

LM8325-1 Mobile I/O Companion Supporting Keyscan, I/O Expansion, PWM, and ACCESS.bus Host Interface

Check for Samples: [LM8325-1](#)

FEATURES

• KEY FEATURES

- Internal RC Oscillator, No External Clock Required
- Internal PWM Clock Generation, No External Clock Required
- Programmable I²C-compatible ACCESS.bus Address (Default 0x88)
- Support for Keypad Matrices of up to of 8 x 12 Keys, Plus 8 Special Function (SF) Keys, for a Full 104 Key Support
- I²C-compatible ACCESS.bus Slave Interface at 100 kHz (Standard-Mode) and 400 kHz (Fast-Mode)
- Three Host-programmable PWM Outputs for Smooth LED Brightness Modulation
- Supports General-purpose I/O Expansion on Pins Not Otherwise Used for Keypad or PWM Output
- 15 Byte Key Event Buffer
- Multiple Key Event Storage
- Key Events, Errors, and Dedicated Hardware Interrupts Request Host Service by Asserting an IRQ Output
- Automatic HALT Mode for Low Power Operation
- Wake-up from HALT Mode on Any Interface (Rising Edge, Falling Edge or Pulse)
- Three PWM Outputs with Dedicated Script Buffer for up to 32 Commands
- Register-based Command Interpreter with Auto Increment Address

• HOST-CONTROLLED FEATURES

- PWM Scripting for Three PWM Outputs
- Period of Inactivity That Triggers Entry into HALT Mode
- Debounce Time for Reliable Key Event Polling
- Configuration of General Purpose I/O Ports
- Various Initialization Options (Keypad Size,

etc.)

• KEY DEVICE FEATURES

- 1.8V ± 10% Single-supply Operation
- On-chip Power-on Reset (POR)
- Watchdog Timer
- Dedicated Slow Clock Input for 32 kHz to 2MHz
- –40°C to +85°C Temperature Range
- 25-pin csBGA Package

APPLICATIONS

- Cordless Phones
- Smart Handheld Devices
- Keyboard Applications

DESCRIPTION

The LM8325-1 GenI/O - Expander and Keypad Controller is a dedicated device to unburden a host processor from scanning a matrix-addressed keypad and to provide flexible and general purpose, host programmable input/output functions. Three independent PWM timer outputs are provided for dynamic LED brightness modulation.

It communicates with a host processor through an I²C-compatible ACCESS.bus serial interface. It can communicate in Standard (100 kHz) - and Fast-Mode (400 kHz) in slave Mode only.

All available input/output pins can alternately be used as an input or an output in a keypad matrix or as a host programmable general purpose input or output.

Any pin programmed as an input can also sense hardware interrupts. The interrupt polarity (“high to low” or “low to high” transition) is thereby programmable.

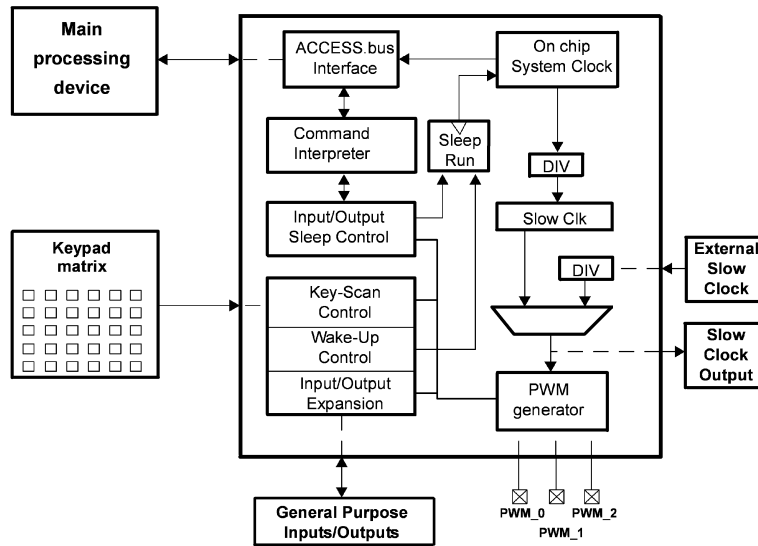
The LM8325-1 follows a predefined register based set of commands. Upon startup (power - on) a configuration file must be sent from the host to setup the hardware of the device.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

LM8325-1 FUNCTION BLOCKS



PIN ASSIGNMENTS

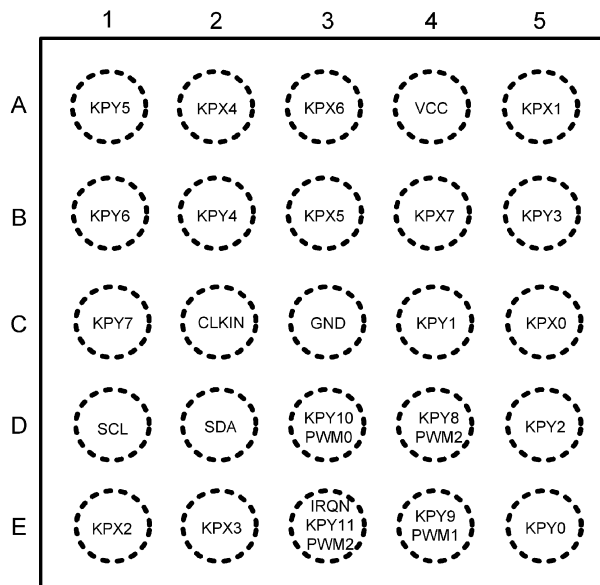


Figure 1. LM8325-1 Pinout — Top View

SIGNAL DESCRIPTIONS

DEVICE PIN FUNCTIONS

KEY AND ALTERNATE FUNCTIONS OF ALL DEVICE PINS

| Ball | Function 0 | Function 1 | Function 2 | Function 3 | Pin Count | Ball Name |
|------|------------------------------|----------------|------------|------------|-----------|-----------|
| C2 | Slow Clock Input | | | | 1 | CLKIN |
| A4 | Supply Voltage | | | | 1 | VCC |
| D1 | Main I ² C - Clk | | | | 1 | SCL |
| D2 | Main I ² C - data | | | | 1 | SDA |
| C5 | Keypad-I/O X0 | Genio0 | | | 1 | KPX0 |
| A5 | Keypad-I/O X1 | Genio1 | | | 1 | KPX1 |
| E1 | Keypad-I/O X2 | Genio2 | | | 1 | KPX2 |
| E2 | Keypad-I/O X3 | Genio3 | | | 1 | KPX3 |
| A2 | Keypad-I/O X4 | Genio4 | | | 1 | KPX4 |
| B3 | Keypad-I/O X5 | Genio5 | | | 1 | KPX5 |
| A3 | Keypad-I/O X6 | Genio6 | | | 1 | KPX6 |
| B4 | Keypad-I/O X7 | Genio7 | | | 1 | KPX7 |
| E5 | Keypad-I/O Y0 | Genio8 | | | 1 | KPY0 |
| C4 | Keypad-I/O Y1 | Genio9 | | | 1 | KPY1 |
| D5 | Keypad-I/O Y2 | Genio10 | | | 1 | KPY2 |
| B5 | Keypad-I/O Y3 | Genio11 | | | 1 | KPY3 |
| B2 | Keypad-I/O Y4 | Genio12 | | | 1 | KPY4 |
| A1 | Keypad-I/O Y5 | Genio13 | | | 1 | KPY5 |
| B1 | Keypad-I/O Y6 | Genio14 | | | 1 | KPY6 |
| C1 | Keypad-I/O Y7 | Genio15 | | | 1 | KPY7 |
| D4 | Keypad-I/O Y8 | Genio16 | ClockOut | PWM2 | 1 | KPY8 |
| | | | | | | PWM2 |
| E4 | Keypad-I/O Y9 | Genio17 | | PWM1 | 1 | KPY9 |
| | | | | | | PWM1 |
| D3 | Keypad-I/O Y10 | Genio18 | | PWM0 | 1 | KPY10 |
| | | | | | | PWM0 |
| E3 | Interrupt | Keypad-I/O Y11 | Genio19 | PWM2 | 1 | IRQN |
| | | | | | | KPY11 |
| | | | | | | PWM2 |
| C3 | Ground | | | | 1 | GND |
| | TOTAL | | | | 25 | |

PIN CONFIGURATION AFTER RESET

Upon power-up or RESET the LM8325-1 will have defined states on all pins. provides a comprehensive overview on the states of all functional pins.

Pin Configuration after Reset

| Pins | Pin States |
|---|--|
| KPX0 KPX1 KPX2 KPX3 KPX4 KPX5 KPX6 KPX7 | Full Buffer mode with an on-chip pull up resistor enabled. |
| KPY0 KPYP1 KPYP2 KPYP3 KPYP4 KPYP5 KPYP6 KPYP7 KPY8 / PWM2 KPY9 / PWM1 KPY10 / PWM0 | Full Buffer mode with an on-chip pull down resistor enabled. |
| KPY11 / IRQN / PWM2 | Open Drain mode with no pull resistor enabled, driven low (IRQN). ⁽¹⁾ |
| SCL SDA | Open Drain mode with no pull resistor enabled. |

(1) The IRQN is driven low after Power-On Reset due to PORIRQ signal. The value 0x01 must be written to the RSTINTCLR register (0x84) to release the IRQN pin.

TYPICAL APPLICATION SETUP

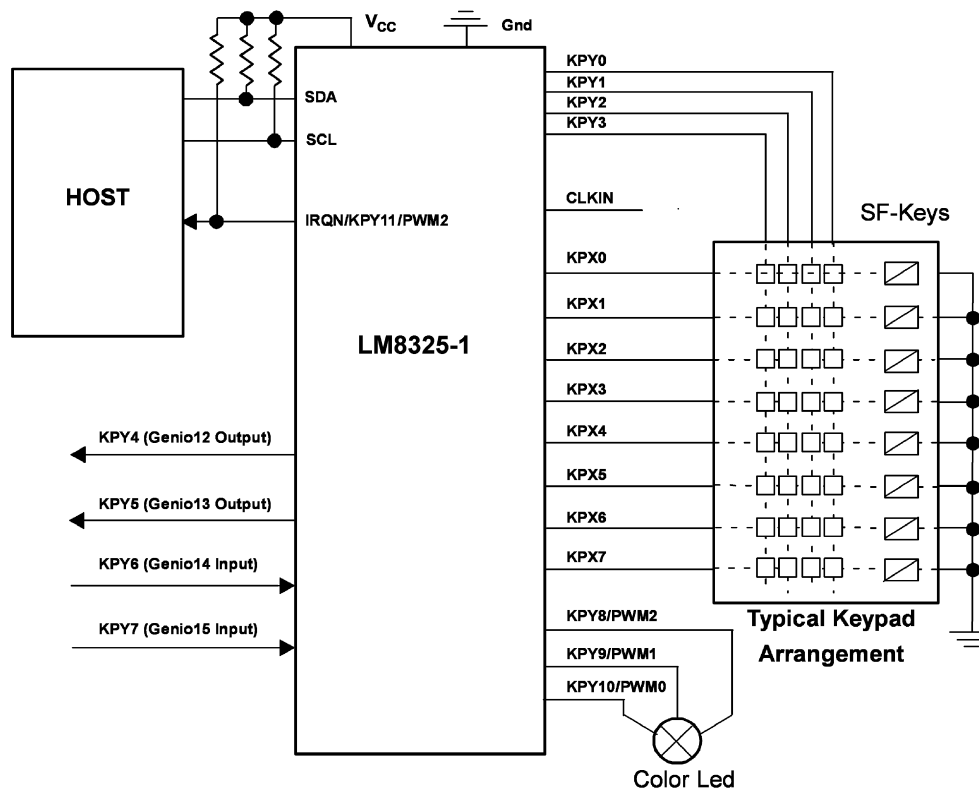


Figure 2. LM8325-1 in a Typical Setup with Standard Handset Keypad

FEATURES

The following features are supported with the application example shown in example above:

Hardware

Hardware

- 4 x 8 keys and 8 Special Function (SF) keys for 40 keys.
- ACCESS.bus interface for communication with a host device.
 - - communication speeds supported are: 100 kHz and 400 kHz fast mode of operation.
- Interrupt signal (IRQN) to indicate any keypad or hardware interrupt events to the host.
- Sophisticated PWM function block with 3 independent channels to control color LED.
- External clock input for accurate PWM clock (not used).
- Two host programmable dedicated general-purpose output pins (GPIOs) supporting IO-expansion capabilities for host device.
- Two host programmable dedicated general-purpose input pins with wake-up supporting IO-expansion capabilities for host device.

Communication Layer

- Versatile register-based command integration supported from on-chip command interpreter.
- Keypad event storage.
- Individual PWM script file storage and execution control for 3 PWM channels.

HALT MODE

HALT MODE DESCRIPTION

The fully static architecture of the LM8325-1 allows stopping the internal RC clock in Halt mode, which reduces power consumption to the minimum level. Figure 3 shows the current in Halt mode at the maximum VCC (1.98V) from 25°C to +85°C.

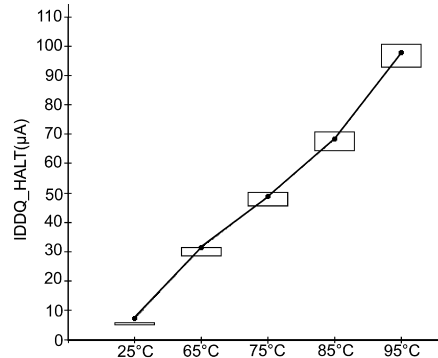


Figure 3. Halt Current vs. Temperature at 1.98V

Halt mode is entered when no key-press event, key-release event, or ACCESS.bus activity is detected for a certain period of time (by default, 1020 milliseconds). The mechanism for entering Halt mode is always enabled in hardware, but the host can program the period of inactivity which triggers entry into Halt mode using the autosleep function. (See Table 49.)

ACCESS.BUS ACTIVITY

When the LM8325-1 is in Halt mode, only activity on the ACCESS.bus interface that matches the LM8325-1 Slave Address will cause the LM8325-1 to exit from Halt mode. However, the LM8325-1 will not be able to acknowledge the first bus cycle immediately following wake-up from Halt mode. It will respond with a negative acknowledgement, and the host should then repeat the cycle. A peripheral that is continuously active can share the bus since this activity will not prevent the LM8325-1 from entering Halt mode.

