Status register of direct key.

DKBDRIS register (0xF0)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	DRELINT	DREVTINT
	Default	0	0	0	0	0	0	0	0

DRELINT Raw Event Lost Interrupt
This bit is cleared by writing into DEVTIC.
0 : No interrupt
1 : More than 8 direct key events are detected and event buffer overflow generates.

DREVTINT Raw direct key Event Interrupt
This interrupt is automatically clear until the buffer is empty to read DEVTCODE.
0 : No interrupt
1 : At least one direct key press or direct key release is in the event buffer.

9.2.16. Direct Key Mask Interrupt register

DKBDMIS register shows Masked Interrupt Status. When the corresponding bit for DKBDMSK is 0, RAW Interrupt Status (DKBDRIS) value is copied into DKBDMIS. When IRQN is enable to use, any bit for DKBDMIS register is set 1, the external IRQN interrupt generates.

DKBDMIS register (0xF1)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	-	-	-	-	-	-	DMELINT	DMEVTINT
	Default	*	*	*	*	*	*	0	0

DMELINT Masked Event Lost Interrupt
0 : No interrupt
1 : More than 8 direct key events are detected and event buffer overflow generate.

DMEVTINT Masked direct key Event Interrupt
0 : No interrupt
1 : At least one direct key press or direct key release is in the event buffer.

9.2.17. Direct Key Interrupt Clear register (WRITE ONLY)

DKBDIC register controls clear for direct key interrupt.

DKBDIC register (0xF2)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	-	-	-	-	-	-	-	DEVTIC
	Default	*	*	*	*	*	*	*	0

DEVTIC Clear event buffer and corresponding interrupts (DREVTINT and DRELINT) when writing 1 to this register.
0 : No action
1 : Clear event buffer and direct key interrupt

9.2.18. Direct Key Mask register

DKBDMSK register controls direct key interrupt mask.

DKBDMSK register (0xF3)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	-	DMSK ELINT	DMSK EINT
	Default	*	*	*	*	*	*	0	0

DMSKELINT Enable keyboard event lost interrupt
0 : Keyboard event lost interrupt is enabled
1 : Keyboard event lost interrupt is disabled

DMSKEINT Enable keyboard event interrupt
0 : Keyboard event interrupt is enabled
1 : Keyboard event interrupt is disabled

9.3. Direct Keypad Initialization

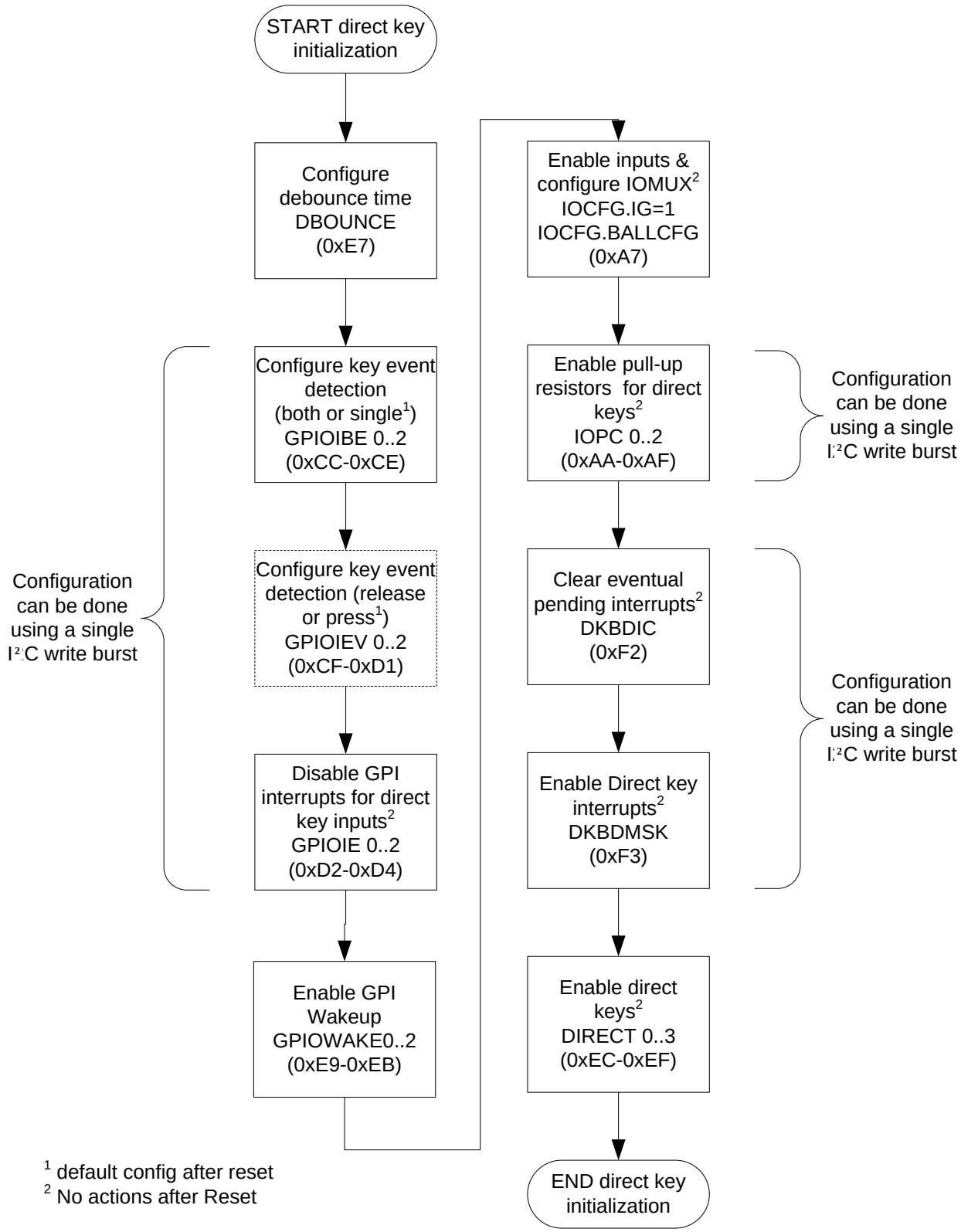


Figure 9.1 Direct Keyboard Initialization

9.4. Function of Interrupt Detection Logic Block

This block enables the GPIO module to generate maskable synchronous and asynchronous interrupts by observing the GPIO ports that are in input mode. The block can be configured to detect either level sensitive or edge sensitive interrupts. In addition, there is also an interrupt mask function by software. The interrupt can be clear via software reset or "IC" register.

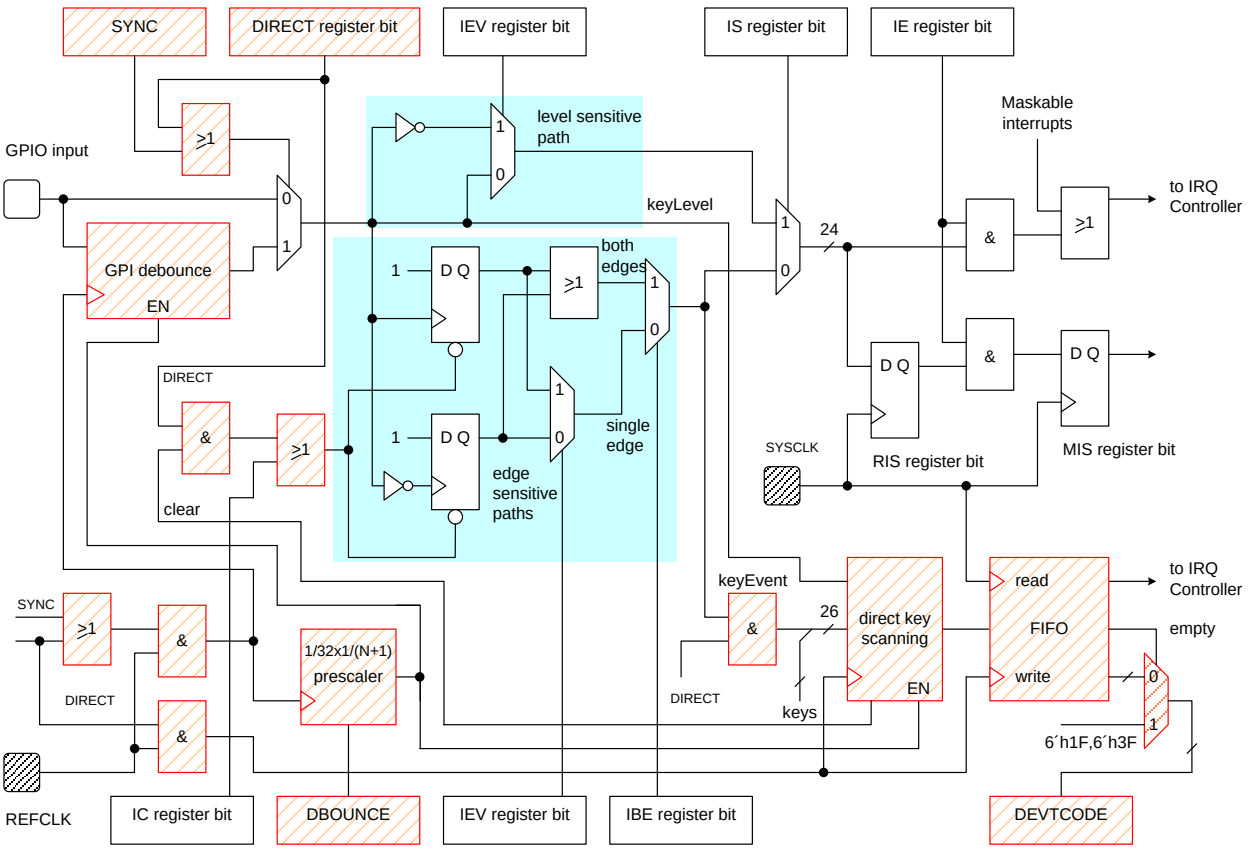


Figure 9.2 Detailed interrupt functionality

9.5. Function of Trigger Logic

The GPIO module generate input signal (refer to Chapter 8.8.8) to TRIGGER pattern and Wake-Up signal. More than one cycle of SYSCLK are necessary for detectable minimum pulse width in TRIGGER signal by Timer Module. WAKE-UP signal generating circuit is shown in Figure 9.3.

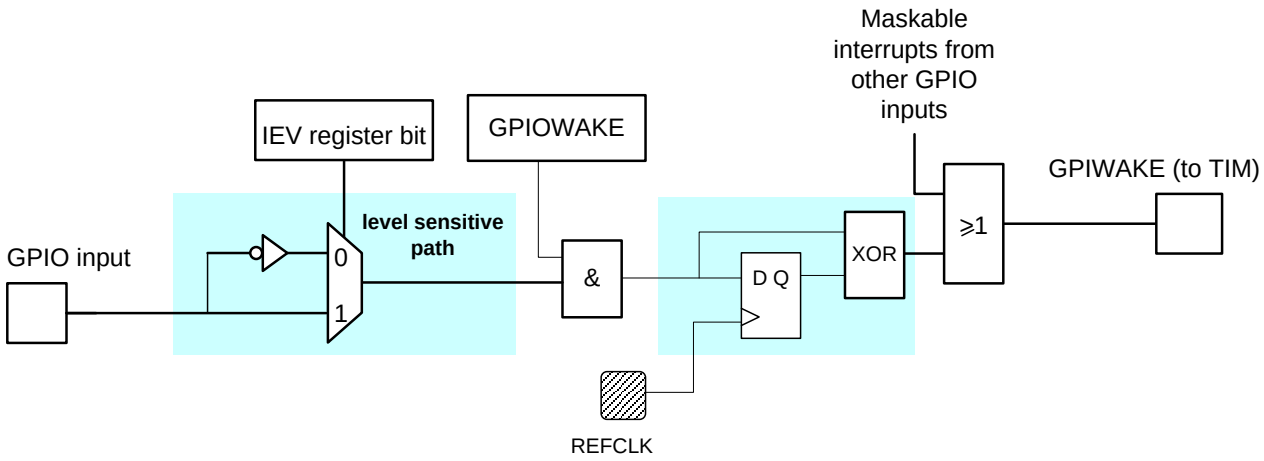


Figure 9.3 Detailed wake up and trigger functionality

9.6. Function of GPIO Control Block and Mode Control

The GPIO control block controls, which GPIO ports should work in input or output mode. In output mode there are three different types of output buffers available. In addition to the normal CMOS buffer output the port can also operate in pseudo open drain mode. The pseudo open drain output is selectable without PMOS or without NMOS transistor by register setting. And this control block can be connected Pull-Up or Pull-Down resistor with each port individually. Due to protecting the external devices, all of GPIOs are input direction after reset.

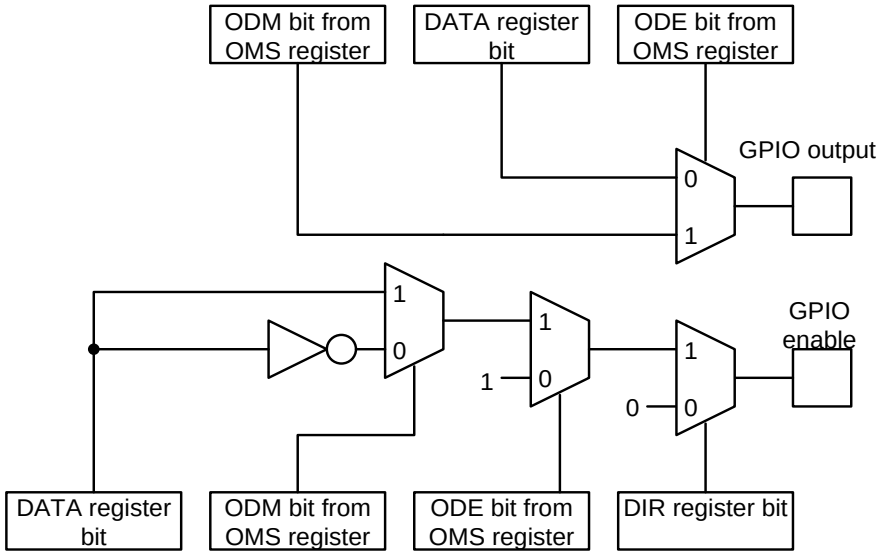


Figure 9.4 Detailed diagram of GPIO output data generation

Note:

Since a software reset resets initializing also the configuration of the Pull-Up/Pull-Down resistors, the system designer has to check that GPIO logic level does not damage the external components.

9.7. GPIO Module Operation

9.7.1. Recommended Configuration Sequence for GPIO functionality

The following initialization sequence is recommended for using the general purpose input/output functionality**:**

- Disable direct key feature by configuring DIRECT register
- Programming DEBOUNCE.SYNC bit, and setting the desired de-bounce value
- Setting edge detecting mode by IBE register
- Setting IEV in case of selecting single edge detection
- Selecting edge・trigger by IS register
- Clear all interrupts by writing 0xFF to the IC register
- Program IE to enable interrupts
- Enable automatic waking up capability by programming WAKE register
- Read or write the DATA register depending on the GPIO configuration

9.7.2. Recommended Configuration Sequence for direct key functionality

The following initialization sequence is recommended for using the direct key functionality**:**

- Program desired de-bounce value in DEBOUNCE register
- Setting Interrupt detecting mode (single (press or release) or both edge (press and release)) by IBE register
- Setting IEV register in case of selecting single edge detecting mode
- Clear all interrupts by writing 0xFF to the IC register (when pending interrupts exists)
- Disable IE register to avoid duplicating with GPIO interrupt
- Enable automatic waking up capability by programming WAKE register
- Enable direct key functionality by configuring DIRECT register

9.7.3. Operation of I/O Lines

The GPIO module controls 24 programmable input/output lines. In GPIO output mode, DATA register value is output value. In case of writing access to DATA register, only the bit without mask setting in upper half word is updated lower half word value. In case of GPIO input mode, READ value for DATA register returns GPIO input value.

Input/output direction for each GPIO line is set by DIR0, DIR1, and DIR2 registers.