LM8325-1 PROGRAMMING INTERFACE

The LM8325-1 operation is controlled from a host device by a complete register set, accessed via the I²C-compatible ACCESS.bus interface. The ACCESS.bus communication is based on a READ/WRITE structure, following the I²C transmission protocol. All functions can be controlled by configuring one or multiple registers. Please refer to [LM8325-1 REGISTER SET](#) for the complete register set.

ACCESS.BUS COMMUNICATION

Figure 4 shows a typical read cycle initiated by the host.)

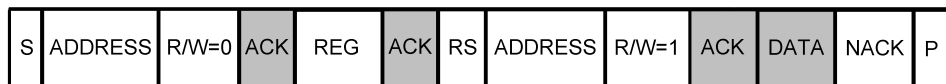


Figure 4. Master/Slave Serial Communication (Host to LM8325-1)

Table 1. Definition of Terms used in Serial Command Example

| Term | Bits | Description |
|---------|------|---|
| S | | START Condition (always generated from the master device) |
| ADDRESS | 7 | Slave address of LM8325-1 sent from the host |

Table 1. Definition of Terms used in Serial Command Example (continued)

| Term | Bits | Description |
|------|------|---|
| R/W | 1 | This bit determines if the following data transfer is from master to slave (data write) or from slave to master (data read). 0: Write 1: Read |
| ACK | 1 | An acknowledge bit is mandatory and must be appended on each byte transfer. The Acknowledge status is actually provided from the slave and indicates to the master, that the byte transfer was successful. |
| REG | 8 | The first byte after sending the slave address is the REGISTER byte which contains the physical address the host wants to read from or write to. |
| RS | | Repeated START condition |
| DATA | 8 | The DATA field contains information to be stored into a register or information read from a register. |
| NACK | 1 | Not Acknowledge Bit. The Not Acknowledge status is assigned from the Master receiving data from a slave. The NACK status will actually be assigned from the master in order to signal the end of a communication cycle transfer |
| P | | STOP condition (always generated from the master device) |

All actions associated with the non-shaded boxes in [Figure 4](#) are controlled from the master (host) device.

All actions associated with the shaded boxes in [Figure 4](#) are controlled from the slave (LM8325-1) device.

The master device can send subsequent REGISTER addresses separated by Repeated START conditions. A STOP condition must be set from the master at the very end of a communication cycle.

It is recommended to use Repeated START conditions in multi-Master systems when sending subsequent REGISTER addresses. This technique will make sure that the master device communicating with the LM8325-1 will not lose bus arbitration.

Starting a Communication Cycle

There are two reasons for the host device to start communication to the LM8325-1:

1. The LM8325-1 device has set the IRQN line low in order to signal a key - event or any other condition which initializes a hardware interrupt from LM8325-1 to the host.
2. The host device wants to set a GENIO port, read from a GENIO port, configure a GENIO port, and read the status from a register or initialize any other function which is supported from the LM8325-1. In case a GENIO shall be read it will be most likely, that the LM8325-1 device will be residing in “sleep mode”. In this mode the system clock will be off to establish the lowest possible current consumption. If the host device starts the communication under this condition the LM8325-1 device will not be able to acknowledge the first attempt of sending the slave address. The LM8325-1 will wake up because of the START condition but it can’t establish the internal timing to scan the first byte received. The master device must therefore apply a second attempt to start the communication with the LM8325-1 device.

Communication Initialized from Host (Restart from Sleep Mode)

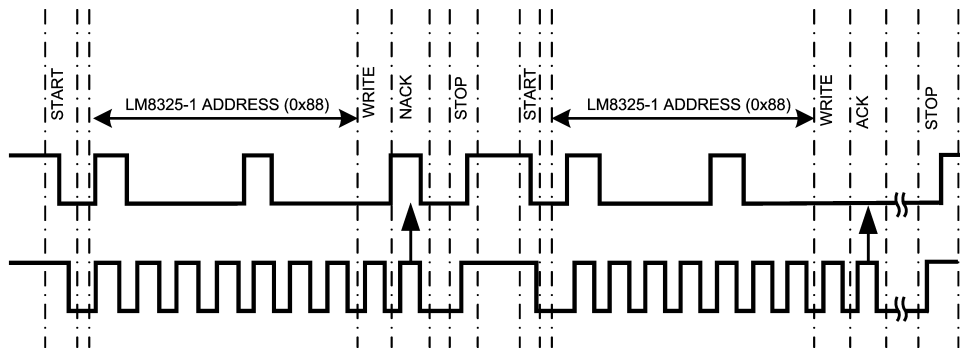


Figure 5. Host Starts Communication While LM8325-1 is in Sleep Mode

- In the timing diagram shown in [Figure 5](#) the LM8325-1 resides in sleep mode. Since the LM8325-1 device can’t acknowledge the slave address the host must generate a STOP condition followed by a second START condition.

- On the second attempt the slave address is being acknowledged from the LM8325-1 device because it is in active mode now.
- The host can send different WRITE and/or READ commands subsequently after each other.
- The host must finally free the bus by generating a STOP condition.

ACCESS.Bus Communication Flow

The LM8325-1 will only be driven in slave mode. The maximum communication speed supported is Fast Mode (FS) which is 400 kHz. The device can be heavily loaded as it is processing different kind of events caused from the human interface and the host device. In such cases the LM8325-1 may temporarily be unable to accept new commands and data sent from the host device.

NOTE

“It is a legitimate measure of the slave device to hold SCL line low in such cases in order to force the master device into a waiting state!. It is therefore the obligation of the host device to detect such cases. Typically there is a control bit set in the master device indicating the Busy status of the bus. As soon as the SCL line is released the host can continue sending commands and data.”

Further Remarks:

- In systems with multiple masters it is recommended to separate commands with Repeat START conditions rather than sending a STOP - and another START - condition to communicate with the LM8325-1 device.
- Delays enforced by the LM8325-1 during very busy phases of operation should typically not exceed a duration of 100 usec.
- Normally the LM8325-1 will clock stretch after the acknowledge bit is transmitted; however, there are some conditions where the LM8325-1 will clock stretch between the SDA Start bit and the first rising edge of SCL.

Auto Increment

In order to improve multi-byte register access, the LM8325-1 supports the auto increment of the address pointer.

A typical protocol access sequence to the LM8325-1 starts with the I²C-compatible ACCESS.bus address, followed by REG, the register to access (see [Figure 4](#)). After a REPEATED START condition the host reads/writes a data byte from/to this address location. If more than one byte is transmitted, the LM8325-1 automatically increments the address pointer for each data byte by 1. The address pointer keeps the status until the STOP condition is received.

The LM8325-1 always uses auto increments unless otherwise noted.

Please refer to [Table 3](#) and [Table 3](#) for the typical ACCESS.bus flow of reading and writing multiple data bytes.

Reserved Registers and Bits

The LM8325-1 includes reserved registers for future implementation options. Please use value 0 on a write to all reserved register bits.

Global Call Reset

The LM8325-1 supports the Global Call Reset as defined in the I²C Specification, which can be used by the host to reset all devices connected to interface. The Global call reset is a single byte ACCESS.bus/I²C write of data byte 0x06 to slave address 0x00.

The Global Call Reset changes the I²C-compatible ACCESS.bus Slave address of the LM8325-1 back to its default value of 0x88.

Table 2. Multi-Byte Write with Auto Increment

| Step | Master/Slave | I ² C Com. | Value | Address Pointer | Comment |
|------|--------------|-----------------------|-------|-----------------|--|
| 1 | M | S | | | START condition |
| 2 | M | ADDR. | 0x88 | | I ² C-compatible ACCESS.bus Address |
| 3 | M | R/W | 0 | | Write |
| 4 | S | ACK | | | Acknowledge |
| 5 | M | REG | 0xAA | 0xAA | Register Address, used as Address Pointer |
| 6 | S | ACK | | 0xAA | Acknowledge |
| 7 | M | DATA | 0x01 | 0xAA | Write Data to Address in Pointer |
| 8 | S | ACK | 0 | 0xAB | Acknowledge, Address pointer incremented |
| 9 | M | DATA | 0x05 | 0xAB | Write Data to address 0xAB |
| 10 | S | ACK | 0 | 0xAC | Acknowledge, Address pointer incremented |
| 11 | M | P | | | STOP condition |

Table 3. Multi-Byte Read with Auto Increment

| Step | Master/Slave | I ² C Com. | Value | Address Pointer | Comment |
|------|--------------|-----------------------|-------|-----------------|--|
| 1 | M | S | | | START condition |
| 2 | M | ADDR. | 0x88 | | I ² C-compatible ACCESS.bus Address |
| 3 | M | R/W | 0 | | Write |
| 4 | S | ACK | | | Acknowledge |
| 5 | M | REG | 0xAA | 0xAA | Register Address, used as Address pointer |
| 6 | S | ACK | | 0xAA | Acknowledge |
| 7 | M | RS | | 0xAA | Repeated Start |
| 8 | M | ADDR. | 0x88 | 0xAA | I ² C-compatible ACCESS.bus Address |
| 9 | M | R/W | 1 | | Read |
| 10 | S | ACK | 0 | 0xAA | Acknowledge |
| 11 | S | DATA | 0x01 | 0xAA | Read Data from Address in Pointer |
| 12 | M | ACK | 0 | 0xAB | Acknowledge, Address Pointer incremented |
| 13 | S | DATA | 0x05 | 0xAB | Read Data from Address in Pointer |
| 14 | M | NACK | 0 | 0xAC | No Acknowledge, stops transmission |
| 15 | M | P | | | STOP condition |

KEYSCAN OPERATION

KEYSCAN INITIALIZATION

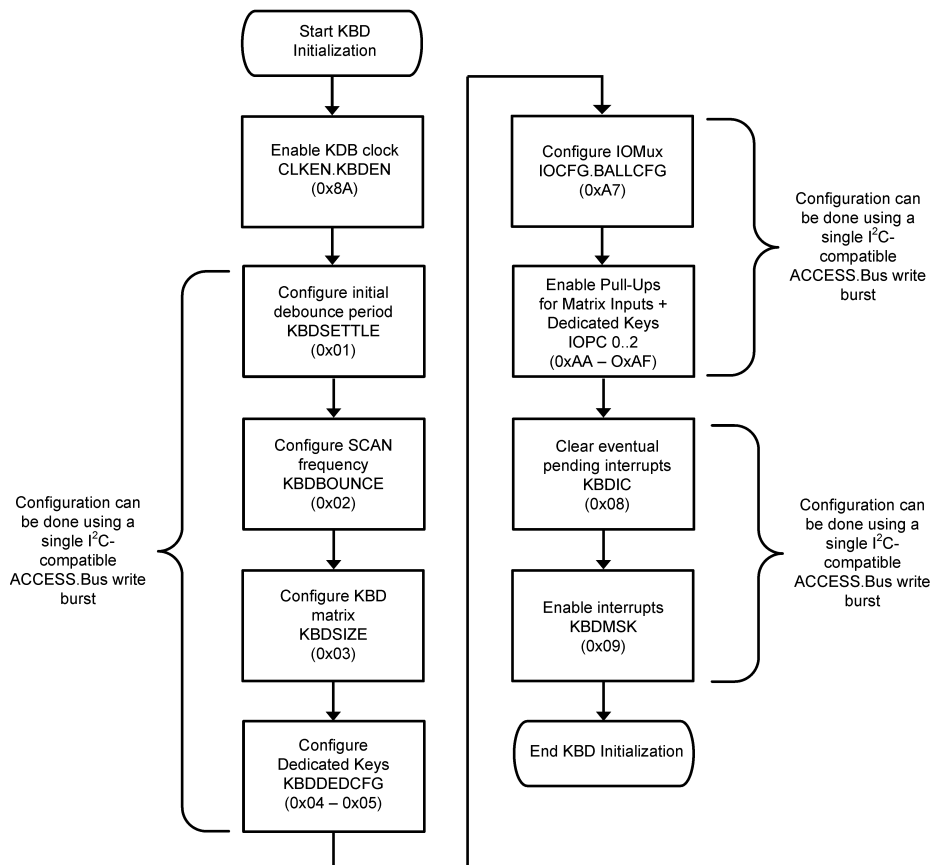


Figure 6. Keyscan Initialization

KEYSCAN INITIALIZATION EXAMPLE

Table 4 shows all the LM8325-1 register configurations to initialize keyscan:

- Keypad matrix configuration is 8 rows x 8 columns.

Table 4. Keyscan Initialization Example

| Register name | adress | Access Type | Value | Comment |
|---------------|--------|-------------|--------|---|
| CLKEN | 0x8A | byte | 0x01 | enable keyscan clock |
| KBDSETTLE | 0x01 | byte | 0x80 | set the keyscan settle time to 12 msec |
| KBDBOUNCE | 0x02 | byte | 0x80 | set the keyscan debounce time to 12 msec |
| KBDSIZE | 0x03 | byte | 0x88 | set the keyscan matrix size to 8 rows x 8 columns |
| KBDEDCFG | 0x04 | word | 0xFC3F | configure KPX[7:2] and KPY[7:2] pins as keyboard matrix |
| IOCFG | 0xA7 | byte | 0xF8 | write default value to enable all pins as keyboard matrix |
| IOPC0 | 0xAA | word | 0xAAAA | configure pull-up resistors for KPX[7:0] |
| IOPC1 | 0xAC | word | 0x5555 | configure pull-down resistors for KPY[7:0] |
| KBDIC | 0x08 | byte | 0x03 | clear any pending interrupts |
| KBDMSK | 0x09 | byte | 0x03 | enable keyboard interrupts |

KEYSCAN PROCESS

The LM8325-1 keyscan functionality is based on a specific scanning procedure performed in a 4ms interval. On each scan all assigned key matrix pins are evaluated for state changes.

In case a key event has been identified, the event is stored in the key event FIFO, accessible via the EVTCODE register. A key event can either be a key press or a key release. In addition, key presses are also stored in the KBDCODE[3:0] registers. As soon as the EVTCODE FIFO includes a event, the device sets the RAW keyboard event interrupt REVTINT. The RSINT interrupt is set anytime the keyboard status has changed.