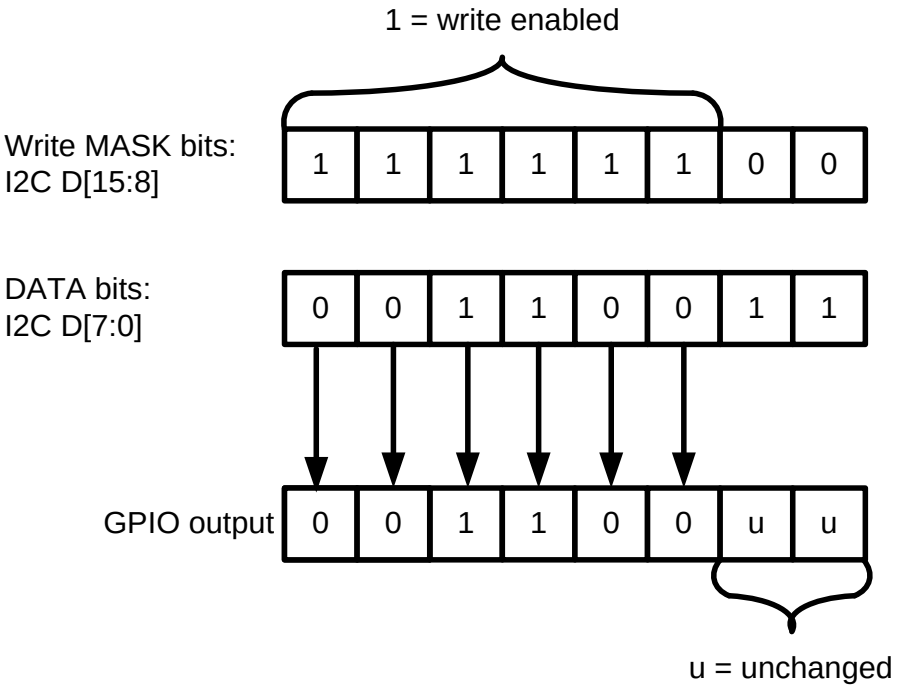


Figure 9.5 Bit masking mechanism for write access to GPIO outputs

9.7.4. Interrupt Operation

The interrupt operating of the GPIO module is programmable by register setting. The interrupt detecting is enable at input signal level or edge by this register setting. Edge is selectable from rising edge, falling edge, or both edges. Level is selectable low level or high level. All of interrupts are enable masking. If at least one unmasked interrupt is active, the GPIO module asserts wake up signal and interrupt signal. For edge detecting interrupts, the software needs to clear the interrupt before starting interrupt detecting.

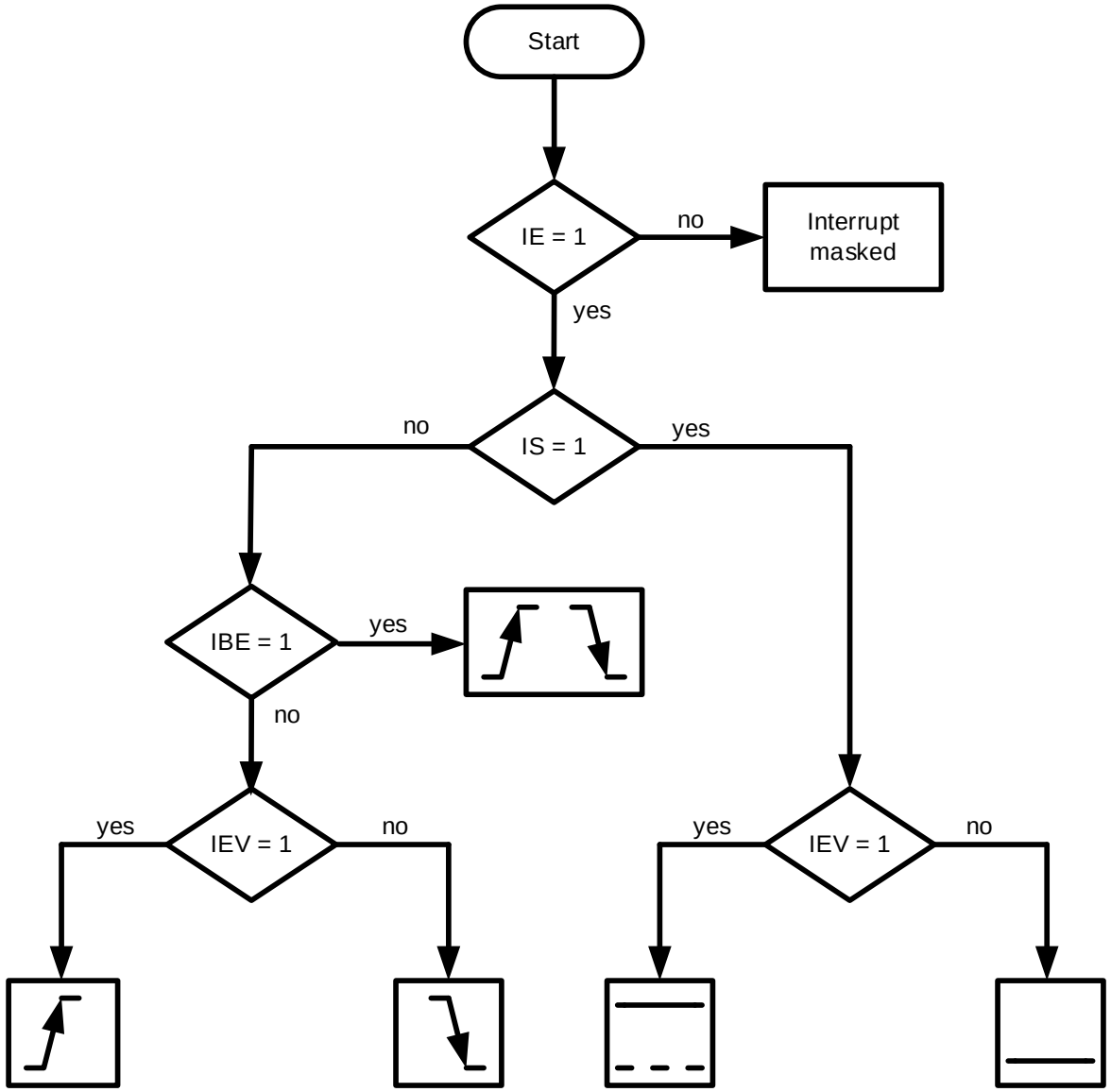


Figure 9.6 Interrupt sensitivity configuration flow

9.7.5. GPIO Mode Control

GPIO ports can operate in some modes by register setting. Output buffer is configurable as standard CMOS output or pseudo open drain output buffer. Figure 9.7, Table 9.1 and Table 9.2 show the relation between GPIOIOMS register setting and output value.

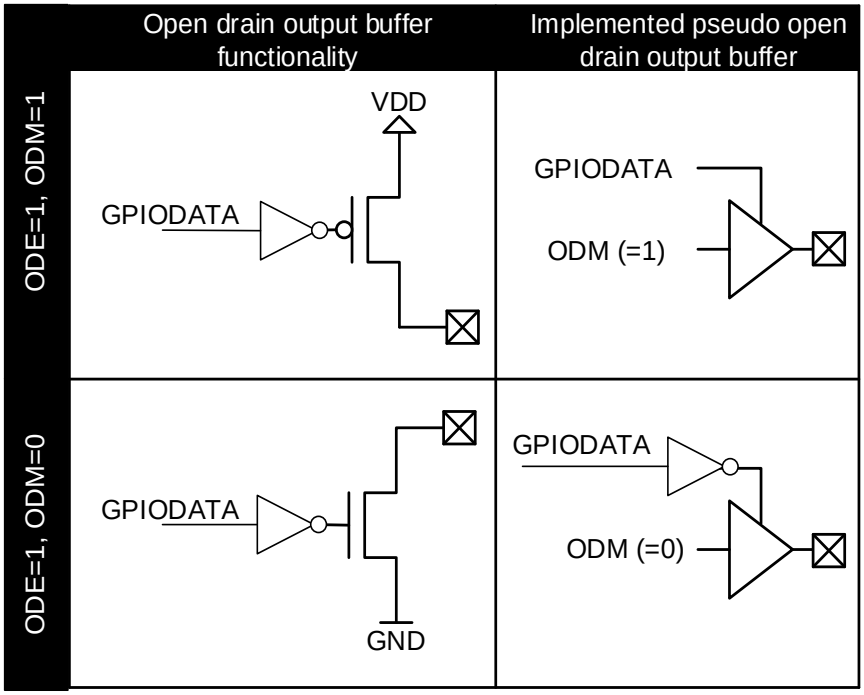


Figure 9.7 Genuine open drain vs. Implemented pseudo-open drain

Table 9.1 Pseudo open drain operation without NMOS resistor, ODM bit = 1

GPIODATA	GPIO port
0	Z
1	1

Table 9.2 Pseudo open drain operation without PMOS resistor, ODM bit = 0

GPIODATA	GPIO port
0	0
1	Z

10. KBD (Keyboard)

The Keypad module is an interface to the externally connected keypad (keyboard matrix or dedicated keys). These are the features of the keyboard controller**:**

- Keyboard matrix of up to 8 × 12 keys plus 8 special function keys
- Support for key code generation of up to 16 so called "dedicated" keys
- Generation of up to four key scan codes indicating which keys are currently pressed
- Event buffer for last 8 key events
- Key release interrupt generation to host
- Built-in key de-bouncing function, and timing adjustable by register setting
- Enable composing various keyboard layout by key composing option
- Enable read out key code easily by event buffer for FIFO composition.

10.1. Keyboard Layout

The layout of the external keyboard can be composed flexibly. It is possible to compose minimum 2 rows by 2 columns and maximum 8 rows by 12 columns by register setting.

The matrix row is connected with KPX[7**:**0], and the matrix column is connected with KPY[11**:**0]. KPX and KPY lines are multi-function with GPIO lines.

Unused KPX and KPY lines as Matrix key are enable for using as dedicated keys. The dedicated keys are de-bounced in the same as matrix keys.

The assignment for the dedicated keys is made by following priority ordering.

1. Assign from smaller row for row index in one matrix unused row.
2. Assign from smaller column for column index in two matrix unused column.

The line neither keyboard matrix nor dedicated key is used for GPIO line, is standard GPIO line.

Figure 10.1 shows an example for a keyboard layout using dedicated keys. (Dedicated key for 5 rows by 6 columns matrix and 4 dedicated keys) As above rule, the dedicated keys are connected with KPX[7**:**5] and KPY6. And two "special function" keys are connected with row KPX0 line and column KPY1 line. Pull-up setting for each input line is operated by IOPC0, IOPC1, and IOPC2 registers.

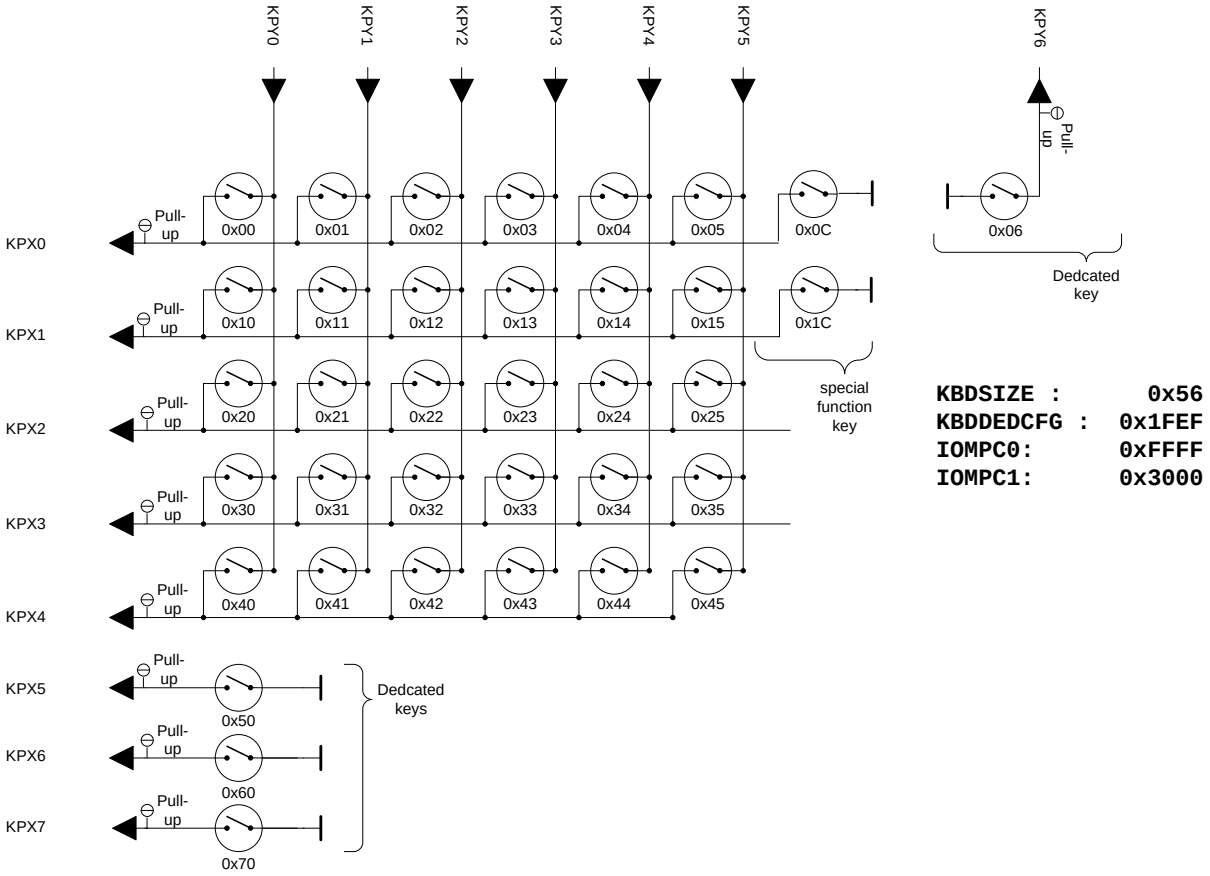


Figure 10.1 Keycode layout example

10.2. Keyboard Scanning

Due to event detecting the keyboard matrix, dedicated and special function keys, the keyboard scan is executed. All of columns for the key matrix are driven 0, all of the row inputs are pulled-up. This situation is called keyboard IDLE state. When any key of the matrix is pressed, the corresponding matrix row input is pulled-down.

When any dedicated key is pressed, the corresponding dedicated key input is pulled-down. In both cases, the Keyboard IDLE state is cancelled and a scanning process is started.

The scanning process specified column and row index of the pressed key. Firstly sampling row input in driving state for high impedance all of columns, "special function" key is detected. When the "special function" key is pressed, the corresponding row is permanently connected with ground. And matrix scan process cannot be detected the standard matrix key press in the specified row. Therefore, the scanning process skips left press "special function" key in key scanning for the same row those keys. And in case of "special function" key, the column index is always 12.

A "special function" keys are scanned, and key matrix are scanned. It is started from KPY[0], each column is continuously connected with ground by one SYSCLK cycle, and released to high impedance again. At each cycle, the matrix row input is sampled and specified a pressed key row/column index.

The pressed row/column index is detected when the row input is row driven. The column index is currently column within "L" driving. The row index is a row to detect "L" input.

It is not necessary to scan dedicated keys. In case of generating key pressing, the corresponding input line is "L", and detected pressing. The column index for dedicated key connected with KPX input is zero. The row index for dedicated key connected with KPY input is zero.

10.3. Keyboard De-bouncing

After key event detecting, timing until key scan starting is adjustable by KBDSETTLE register. Therefore, it is possible to delay key scan starting until the physical contact is stable.