Depending on the interrupt masking for the keyboard events (KBDMSK) and the masked interrupt handling (KBDMIS), the pin IRQN/KPY11/PWM2 will follow the IRQST.KBDIRQ status, which is set as soon as one interrupt in KBDRIS is set.

Figure 7 shows the basic flow of a scanning process and which registers are affected.

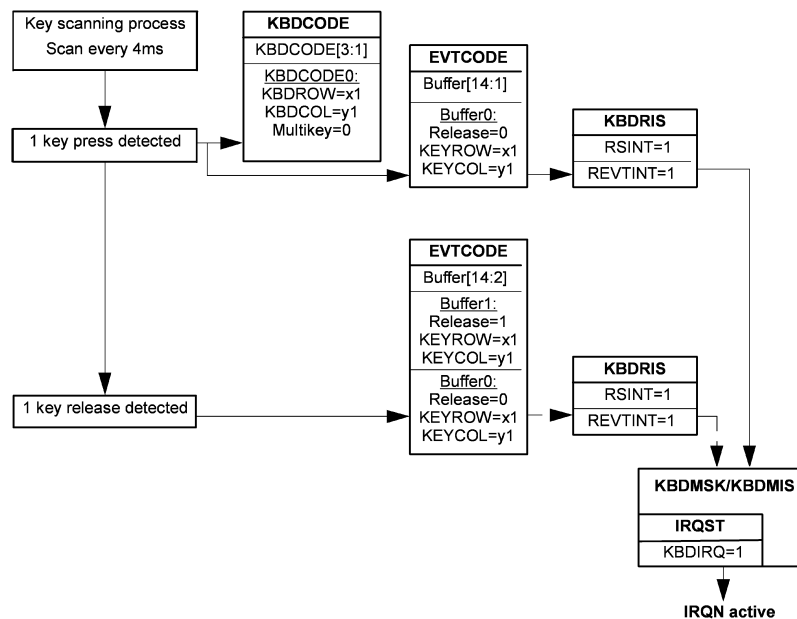


Figure 7. Example Keyscan Operation for 1 Key Press and Release

READING KEYSKAN STATUS BY THE HOST

In order to keep track of the keyscan status, the host either needs to regularly poll the EVTCODE register or needs to react on the Interrupt signalled by the IRQN/KPY11/PWM2 pin, in case the ball is configured for interrupt functionality. (See [GPIO FEATURE CONFIGURATION](#)).

Figure 8 gives an example on which registers to read to get the keyboard events from the LM8325-1 and how they influence the interrupt event registers. The example is based on the assumption that the LM8325-1 has indicated the keyboard event by the IRQN/KPY11/PWM2 pin.

Since the interrupt pin has various sources, the host first checks the IRQST register for the interrupt source. If KBDIRQ is set, the host can check the KBDMIS register to define the exact interrupt source. KBDMIS contains the masked status of KBDRIS and reflects the source for raising the interrupt pin. The interrupt mask is defined by KBDMSK. The complete status of all pending keyboard interrupts is available in the raw interrupt register KBDRIS.

After evaluating the interrupt source the host starts reading the EVTCODE or KBDCODE register. In this example the host first reads the KBDCODE to get possible key press events and afterwards reads the complete event list by reading the EVTCODE register until all events are captured (0x7F indicates end of buffer).

Reading KBDCODE clears the RSINT interrupt bit if all keyboards events are emptied. In the same way, REVTINT is cleared in case the EVTCODE FIFO reaches its empty state on read.

The event buffer content and the REVTINT and RELINT (lost event) interrupt bits are also cleared if the KBDIC.EVTIC bit is set.

Interrupt bits in the masked interrupt register KBDMIS follow the masked KBDRIS status.

In order to support efficient Multi-byte reads from EVTCODE, the autoincrement feature is turned off for this register. Therefore the host can continuously read the complete EVTCODE buffer by sending one command.

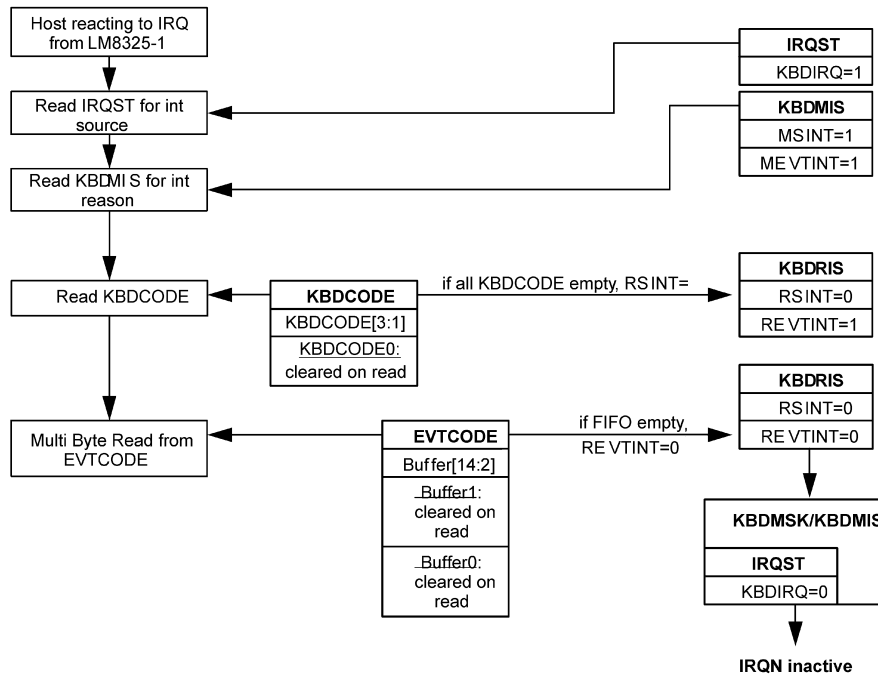


Figure 8. Example Host Reacting to Interrupt for Keypad Event

MULTIPLE KEY PRESSES

The LM8325-1 supports up to four simultaneous key presses. Any time a single key is pressed KBDCODE0 is set with the appropriate key code. If a second key is pressed, the key is stored in KBDCODE1 and the MULTIKEY flag of KBDCODE0 is set. Additional key presses are stored in KBDCODE2 and KBDCODE3 accordingly. The four registers signal the last multi key press events.

All events are stored in parallel in the EVTCODE register for the complete set of events.

All KBDCODE[3:0] registers are cleared on read.

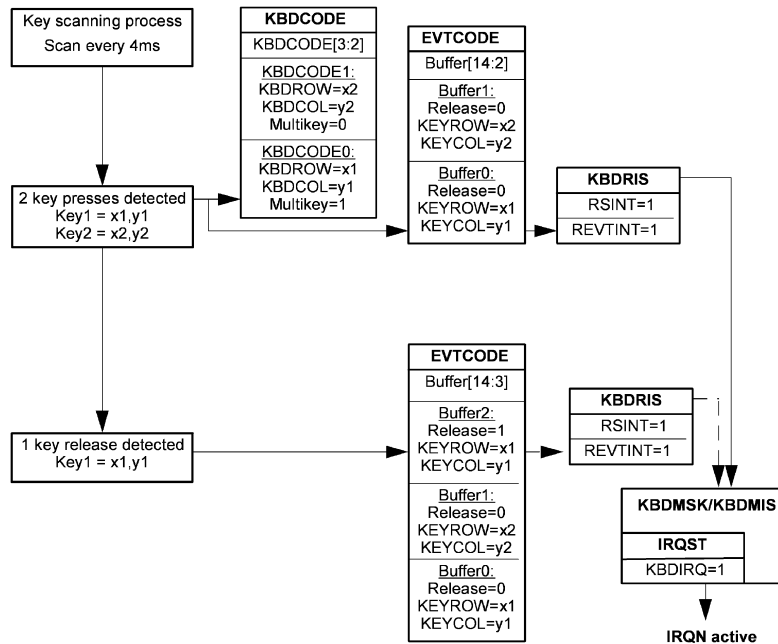


Figure 9. Example Keyscan Operation for 2 Key Press Events and 1 Key Release Event

PWM TIMER

The LM8325-1 supports a timer module dedicated to smooth LED control techniques (lighting controls).

The PWM timer module consists of three independent timer units of which each can generate a PWM output with a fixed period and automatically incrementing or decrementing variable duty cycle. The timer units are all clocked with a slow (32.768 kHz) clock whereas the interface operates with the main system clock.

OVERVIEW OF PWM FEATURES

- Each PWM can establish fixed — or variable — duty-cycle signal sequences on its output.
- Each PWM can trigger execution of any pre-programmed task on another PWM channel.
- The execution of any pre-programmed task is self-sustaining and does not require further interaction from the host.
- 64-byte script buffer for each PWM for up to 32 consecutive instructions.
- Direct addressing within script buffer to support multiple PWM tasks in one buffer.

OVERVIEW ON PWM SCRIPT COMMANDS

The commands listed in [Table 5](#) are dedicated to the slow PWM timers.

NOTE

The PWM Script commands are not part of the command set supported by the LM8325-1 command interpreter. These commands must be transferred from the host with help of the register-based command set.

Table 5. PWM Script Commands

| Command | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----------------|----|----------|----------|-------------|-----|----|---|----------|-------------|------------|---|---|---|---|---|---|--|
| RAMP | 0 | PRESCALE | STEPTIME | | | | | SIGN | INCREMENT | | | | | | | | |
| SET_PWM | 0 | 1 | 0 | | | | | PWMVALUE | | | | | | | | | |
| GO_TO_ START | 0 | | | | | | | | | | | | | | | | |
| BRANCH | 1 | 0 | 1 | LOOPCOUNT | | | | | ADDR | STEPNUMBER | | | | | | | |
| END | 1 | 1 | 0 | 1 | INT | X | | | | | | | | | | | |
| TRIGGER | 1 | 1 | 1 | WAITTRIGGER | | | | | SENDTRIGGER | | | | | | | 0 | |

RAMP COMMAND

A RAMP command will vary the duty cycle of a PWM output in either direction (up or down). The INCREMENT field specifies the amount of steps for the RAMP. The maximum amount of steps which can be executed with one RAMP Command is 126 which is equivalent to 50%. The SIGN bit field determines the direction of a RAMP (up or down). The STEPTIME field and the PRESCALE bit determine the duration of one step. Based on a 32.768 kHz clock, the minimum time resulting from these options would be 0.49 milliseconds and the maximum time for one step would be 1 second.

Table 6. RAMP Command Bit and Building Fields

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|----|----------|----------|----|----|----|---|------|-----------|---|---|---|---|---|---|---|--|
| 0 | PRESCALE | STEPTIME | | | | | SIGN | INCREMENT | | | | | | | | |

Table 7. Description of Bit and Building Fields of the RAMP Command

| Bit or Field | Value | Description |
|--------------|---------|--|
| PRESCALE | 0 | Divide the 32.768 kHz clock by 16 |
| | 1 | Divide the 32.768 kHz clock by 512 |
| STEPTIME | 1 - 63 | Number of prescaled clock cycles per step |
| SIGN | 0 | Increment RAMPcounter |
| | 1 | Decrement RAMPcounter |
| INCREMENT | 0 - 126 | Number of steps executed by this instruction; a value of 0 functions as a WAIT determined by STEPTIME. |

SET_PWM COMMAND

The SET_PWM command will set the starting duty cycle MIN SCALE or FULL SCALE (0% or 100%). A RAMP command following the SET_PWM command will finally establish the desired duty cycle on the PWM output.

Table 8. SET_PWM Command Bit and Building Fields

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|-----------|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | DUTYCYCLE | | | | | | | |

Table 9. Description of Bit and Building Fields of the SET_PWM Command

| Bit or Field | Value | Description |
|--------------|-------|---------------------|
| DUTYCYCLE | 0 | Duty cycle is 0%. |
| | 255 | Duty cycle is 100%. |

GO_TO_START COMMAND

The GO_TO_START command jumps to the first command in the script command file.

Table 10. GO_TO_START Command Bit and Building Fields

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

BRANCH COMMAND

The BRANCH command jumps to the specified command in the script command file. The branch is executed with either absolute or relative addressing. In addition, the command gives the option of looping for a specified number of repetitions.

NOTE

Nested loops are not allowed.

Table 11. BRANCH Command Bit and Building Fields

| | | | | | | | | | | | | | | | | |
|----|----|----|-----------|----|----|---|---|------|------------|---|---|---|---|---|---|--|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 1 | 0 | 1 | LOOPCOUNT | | | | | ADDR | STEPNUMBER | | | | | | | |

Table 12. Description of Bit and Building Fields of the BRANCH Command

| Bit or Field | Value | Description |
|--------------|--------|--|
| LOOPCOUNT | 0 | Loop until a STOP PWM SCRIPT command is issued by the host. |
| | 1 - 63 | Number of loops to perform. |
| ADDR | 0 | Absolute addressing |
| | 1 | Relative addressing |
| STEPNUMBER | 0 - 63 | Depending on ADDR: ADDR = 0; Addr to jump to ADDR = 1 - Number of backward steps |

TRIGGER COMMAND

Triggers are used to synchronize operations between PWM channels. A TRIGGER command that sends a trigger takes sixteen 32.768 kHz clock cycles, and a command that waits for a trigger takes at least sixteen 32.768 kHz clock cycles.

A TRIGGER command that waits for a trigger (or triggers) will stall script execution until the trigger conditions are satisfied. On trigger it will clear the trigger(s) and continue to the next command.

When a trigger is sent, it is stored by the receiving channel and can only be cleared when the receiving channel executes a TRIGGER command that waits for the trigger.

Table 13. TRIGGER Command Bit and Building Fields

| | | | | | | | | | | | | | | | | |
|----|----|----|-------------|----|----|---|---|-------------|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 1 | 1 | 1 | WAITTRIGGER | | | | | SENDTRIGGER | | | | | | | | 0 |

Table 14. Description of Bit and Building Fields

| Field | Value | Description |
|-------------|--------|---------------------------------|
| WAITTRIGGER | 000xx1 | Wait for trigger from channel 0 |
| | 000x1x | Wait for trigger from channel 1 |
| | 0001xx | Wait for trigger from channel 2 |
| SENDTRIGGER | 000xx1 | Send trigger to channel 0 |
| | 000x1x | Send trigger to channel 1 |
| | 0001xx | Send trigger to channel 2 |

END COMMAND

The END command terminates script execution. It will only assert an interrupt to the host if the INT bit is set to “1”.

When the END command is executed, the PWM output will be set to the level defined by PWMCFG.PWMPOL for this channel. Also, the script counter is reset back to the beginning of the script command buffer.

NOTE

If a PWM channel is waiting for the trigger (last executed command was "TRIGGER") and the script execution is halted then the "END" command can't be executed because the previous command is still pending. This is an exception - in this case the IRQ signal will not be asserted.

Table 15. END Command Bit and Building Fields

| | | | | | | | | | | | | | | | |
|----|----|----|----|-----|----|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 1 | 0 | 1 | INT | | | | | | 0 | | | | | |

Table 16. Description of Bit and Building Fields of the END Command

| Field | Value | Description |
|-------|-------|---|
| INT | 0 | No interrupt will be sent. |
| | 1 | Set TIMRIS.CDIRQ for this PWM channel to notify that program has ended. |

LM8325-1 REGISTER SET**KEYBOARD REGISTERS AND KEYBOARD CONTROL**

Keyboard selection and control registers are mapped in the address range from 0x01 to 0x10. This paragraph describes the functions of the associated registers down to the bit level.

KBDSETTLE - Keypad Settle Time Register**Table 17. KBDSETTLE - Keypad Settle Time Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| KBDSETTLE | 0x01 | R/W | Initial time for keys to settle, before the key-scan process is started. |
| Bit - Name | Bit | Default | Bit Function |
| WAIT[7:0] | 7:0 | 0x80 | The default value 0x80 : 0xBF sets a time target of 12 msec Further time targets are as follows: 0xC0 - 0xFF: 16 msec 0x80 - 0xBF: 12 msec 0x40 - 0x7F: 8 msec 0x01 - 0x3F: 4 msec 0x00 : no settle time |

KBDBOUNCE - Debounce Time Register**Table 18. KBDBOUNCE - Debounce Time Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| KBDBOUNCE | 0x02 | R/W | Time between first detection of key and final sampling of key |
| Bit - Name | Bit | Default | Bit Function |

Table 18. KBDBOUNCE - Debounce Time Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|---|
| WAIT[7:0] | 7:0 | 0x80 | The default value 0x80 : 0xBF sets a time target of 12 msec Further time targets are as follows: 0xC0 - 0xFF: 16 msec 0x80 - 0xBF: 12 msec 0x40 - 0x7F: 8 msec 0x01 - 0x3F: 4 msec 0x00: no debouncing time |

KBDSIZE - Set Keypad Size Register
Table 19. KBDSIZE - Set Keypad Size Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| KBDSIZE | 0x03 | R/W | Defines the physical keyboard matrix size |
| Bit - Name | Bit | Default | Bit Function |
| ROWSIZE[3:0] | 7:4 | 0x2 | Number of rows in the keyboard matrix 0x0: free all rows to become GPIO, KPX[1:0] used as dedicated key inputs if scanning is enabled by CLKEN.KBEN 0x1: (illegal value) 0x2 - 0x8: Number of rows in the matrix |
| COLSIZE[3:0] | 3:0 | 0x2 | Number of columns in the keyboard matrix 0x0: free all rows to become GPIO, KPY[1:0] used as dedicated key inputs if scanning is enabled by CLKEN.KBEN 0x1: (illegal value) 0x2 - 0xC: Number of columns in the matrix |

KBDEDCFG - Dedicated Key Register
Table 20. KBDEDCFG - Dedicated Key Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| KBDEDCFG | 0x04 | R/W | Defines if a key is used as a standard keyboard/GPIO pin or whether it is used as dedicated key input. |
| Bit - Name | Bit | Default | Bit Function |
| ROW[7:2] | 15:10 | 0x3F | Each bit in ROW [7:2] corresponds to ball KPX7 : KPX2. Bit=0: the dedicated key function applies. Bit=1: no dedicated key function is selected. The standard GPIO functionality applies according to register IOCFG or defined keyboard matrix. |
| COL[11:10] | 9:8 | 0x03 | Each bit in COL [11:10] corresponds to ball KPY11 : KPY10. Bit=0: the dedicated key function applies. Bit=1: no dedicated key function is selected. The standard GPIO functionality applies according to register IOCFG or defined keyboard matrix. |
| COL[9:2] | 7:0 | 0xFF | Each bit in COL [9:2] corresponds to ball KPY9 : KPY2 and can be configured individually. Bit=0: the dedicated key function applies. Bit=1: no dedicated key function is selected. The standard GPIO functionality applies according to register IOCFG or defined keyboard matrix. |

KBDRIS - Keyboard Raw Interrupt Status Register**Table 21. KBDRIS - Keyboard Raw Interrupt Status Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| KBDRIS | 0x06 | R | Returns the status of stored keyboard interrupts. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| RELINT | 3 | 0x0 | Raw event lost interrupt. More than 8 keyboard events have been detected and caused the event buffer to overflow. This bit is cleared by setting bit EVTIC of the KBDIC register. |
| REVTINT | 2 | 0x0 | Raw keyboard event interrupt. At least one key press or key release is in the keyboard event buffer. Reading from EVTICODE until the buffer is empty will clear this interrupt. |
| RKLINT | 1 | 0x0 | Raw key lost interrupt indicates a lost key-code. This interrupt is asserted 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. |
| RSINT | 0 | 0x0 | Raw scan interrupt. Interrupt generated after keyboard scan, if the keyboard status has changed. |

KBDMIS - Keypad Masked Interrupt Status Register**Table 22. KBDMIS - Keypad Masked Interrupt Status Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| KBDMIS | 0x07 | R | Returns the status on masked keyboard interrupts after masking with the KBDMSK register. |
| Bit - Name | Bit | Default | Bit Functions |
| (reserved) | 7:4 | | (reserved) |
| MELINT | 3 | 0x0 | Masked event lost interrupt. More than 8 keyboard events have been detected and caused the event buffer to overflow. This bit is cleared by setting bit EVTIC of the KBDIC register. |
| MEVTINT | 2 | 0x0 | Masked keyboard event interrupt. At least one key press or key release is in the keyboard event buffer. Reading from EVTICODE until the buffer is empty will clear this interrupt. |
| MKLINT | 1 | 0x0 | Masked key lost interrupt. Indicates a lost key-code. This interrupt is asserted 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. |
| MSINT | 0 | 0x0 | Masked scan interrupt. Interrupt generated after keyboard scan, if the keyboard status has changed, after masking process. |

KBDIC - Keypad Interrupt Clear Register**Table 23. KBDIC - Keypad Interrupt Clear Register**

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|--|
| KBDIC | 0x08 | W | Setting these bits clears Keypad active Interrupts |
| Bit - Name | Bit | Default | Bit Function |

Table 23. KBDIC - Keypad Interrupt Clear Register (continued)

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|---|
| SFOFF | 7 | | Switches off scanning of special function (SF) keys, when keyboard has no special function layout. 0: keyboard layout and SF keys are scanned 1: only keyboard layout is scanned, SF keys are not scanned |
| (reserved) | 6:2 | | (reserved) |
| EVTIC | 1 | | Clear event buffer and corresponding interrupts REVTINT and RELINT by writing a 1 to this bit position |
| KBDIC | 0 | | Clear RSINT and RKLINT interrupt bits by writing a 1 to this bit position. |

KBDMSK - Keypad Interrupt Mask Register
Table 24. KBDMSK - Keypad Interrupt Mask Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| KBDMSK | 0x09 | R/W | Configures masking of keyboard interrupts. Masked interrupts do not trigger an event on the Interrupt output. In case the interrupt processes registers KBDCODE[3:0], MSKELINT and MSKEINT should be set to 1. When the Event FIFO is processed, MSKLINT and MSKSINT should be set. For keyboard polling operations, all bits should be set and the polling operation consists of reading out the EVTCODE. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| MSKELINT | 3 | 0x0 | 0: keyboard event lost interrupt RELINT triggers IRQ line 1: keyboard event lost interrupt RELINT is masked |
| MSKEINT | 2 | 0x0 | 0: keyboard event interrupt REVINT triggers IRQ line 1: keyboard event interrupt REVINT is masked |
| MSKLINT | 1 | 0x1 | 0: keyboard lost interrupt RKLINT triggers IRQ line 1: keyboard lost interrupt RKLINT is masked |
| MSKSINT | 0 | 0x1 | 0: keyboard status interrupt RSINT triggers IRQ line 1: keyboard status interrupt RSINT is masked |

KBDCODE0 - Keyboard Code Register 0

The key code detected by the keyboard scan can be read from the registers KBDCODE0: KBDCODE3. Up to 4 keys can be detected simultaneously. Each KBDCODE register includes a bit (MULTIKEY) indicating if another key has been detected.

NOTE

Reading out all key code registers (KBDCODE0 to KBDCODE3) will automatically reset the keyboard scan interrupt RSINT the same way as an active write access into bit KBDIC of the interrupt clear register does. Reading 0x7F from the KBDCODE0 register means that no key was pressed.

Table 25. KBDCODE0 - Keyboard Code Register 0

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|--|
| KBDCODE0 | 0x0B | R | Holds the row and column information of the first detected key |
| Bit - Name | Bit | Default | Bit Function |
| MULTIKEY | 7 | 0x0 | if this bit is 1 another key is available in KBDCODE1 register |
| KEYROW[2:0] | 6:4 | 0x7 | ROW index of detected key (0 to 7) |
| KEYCOL[3:0] | 3:0 | 0xF | Column index of detected (0 to 11, 12 for special function key). |

KBDCODE1 - Keyboard Code Register 1**Table 26. KBDCODE1 - Keyboard Code Register 1**

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|---|
| KBDCODE1 | 0x0C | R | Holds the row and column information of the second detected key |
| Bit - Name | Bit | Default | Bit Function |
| MULTIKEY | 7 | 0x0 | if this bit is 1 another key is available in KBDCODE2 register |
| KEYROW[2:0] | 6:4 | 0x7 | ROW index of detected key (0 to 7) |
| KEYCOL[3:0] | 3:0 | 0xF | Column index of detected key (0 to 11, 12 for special function key. |

KBDCODE2 - Keyboard Code Register 2**Table 27. KBDCODE2 - Keyboard Code Register 2**

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|---|
| KBDCODE2 | 0x0D | R | Holds the row and column information of the third detected key |
| Bit - Name | Bit | Default | Bit Function |
| MULTIKEY | 7 | 0x0 | if this bit is 1 another key is available in KBDCODE3 register |
| KEYROW[2:0] | 6:4 | 0x7 | ROW index of detected key (0 to 7) |
| KEYCOL[3:0] | 3:0 | 0xF | Column index of detected key (0 to 11, 12 for special function key. |

KBDCODE3 - Keyboard Code Register 3**Table 28. KBDCODE3 - Keyboard Code Register 3**

| Register - Name | Address | Default | Register Function |
|-----------------|---------|---------|---|
| KBDCODE3 | 0x0E | R | Holds the row and column information of the fourth detected key |
| Bit - Name | Bit | Default | Bit Function |
| MULTIKEY | 7 | 0x0 | if this bit is set to "1" then more than 4 keys are pressed simultaneously. |
| KEYROW[2:0] | 6:4 | 0x7 | ROW index of detected key (0 to 7) |
| KEYCOL[3:0] | 3:0 | 0xF | Column index of detected key (0 to 11, 12 for special function key. |

EVTCODE - Key Event Code Register**Table 29. EVTCODE - Key Event Code Register**

| Register - Name | Address | Default | Bit Function |
|-----------------|---------|---------|--|
| EVTCODE | 0x10 | R | With this register a FIFO buffer is addressed storing up to 15 consecutive events. Reading the value 0x7F from this address means that the FIFO buffer is empty. See further details below. NOTE: Auto increment is disabled on this register. Multi-byte read will always read from the same address. |
| Bit - Name | Bit | Default | Bit Function |
| RELEASE | 7 | 0x0 | This bit indicates, whether the keyboard event was a key press or a key release event. 0: key was pressed 1: key was released |
| KEYROW[2:0] | 6:4 | 0x7 | Row index of key that is pressed or released. |
| KEYCOL[3:0] | 3:0 | 0xF | Column index of key that is pressed (0...11, 12 for special function key) or released. |

PWM TIMER CONTROL REGISTERS

The LM8325-1 provides three host-programmable PWM outputs useful for smooth LED brightness modulation. All PWM timer control registers are mapped in the range from 0x60 to 0x7F. This paragraph describes the functions of the associated registers down to the bit level.