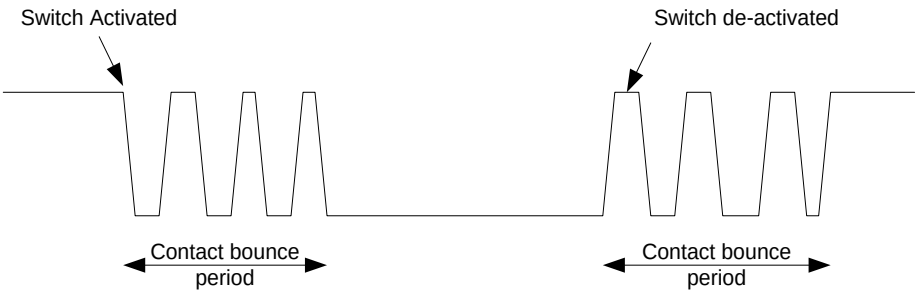


Figure 10.2 Keyboard de-bouncing

10.4. Detection of Multiple Key-presses

The keyboard interface is able to detect multiple key-presses.

- Two or more keys pressed within keyboard matrix
- Four dedicated keys pressed

The detection of more than 2 multiple key-pressed in the keyboard matrix is enable when only "ghost key detection" does not occur. The example for ghost keys is shown at Figure 10.3. When three keys([KPX0, KPY0], [KPX0, KPY2], [KPX3, KPY2]) for composing rectangular triangle are pressed, [KPX3, KPY0] in position for composing rectangle with these three keys is detected as pressed.

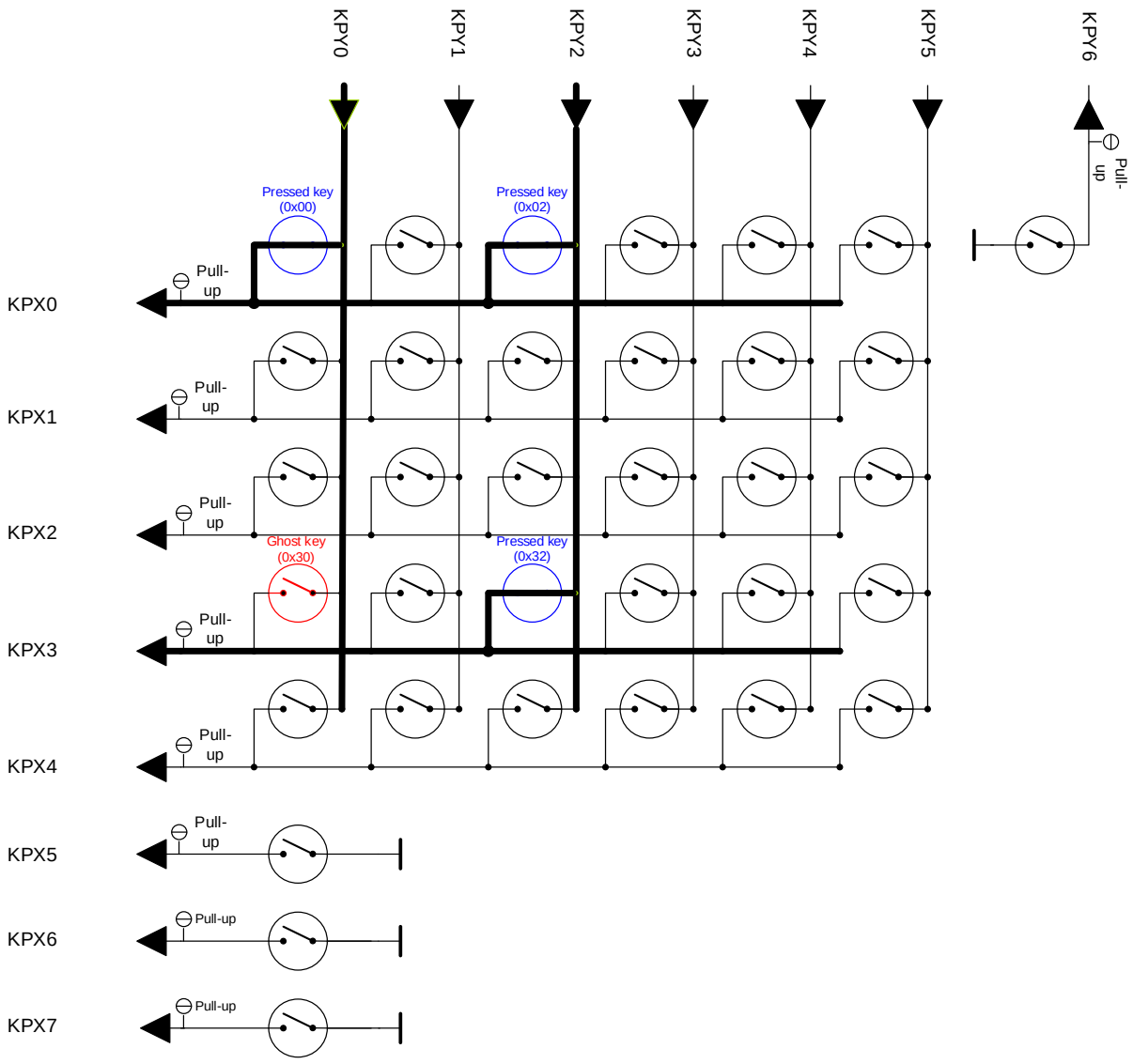


Figure 10.3 Ghost key generation

10.5. Software Interface for Keypad

The registers KBDSETTLE, KBDBOUNCE, KBDSIZE and KBDDEDCFG are used for keyboard module setup. In addition, the registers IOPC setting is necessary for pull-ups on all keyboard inputs. Due to reading key code, there are two methods.

a. Read from event FIFO (EVTCODE register) (recommended)

b. Read from Keycode register (For KBDCODE3 from KBDCODE0)

10.5.1. Setup of Initial Wait Period

When the event is detected in the key matrix, keyboard scan is started. This register defines a wait time until first key scan is started after detecting the event.

KBDSETTLE register (0x01)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	WAIT07	WAIT06	WAIT05	WAIT04	WAIT03	WAIT02	WAIT01	WAIT00
	Default	1	0	1	0	0	0	1	1

WAIT0_7:0 Initial wait time (Twait) until the key is stable, before key scan is started. Twait is calculated by the following equation.

Twait = 4*N/fsysclk

0xFF : N = 255 (fsysclk = 64 kHz, Twait = 15.9 ms)
0xA3 : **N = 163 (**fsysclk = 64 kHz, Twait = **9.68 ms)**
0x7F : N = 127 (fsysclk = 64 kHz, Twait = 7.8 ms)
0x52 : N = 82 (fsysclk = 64 kHz, Twait = 5.0 ms)
0x40 : N = 64 (fsysclk = 64 kHz, Twait = 3.9 ms)
0x00 : N = 0 (fsysclk = 64 kHz, Twait = 0 ms)

10.5.2. Setup of De-bouncing

The KBDBOUNCE register configures the de-bounce time. After an initial keyboard scan, the keyboard is subsequently scanned in intervals defined by register KBDBOUNCE. Keyboard scanning is stopped when the same two codes are detected consecutive.

KBDBOUNCE register (0x02)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	BOUNCE TIM7	BOUNCE TIM6	BOUNCE TIM5	BOUNCE TIM4	BOUNCE TIM3	BOUNCE TIM2	BOUNCE TIM1	BOUNCE TIM0
	Default	1	0	1	0	0	0	1	1

BOUNCETIM 7:0 Setting for keyboard scan interval (Tdebounce). Tdebounce is calculated by the following equation.

Tdebounce = 4*N/fsysclk

0xFF : N = 255 (fsysclk = 64 kHz, Tdebounce = 15.9 ms)
0xA3 : **N = 163 (**fsysclk = 64 kHz, Tdebounce = **9.68 ms)**
0x7F : N = 127 (fsysclk = 64 kHz, Tdebounce = 7.8 ms)
0x52 : N = 82 (fsysclk = 64 kHz, Tdebounce = 5.0 ms)
0x40 : N = 64 (fsysclk = 64 kHz, Tdebounce = 3.9 ms)
0x00 : N = 0 (fsysclk = 64 kHz, Tdebounce = 0 ms)

10.5.3. Keyboard Matrix Setup

Keyboard matrix layout is set in KBDSIZE register. For the example layout indicating in Figure 10.1, this register setting value is 0x56.

KBDSIZE register (0x03)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	ROWSIZE3	ROWSIZE2	ROWSIZE1	ROWSIZE0	COLSIZE3	COLSIZE2	COLSIZE1	COLSIZE0
	Default	0	0	0	0	0	0	0	0

ROWSIZE3:0 Number of rows in the keyboard matrix, between 2 and 8.
0x0 : Keyboard matrix is not used.
0x1 : Inhibition
0x2
| Number of rows
0x8

COLSIZE3:0 Number of columns in the keyboard matrix, between 2 and 12.
0x0 : Keyboard matrix is not used.
0x1 : Inhibition
0x2
| Number of columns
0xC

10.5.4. Dedicated Key Setup

The KBDDEDCFG register configures the use of dedicated keys. GPIO can also be used as output, but dedicated keys are enable as only input. And dedicated keys are different from GPIO in terms of that dedicated key are de-bounced and generates key code. For the layout example of Figure 10.1, this register is set 0x1FEF.

KBDDEDCFG register (0x04)

R/W	ITEM	BIT15	BIT14	BIT13	BIT12	BIT11	BIT10	BIT9	BIT8
		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	ROW7	ROW6	ROW5	ROW4	ROW3	ROW2	COL11	COL10
		COL9	COL8	COL7	COL6	COL5	COL4	COL3	COL2
	Default	1	1	1	1	1	1	1	1
		1	1	1	1	1	1	1	1

ROW7:2 Each bit in ROW [7:2] corresponds to pin KPX7...KPX2.
0 : Dedicated key
1 : No dedicated key (standard GPIO or keyboard matrix)

COL11:2 Each bit in COL [11:2] corresponds to pin KPY11...KPY2.
0 : Dedicated key
1 : No dedicated key (standard GPIO or keyboard matrix)

10.5.5. KBDCODE and EVTCODE register

The Key code detected by the keyboard scan can be read from the registers KBDCODE0 to KBDCODE3 or by the EVTCODE register.

KBDCODE 0 to 3 register (0x0B-0x0E)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	MULTIKEY	KEYROW2	KEYROW1	KEYROW0	KEYCOL3	KEYCOL2	KEYCOL1	KEYCOL0
	Default	0	1	1	1	1	1	1	1

MULTIKEY Multiple key press. Another key code is available in KBDCODE (x+1) register.
0 : No key code in KBDCODE (x+1)
1 : Key code in KBDCODE (x+1)
KEYROW2:0 Row index (0,7)of key pressed
KEYCOL3:0 Column index (0..11, 12 for special function key index) of key pressed

When first key press is detected, key code is stored in KBDCODE0, and MULTIKEY bit is set to 0. When generated situation that two keys are pressed simultaneously by second key pressed, new created key code is stored in KBDCODE1 and set 1 to MULTIKEY for KBDCODE0. In case of no key pressed, read value for KBDCODE0 register is 0x7F.When all of KBDCODE registers are read or write access is done to interrupt clear register KBDIC, keyboard scanning interrupt RSINT is clear.

EVTCODE register (0x10)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	RELEASE	KEYROW2	KEYROW1	KEYROW0	KEYCOL3	KEYCOL2	KEYCOL1	KEYCOL0
	Default	1	1	1	1	1	1	1	1

RELEASE Indicates, whether keyboard event is a key press or a key release.
0 : Key pressed
1 : Key released
KEYROW2:0 Row index of key that is pressed (0…7).
KEYCOL3:0 Column index of key that is pressed (0…11 and 12 for special function key).

When event is generated in keyboard, the key code is stored into an 8-byte deep event buffer. The Event buffer is composed as FIFO, and can be read out from register EVTCODE. When FIFO is read and empty, REVTINT is cleared soon. In this time, read value for EVTCODE is 0x7F.

10.5.6. KBD Raw Interrupt register

In the KBDRIS register, unmasked keyboard interrupt status (Raw Interrupt Status) is stored.

KBDRIS register (0x06)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	-	-	-	-	RELINT	REVTINT	PKLINT	RSINT
	Default	*	*	*	*	0	0	0	0

RELINT Raw Event Lost Interrupt
This bit is cleared by writing into EVTIC.
0 : No interrupt
1 : More than 8 keyboard events have been detected and caused the event buffer to overflow.

REVTINT Raw keyboard Event Interrupt
Reading from EVTCODE until the buffer is empty will automatically clear this interrupt.
0 : No interrupt
1 : At least one key press or key release is in the keyboard event buffer.

RKLINT Raw Key Lost interrupt.
The meaning of this interrupt bit changes depending on configuration of the KBDMFS (Keyboard Modified Feature Set) register.
0 : No interrupt
1 : If KBDMFS is set to 0:
When RSINT has not been cleared upon detection of a new key press or key release, or when more than 4 keys are pressed simultaneously.
If KBDMFS is set to 1 (default):
Indicates that more than 4 keys are pressed simultaneously.

RSINT Raw Scan Interrupt
0 : No interrupt
1 : Interrupt generated after keyboard scan. (Detecting key pressed or key release)

10.5.7. KBD Mask Interrupt register

In the KBDMIS register, the status of the masked interrupts is given. The raw interrupt status from register (KBDRIS) is copied into KBDMIS, if the corresponding bit in KBDMSK is 0. When IRQN is enable, IRQN interrupt is generated if setting 1 into KBDMIS register any bit.

KBDMIS register (0x07)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	-	-	-	-	MELINT	MEVTINT	MKLINT	MSINT
	Default	*	*	*	*	0	0	0	0

MELINT Masked Event Lost Interrupt
0 : No interrupt
1 : More than 8 keyboard events have been detected and caused the event buffer to overflow.

MEVTINT Masked keyboard Event Interrupt
0 : No interrupt
1 : At least one key press or key release is in the keyboard event buffer.

MKLINT Masked Key Lost Interrupt
0 : No interrupt
1 : Masked Key lost interrupt

MSINT Masked Scan Interrupt
0 : No interrupt
1 : Masked scan interrupt. Key event (press/release) is detected by key scanning.

10.5.8. KBD Interrupt clear register (WRITE ONLY)

In the KBDIC register, an active keyboard interrupt is cleared.

KBDIC register (0x08)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
W	MNEMONIC	SFOFF	-	-	-	-	-	EVTIC	KBDIC
	Default	0	*	*	*	*	*	0	0

SFOFF Switches off scanning of special function keys.
0 : In key scanning, subject to scanning for special function keys and dedicated keys.
1 : In key scanning, no subject to scanning for special function keys and dedicated keys.

EVTIC Clear event buffer and corresponding interrupts REVTINT and RELINT.
0 : No action
1 : Clear event buffer and corresponding interrupts REVTINT and RELINT

KBDIC Clear RSINT and RKLINT interrupts bits.
0 : No action
1 : Clear RSINT and RKLINT interrupts bits

10.5.9. KBD Mask register

The KBDMSK register is set masking for a keyboard interrupt. In case of generating interrupt, there are several methods for the interrupt handler process flow. When interrupt handler reads key code from EVTCODE, the bit 0 and bit 1 have to be set into "1." (Recommend) On the other hand, in case of reading key code from KBDCODE0 to KBDCODE3, the bit 3 and bit 2 have to be set into "1."

KBDMSK register (0x09)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	MSKELNT	MSKEINT	MSKKLI	MSKSINT
	Default	*	*	*	*	1	1	0	0

MSKELNT Enable keyboard event lost interrupt (RELINT)
0 : Keyboard event lost interrupt is enabled
1 : Keyboard event lost interrupt is disabled

MSKEINT Enable keyboard event interrupt (REVTINT)
0 : Keyboard event interrupt is enabled
1 : Keyboard event interrupt is disabled

MSKKLI Enable keycode lost interrupt (RKLINT)
0 : Keycode lost interrupt is enabled
1 : Keycode lost interrupt is disabled

MSKSINT Enable keyboard status interrupt (RSINT)
0 : Keyboard status interrupt is enabled
1 : Keyboard status interrupt is disabled

10.5.10. KBD feature correcting register (WRITE ONLY)

It is recommended to always set the KBDMFS.MFSEN bit to 1 (enable). The timing restriction is eased for keyboard handling by this setting.