TIMCFGx - PWM Timer 0, 1 and 2 Configuration Registers

Table 30. TIMCFGx - PWM Timer 0, 1 and 2 Configuration Registers

| Register - Name | Address | Type | Register Function |
|-------------------------------|---------|---------|--|
| TIMCFG0 | 0x60 | R/W | This register configures interrupt masking and handles PWM start/stop control of the associated PWM channel. |
| TIMCFG1 | 0x68 | | |
| TIMCFG2 | 0x70 | | |
| Bit - Name (x = 0, 1 or 2) | Bit | Default | Bit Function |
| CYCIRQxMSK | 4 | 0x0 | Interrupt mask for PWM CYCIRQx (see register TIMRIS) 0: interrupt enabled 1: interrupt masked |
| (reserved) | 3:0 | 0x0 | (reserved) |

PWMCFGx - PWM Timer 0, 1 and 2 Configuration Control Registers

Table 31. PWMCFGx - PWM Timer 0, 1 and 2 Configuration Control Registers

| Register - Name | Address | Type | Register Function |
|-------------------------------|---------|---------|--|
| PWMCFG0 | 0x61 | R/W | This register defines interrupt masking and the output behavior for the associated PWM channel. PGE _x is used to start and stop the PWM script execution. PWMEN _x sets the PWM output to either reflect the generated pattern or the value configured in PWMPOL _x . |
| PWMCFG1 | 0x69 | | |
| PWMCFG2 | 0x71 | | |
| Bit - Name (x = 0, 1 or 2) | Bit | Default | Bit Function |
| CDIRQxMSK | 3 | 0x0 | Mask for CDIRQ _x 0: CDIRQ _x enabled 1: CDIRQ _x disabled/masked |
| PGE _x | 2 | 0x0 | Pattern Generator Enable. Start/Stop PWM command processing for this channel. Script execution is started always from beginning. 0: Pattern Generator disabled 1: Pattern Generator enabled |
| PWMEN _x | 1 | 0x0 | 0: PWM disabled. PWM timer output assumes value programmed in PWMPOL. 1: PWM enabled |
| PWMPOL _x | 0 | 0x0 | Off-state of PWM output, when PWMEN=0. 0: PWM off-state is low 1: PWM off-state is high |

TIMSCALx - PWM Timer 0, 1 and 2 Prescale Registers**Table 32. TIMSCALx - PWM Timer 0, 1 and 2 Prescale Registers**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| TIMSCAL0 | 0x62 | R/W | These registers determine the divider of the CLKIN external clock. The resulting clock is only used for PWM Generation. The value should only be changed while PWM is stopped. Since all 3 PWM channels use the same slow clock, TIMSCAL0 affects all 3 PWM channels. TIMSCAL1 and 2 are directly linked to TIMSCAL0. |
| TIMSCAL1 | 0x6A | | |
| TIMSCAL2 | 0x72 | | |
| Bit - Name | Bit | Default | Bit Function |
| SCAL[7:0] | 7:0 | 0x0 | CLKIN is divided by (SCAL+1). |

TIMSWRES - PWM Timer Software Reset Registers**Table 33. TIMSWRES - PWM Timer Software Reset Registers**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| TIMSWRES | 0x78 | W | Reset control on all PWM timers A reset forces the pattern generator to fetch the first pattern and stops it. Each reset stops all state-machines and timer. Patterns stored in the pattern configuration register remain unaffected. Interrupts on each timer are not cleared, they need to be cleared writing into register TIMIC |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:3 | | (reserved) |
| SWRES2 | 2 | | Software reset of timer 2. 0: no action 1: Software reset on timer 2, needs not to be written back to 0. |
| SWRES1 | 1 | | Software reset of timer 1. 0: no action 1: Software reset on timer 1, needs not to be written back to 0. |
| SWRES0 | 0 | | Software reset of timer 0. 0: no action 1: software reset on timer 0, needs not to be written back to 0. |

TIMRIS - PWM Timer Interrupt Status Register**Table 34. TIMRIS - PWM Timer Interrupt Status Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| TIMRIS | 0x7A | R | This register returns the raw interrupt status from the PMW timers 0,1 and 2. CYCIRQx - Interrupt from the timers when PWM cycle is complete (applies to the current PWM command residing in the active command register of a PWM block). CDIRQx - Interrupt from the pattern generator when PWM pattern code is complete (applies to a completed task residing in the script buffer of a PWM block). |
| Bit - Name | Bit | Default | Bit Functions |
| (reserved) | 7:6 | | (reserved) |
| CDIRQ2 | 5 | 0x0 | Raw interrupt status for CDIRQ timer2 0: no interrupt pending 1: unmasked interrupt generated |

Table 34. TIMRIS - PWM Timer Interrupt Status Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| CDIRQ1 | 4 | 0x0 | Raw interrupt status for CDIRQ timer1 0: no interrupt pending 1: unmasked interrupt generated |
| CDIRQ0 | 3 | 0x0 | Raw interrupt status for CDIRQ timer0 0: no interrupt pending 1: unmasked interrupt generated |
| CYCIRQ2 | 2 | 0x0 | Raw interrupt status for CYCIRQ timer2 0: no interrupt pending 1: unmasked interrupt generated |
| CYCIRQ1 | 1 | 0x0 | Raw interrupt status for CYCIRQ timer1 0: no interrupt pending 1: unmasked interrupt generated |
| CYCIRQ0 | 0 | 0x0 | Raw interrupt status for CYCIRQ timer0 0: no interrupt pending 1: unmasked interrupt generated |

TIMMIS - PWM Timer Masked Interrupt Status Register
Table 35. TIMMIS - PWM Timer Masked Interrupt Status Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| TIMMIS | 0x7B | R | This register returns the masked interrupt status from the PMW timers 0,1 and 2. The raw interrupt status (TIMRIS) is masked with the associated TIMCFGx.CYCIRQxMSK and PWMCFGx.CDIRQxMSK bits to get the masked interrupt status of this register. CYCIRQ - Interrupt from the timers when PWM cycle is complete (applies to the current PWM command residing in the active command register of a PWM block) CDIRQ - Interrupt from the pattern generator when PWM pattern code is complete (applies to a completed task residing in the script buffer of a PWM block) |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:6 | | (reserved) |
| CDIRQ2 | 5 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CDIRQ timer2. 0: no interrupt pending 1: interrupt generated |
| CDIRQ1 | 4 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CDIRQ timer1. 0: no interrupt pending 1: interrupt generated |
| CDIRQ0 | 3 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CDIRQ timer0. 0: no interrupt pending 1: interrupt generated |
| CYCIRQ2 | 2 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CYCIRQ timer2. 0: no interrupt pending 1: interrupt generated |

Table 35. TIMMIS - PWM Timer Masked Interrupt Status Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| CYCIRQ1 | 1 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CYCIRQ timer1. 0: no interrupt pending 1: interrupt generated |
| CYCIRQ0 | 0 | 0x0 | Interrupt after masking, indicates active contribution to the interrupt ball, when set. Status for CYCIRQ timer0. 0: no interrupt pending 1: interrupt generated |

TIMIC - PWM Timer Interrupt Clear Register**Table 36. TIMIC - PWM Timer Interrupt Clear Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| TIMIC | 0x7C | W | This register clears timer and pattern interrupts. CYCIRQ - Interrupt from the timers when PWM cycle is complete (applies to the current PWM command residing in the active command register of a PWM block). CDIRQ - Interrupt from the pattern generator when PWM pattern code is complete (applies to a completed task residing in the script buffer of a PWM block) |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:6 | | (reserved) |
| CDIRQ2 | 5 | | Clears interrupt CDIRQ timer2. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |
| CDIRQ1 | 4 | | Clears interrupt CDIRQ timer1. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |
| CDIRQ0 | 3 | | Clears interrupt CDIRQ timer0. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |
| CYCIRQ2 | 2 | | Clears interrupt CYCIRQ timer2. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |
| CYCIRQ1 | 1 | | Clears interrupt CYCIRQ timer1. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |
| CYCIRQ0 | 0 | | Clears interrupt CYCIRQ timer0. 0: no effect 1: interrupt is cleared. Does not need to be written back to 0 |

PWMWP - PWM Timer Pattern Pointer Register**Table 37. PWMWP - PWM Timer Pattern Pointer Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| PWMWP | 0x7D | R/W | Pointer to the pattern position inside the configuration register, which will be overwritten by the next write access to be PWMCFG register. NOTE: 1 pattern consist of 2 bytes and not the byte position (low or high). It is incremented by 1 every time a full PWMCFG register access (word) is performed. |

Table 37. PWMWP - PWM Timer Pattern Pointer Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7 | 0x0 | (reserved) |
| POINTER[6:0] | 6:0 | 0x0 | 0 ≤ POINTER < 32 : timer0 patterns 0 to 31 32 ≤ POINTER < 64 : timer1 patterns 0 to 31 64 ≤ POINTER < 96 : timer2 patterns 0 to 31 96 ≤ POINTER < 128: not valid |

PWMCFG - PWM Script Register
Table 38. PWMCFG - PWM Script Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| PWMCFG | 0x7E | W | Two byte pattern storage register for a PWM script command indexed by PWMWP. PWMWP is automatically incremented. To be applied by two consecutive parameter bytes in one I ² C Write Transaction. NOTE: Autoincrement is disabled on this register. Address will stay at 0x7E for each word access. |
| Bit - Name | Bit | Default | Bit Function |
| CMD[15:8] | 15:8 | | High byte portion of a PWM script command |
| CMD[7:0] | 7:0 | | Low byte portion of a PWM script command |

INTERFACE CONTROL REGISTERS

The following section describes the functions of special control registers provided for the main controller.

The manufacturer code MFGCODE and the software revision number SWREV tell the main device which configuration file has to be used for this device.

NOTE

I2CSA and MFGCODE use the same address. They just differentiate in the access type:

- Write - I2CSA
- Read - MFGCODE

I2CSA - I²C-Compatible ACCESS.bus Slave Address Register
Table 39. I2CSA - I²C-Compatible ACCESS.bus Slave Address Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| I2CSA | 0x80 | W | I ² C-compatible ACCESS.bus Slave Address. The address is internally applied after the next I ² C STOP. |
| Bit - Name | Bit | Default | Bit Function |
| SLAVEADDR[7:1] | 7:1 | 0x44 | 7-bit address field for the I ² C-compatible ACCESS.bus slave address. |
| (reserved) | 0 | | (reserved) |

MFGCODE - Manufacturer Code Register
Table 40. MFGCODE - Manufacturer Code Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|-----------------------------------|
| MFGCODE | 0x80 | R | Manufacturer code of the LM8325-1 |

Table 40. MFGCODE - Manufacturer Code Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| Bit - Name | Bit | Default | Bit Function |
| MFGBIT | 7:0 | 0x00 | 8 - bit field containing the manufacturer code |

SWREV - Software Revision Register**Table 41. SWREV - Software Revision Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| Bit - Name | Bit | Default | Bit Function |
| SWREV | 0x81 | R | Software revision code of the LM8325-1. NOTE: writing the SW revision with the inverted value triggers a reset (see SWRESET) |
| SWBIT | 7:0 | 0x84 | 8 - bit field containing the SW Revision number. |

SWRESET - Software Reset**Table 42. SWRESET - Software Reset Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| Bit - Name | Bit | Default | Bit Function |
| SWRESET | 0x81 | W | Software reset NOTE: the reset is only applied if the supplied parameter has the inverted value as SWBIT. Reading this register provides the software revision. (see SWREV) |
| SWBIT | 7:0 | | Reapply inverted value for software reset. |

RSTCTRL - System Reset Register

This register allows to reset specific blocks of the LM8325-1. For global reset of the IOExpander the I²C command 'General Call reset' is used (see [Global Call Reset](#)). This will reset the slave address back to 88H. During an active reset of a module, the LM8325-1 blocks the access to the module registers. A read will return 0, write commands are ignored.

Table 43. RSTCTRL - System Reset Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| Bit - Name | Bit | Default | Bit Function |
| RSTCTRL | 0x82 | R/W | Software reset of specific parts of the LM8325-1 |
| (reserved) | 7:5 | | (reserved) |
| IRQRST | 4 | 0x0 | Interrupt controller reset. Does not change status on IRQN ball. Only controls IRQ module register. Interrupt status read out is not possible when this bit is set. 0: interrupt controller not reset 1: interrupt controller reset |
| TIMRST | 3 | 0x0 | Timer reset for Timers 0, 1, 2 0: timer not reset 1: timer is reset |
| (reserved) | 2 | 0x0 | (reserved) |
| KBDRST | 1 | 0x0 | Keyboard interface reset 0: keyboard is not reset 1: keyboard is reset |

Table 43. RSTCTRL - System Reset Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|---|
| GPIRST | 0 | 0x0 | GENIO reset 0: GENIO not reset 1: GENIO is reset. |

RSTINTCLR - Clear NO Init/Power-On Interrupt Register
Table 44. RSTINTCLR - Clear NO Init/Power-On Interrupt Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| RSTINTCLR | 0x84 | W | This register allows to de-assert the POR/No Init Interrupt set every time the device returns from RESET (either POR, HW or SW Reset), the IRQN line is assigned active (low) and the IRQST.PORIRQ bit is set. |
| Bit - Name | Bit | Default | Bit Function |
| reserved | 7:1 | | (reserved) |
| IRQCLR | 0 | | 1: Clears the PORIRQ Interrupt signalled in IRQST register. 0: is ignored |

CLKMODE - Clock Mode Register
Table 45. CLKMODE - Clock Mode Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| CLKMODE | 0x88 | R/W | This register controls the current operating mode of the LM8325-1 device |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:2 | | (reserved) |
| MODCTL[1:0] | 1:0 | 0x01 | Writing to 00 forces the device to immediately enter sleep mode, regardless of any autosleep configuration. Reading this bit returns the current operating mode, which should always be 01. 00: SLEEP Mode 01: Operation Mode 1x: Future modes |

CLKCFG - Clock Configuration Register
Table 46. CLKCFG - Clock Configuration Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| CLKCFG | 0x89 | R/W | Configures clock sources and power options of the device. Note: Don't change while a PWM script is in progress |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7 | 0x0 | (reserved) |
| CLKSRCSEL[1:0] | 6:5 | 0x2 | 00: (reserved) 01: use external generated clock from CLKIN pin as PWM slow clock 1x: use internally generated PWM slow clock |
| (reserved) | 4:0 | 0x00 | (reserved) |

CLKEN - Clock Enable Register
Table 47. CLKEN - Clock Enable Register

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| CLKEN | 0x8A | R/W | Controls the clock to different functional units. It shall be used to enable the functional blocks globally and independently. |

Table 47. CLKEN - Clock Enable Register (continued)

| Register - Name | Address | Type | Register Function |
|-------------------|------------|----------------|--|
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:3 | | (reserved) |
| TIMEN | 2 | 0x0 | PWM Timer 0, 1, 2 clock enable 0: Timer 0, 1, 2 clock disabled 1: Timer 0, 1, 2 clock enabled. |
| (reserved) | 1 | | (reserved) |
| KBDEN | 0 | 0x0 | Keyboard clock enable (starts/stops key scan) 0: Keyboard clock disabled 1: Keyboard clock enabled |

AUTOSLP - Autosleep Enable Register**Table 48. AUTOSLP - Autosleep Enable Register**

| Register - Name | Address | Type | Register Function |
|-------------------|------------|----------------|--|
| AUTOSLP | 0x8B | R/W | This register controls the Auto Sleep function of the LM8325-1 device |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:1 | | (reserved) |
| ENABLE | 0 | 0x00 | Enables automatic sleep mode after a defined activity time stored in the AUTOSLPTI register 1: Enable entering auto sleep mode 0: Disable entering auto sleep mode |

AUTOSLPTI - Auto Sleep Time Register**Table 49. AUTOSLPTI - Auto Sleep Time Register**

| Register - Name | Address | Type | Register Function |
|-----------------------------|--------------|----------------|---|
| AUTOSLPTIL AUTOSLPTIH | 0x8C 0x8D | R/W | This register defines the activity time. If this time passes without any processing events then the device enters into sleep-mode, but only if AUTOSLP.ENABLE bit is set to 1. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 15:11 | | (reserved) |
| UPTIME[10:8] UPTIME[7:0] | 10:8 7:0 | 0x00 0xFF | Values of UPTIME[10:0] match to multiples of 4ms: 0x00: no autosleep, regardless if AUTOSLP.ENABLE is set 0x01: 4ms 0x02: 8ms 0x7A: 500 ms 0xFF: 1020 ms (default after reset) 0x100: 1024 ms 0x7FF: 8188 ms |

IRQST - Global Interrupt Status Register**Table 50. IRQST - Global Interrupt Status Register**

| Register - Name | Address | Type | Register Function |
|-------------------|------------|----------------|---|
| IRQST | 0x91 | R | Returns the interrupt status from various on-chip function blocks. If any of the bits is set and an IRQN line is configured, the IRQN line is asserted active |
| Bit - Name | Bit | Default | Bit Function |

Table 50. IRQST - Global Interrupt Status Register (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| PORIRQ | 7 | 0x1 | Supply failure on VCC. Also power-on is considered as an initial supply failure. Therefore, after power-on, the bit is set. 0: no failure recorded 1: Failure, device was completely reset and requires re-programming. |
| KBDIRQ | 6 | 0x0 | Keyboard interrupt (further key selection in keyboard module) 0: inactive 1: active |
| (reserved) | 5:4 | | (reserved) |
| TIM2IRQ | 3 | 0x0 | Timer2 expiry (CDIRQ or CYCIRQ) 0: inactive 1: active |
| TIM1IRQ | 2 | 0x0 | Timer1 expiry (CDIRQ or CYCIRQ) 0: inactive 1: active |
| TIM0IRQ | 1 | 0x0 | Timer0 expiry (CDIRQ or CYCIRQ) 0: inactive 1: active |
| GPIOIRQ | 0 | 0x0 | GPIO interrupt (further selection in GPIO module) 0: inactive 1: active |

GPIO FEATURE CONFIGURATION

GPIO Feature Mapping

The LM8325-1 has a flexible IO structure which allows to dynamically assign different functionality to each ball. The functionality of each ball is determined by the complete configuration of the balls.

In general the following priority is given:

- Keypad
- GPIO/PWM/Interrupt

With this, each ball will be available as GPIO, PWM or interrupt unless it is specified to be part of the keypad matrix. The configuration for keypad or PWM/interrupt usage is defined by the following registers:

- KBDSize and KBDDEDCFG
 - Both registers define a ball as either part of the keypad matrix or as dedicated key input. These settings have highest priority and will overwrite settings made in other registers.
- IOCFG
 - This register is used to define the usage of KPY[11:8] if not configured to be part of the keymatrix, to be used as GPIO.

Table 51. Ball Configuration Options

| BALL | Module connectivity | | | | | | | | |
|-----------|---------------------|------------|---------------------|------------|-----|-----|-----|-----|-----|
| | GPIOSEL | BALLCFG | | | | | | | |
| | | 0x0 | 0x1 | 0x2 | 0x3 | 0x4 | 0x5 | 0x6 | 0x7 |
| KPX[7:0] | not used | GPIO[7:0] | | | | | | | |
| KPY[7:0] | not used | GPIO[15:8] | | | | | | | |
| KPY8/PWM2 | not used | GPIO16 | PWM2 ⁽¹⁾ | (reserved) | - | - | - | - | - |

(1) **Note 1:** PWM2 functionality is mutually exclusive — one pin at a time only (KPY8 or KPY11) depending on interrupt enable Bit 4 of IOCFG.

Table 51. Ball Configuration Options (continued)

| BALL | Module connectivity | | | | | | | | |
|-----------------|---------------------------|--------|---------------------|------|---|---|---|---|---|
| | | | | | | | | | |
| KPY9/PWM1 | not used | GPIO17 | PWM1 | - | - | - | - | - | - |
| KPY10/PWM0 | not used | GPIO18 | PWM0 | - | - | - | - | - | - |
| IRQN/KPY11/PWM2 | see IOCFG | GPIO19 | PWM2 ⁽¹⁾ | PWM2 | - | - | - | - | - |

IOCGF - Input/Output Pin Mapping Configuration Register**Table 52. IOCGF - Input/Output Pin Mapping Configuration Register**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| IOCFG | 0xA7 | W | Configures usage of KPY[11:8] if not used for Keypad. On each write to this register, BALLCFG defines the column of Table 51 to configure. |
| Bit - Name | Bit | Default | Bit Function |
| GPIOSEL | 7:4 | | Configures the IRQN/KPY11/PWM2 ball Bit 4: Interrupt enabled Bit [7:5]: not used |
| (reserved) | 3 | | (reserved) |
| BALLCFG | 2:0 | | Select column to configure, see Ball configuration options |

IOPC0 - Pull Resistor Configuration Register 0**Table 53. IOPC0 - Pull Resistor Configuration Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| IOPC0* | 0xAA | R/W | Defines the pull resistor configuration for balls KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX7PR[1:0] | 15:14 | 0x2 | Resistor enable for KPX7 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX6PR[1:0] | 13:12 | 0x2 | Resistor enable for KPX6 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX5PR[1:0] | 11:10 | 0x2 | Resistor enable for KPX5 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX4PR[1:0] | 9:8 | 0x2 | Resistor enable for KPX4 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX3PR[1:0] | 7:6 | 0x2 | Resistor enable for KPX3 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX2PR[1:0] | 5:4 | 0x2 | Resistor enable for KPX2 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |

Table 53. IOPC0 - Pull Resistor Configuration Register 0 (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|---|
| KPX1PR[1:0] | 3:2 | 0x2 | Resistor enable for KPX1 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPX0PR[1:0] | 1:0 | 0x2 | Resistor enable for KPX0 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |

IOPC1 - Pull Resistor Configuration Register 1
Table 54. IOPC1 - Pull Resistor Configuration Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| IOPC1** | 0xAC | R/W | Defines the pull resistor configuration for balls KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY7PR[1:0] | 15:14 | 0x1 | Resistor enable for KPY7 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY6PR[1:0] | 13:12 | 0x1 | Resistor enable for KPY6 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY5PR[1:0] | 11:10 | 0x1 | Resistor enable for KPY5 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY4PR[1:0] | 9:8 | 0x1 | Resistor enable for KPY4 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY3PR[1:0] | 7:6 | 0x1 | Resistor enable for KPY3 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY2PR[1:0] | 5:4 | 0x1 | Resistor enable for KPY2 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY1PR[1:0] | 3:2 | 0x1 | Resistor enable for KPY1 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY0PR[1:0] | 1:0 | 0x1 | Resistor enable for KPY0 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |

IOPC2 - Pull Resistor Configuration Register 2**Table 55. IOPC2 - Pull Resistor Configuration Register 2**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| IOPC2*** | 0xAE | R/W | Defines the pull resistor configuration for balls KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 15:8 | 0x5A | (reserved) |
| KPY11PR[1:0] | 7:6 | 0x0 | Resistor enable for KPY11 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY10PR[1:0] | 5:4 | 0x1 | Resistor enable for KPY10 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY9PR[1:0] | 3:2 | 0x1 | Resistor enable for KPY9 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |
| KPY8PR[1:0] | 1:0 | 0x1 | Resistor enable for KPY8 ball 00: no pull resistor at ball 01: pull down resistor programmed 1x: pull up resistor programmed |

GPIOOME0 - GPIO Open Drain Mode Enable Register 0**Table 56. GPIOOME0 - GPIO Open Drain Mode Enable Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOOME0 | 0xE0 | R/W | Configures KPX[7:0] for Open Drain or standard output functionality. The Open Drain drive source is configured by GPIOOMS0. |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]ODE | 7:0 | 0x0 | Open Drain Enable on KPX[7:0] 0: full buffer 1: open drain functionality |

GPIOOMS0 - GPIO Open Drain Mode Select Register 0**Table 57. GPIOOMS0 - GPIO Open Drain Mode Select Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOOMS0 | 0xE1 | R/W | Configures the Open Drain drive source on KPX[7:0] if selected by GPIOOME0. |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]ODM | 7:0 | 0x0 | 0: Only nmos transistor is active in output driver stage. Output can be driven to gnd or Hi-Z 1: Only pmos transistor is active in output driver stage. Output can be driven to VCC or Hi-Z |

GPIOOME1 - GPIO Open Drain Mode Enable Register 1
Table 58. GPIOOME1 - GPIO Open Drain Mode Enable Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOOME1 | 0xE2 | R/W | Configures KPY[7:0] for Open Drain or standard output functionality. The Open Drain drive source is configured by GPIOOMS1. |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]ODE | 7:0 | 0x0 | Open Drain Enable on KPY[7:0] 0: full buffer 1: open drain functionality |

GPIOOMS1 - GPIO Open Drain Mode Select Register 1
Table 59. GPIOOMS1 - GPIO Open Drain Mode Select Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOOMS1 | 0xE3 | R/W | Configures the Open Drain drive source on KPY[7:0] if selected by GPIOOME1. |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]ODM | 7:0 | 0x0 | 0: Only nmos transistor is active in output driver stage. Output can be driven to gnd or Hi-Z 1: Only pmos transistor is active in output driver stage. Output can be driven to VCC or Hi-Z |

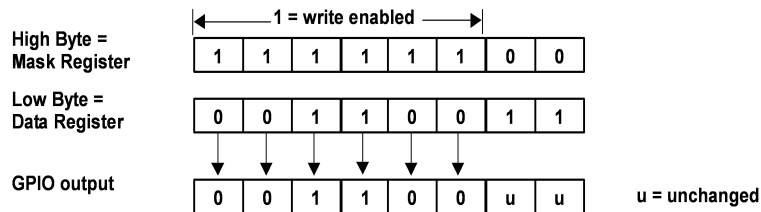
GPIOOME2 - GPIO Open Drain Mode Enable Register 2
Table 60. GPIOOME2 - GPIO Open Drain Mode Enable Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOOME2 | 0xE4 | R/W | Configures KPY[11:8] for Open Drain or standard output functionality. The Open Drain drive source is configured by GPIOOMS2. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | 0x0 | (reserved) |
| KPY[11:8]ODE | 3:0 | 0x8 | Open Drain Enable on KPY[11:8] 0: full buffer 1: open drain functionality Note: KPY11/IRQN ball defaults to Open Drain Mode Enable after reset. |

GPIOOMS2 - GPIO Open Drain Mode Select Register 2
Table 61. GPIOOMS2 - GPIO Open Drain Mode Select Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOOMS2 | 0xE5 | R/W | Configures the Open Drain drive source on KPY[11:8] if selected by GPIOOME2. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]ODM | 3:0 | 0x0 | 0: Only nmos transistor is active in output driver stage. Output can be driven to gnd or Hi-Z 1: Only pmos transistor is active in output driver stage. Output can be driven to VCC or Hi-Z |

GPIO DATA INPUT/OUTPUT



GPIO_DATA0 - GPIO Data Register 0

Table 62. GPIO_DATA0 - GPIO Data Register 0

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIO_DATA0 | 0xC0 | R/W | This register is used for data input/output of KPX[7:0]. Every data I/O is masked with the associated MASK register. If one of the I/Os is defined as output (see Table 65) values written to this register are masked with MASK and then applied to the associated pin. If one of the I/Os is defined as input (see Table 65) values read from this register hold the masked input value of the associated pin. |
| Bit - Name | Bit | Default | Bit Function |
| MASK7 | 15 | 0x0 | Mask Status for KPX7 when enabled as GPIO 1: KPX7 enabled 0: KPX7 disabled |
| MASK6 | 14 | 0x0 | Mask Status for KPX6 when enabled as GPIO 1: KPX6 enabled 0: KPX6 disabled |
| MASK5 | 13 | 0x0 | Mask Status for KPX5 when enabled as GPIO 1: KPX5 enabled 0: KPX5 disabled |
| MASK4 | 12 | 0x0 | Mask Status for KPX4 when enabled as GPIO 1: KPX4 enabled 0: KPX4 disabled |
| MASK3 | 11 | 0x0 | Mask Status for KPX3 when enabled as GPIO 1: KPX3 enabled 0: KPX3 disabled |
| MASK2 | 10 | 0x0 | Mask Status for KPX2 when enabled as GPIO 1: KPX2 enabled 0: KPX2 disabled |
| MASK1 | 9 | 0x0 | Mask Status for KPX1 when enabled as GPIO 1: KPX1 enabled 0: KPX1 disabled |
| MASK0 | 8 | 0x0 | Mask Status for KPX0 when enabled as GPIO 1: KPX0 enabled 0: KPX0 disabled |
| DATA7 | 7 | 0x0 | Pin Status for KPX7 when enabled as GPIO |
| DATA6 | 6 | 0x0 | Pin Status for KPX6 when enabled as GPIO |
| DATA5 | 5 | 0x0 | Pin Status for KPX5 when enabled as GPIO |
| DATA4 | 4 | 0x0 | Pin Status for KPX4 when enabled as GPIO |
| DATA3 | 3 | 0x0 | Pin Status for KPX3 when enabled as GPIO |
| DATA2 | 2 | 0x0 | Pin Status for KPX2 when enabled as GPIO |
| DATA1 | 1 | 0x0 | Pin Status for KPX1 when enabled as GPIO |

Table 62. GPIODATA0 - GPIO Data Register 0 (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| DATA0 | 0 | 0x0 | Pin Status for KPX0 when enabled as GPIO |