KBDMFS register (0x8F)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R/W	MNEMONIC	-	-	-	-	-	-	-	MFSEN
	Default	*	*	*	*	*	*	*	1

MFSEN KBD function correcting enable
0 : KBD function correcting disabled
1 : KBD function correcting enabled (**recommend**)

10.6. Keyboard Interface Operation

10.6.1. Single Key-press

The following figure indicates keyboard scanning operation.

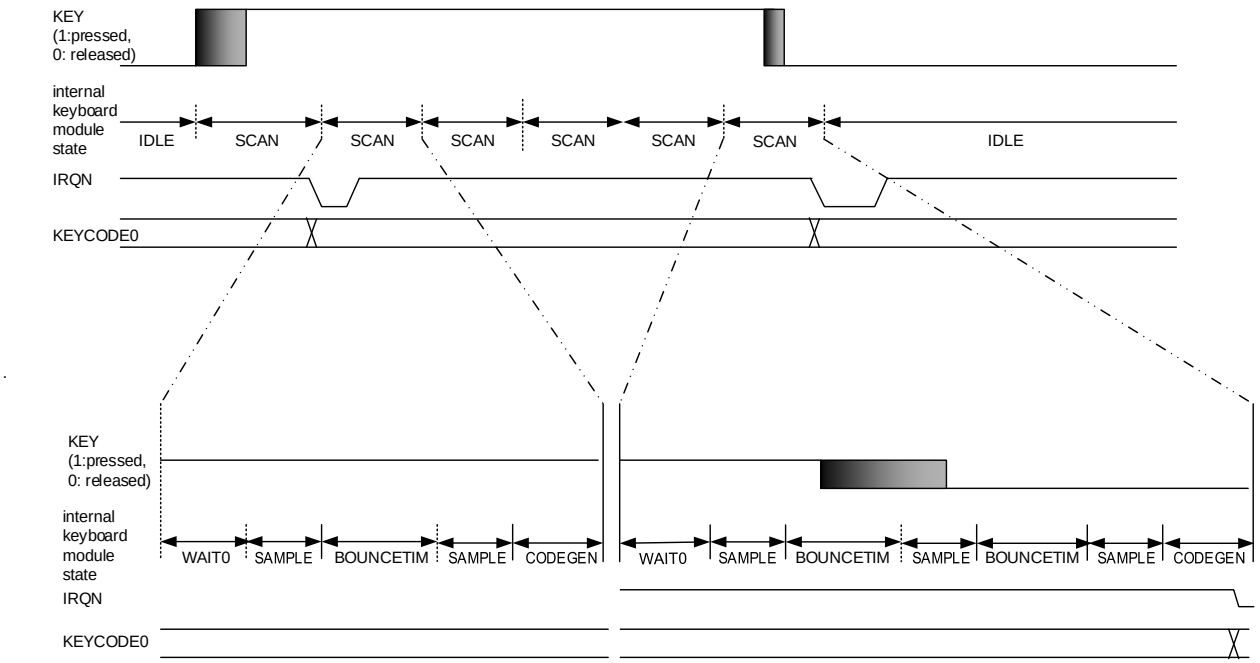


Figure 10.4 Key scan with a single key press

When no key is pressed the keyboard scanner is in the IDLE state. In IDLE state, all keyboard rows are at input, and connected with internal pull-up resistor by IOPC register setting. All keyboard columns are at output and output a low level.

When key event is detected, the scanner starts scanning process after waiting WAIT0 period for de-bounce, and creates a temporary key code. After that, second scanning is executed after BOUNCETIM period, and new key code is created. When two series of the same key code is detected by key scanning process, this is stored on KBDCODE register. The scanning process is executed repetition as far as the key is pressed, and interrupt is created in case of changing into keyboard status.

In example Figure 10.4, three times of sample processes are executed in key release context. Different key codes are detected between first and second sample process, the same key codes are detected between second and third sample process, and the scanning process is finished after that.

When new event is generated, key code is stocked on event FIFO and KBDIRQ is active. The host controller is cleared interrupt by reading EVTCODE register. When keyboard interrupt is generated, the interrupt handler is read key code from event buffer (EVTCODE). The event buffer is eased largely the overflow timing restriction for storable maximum eight events.

Minimum time for key scanning is calculated in the following equation for setting WAIT0 and BOUBCETIM;

$$\textbf{MinSCAN (period)} = 2 \times \textbf{SAMPLE} + \textbf{WAIT0} + \textbf{BOUNCETIM} + \textbf{CODEGEN}$$

One sample period is depended on keyboard composition and pressed key numbers. Sample period minimum value is zero (in case of no key press), and sample period maximum value is ROWSIZE × COLSIZE × SYSCLK cycle period. When four keys are pressed within matrix, the sample period is 4 × COLSIZE.

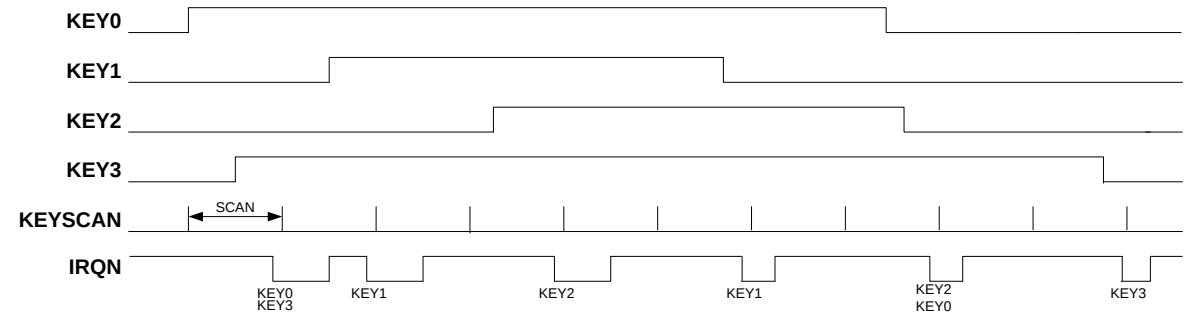
CODEGEN period is calculated in the following equation;

CODEGEN period = $8T_{SYSCLK}$ × (key pressed numbers in matrix) + $16T_{SYSCLK}$ × (only using dedicated key)

CODEGEN minimum period is zero. The maximum period is $3 \times 8 + 16 = 40T_{SYSCLK}$ in case of detected three key presses and one dedicated key presses within matrix.

10.6.2. Multiple Key-press

The following diagram shows an example of multiple key-presses**:**



KEYCODE0	$7F	KEY0	KEY0	KEY0	KEY0	KEY3	$7F
KEYCODE1	$7F	KEY3	KEY1	KEY1	KEY2	$7F	$7F
KEYCODE2	$7F	$7F	KEY3	KEY2	KEY3	$7F	$7F
KEYCODE3	$7F	$7F	$7F	KEY3	$7F	$7F	$7F

EVTCODE						
empty	P, KEY3	P, KEY1	P, KPY2	R, KEY1	R, KEY0	R, KEY3
empty	P, KEY0	P, KEY3	P, KEY1	P, KPY2	R, KEY2	R, KEY0
empty	empty	P, KEY0	P, KEY3	P, KEY1	R, KEY1	R, KEY2
empty	empty	empty	P, KEY0	P, KEY3	P, KPY2	R, KEY1
empty	empty	empty	empty	P, KEY0	P, KEY1	P, KPY2
empty	empty	empty	empty	empty	P, KEY3	P, KEY1
empty	empty	empty	empty	empty	P, KEY0	P, KEY3
empty	empty	empty	empty	empty	empty	P, KEY0

Figure 10.5 Multiple Key-press

After keyboard initializing, KBDCODE register value is 0x7F in case of no key press. Keyboard interface is idle status and waits key event. Firstly KEY0 is pressed, and secondly KEY3 is pressed. A delay between KEY0 press and KEY3 press is shorter than key scanning period, so detected as simultaneous key pressing and interrupt is created. After that, the host reads EVTCODE register and clears interrupt.

After that, when KEY1 is pressed for pressing KEY0 and KEY3, interrupt is created after key scanning, and three key codes (KEY0, KEY1, and KEY3 pressing) are stocked on EVTCODE. The host is detected new KEY1 pressing by reading EVTCODE register, and the interrupt is cleared.

After that, KEY2 is pressed and the interrupt is created in the same terminal of SCAN period. The host reads EVTCODE register, and detects KEY2 for new pressed.

Secondly, KEY1 is released and new interrupt is set in the terminal scanning period. The host recognizes KEY1 release by reading EVTCODE register and the interrupt is cleared simultaneously.

The same processes are executed for KEY0, KEY2, and KEY3 release, and when all of keys are released, the keyboard is returned to IDLE status.

10.6.3. Keyboard Initialization Flow

The following flowchart indicates the necessary steps for initializing the keyboard**:**

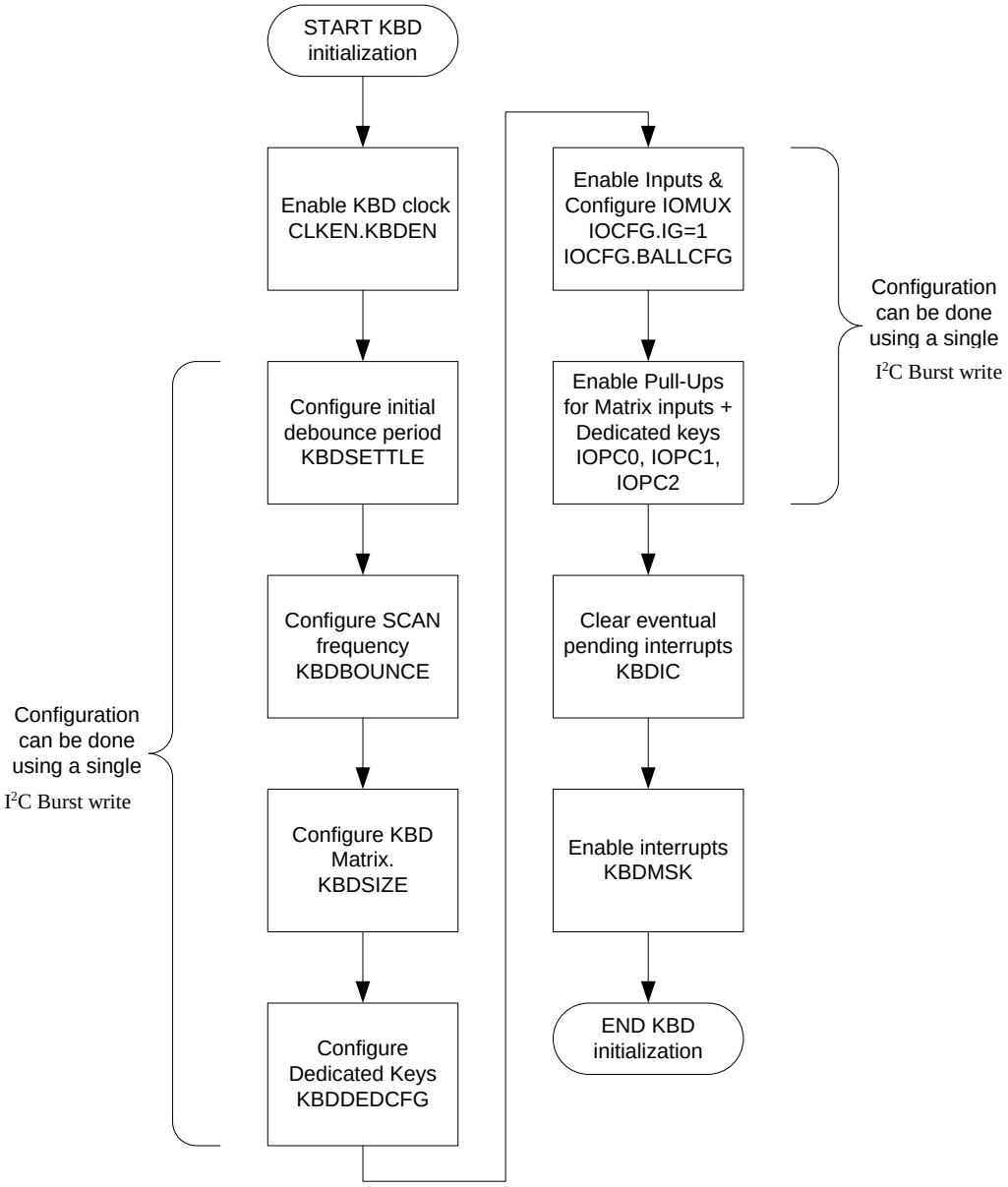


Figure 10.6 Keyboard initialization flow

Firstly, it is necessary to enable the keyboard interface clock. Secondly, set KBDSETTLE and KBDBOUNCE values considering of keyboard mechanical characteristics. The keyboard matrix is composed by KBDSIZE and KBDDEDCFG register programming.

KBDSETTLE, KBDBOUNCE, KBDSIZE, and KBDDEDCFG are set of using single I^2C burst write. In case of using TC35894FG keyboard interface, it is necessary to set appropriately IOCFG register. And regarding KPX input and dedicated key input, it is necessary to set internal pull-up by IOPC0, IOPC1, and IOPC2 register.

It is possible to set of using single I^2C burst write for IOCFG, IOPC0, IOPC1, and IOPC2 register. After changing I/O multiple setting for these registers, it is necessary to clear the pending interrupt. Before enable keyboard interrupt for this, write KBDIC register surely and clear the interrupt.

10.6.4. Keyboard Interrupts Handling

A process flow of keyboard interrupt is indicated as following; (Process flow of using EVTCODE register)

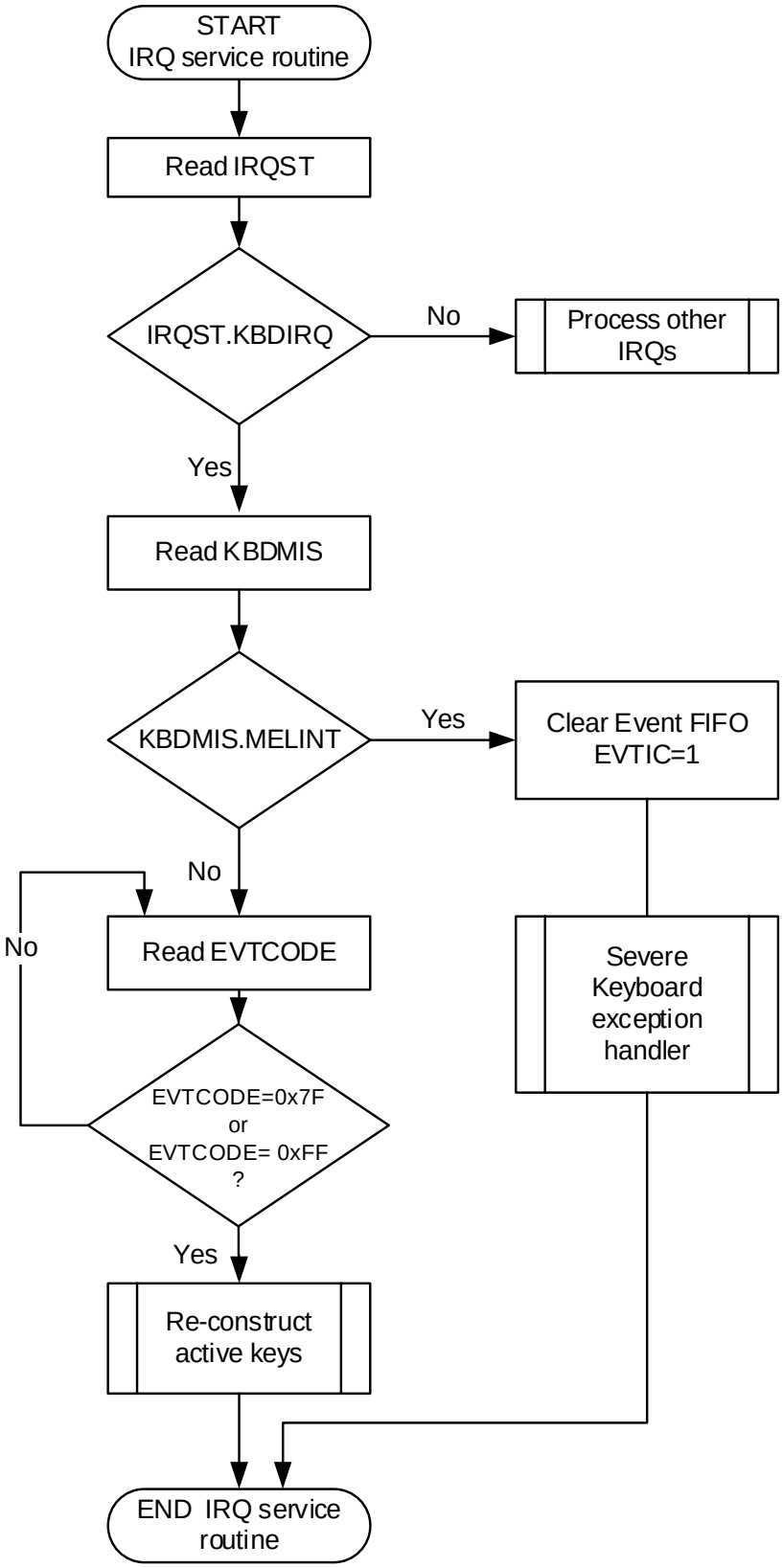


Figure 10.7 Interrupt handler for Event FIFO

10.6.5. Using GPI together with Keyboard

Keyboard input signal is also connected with internal GPIO module.