GPIODATA1 - GPIO Data Register 1
Table 63. GPIODATA1 - GPIO Data Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIODATA1 | 0xC2 | R/W | <p>This register is used for data input/output of KPY[7:0]. Every data I/O is masked with the associated MASK register.</p> <p>If one of the I/Os is defined as output (see Table 66) values written to this register are masked with MASK and then applied to the associated pin.</p> <p>If one of the I/Os is defined as input (see Table 66) values read from this register hold the masked input value of the associated pin.</p> |
| Bit - Name | Bit | Default | Bit Function |
| MASK15 | 15 | 0x0 | Mask Status for KPY7 when enabled as GPIO 1: KPY7 enabled 0: KPY7 disabled |
| MASK14 | 14 | 0x0 | Mask Status for KPY6 when enabled as GPIO 1: KPY6 enabled 0: KPY6 disabled |
| MASK13 | 13 | 0x0 | Mask Status for KPY5 when enabled as GPIO 1: KPY5 enabled 0: KPY5 disabled |
| MASK12 | 12 | 0x0 | Mask Status for KPY4 when enabled as GPIO 1: KPY4 enabled 0: KPY4 disabled |
| MASK11 | 11 | 0x0 | Mask Status for KPY3 when enabled as GPIO 1: KPY3 enabled 0: KPY3 disabled |
| MASK10 | 10 | 0x0 | Mask Status for KPY2 when enabled as GPIO 1: KPY2 enabled 0: KPY2 disabled |
| MASK9 | 9 | 0x0 | Mask Status for KPY1 when enabled as GPIO 1: KPY1 enabled 0: KPY1 disabled |
| MASK8 | 8 | 0x0 | Mask Status for KPY0 when enabled as GPIO 1: KPY0 enabled 0: KPY0 disabled |
| DATA15 | 7 | 0x0 | Pin Status for KPY7 when enabled as GPIO |
| DATA14 | 6 | 0x0 | Pin Status for KPY6 when enabled as GPIO |
| DATA13 | 5 | 0x0 | Pin Status for KPY5 when enabled as GPIO |
| DATA12 | 4 | 0x0 | Pin Status for KPY4 when enabled as GPIO |
| DATA11 | 3 | 0x0 | Pin Status for KPY3 when enabled as GPIO |
| DATA10 | 2 | 0x0 | Pin Status for KPY2 when enabled as GPIO |
| DATA9 | 1 | 0x0 | Pin Status for KPY1 when enabled as GPIO |
| DATA8 | 0 | 0x0 | Pin Status for KPY0 when enabled as GPIO |

GPIOPDATA2 - GPIO Data Register 2**Table 64. GPIOPDATA2 - GPIO Data Register 2**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOPDATA2 | 0xC4 | R/W | This register is used for data input/output of KPY[11:8]. Every data I/O is masked with the associated MASK register. If one of the I/Os is defined as output (see Table 67) values written to this register are masked with MASK and then applied to the associated pin. If one of the I/Os is defined as input (see Table 67) values read from this register hold the masked input value of the associated pin. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 15:12 | 0x0 | (reserved) |
| MASK19 | 11 | 0x0 | Mask Status for KPY11 when enabled as GPIO 1: KPY11 enabled 0: KPY11 disabled |
| MASK18 | 10 | 0x0 | Mask Status for KPY10 when enabled as GPIO 1: KPY10 enabled 0: KPY10 disabled |
| MASK17 | 9 | 0x0 | Mask Status for KPY9 when enabled as GPIO 1: KPY9 enabled 0: KPY9 disabled |
| MASK16 | 8 | 0x0 | Mask Status for KPY8 when enabled as GPIO 1: KPY8 enabled 0: KPY8 disabled |
| reserved | 7:4 | 0x0 | (reserved) |
| DATA19 | 3 | 0x0 | Pin Status for KPY11 when enabled as GPIO |
| DATA18 | 2 | 0x0 | Pin Status for KPY10 when enabled as GPIO |
| DATA17 | 1 | 0x0 | Pin Status for KPY9 when enabled as GPIO |
| DATA16 | 0 | 0x0 | Pin Status for KPY8 when enabled as GPIO |

GPIOPDIR0 - GPIO Port Direction Register 0**Table 65. GPIOPDIR0 - GPIO Port Direction Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOPDIR0 | 0xC6 | R/W | Port direction for KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]DIR | 7:0 | 0x00 | Direction bits for KPX[7:0] 0: input mode 1: output mode |

GPIOPDIR1 - GPIO Port Direction Register 1**Table 66. GPIOPDIR1 - GPIO Port Direction Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOPDIR1 | 0xC7 | R/W | Port direction for KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]DIR | 7:0 | 0x00 | Direction bits for KPY[7:0] 0: input mode 1: output mode |

GPIODIR2 - GPIO Port Direction Register 2
Table 67. GPIODIR2 - GPIO Port Direction Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIODIR2 | 0xC8 | R/W | Port direction for KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]DIR | 3:0 | 0x08 | Direction bits for KPY[11:8] 0: input mode 1: output mode |

GPIO INTERRUPT CONTROL
GPIOS0 - Interrupt Sense Configuration Register 0
Table 68. GPIOS0 - Interrupt Sense Configuration Register 0

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOS0 | 0xC9 | R/W | Interrupt type on KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]IS | 7:0 | 0x0 | Interrupt type bits for KPX[7:0] 0: edge sensitive interrupt 1: level sensitive interrupt |

GPIOS1 - Interrupt Sense Configuration Register 1
Table 69. GPIOS1 - Interrupt Sense Configuration Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOS1 | 0xCA | R/W | Interrupt type on KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]IS | 7:0 | 0x0 | Interrupt type bits for KPY[7:0] 0: edge sensitive interrupt 1: level sensitive interrupt |

GPIOS2 - Interrupt Sense Configuration Register 2
Table 70. GPIOS2 - Interrupt Sense Configuration Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOS2 | 0xCB | R/W | Interrupt type on KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]IS | 3:0 | 0x0 | Interrupt type bits for KPY[11:8] 0: edge sensitive interrupt 1: level sensitive interrupt |

GPIOIBE0 - GPIO Interrupt Edge Configuration Register 0**Table 71. GPIOIBE0 - GPIO Interrupt Edge Configuration Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIBE0 | 0xCC | R/W | Defines whether an interrupt on KPX[7:0] is triggered on both edges or on a single edge. See Table 74 for the edge configuration. |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]IBE | 7:0 | 0x0 | Interrupt both edges bits for KPX[7:0] 0: interrupt generated at the active edge 1: interrupt generated after both edges |

GPIOIBE1 - GPIO Interrupt Edge Configuration Register 1**Table 72. GPIOIBE1 - GPIO Interrupt Edge Configuration Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIBE1 | 0xCD | R/W | Defines whether an interrupt on KPY[7:0] is triggered on both edges or on a single edge. See Table 75 for the edge configuration. |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]IBE | 7:0 | 0x0 | Interrupt both edges bits for KPY[7:0] 0: interrupt generated at the configured edge 1: interrupt generated after both edges |

GPIOIBE2 - GPIO Interrupt Edge Configuration Register 2**Table 73. GPIOIBE2 - GPIO Interrupt Edge Configuration Register 2**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOIBE2 | 0xCE | R/W | Defines whether an interrupt on KPY[11:8] is triggered on both edges or on a single edge. See Table 76 for the edge configuration. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]IBE | 3:0 | 0x0 | Interrupt both edges bits for KPY[11:8] 0: interrupt generated at the active edge 1: interrupt generated after both edges |

GPIOIEV0 - GPIO Interrupt Edge Select Register 0**Table 74. GPIOIEV0 - GPIO Interrupt Edge Select Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIEV0 | 0xCF | R/W | Select Interrupt edge for KPX[7:0]. |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]EV | 7:0 | 0xFF | Interrupt edge select from KPX[7:0] 0: interrupt at low level or falling edge 1: interrupt at high level or rising edge |

GPIOIEV1 - GPIO Interrupt Edge Select Register 1**Table 75. GPIOIEV1 - GPIO Interrupt Edge Select Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|-------------------------------------|
| GPIOIEV1 | 0xD0 | R/W | Select Interrupt edge for KPY[7:0]. |
| Bit - Name | Bit | Default | Bit Function |

Table 75. GPIOIEV1 - GPIO Interrupt Edge Select Register 1 (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|---|
| KPY[7:0]EV | 7:0 | 0xFF | Interrupt edge select from KPY[7:0] 0: interrupt at low level or falling edge 1: interrupt at high level or rising edge |

GPIOIEV2 - GPIO Interrupt Edge Select Register 2
Table 76. GPIOIEV2 - GPIO Interrupt Edge Select Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOIEV2 | 0xD1 | R/W | Select Interrupt edge for KPY[11:8]. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]EV | 3:0 | 0xFF | Interrupt edge select from KPY[11:8] 0: interrupt at low level or falling edge 1: interrupt at high level or rising edge |

GPIOIE0 - GPIO Interrupt Enable Register 0
Table 77. GPIOIE0 - GPIO Interrupt Enable Register 0

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIE0 | 0xD2 | R/W | Enable/disable interrupts on KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]IE | 7:0 | 0x0 | Interrupt enable on KPX[7:0] 0: disable interrupt 1: enable interrupt |

GPIOIE1 - GPIO Interrupt Enable Register 1
Table 78. GPIOIE1 - GPIO Interrupt Enable Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIE1 | 0xD3 | R/W | Enable/disable interrupts on KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]IE | 7:0 | 0x0 | Interrupt enable on KPY[7:0] 0: disable interrupt 1: enable interrupt |

GPIOIE2 - GPIO Interrupt Enable Register 2
Table 79. GPIOIE2 - GPIO Interrupt Enable Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOIE2 | 0xD4 | R/W | Enable/disable interrupts on KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]IE | 3:0 | 0x0 | Interrupt enable on KPY[11:8] 0: disable interrupt 1: enable interrupt |

GPIOIC0 - GPIO Clear Interrupt Register 0**Table 80. GPIOIC0 - GPIO Clear Interrupt Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIC0 | 0xDC | W | Clears the interrupt on KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]IC | 7:0 | | Clear Interrupt on KPX[7:0] 0: no effect 1: Clear corresponding interrupt |

GPIOIC1 - GPIO Clear Interrupt Register 1**Table 81. GPIOIC1 - GPIO Clear Interrupt Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOIC1 | 0xDD | W | Clears the interrupt on KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]IC | 7:0 | | Clear Interrupt on KPY[7:0] 0: no effect 1: Clear corresponding interrupt |

GPIOIC2 - GPIO Clear Interrupt Register 2**Table 82. GPIOIC2 - GPIO Clear Interrupt Register 2**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOIC2 | 0xDE | W | Clears the interrupt on KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]IC | 3:0 | | Clear Interrupt on KPY[11:8] 0: no effect 1: Clear corresponding interrupt |

GPIO INTERRUPT STATUS***GPIORIS0 - Raw Interrupt Status Register 0*****Table 83. GPIORIS0 - Raw Interrupt Status Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIORIS0 | 0xD6 | R | Raw interrupt status on KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]RIS | 7:0 | 0x0 | Raw Interrupt status data on KPX[7:0] 0: no interrupt condition at GPIO 1: interrupt condition at GPIO |

GPIORIS1 - Raw Interrupt Status Register 1**Table 84. GPIORIS1 - Raw Interrupt Status Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|----------------------------------|
| GPIORIS1 | 0xD7 | R | Raw interrupt status on KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |

Table 84. GPIORIS1 - Raw Interrupt Status Register 1 (continued)

| Register - Name | Address | Type | Register Function |
|-----------------|---------|------|--|
| KPY[7:0]RIS | 7:0 | 0x0 | Raw Interrupt status data on KPY[7:0] 0: no interrupt condition at GPIO 1: interrupt condition at GPIO |

GPIORIS2 - Raw Interrupt Status Register 2
Table 85. GPIORIS2 - Raw Interrupt Status Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIORIS2 | 0xD8 | R | Raw interrupt status on KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]RIS | 3:0 | 0x0 | Raw Interrupt status data on KPY[11:8] 0: no interrupt condition at GPIO 1: interrupt condition at GPIO |

GPIOMIS0 - Masked Interrupt Status Register 0
Table 86. GPIOMIS0 - Masked Interrupt Status Register 0

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOMIS0 | 0xD9 | R | Masked interrupt status on KPX[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]MIS | 7:0 | 0x0 | Masked Interrupt status data on KPX[7:0] 0: no interrupt contribution from GPIO 1: interrupt GPIO is active |

GPIOMIS1 - Masked Interrupt Status Register 1
Table 87. GPIOMIS1 - Masked Interrupt Status Register 1

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOMIS1 | 0xDA | R | Masked interrupt status on KPY[7:0] |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]MIS | 7:0 | 0x0 | Masked Interrupt status data on KPY[7:0] 0: no interrupt contribution from GPIO 1: interrupt GPIO is active |

GPIOMIS2 - Masked Interrupt Status Register 2
Table 88. GPIOMIS2 - Masked Interrupt Status Register 2

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOMIS2 | 0xDB | R | Masked interrupt status on KPY[11:8] |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]MIS | 3:0 | 0x0 | Masked Interrupt status data on KPY[11:8] 0: no interrupt contribution from GPIO 1: interrupt GPIO is active |

GPIO WAKE-UP CONTROL***GPIOWAKE0 - GPIO Wake-Up Register 0*****Table 89. GPIOWAKE0 - GPIO Wake-Up Register 0**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOWAKE0 | 0xE9 | R/W | Configures wake-up conditions for KPX[7:0] Each bit corresponds to a ball. When bit set, the corresponding ball contributes to wakeup from auto sleep mode. |
| Bit - Name | Bit | Default | Bit Function |
| KPX[7:0]WAKE | 7:0 | 0x0 | Bit 7: KPX7 ... Bit 0: KPX0 |

GPIOWAKE1 - GPIO Wake-Up Register 1**Table 90. GPIOWAKE1 - GPIO Wake-Up Register 1**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|--|
| GPIOWAKE1 | 0xEA | R/W | Configures wake-up conditions for KPY[7:0] Each bit corresponds to a ball. When bit set, the corresponding ball contributes to wakeup from auto sleep mode. |
| Bit - Name | Bit | Default | Bit Function |
| KPY[7:0]WAKE | 7:00 | 0x0 | Bit 7: KPY7 ... Bit 0: KPY0 |

GPIOWAKE2 - GPIO Wake-Up Register 2**Table 91. GPIOWAKE2 - GPIO Wake-Up Register 2**

| Register - Name | Address | Type | Register Function |
|-----------------|---------|---------|---|
| GPIOWAKE2 | 0xEB | R/W | Configures wake-up conditions for KPY[11:8] Each bit corresponds to a ball. When bit set, the corresponding ball contributes to wakeup from auto sleep mode. |
| Bit - Name | Bit | Default | Bit Function |
| (reserved) | 7:4 | | (reserved) |
| KPY[11:8]WAKE | 3:0 | 0x0 | Bit 3: KPY11 ... Bit 0: KPY8 |



These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

ABSOLUTE MAXIMUM RATINGS ⁽¹⁾⁽²⁾

| | | |
|--|-----------------------|--------------------------|
| Supply Voltage (V_{CC}) | | -0.3V to 2.2V |
| Voltage at Generic IOs | | -0.2V to $V_{CC} + 0.2V$ |
| Voltage at Backdrive/Overvoltage IOs | | -0.3V to +4.25V |
| Maximum Input Current Without Latchup | | ±100 mA |
| ESD Protection Level | (Human Body Model) | 2kV |
| | (Machine Model) | 200V |
| | (Charge Device Model) | 750V |
| Total Current into V_{CC} Pin (Source) | | 100 mA |
| Total Current out of GND Pin (Sink) | | 100 mA |
| Storage Temperature Range | | -65°C to +140°C |

- (1) Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is intended to be functional, but specific performance is not ensured. For ensured specifications and test conditions, see the Electrical Characteristics tables.
- (2) If Military/Aerospace specified devices are required, please contact the TI Sales Office/ Distributors for availability and specifications.