In case of using internal RC oscillator, due to SYSCLK stop state in SLEEP mode, keyboard scanning is not executed. However, key pressed detecting is enable by valid for GPIO wake-up function (GPIOWAKE register).

When key is pressed in SLEEP mode state, GPIO module executes wake-up for device. By this, internal RC oscillator is started immediately, key press and release are detected exactly.

11. IRQ (Interrupt module)

Each interrupt factor is collected inside the interrupt controller and connected with host controller via IRQN pin.
IRQN pin is a fail-safe open drain and enable as Wired-OR logic.
The followings are TC35894FG interrupt sources.

- GPIO input trigger (Logic or Edge)
- Direct key event detecting
- Key press in the keyboard module
- Timer expiry
- Error detecting by power watchdog function (VCC power spike generating)

There is creating in only SYSCLK operating for interrupt factor. There is no prior ordering for interrupt. The interrupt needs to be cleared by I^2C programming before asserted again. The TC35894FG interrupt output circuit composing is indicated in the following figure. As the interrupt created from each module is connected with IRQ module in the combination logic, it is possible to output the interrupt for GPIO module from IRQN ball even if SLEEP mode (SYSCLK stop).

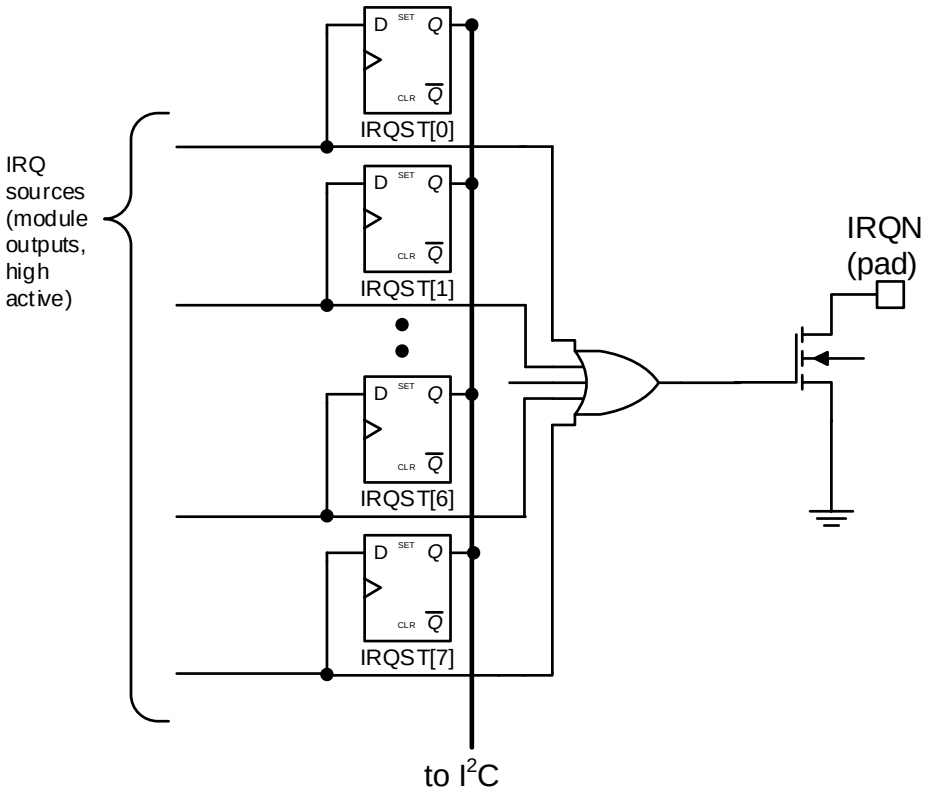


Figure 11.1 Interrupt output circuit composing

The IRQN pin is open drain. Externally pull-up resistor must be connected for exact operating.

IRQST register (0x91)

R/W	ITEM	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
R	MNEMONIC	PORIRQ	KBDIRQ	DKBDIRQ	-	TI2IRQ	TI1IRQ	TI0IRQ	GPIIRQ
	Default	1	0	0	*	0	0	0	0

PORIRQ VCC supply error
0 : No VCC supply error
1 : VCC error detecting.
Entire LSI is reset and requires re-programming.

KBDIRQ Keyboard interrupt
0 : Inactive
1 : Active

DKBDIRQ Direct keyboard interrupt
0 : Inactive
1 : Active

TI2IRQ Timer2 expiry (CDIRQ or CYCIRQ)
0 : Inactive
1 : Active

TI1IRQ Timer1 expiry (CDIRQ or CYCIRQ)
0 : Inactive
1 : Active

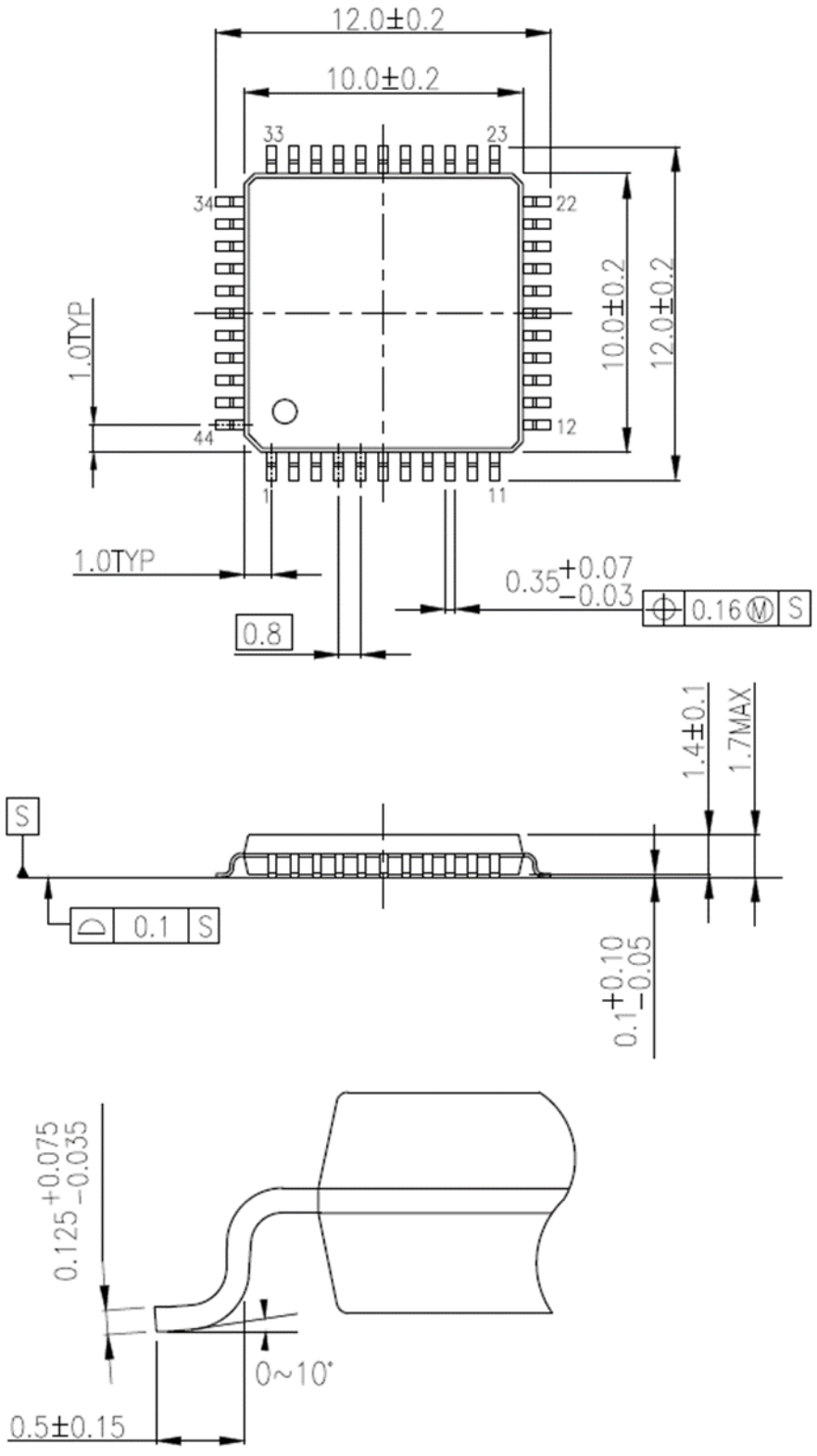
TI0IRQ Timer0 expiry (CDIRQ or CYCIRQ)
0 : Inactive
1 : Active

GPIIRQ GPIO interrupt
0 : Inactive
1 : Active

12. Package Mechanical Dimensions

The following drawing shows the dimensions of the P-LQFP44-1010-0.80-001, 0.8mm pitch package.

Unit: mm



Weight: 0.35g (Typ.)

Figure 12.1 P-LQFP44-1010-0.80-001, 0.8mm pin pitch (10mm x 10mm)

13. Electrical Parameters

13.1. I²C AC Timing

The following diagram specifies the standard I²C timings for "fast" mode.

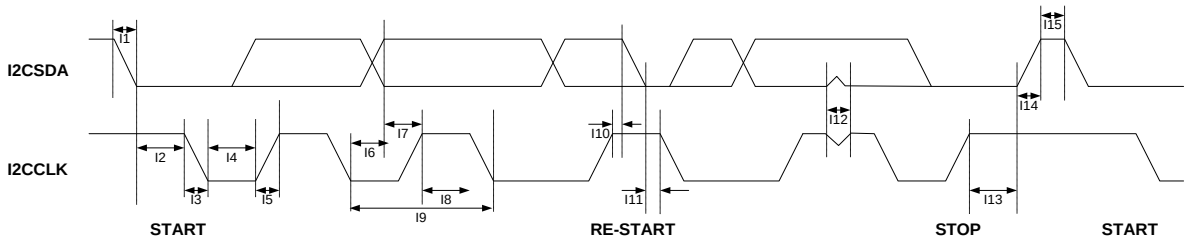


Figure 13.1 I²C AC Timing

Table 13.1 I²C AC Timing

Symbol	Description [a]	Min	Max
I9	I²C CLK frequency (1/I9)	-	400 kHz
I4	I²C CLK low time	1.3 µs	-
I8	I²C CLK high time	0.6 µs	-
I3	I²C CLK fall time	-	0.3 µs
I5	I²C CLK rise time	-	0.3 µs
I1	I²C SDA fall time	-	0.3 µs
I14	I²C SDA rise time	-	0.3 µs
I7	Data setup time	0.1 µs	-
I6	Data hold time	0.3 µs	-
I2	Hold time start condition	0.6 µs	-
I10	Setup time re-start condition	0.6 µs	-
I13	Setup time for stop condition	0.6 µs	-
I11	Hold time for restart	0.6 µs	-
I12	Spike length	-	50 ns
I15	Guard time (Bus free period between Stop condition and Start condition)	1.3 µs	-

Note:

a. Output timings depend on the value of the externally used pull-up resistor. The above value is the maximum allowed values from Chapter 15 reference material [1].

13.2. External clock input timing

In case of using external clock, input the following pulse into DIR24.

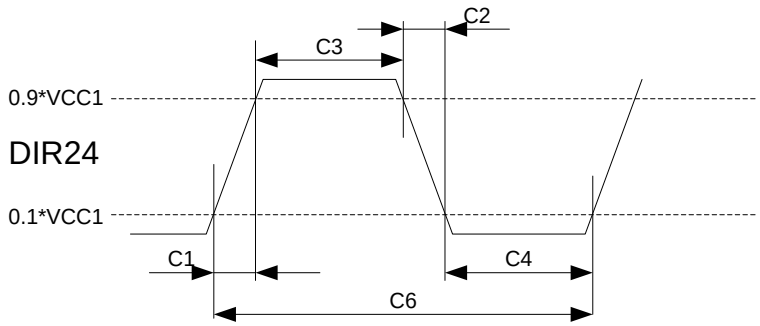


Figure 13.2 Direct clock input timing

The following table defines the timing.

Table 13.2 Clock Input Timing

Symbol	Description	Min	Max
C6	DIR24 frequency	32 kHz	20 MHz
C1	DIR24 input rise time	-	4 ns
C2	DIR24 input fall time	-	4 ns
C3/C4	Duty cycle high/low	45/55	55/45

Note: DIR24 < 32 kHz will not lead to damage of the device but the operation of modules working at system clock can no longer be guaranteed.

13.3. Internal Oscillator

Table 13.3 Internal RC Oscillating Clock Frequency Range

Symbol	Description	Min	Max
Fosc	Oscillator frequency	1.54 MHz	2.86 MHz

13.4. Power Supply Timing

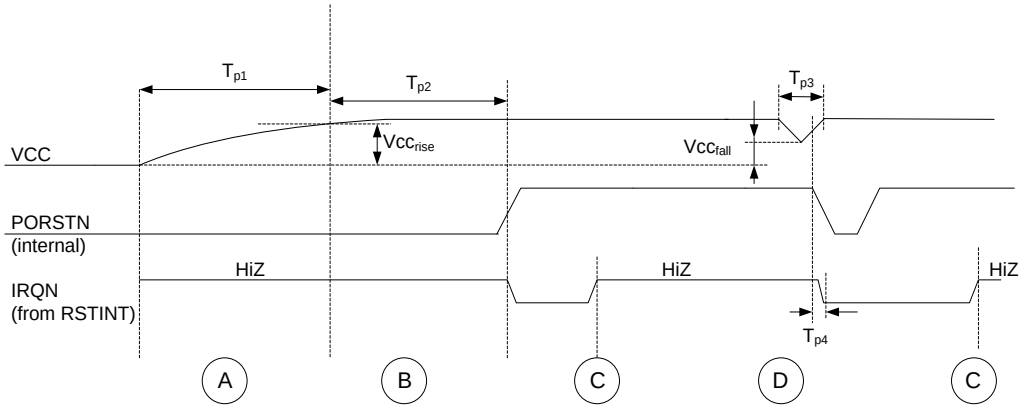


Figure 13.3 Power up and Supply watchdog control

Table 13.4 AC-parameters for power up and power watchdog

Symbol	Description	Min	Max
T_{p1}	Rise time for VCC (Note 1)	-	80 µs
T_{p2}	Time from VCC = VCC_{rise} to release of PORSTN	20 µs	120 µs
T_{p3}	Time for VCC ≤ $VCC_{fall.}$ In case of detecting spike more than $T_{p3,}$ PORSTN is taken by watchdog function.	2 µs	-
T_{p4}	Time from PORSTN trigger detecting to IRQN generating	-	30 µs
VCC_{rise}	Voltage threshold for PORSTN released in VCC on	1.00 V	1.55 V
VCC_{fall}	Voltage threshold for PORSTN released again in VCC on again	1.00 V	1.55 V

VCC_{rise} and VCC_{fall} can be adjusted by register PORTRIM.

Note 1: We assume any problem is not occurred until around 2 ms. But, to make sure, we recommend to have "software reset" register setting after Vcc power-on.

13.5. GPIO pads

The standard GPIO pad involves the following functionality**:**

- Programmable input/output/bidirectional
- Programmable pull-up/pull-down, only effective, if pin in input direction.
- Programmable I/O output drive-strength (DR)
- Programmable pseudo open drain output
- CMOS Schmidt input for switch noise cancellation

EN	ouput enable, low active
A	ouput line
PU	pull up, configurable
PD	pull down, configurable
IO	IO pad, pin, ball
Z	input line
DR	ouput drive

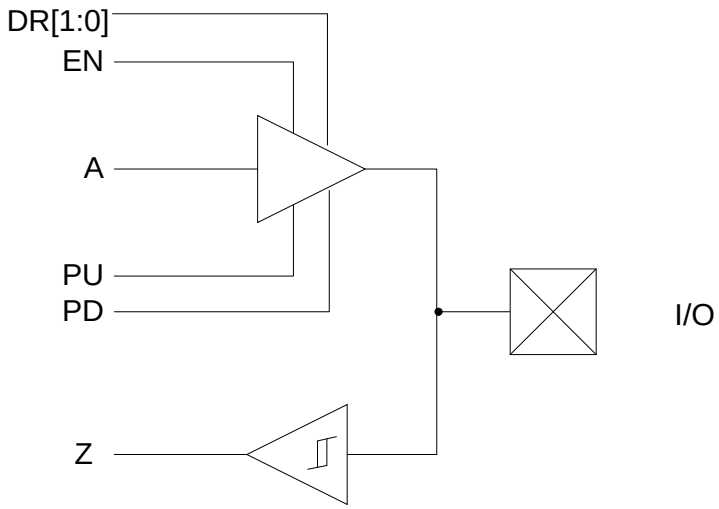


Figure 13.4 GPIO Symbol

The Figure 13.4 shows the symbol of the GPIO pads. The typical signal lines of a bidirectional pad driver are complemented by pull-up/pull down pins (PU/PD) and drive strength settings (DR [1**:**0]). The GPIO pad is used for Keyboard (KPX0...KPX7, KPY0...KPY10).

13.5.1. GPIO AC-parameters

The following figures illustrate the drive characteristics of the GPIO (N-MOS and P-MOS transistors)**:**

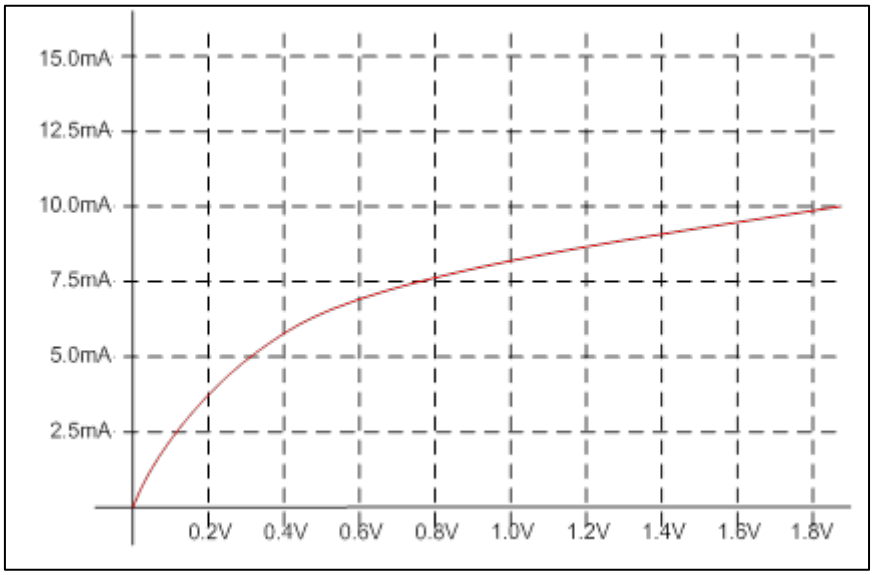


Figure 13.5 GPIO Output Voltage vs. Output Current (VOL-IOL @ VCC = 1.8 V, Temp = 25°C)

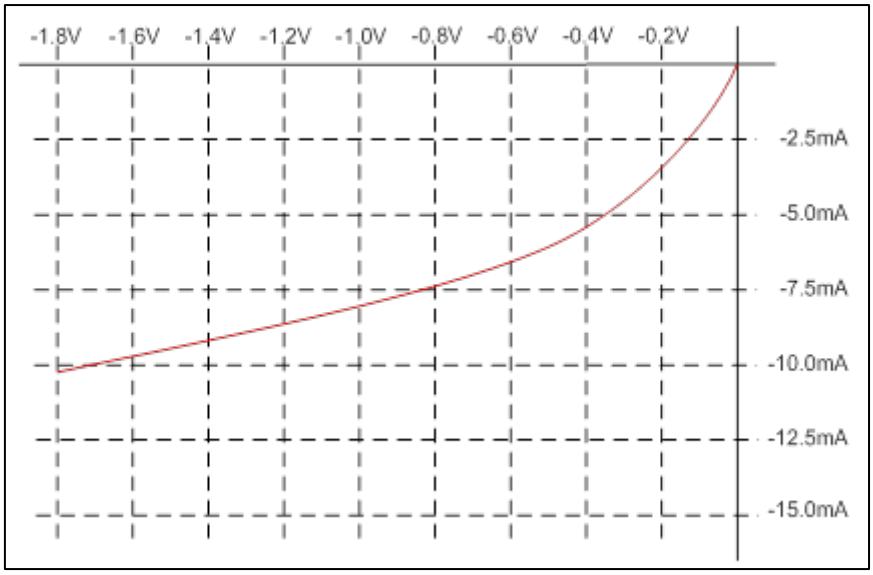


Figure 13.6 GPIO Output Voltage vs. Output Current (VOH-IOH @ VCC = 1.8 V, Temp = 25°C)

The GPIO input characteristics corresponds to a CMOS Schmitt trigger with minimum hysteresis of 0.15 V. The diagram below shows the switching points for both input transitions.

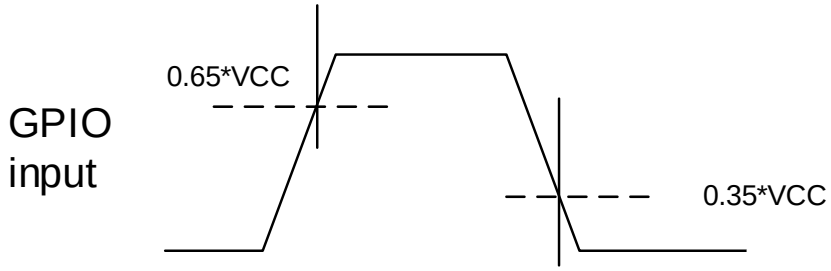


Figure 13.7 GPIO Input Characteristics (Vin switching points)

Table 13.5 GPIO Input Voltage Threshold Level

Symbol	Description	Min	Max
VIH	Input voltage for safe high detection	0.65 * VCC	-
VIL	Input voltage for safe low detection	-	0.35 * VCC
-	Minimum switching hysteresis	50 mV	-

When the pad is switched into input mode, pull up or pull down resistors being added via I^2C programming are activated. The characteristics for pull down and pull up are not symmetric**:**

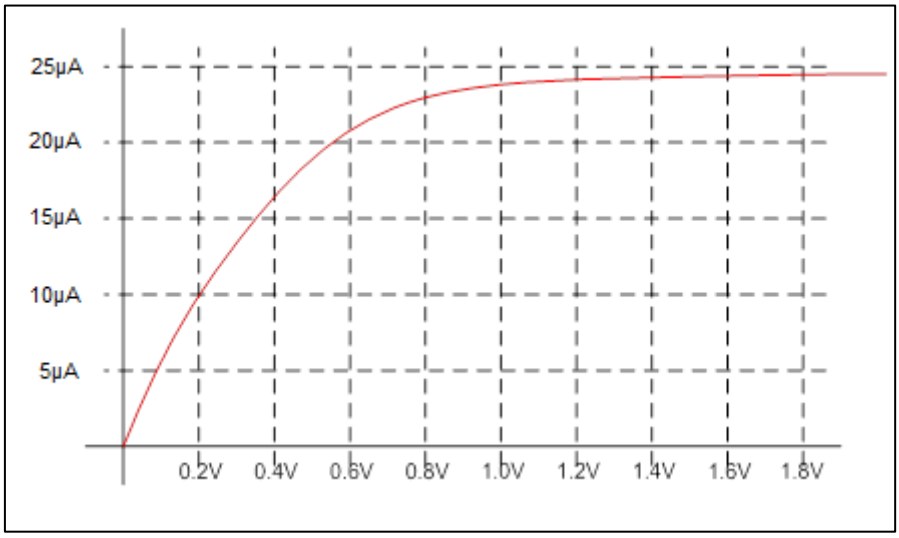


Figure 13.8 Pull-down (VCC = 1.8 V, Temp = 25°C)

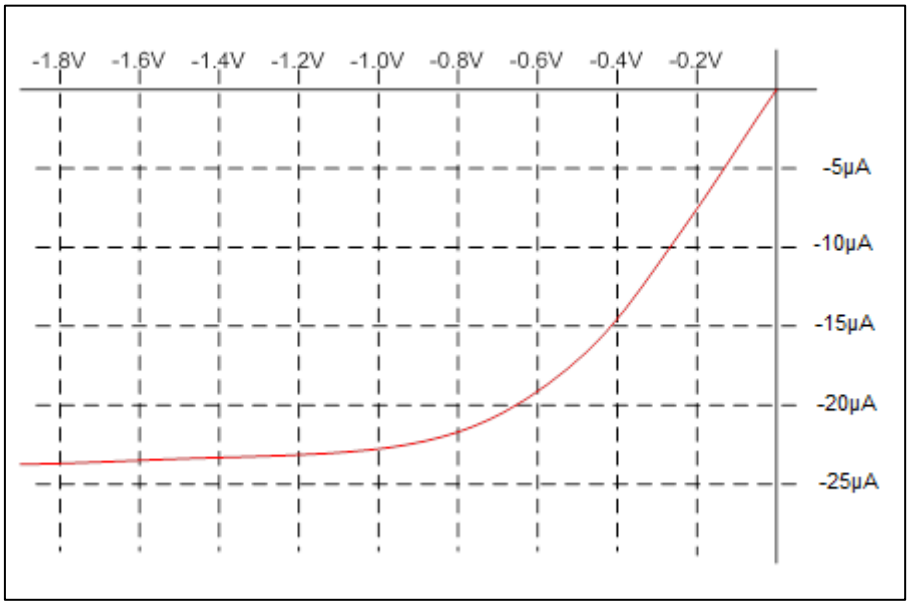


Figure 13.9 Pull-up (VCC = 1.8 V, Temp = 25°C)

13.6. Fail-safe Pads

Regarding characteristic, the pads used for the IRQN output, the RESETN inputs and the I²C bus SDA and SCL lines are different from the GPIO pads. They are implemented as true open drain pad types with removed PMOS transistor. This allows fail-safe operation on the I²C bus, the IRQN even in those cases where the TC35894FG is not powered.

DC characteristics are the same as for GPIO pads.

13.6.1. I²C/IRQN AC-parameters

The output characteristics for IRQN and SDA have higher drive strength than the standard GPIO output drivers.

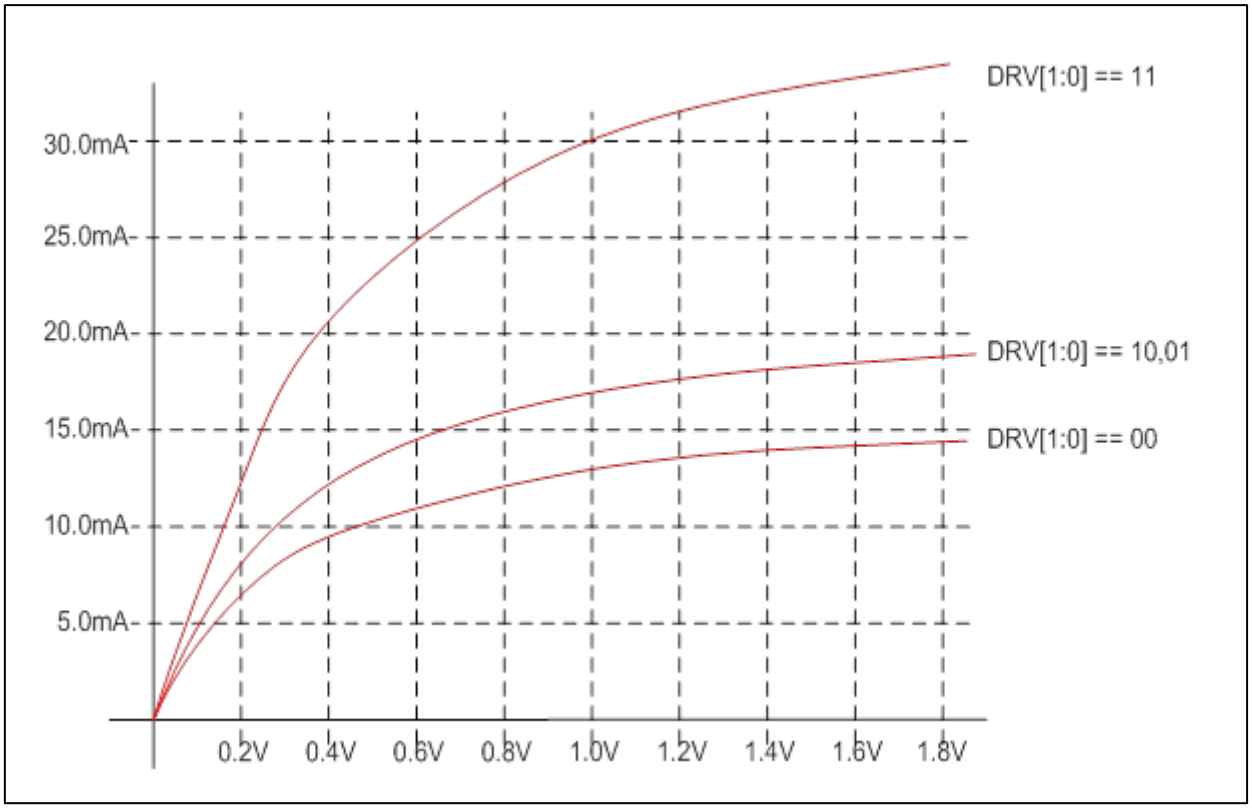


Figure 13.10 IRQN/SDA Output Voltage vs Output Current (VOL-IOL@VCC = 1.8 V, Temp = 25°C)

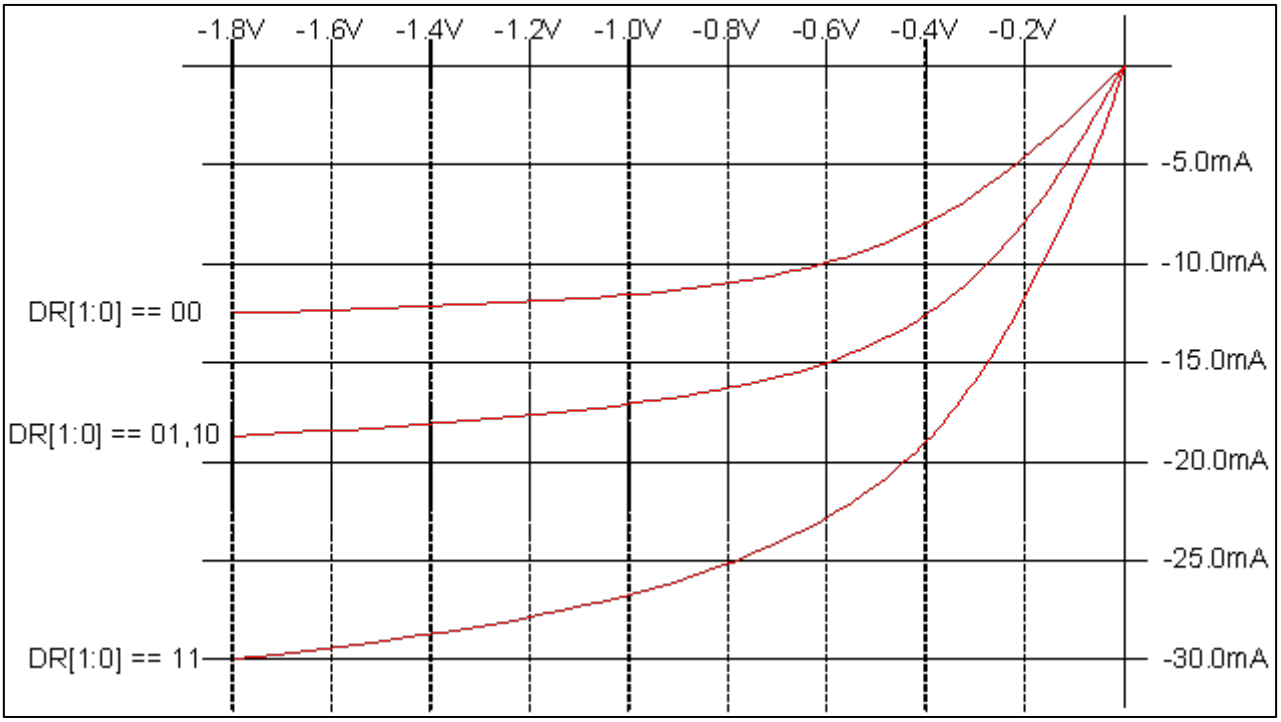


Figure 13.11 IRQN/SDA Output Voltage vs Output Current (VOH-IOH@VCC = 1.8 V, Temp = 25°C)

For Inputs RESETN, SCL and SDA, the input characteristics differ from the standard GPIO characteristics. These inputs have no internal pull resistors attached as they are failsafe with respect to a shut down power supply.

14. Conditions

14.1. Operating Conditions

The following ranges of voltage and temperature levels have to be respected for correct operation.

Table 14.1 Operating Characteristics

Symbol	Description	Min	Typ.	Max	Unit
TEMP	Ambient temperature range	-40	25	85	°C
VCC	Power supply	1.62	1.8	3.6	V
V_{NOISE}	Input stability of the supply voltage	-	-	0.05	%/V
		-	-	100	mV
VOH	DC output voltage of IOs	-	-	VCC	V
	V_{OUT} @ 1 mA I_{OUT} and DRV = 00, Ta = 25°C	0.75*VCC	0.8*VCC	-	
	V_{OUT} @ 2 mA I_{OUT} and DRV = 01, Ta = 25°C	0.75*VCC	0.8*VCC	-	
	V_{OUT} @ 4 mA I_{OUT} and DRV = 11, Ta = 25°C	0.75*VCC	0.8*VCC	-	
VOL	DC output voltage of IOs	GND	-	-	
	V_{OUT} @ -1 mA I_{OUT} and DRV = 00, Ta = 25°C	-	0.2*VCC	0.25*VCC	
	V_{OUT} @ -2 mA I_{OUT} and DRV = 01, Ta = 25°C	-	0.2*VCC	0.25*VCC	
	V_{OUT} @ -4 mA I_{OUT} and DRV = 11, Ta = 25°C	-	0.2*VCC	0.25*VCC	
IIL	Input current	-	-	-10	µA
	Input current with pull-up, VCC = 1.8 V	-25	-	-110	
IIH	Input current	-	-	10	
	Input current with pull-down, VCC = 1.8 V	25	-	110	
Iquies	Total leakage current in SLEEP mode (VCC = 1.8 V, Ta = 25°C)	-	15	-	µA
$I^{a, b}$	Supply current in OPERATION mode (VCC = 1.8 V, Ta = 25°C)	-	122	-	

Note:

a. Measured using typical applications on PWM and Keyboard. Measured values are indications, only. Values cannot be guaranteed, because they depend on external circuitry and on use case.

b. When using AUTOSLEEP function and no PWM in use, the average current figures for operation mode get close to the ones specified in I_{quies}

14.2. Absolute Maximum Ratings

The absolute maximum ratings as defined in below table are not meant to be exceeded during any time of the product life cycle. Exceeding maximum ratings will potentially result in permanent damage to the device.

Table 14.2 Absolute Maximum Ratings

Symbol	Description	Min	Typ.	Max	Unit
VCC	Supply Voltage	-0.3	-	3.9	V
Vi	CMOS input voltage	-0.3	-	VCC+0.3	V
Vout	DC output voltage	-0.3	-	VCC+0.3	
V_{NOISE}	Supply Noise Voltage (peak to peak)	-	-	190	mV
I_{in}	Maximum input current	-10	-	10	mA
Ta	Ambient Temperature	-40	25	85	°C
Ts	Storage Temperature	-40	-	125	

15. References

[1] I^2C Bus Specification

"The I^2C -Bus Specification" Version 2.1 January 2000, Philips Semiconductor

16. Register Map

The following table shows the entire register map of the TC35894FG.

Regarding all addresses that are not defined, don't access.

Changing default value in the reserved bits will not guarantee the normal operation.

All bits in defined addresses, which are not in use (dashed in table), shall be written to "0."

Reading from undefined bits results in undefined return values.

Addresses in the range from 0x80 to 0xFF can be accessed without the need of a system clock. Access to addresses from 0x00 to 0x7F requires a running system clock for being successful. The I^2C bus allows data burst of arbitrary length but bursting beyond the address 0xFF is not permitted. A burst beyond the address 0x7F wraps the address back to the value 0x7E.

Some registers are required a 2-bytes unit access. Those registers are updated only at the end of the full 2-bytes transfer. Before an I^2C command is transferred to be finished, I^2C address is reset into I^2C address value to use in last access before starting 2-bites transferred..

16.1. Register Map

Table 16.1 Register Map

NAME	Address	R/W	Reset value	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
KBDSETTLE	0x01	R/W	0xA3	WAIT [7:0]							
KBDBOUNCE	0x02	R/W	0xA3	BOUNCETIM [7:0]							
KBDSIZE	0x03	R/W	0x00	ROWSIZE [3:0]				COLSIZE [3:0]			
KBDDEDCFG	0x04	R/W	0xFF	COL9	COL8	COL7	COL6	COL5	COL4	COL3	COL2
	0x05	R/W	0xFF	ROW7	ROW6	ROW5	ROW4	ROW3	ROW2	COL11	COL10
KBDRIS	0x06	R	0x00	—	—	—	—	RELINT	REVT INT	RKLINT	RSINT
KBDMIS	0x07	R	0x00	—	—	—	—	MELINT	MEVT INT	MKLINT	MSINT
KBDIC	0x08	W	0x00	SFOFF	—	—	—	—	—	EVTIC	KBDIC
KBDMSK	0x09	R/W	0x0C	—	—	—	—	MSKELINT	MSKEINT	MSKKLINT	MSKSINT
KBDCODE0	0x0B	R	0x7F	MULTIKEY	KEYROW [2:0]			KEYCOL [3:0]			
KBDCODE1	0x0C	R	0x7F	MULTIKEY	KEYROW [2:0]			KEYCOL [3:0]			
KBDCODE2	0x0D	R	0x7F	MULTIKEY	KEYROW [2:0]			KEYCOL [3:0]			
KBDCODE3	0x0E	R	0x7F	MULTIKEY	KEYROW [2:0]			KEYCOL [3:0]			
EVTCODE	0x10	R	0xFF	RELEASE	KEYROW [2:0]			KEYCOL [3:0]			
TIMCFG0	0x60	R/W	0x00	—	—	—	IRQMASK	CYCCTRL	FREE	SYNC	START
PWMCFG0	0x61	R/W	0x00	—	—	—	—	IRQMASK	PGE	PWMEN	PWMPOL
TIMSCAL0	0x62	R/W	0x00	SCAL [7:0]							
TIMCYCLE0	0x63	R/W	0x00	CYCLE [7:0]							
TIMLOAD0	0x64	R/W	0xFF	LOAD [7:0]							
TIMCFG1	0x68	R/W	0x00	—	—	—	IRQMASK	CYCCTRL	FREE	SYNC	START
PWMCFG1	0x69	R/W	0x00	—	—	—	—	IRQMASK	PGE	PWMEN	PWMPOL
TIMSCAL1	0x6A	R/W	0x00	SCAL [7:0]							
TIMCYCLE1	0x6B	R/W	0x00	CYCLE [7:0]							
TIMLOAD1	0x6C	R/W	0xFF	LOAD [7:0]							
TIMCFG2	0x70	R/W	0x00	—	—	—	IRQMASK	CYCCTRL	FREE	SYNC	START
PWMCFG2	0x71	R/W	0x00	—	—	—	—	IRQMASK	PGE	PWMEN	PWMPOL
TIMSCAL2	0x72	R/W	0x00	SCAL [7:0]							
TIMCYCLE2	0x73	R/W	0x00	CYCLE [7:0]							
TIMLOAD2	0x74	R/W	0xFF	LOAD [7:0]							
TIMSWRES	0x78	W	0x00	—	—	—	—	—	SWRES2	SWRES1	SWRES0
TIMRIS	0x7A	R	0x00	—	—	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
TIMMIS	0x7B	R	0x00	—	—	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
TIMIC	0x7C	W	0x00	—	—	CDIRQ2	CDIRQ1	CDIRQ0	CYCIRQ2	CYCIRQ1	CYCIRQ0
PWMWP	0x7D	R/W	0x00	—	POINTER[6:0]						
PWMPAT [a]	0x7E	W	0x00	PAT [7:0]							
	0x7F		0x00	PAT [15:0]							
Manufacture code	0x80	R	0x03	0	0	0	0	0	0	1	1
SW version	0x81 b	R	0xC0	1	1	0	0	0	0	0	0
I2CSA	0x80	W	0x8A	SLAVEADDR [7:1]							-
RSTCTRL	0x82	R/W	0x00	—	—	reserved	IRQRST	TIMRST	reserved	KBDRST	GPIRST
EXTRSTN	0x83	R/W	0x1F	—	—	—	reserved	reserved	reserved	reserved	EXTRSTN
RSTINTCLR	0x84	W	0x00	—	—	—	—	—	—	—	IRQCLR
PORTRIM	0x85	R/W	0x00	POR_SEL	—	—	POR_TRIM [4:0]				
Reserved	0x86	R/W	0x00	reserved							
Reserved	0x87	R/W	0x00	reserved							
CLKMODE	0x88	R/W	0x01	—	—	—	—	—	—	reserved	MODCTL
CLKCFG	0x89	R/W	0x40	reserved	CLKSRCSE	reserved	CLKFDEN	CLKDIV [3:0]			
CLKEN	0x8A	R/W	0x00	CLKOUTEN		—	—	—	TIMEN	reserved	KBDEN
AUTOSLPENA	0x8B	R/W	0x00	—	—	—	—	—	—	—	ENABLE
AUTOSLPTIMER	0x8C	R/W	0xFF	UPTIME [7:0]							
	0x8D		0x07	—	—	—	—	—		UPTIME [10:8]	
I2CWAKEUPEN	0x8E	R/W	0x01	—	—	—	—	—	—	—	I2CWEN
KBDMFS	0x8F	R/W	0x01	—	—	—	—	—	—	—	MFSEN
IRQST	0x91	R/W	0x80	PORIRQ	KBDIRQ	DKBDIRQ	—	TIM2IRQ	TIM1IRQ	TIM0IRQ	GPIIRQ
DRIVE0	0xA0	R/W	0x00	KPX3DRV [1:0]		KPX2DRV [1:0]		KPX1DRV [1:0]		KPX0DRV [1:0]	
	0xA1		0x00	KPX7DRV [1:0]		KPX6DRV [1:0]		KPX5DRV [1:0]		KPX4DRV [1:0]	
DRIVE1	0xA2	R/W	0x00	KPY3DRV [1:0]		KPY2DRV [1:0]		KPY1DRV [1:0]		KPY0DRV [1:0]	
	0xA3		0x00	KPY7DRV [1:0]		KPY6DRV [1:0]		KPY5DRV [1:0]		KPY4DRV [1:0]	
DRIVE2	0xA4	R/W	0x00	KPY11DRV[1:0]		KPY10DRV[1:0]		KPY9DRV[1:0]		KPY8DRV[1:0]	
	0xA5		0x00	EXTIO0DRV[1:0]		PWM2DRV[1:0]		PWM1DRV[1:0]		PWM0DRV[1:0]	
DRIVE3	0xA6	R/W	0x00	—	—	—	—	IRQNDR1	IRQNDR0	SDADRV1	SDADRV0
IOCFG	0xA7	R/W	0xF8	GPIOSEL[3:0]		IG		reserved		BALLCFG	
IOPCEXT	0xA8	R/W	0x0A	—		—		DIR25PR[1:0]		DIR24PR[1:0]	
IOPC0	0xAA	R/W	0xAA	KPX3PR [1:0]		KPX2PR [1:0]		KPX1PR [1:0]		KPX0PR [1:0]	
	0xAB		0xAA	KPX7PR [1:0]		KPX6PR [1:0]		KPX5PR [1:0]		KPX4PR [1:0]	

NAME	Address	R/W	Reset value	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IOPC1	0xAC	R/W	0xAA	KPY3PR [1:0]		KPY2PR [1:0]		KPY1PR [1:0]		KPY0PR [1:0]	
	0xAD		0xAA	KPY7PR [1:0]		KPY6PR [1:0]		KPY5PR [1:0]		KPY4PR [1:0]	
IOPC2	0xAE	R/W	0xAA	KPY11PR[1:0]		KPY10PR[1:0]		KPY9PR[1:0]		KPY8PR[1:0]	
	0xAF		0xAA	EXTIO0PR[1:0]		PWM2PR[1:0]		PWM1PR[1:0]		PWM0PR[1:0]	
GPIODATA0 [a]	0xC0	R/W	0x00	KPX7DATA	KPX6DATA	KPX5DATA	KPX4DATA	KPX3DATA	KPX2DATA	KPX1DATA	KPX0DATA
	0xC1	W	0x00	MASK [7:0]							
GPIODATA1 [a]	0xC2	R/W	0x00	KPY7DATA	KPY6DATA	KPY5DATA	KPY4DATA	KPY3DATA	KPY2DATA	KPY1DATA	KPY0DATA
	0xC3	W	0x00	MASK [15:8]							
GPIODATA2 [a]	0xC4	R/W	0x00	EXTIO0DATA	PWM2DATA	PWM1DATA	PWM0DATA	KPY11DATA	KPY10DATA	KPY9DATA	KPY8DATA
	0xC5	W	0x00	MASK [23:16]							
GPIODIR0	0xC6	R/W	0x00	KPX7DIR	KPX6DIR	KPX5DIR	KPX4DIR	KPX3DIR	KPX2DIR	KPX1DIR	KPX0DIR
GPIODIR1	0xC7	R/W	0x00	KPY7DIR	KPY6DIR	KPY5DIR	KPY4DIR	KPY3DIR	KPY2DIR	KPY1DIR	KPY0DIR
GPIODIR2	0xC8	R/W	0x00	EXTIO0DIR	PWM2DIR	PWM1DIR	PWM0DIR	KPY11DIR	KPY10DIR	KPY9DIR	KPY8DIR
GPIOIS0	0xC9	R/W	0x00	KPX7IS	KPX6IS	KPX5IS	KPX4IS	KPX3IS	KPX2IS	KPX1IS	KPX0IS
GPIOIS1	0xCA	R/W	0x00	KPY7IS	KPY6IS	KPY5IS	KPY4IS	KPY3IS	KPY2IS	KPY1IS	KPY0IS
GPIOIS2	0xCB	R/W	0x00	EXTIO0IS	PWM2IS	PWM1IS	PWM0IS	KPY11IS	KPY10IS	KPY9IS	KPY8IS
GPIOIBE0	0xCC	R/W	0x00	KPX7IBE	KPX6IBE	KPX5IBE	KPX4IBE	KPX3IBE	KPX2IBE	KPX1IBE	KPX0IBE
GPIOIBE1	0xCD	R/W	0x00	KPY7IBE	KPY6IBE	KPY5IBE	KPY4IBE	KPY3IBE	KPY2IBE	KPY1IBE	KPY0IBE
GPIOIBE2	0xCE	R/W	0x00	EXTIO0IBE	PWM2IBE	PWM1IBE	PWM0IBE	KPY11IBE	KPY10IBE	KPY9IBE	KPY8IBE
GPIOIEV0	0xCF	R/W	0x00	KPX7IEV	KPX6IEV	KPX5IEV	KPX4IEV	KPX3IEV	KPX2IEV	KPX1IEV	KPX0IEV
GPIOIEV1	0xD0	R/W	0x00	KPY7IEV	KPY6IEV	KPY5IEV	KPY4IEV	KPY3IEV	KPY2IEV	KPY1IEV	KPY0IEV
GPIOIEV2	0xD1	R/W	0x00	EXTIO0IEV	PWM2IEV	PWM1IEV	PWM0IEV	KPY11IEV	KPY10IEV	KPY9IEV	KPY8IEV
GPIOIE0	0xD2	R/W	0x00	KPX7IE	KPX6IE	KPX5IE	KPX4IE	KPX3IE	KPX2IE	KPX1IE	KPX0IE
GPIOIE1	0xD3	R/W	0x00	KPY7IE	KPY6IE	KPY5IE	KPY4IE	KPY3IE	KPY2IE	KPY1IE	KPY0IE
GPIOIE2	0xD4	R	0x00	EXTIO0 IE	PWM2IE	PWM1IE	PWM0IE	KPY11IE	KPY10IE	KPY9IE	KPY8IE
GPIORIS0	0xD6	R	0x00	KPX7RIS	KPX6RIS	KPX5RIS	KPX4RIS	KPX3RIS	KPX2RIS	KPX1RIS	KPX0RIS
GPIORIS1	0xD7	R	0x00	KPY7RIS	KPY6RIS	KPY5RIS	KPY4RIS	KPY3RIS	KPY2RIS	KPY1RIS	KPY0RIS
GPIORIS2	0xD8	R	0x00	EXTIO0RIS	PWM2 RIS	PWM1 RIS	PWM0 RIS	KPY11 RIS	KPY10 RIS	KPY9 RIS	KPY8 RIS
GPIOMIS0	0xD9	R	0x00	KPX7MIS	KPX6MIS	KPX5MIS	KPX4MIS	KPX3MIS	KPX2MIS	KPX1MIS	KPX0MIS
GPIOMIS1	0xDA	R	0x00	KPY7MIS	KPY6MIS	KPY5MIS	KPY4MIS	KPY3MIS	KPY2MIS	KPY1MIS	KPY0MIS
GPIOMIS2	0xDB	W	0x00	EXTIO0MIS	PWM2MIS	PWM1MIS	PWM0MIS	KPY11MIS	KPY10MIS	KPY9MIS	KPY8MIS
GPIOIC0	0xDC	W	0x00	KPX7IC	KPX6IC	KPX5IC	KPX4IC	KPX3IC	KPX2IC	KPX1IC	KPX0IC
GPIOIC1	0xDD	W	0x00	KPY7IC	KPY6IC	KPY5IC	KPY4IC	KPY3IC	KPY2IC	KPY1IC	KPY0IC
GPIOIC2	0xDE	R/W	0x00	EXTIO0IC	PWM2IC	PWM1IC	PWM0IC	KPY11IC	KPY10IC	KPY9IC	KPY8IC
GPIOOMS0 [a]	0xE0	R/W	0x00	KPX7ODE	KPX6ODE	KPX5ODE	KPX4ODE	KPX3ODE	KPX2ODE	KPX1ODE	KPX0ODE
GPIOOMS0 [a]	0xE1	R/W	0x00	KPX7ODM	KPX6ODM	KPX5ODM	KPX4ODM	KPX3ODM	KPX2ODM	KPX1ODM	KPX0ODM
GPIOOMS1 [a]	0xE2	R/W	0x00	KPY7ODE	KPY6ODE	KPY5ODE	KPY4ODE	KPY3ODE	KPY2ODE	KPY1ODE	KPY0ODE
GPIOOMS1 [a]	0xE3	R/W	0x00	KPY7ODM	KPY6ODM	KPY5ODM	KPY4ODM	KPY3ODM	KPY2ODM	KPY1ODM	KPY0ODM
GPIOOMS2 [a]	0xE4	R/W	0x00	EXTIO0ODE	PWM2ODE	PWM1ODE	PWM0ODE	KPY11ODE	KPY10ODE	KPY9ODE	KPY8ODE
GPIOOMS2 [a]	0xE5	-	0x00	EXTIO0ODM	PWM2ODM	PWM1ODM	PWM0ODM	KPY11ODM	KPY10ODM	KPY9ODM	KPY8ODM
DEVTCODE	0xE6	R	0x3F	—	—	DKEYSTAT	DKEYCODE[4:0]				
Reserved	0xE7	-	-	reserved							
DBOUNCE	0XE8	R/W	0x06	—	—	SYNC	DBOUNCE[4:0]				
GPIOWAKE0	0xE9	R/W	0x00	KPX7WAKE	KPX6WAKE	KPX5WAKE	KPX4WAKE	KPX3WAKE	KPX2WAKE	KPX1WAKE	KPX0WAKE
GPIOWAKE1	0xEA	R/W	0x00	KPY7WAKE	KPY6WAKE	KPY5WAKE	KPY4WAKE	KPY3WAKE	KPY2WAKE	KPY1WAKE	KPY0WAKE
GPIOWAKE2	0xEB	R/W	0x00	EXTIO0WAKE	PWM2WAKE	PWM1WAKE	PWM0WAKE	KPY11WAKE	KPY10WAKE	KPY9WAKE	KPY8WAKE
DIRECT0	0xEC	R/W	0xFF	Direct 7	Direct 6	Direct 5	Direct 4	Direct 3	Direct 2	Direct 1	Direct 0
DIRECT1	0xED	R/W	0xFF	Direct 15	Direct 14	Direct 13	Direct 12	Direct 11	Direct 10	Direct 9	Direct 8
DIRECT2	0xEE	R/W	0xFF	Direct 23	Direct 22	Direct 21	Direct 20	Direct 19	Direct 18	Direct 17	Direct 16
DIRECT3	0xEF	R/W	0x03	—	—	—	—	—	—	Direct 25	Direct 24
DKBDRIS	0xF0	R	0x00	—	—	—	—	—	—	DRELINT	DREVTINT
DKBDMIS	0xF1	R	0x00	—	—	—	—	—	—	DMELINT	DMEVTINT
DKBDIC	0xF2	W	0x00	—	—	—	—	—	—	—	DEVTIC
DKBDMSK	0xF3	R/W	0x00	—	—	—	—	—	—	DMSKELINT	DMSKEINT

Note:

a. Only updated as 16-bit register after having written both bytes.

b. Write access into this register results in undefined behavior.

17. System Integration

The following figures show a recommended printed circuit board design. The global reset is issued via I²C, via power-on reset or dedicated RESETN pin. An I²C default address is at binary "1000101". The functional pins KPX, KPY, PWM, EXTIO0, DIR24 and DIR25 have an internal pull-up resistor programmed (register IOPC) to avoid damage through static discharges when no key is pressed.

17.1. In case of connected with external CMOS level oscillator as clock input

The clock is provided by DIR24 ball (IOCFG.BALLCFG = 0x2).

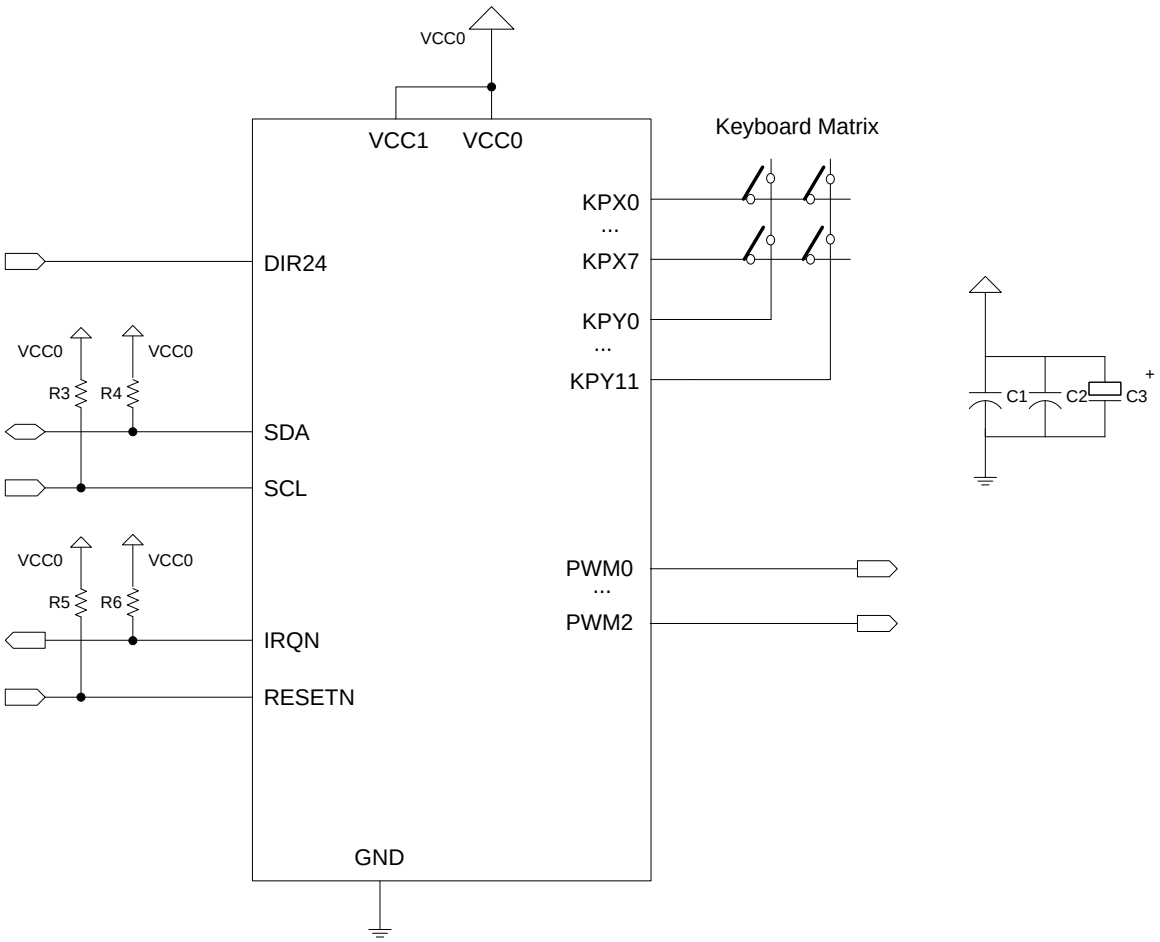


Figure 17.1 Application circuit example (External CMOS clock input case)

Table 17.1 Recommended Values1

Component	R3, R4	R5, R6	C1	C2	C3
VCC = 1.8 V	3.3 kΩ	10 kΩ	1 nF	10 nF	10 µF

17.2. In case of using internal RC oscillating clock Internal clock

In Figure 17.2, the DIR24 pin is used as direct key input (IOCFG.BALLCFG = 0x0). The internal RC-oscillator provides the SYSCLK.

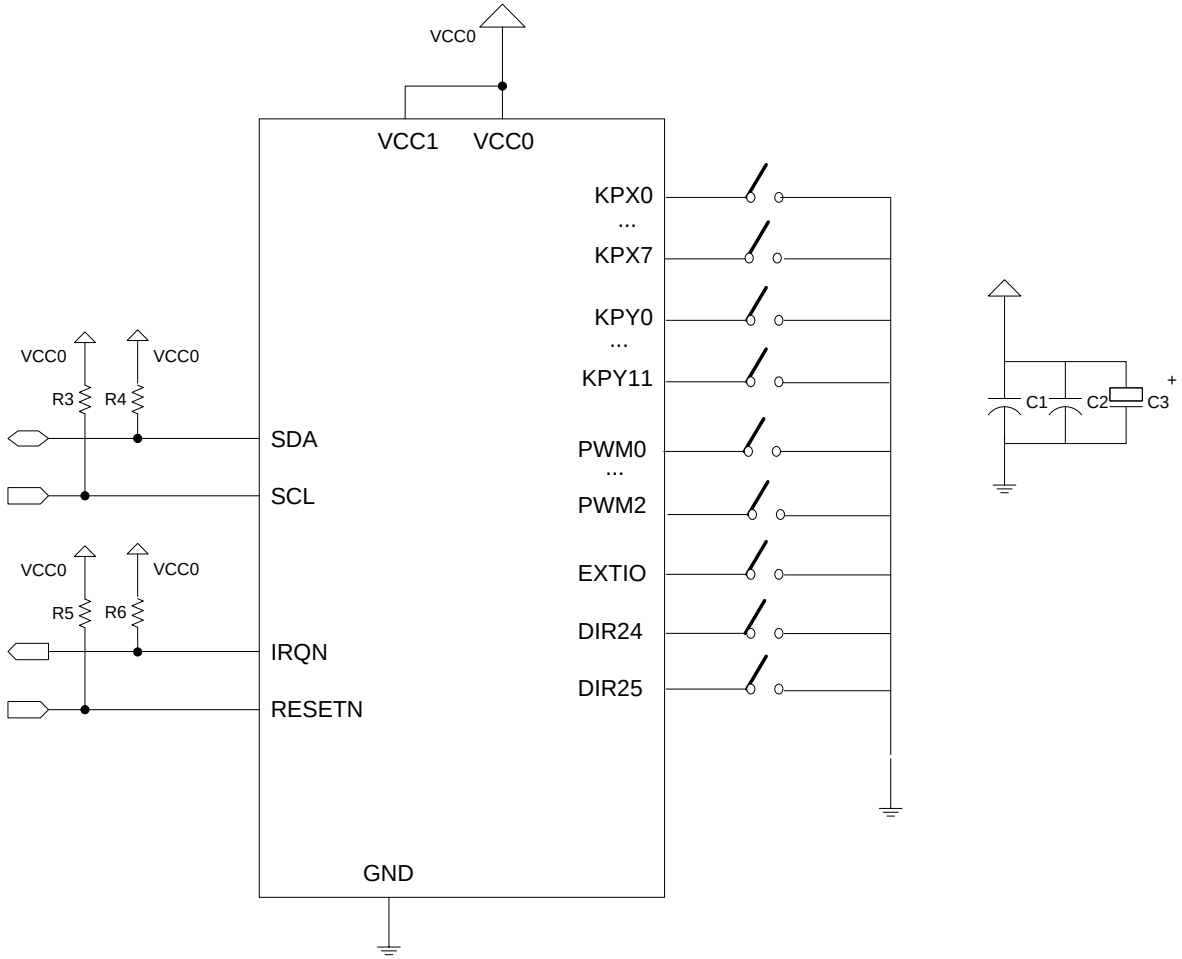


Figure 17.2 Application Circuit example (Internal RC oscillator using case)

Table 17.2 Recommended Values2

Component	R3, R4	R5, R6	C1	C2	C3
VCC = 1.8 V	3.3 kΩ	10 kΩ	1 nF	10 nF	10 µF

18. Revision History

Table 18.1 Revision History

Revision	Date	Description
1.1	2015-12-11	Newly released
1.2	2016-09-07	Modified DRIVE3 register in section 6.3.
1.3	2017-11-15	Corrected typos. Added Revision History. Changed header, footer and the last page. Changed corporate name.
1.4	2018-03-12	Added Supply Voltage in Table 14.2.
1.41	2018-11-01	Changed the package figure in the cover page. Added description of trademark. Modified descriptions in section 1. Corrected typos. Changed from QFP to LQFP in section 3.1. Modified Figure 12.1. Modified Table 13.1 and Table 13.2. Revised the last page “RESTRICTIONS ON PRODUCT USE”, and added URL.
1.42	2019-05-14	Modified CLKEN register in section 5.5. Modified DBOUNCE register in section 9.2.14.

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as “TOSHIBA”.
Hardware, software and systems described in this document are collectively referred to as “Product”.

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE"). ** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**