DC ELECTRICAL CHARACTERISTICS

Datasheet min/max specification limits are specified by design, test, or statistical analysis.
(Temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$, unless otherwise specified)

| Parameter | Conditions | Min | Typ | Max | Units |
|---|---|------|-----|------|---------------|
| Operating Voltage (V_{CC}) | Core Supply Voltage | 1.62 | | 1.98 | V |
| Maximum Input voltage for Backdrive/Overvoltage IOs | | | | 3.60 | V |
| Supply Current (I_{DD}) ⁽¹⁾ | Internal Clock = ON, all internal functional blocks running No loads on pins, $V_{CC} = 1.8V$, $T_C = 0.5 \mu\text{s}$ $T_A = 25^{\circ}\text{C}$ | | 1.9 | 3.0 | mA |
| Sleep Mode HALT Current (I_{HALT}) ⁽²⁾ | $V_{CC} = 1.8V$, $T_A = 25^{\circ}\text{C}$; Internal Clock = OFF, no internal functional blocks running | | <9 | 40 | μA |
| IDLE Current | Internal Clock = ON, no internal functional blocks running | | 1 | | mA |

- (1) Supply and IDLE current is measured with inputs connected to V_{CC} and outputs driven low but not connected to a load.
- (2) In sleep mode, the internal clock is switched off. Supply current in sleep mode is measured with inputs connected to V_{CC} and outputs driven low but not connected to a load.

AC ELECTRICAL CHARACTERISTICS

(Temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$)

Data sheet specification limits are specified by design, test, or statistical analysis.

| Parameter | Conditions | Min | Typ | Max | Units |
|--|--------------------------------|-----|-----|-----|---------------|
| System Clock Frequency | Internal RC | | 21 | | MHz |
| System Clock Period (mclk) | $1.62V \leq V_{CC} \leq 1.98V$ | | 48 | | ns |
| Internal RC Oscillator (t_C) | $1.62V \leq V_{CC} \leq 1.98V$ | | 0.5 | | μs |
| Internal RC Oscillator Frequency Variation | | | | ±7 | % |

AC ELECTRICAL CHARACTERISTICS (continued)

(Temperature: $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$)

Data sheet specification limits are specified by design, test, or statistical analysis.

| Parameter | Conditions | Min | Typ | Max | Units |
|--|-----------------------------|-----|-----|-----|-------|
| ACCESS.bus Input Signals | | | | | |
| Bus Free Time Between Stop and Start Condition (t_{BUF}) ⁽¹⁾ | | 16 | | | |
| SCL Setup Time (t_{CSTOsi}) ⁽¹⁾ | Before Stop Condition | 8 | | | mclk |
| SCL Hold Time (t_{CSTRhi}) ⁽¹⁾ | After Start Condition | 8 | | | |
| SCL Setup Time (t_{CSTRsi}) ⁽¹⁾ | Before Start Condition | 8 | | | |
| Data High Setup Time (t_{DHCsi}) ^{(1) (2)} | Before SCL Rising Edge (RE) | 2 | | | |
| Data Low Setup Time (t_{DLCsi}) ^{(1) (2)} | Before SCL RE | 2 | | | |
| SCL Low Time (t_{SCLowi}) ⁽¹⁾ | After SCL Falling Edge (FE) | 12 | | | |
| SCL High Time (t_{SCLhighi}) ^{(1) (2)} | After SCL FE | 12 | | | |
| SDA Hold Time (t_{SDAhi}) ⁽¹⁾ | After SCL FE | 0 | | | |
| SDA Setup Time (t_{SDAsi}) ^{(1) (2)} | Before SCL RE | 2 | | | |
| ACCESS.bus Output Signals | | | | | |
| SDA Hold Time (t_{SDAho}) ⁽¹⁾ | After SCL Falling Edge | 2 | | | mclk |

(1) Specified by design, not tested.

(2) The ACCESS.bus interface implements and meets the timings necessary for interface to the I²C and SMBus protocols at logic levels. The bus drivers have open-drain outputs for bidirectional operation. Due to Internal RC Oscillator Frequency Variation, this specification may not meet the AC timing and current/voltage drive requirements of the full-bus specifications.

GENERAL GPIO CHARACTERISTICS

Characteristics for all pins except CLKIN, IRQN/KPY11/PWM2, SDA, and SCL in GPIO mode.

| Parameter | Conditions | Min | Typ | Max | Units |
|--|--|----------------------------|-----|----------------------------|-------|
| V_{IH} (Min. Input High Voltage) | | $0.7 \times V_{\text{CC}}$ | | | V |
| V_{IL} (Max. Input Low Voltage) | | | | $0.3 \times V_{\text{CC}}$ | V |
| I_{Source} | $V_{\text{CC}} = 1.62$ $V_{\text{OH}} = 0.7 \times V_{\text{CC}}$ | | | -16 | mA |
| I_{Sink} | $V_{\text{CC}} = 1.62$ $V_{\text{OL}} = 0.3 \times V_{\text{CC}}$ | 16 | | | mA |
| Allowable Sink current per pin ⁽¹⁾ | | | | 16 | mA |
| I_{PU} (Weak Pull-UP Current) ⁽²⁾ | $V_{\text{OUT}} = 0\text{V}$ | -30 | | -160 | μA |
| I_{PD} (Weak Pull-DOWN Current) ⁽²⁾ | $V_{\text{OUT}} = V_{\text{CC}}$ | 30 | | 160 | |
| I_{OZ} (Input Leakage Current) | GPIO output disabled $V_{\text{pin}} = 0$ to V_{CC} | | | ±2 | |
| $t_{\text{Rise/Fall}}$ (Max. Rise and Fall times) ⁽³⁾ | $C_{\text{LOAD}} = 50$ pF | | | 15 | ns |

(1) The sum of all I/O sink/source current must not exceed the maximum total current into V_{CC} and out of GND as specified in the absolute maximum ratings.

(2) This is the internal weak pull-up (pull-down) current when driver output is disabled. If enabled, during receiving mode, this is the current required to switch the input from one state to another.

(3) Specified by design, not tested.

BACKDRIVE/OVERVOLTAGE I/O DC CHARACTERISTICS

Characteristics for pins CLKIN, IRQN/KPY11/PWM2, SDA and SCL

| Parameter | Conditions | Min | Typ | Max | Units |
|--|--|---------------------|-----|---------------------|-------|
| V_{IH} (Min. Input High Voltage) | | $0.7 \times V_{CC}$ | | | V |
| V_{IL} (Max. Input Low Voltage) | | | | $0.3 \times V_{CC}$ | |
| I_{Source} | $V_{CC} = 1.62V$ $V_{OH} = 1.5V$ | | | -6 | mA |
| I_{Sink1} (as GPIO) | $V_{CC} = 1.62V$ $V_{OL} = 0.4V$ | 12 | | | mA |
| I_{Sink2} (as ACCESS.bus) | $V_{CC} = 1.62V$ $V_{OL} = 0.4V$ | 3 | | | mA |
| I_{Sink3} (as ACCESS.bus) | $V_{CC} = 1.62V$ $V_{OL} = 0.6V$ | 4 | | | mA |
| Allowable Sink current per pin ⁽¹⁾ | | | | 12 | mA |
| I_{PU} (Weak Pull-UP Current) ⁽²⁾ | $V_{OUT} = 0V$ | -7 | | -40 | μA |
| I_{PD} (Weak Pull-Down Current) ⁽²⁾ | $V_{OUT} = V_{CC}$ | 7 | | 40 | |
| I_{OZ1} (Input Leakage Current) ⁽³⁾ | GPIO output disabled $V_{CC} = 1.62V$ to $1.98V$ $V_{pin} = 0$ to V_{CC} $V_{pin} = V_{CC}$ to $3.6V$ | | | ±2 ±10 | |
| I_{OZ2} (Input Backdrive Leakage Current) | $0 \leq V_{CC} \leq 0.5V$ $V_{pin} = 0$ to $3.6V$ | | | ±10 | μA |

- (1) The sum of all I/O sink/source current must not exceed the maximum total current into V_{CC} and out of GND as specified in the absolute maximum ratings.
- (2) This is the internal weak pull-up (pull-down) current when driver output is disabled. If enabled, during receiving mode, this is the current required to switch the input from one state to another.
- (3) I_{OZ1} for CLKIN is max 60 μA if $V_{PIN} > V_{CC}$ because the weak pull-down is enabled.

BACKDRIVE/OVERVOLTAGE I/O AC CHARACTERISTICS

Characteristics for pins CLKIN, IRQN/KPY11/PWM2, SDA and SCL

| Parameter | Conditions | Min | Typ | Max | Units |
|---|--|-----|-----|-----|-------|
| $t_{Rise/Fall}$ (Max. Rise and Fall time) ⁽¹⁾ | $C_{LOAD} = 50$ pF @ 1MHz | | | 70 | ns |
| t_{Fall} (Max. Fall time) as ACCESS.bus (SDA, SCL only) ⁽¹⁾ | $C_{LOAD} = 10$ pF to 100 pF V_{IHmin} to V_{ILmax} | 10 | | 120 | |

- (1) Specified by design, not tested.

REGISTER

REGISTER MAPPING

Keyboard Registers

shows the register map for keyboard functionality. In addition to Global Call Reset (see [Global Call Reset](#)) or Software Reset using SWRESET (see [Table 42](#)), these registers are reset to 0x00 values by a module reset using RSTCTRL.KBDRST and should be rewritten for desired settings (see [Table 43](#)).

Register Map for Keyboard Functionality

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|--------------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| KBDSETTLE | Keypad Settle Time | 0x01 | R/W | byte | 0x80 | 0x02 |
| KBDBOUNCE | Keypad Debounce Time | 0x02 | R/W | byte | 0x80 | 0x03 |
| KBDSIZE | Keypad Size Configuration | 0x03 | R/W | byte | 0x22 | 0x04 |
| KBDEDCFG0 | Keypad Dedicated Key 0 | 0x04 | R/W | byte | 0xFF | 0x05 |
| KBDEDCFG1 | Keypad Dedicated Key 1 | 0x05 | R/W | byte | 0xFF | 0x06 |
| KBDRIS | Keypad Raw Interrupt Status | 0x06 | R | byte | 0x00 | 0x07 |
| KBDMIS | Keypad Masked Interrupt Status | 0x07 | R | byte | 0x00 | 0x08 |
| KBDIC | Keypad Interrupt Clear | 0x08 | W | byte | | 0x09 |
| KBDMSK | Keypad Interrupt Mask | 0x09 | R/W | byte | 0xF3 | 0x0A |
| KBDCODE0 | Keypad Code 0 | 0x0B | R | byte | 0x7F | 0x0C |
| KBDCODE1 | Keypad Code 1 | 0x0C | R | byte | 0x7F | 0x0D |
| KBDCODE2 | Keypad Code 2 | 0x0D | R | byte | 0x7F | 0x0E |
| KBDCODE3 | Keypad Code 3 | 0x0E | R | byte | 0x7F | 0x0F |
| EVTCODE | Key Event Code | 0x10 | R | byte | 0x7F | 0x10 |

PWM Timer Registers

shows the register map for PWM Timer functionality. In addition to Global Call Reset (see [Global Call Reset](#)) or Software Reset using SWRESET (see [Table 42](#)), these registers are reset to default values by a module reset using RSTCTRL.TIMRST (see [Table 43](#)).

Register Map for PWM Timer Functionality

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|---------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| TIMCFG0 | PWM Timer Configuration 0 | 0x60 | R/W | byte | 0x00 | 0x61 |
| PWMCFG0 | PWM Configuration 0 | 0x61 | R/W | byte | 0x00 | 0x62 |
| TIMSCAL0 | PWM Timer Prescaler 0 | 0x62 | R/W | byte | 0x00 | 0x63 |
| TIMCFG1 | PWM Timer Configuration 1 | 0x68 | R/W | byte | 0x00 | 0x69 |
| PWMCFG1 | PWM Configuration 1 | 0x69 | R/W | byte | 0x00 | 0x6A |
| TIMSCAL1 | PWM Timer Prescaler 1 | 0x6A | R/W | byte | 0x00 | 0x6B |

Register Map for PWM Timer Functionality (continued)

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|------------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| TIMCFG2 | PWM Timer Configuration 2 | 0x70 | R/W | byte | 0x00 | 0x71 |
| PWMCFG2 | PWM Configuration 2 | 0x71 | R/W | byte | 0x00 | 0x72 |
| TIMSCAL2 | PWM Timer Prescaler 2 | 0x72 | R/W | byte | 0x00 | 0x73 |
| TIMSWRES | PWM Timer SW Reset | 0x78 | W | byte | | 0x79 |
| TIMRIS | PWM Timer Interrupt Status | 0x7A | R | byte | 0x00 | 0x7B |
| TIMMIS | PWM Timer Masked Int. Status | 0x7B | R | byte | 0x00 | 0x7C |
| TIMIC | Timer Interrupt Clear | 0x7C | W | byte | | 0x7D |
| PWMWP | PWM Command Write Pointer | 0x7D | R/W | byte | 0x00 | 0x7E |
| PWMCFG | PWM Command Script | 0x7E | W | word | | 0x7F |

System Registers

shows the register map for general system registers. These registers are not affected by any of the module resets addressed by RSTCTRL (see [Table 43](#)). These registers can only be reset to default values by a Global Call Reset (see [Global Call Reset](#)) or by a complete Software Reset using SWRESET (see [Table 42](#)).

Register Map for System Control Functionality

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|--|-----------------------|---------------|-------------|---------------|-----------------|
| I2CSA | I ² C-compatible ACCESS.bus Slave Address | 0x80 | W | byte | 0x88 | 0x81 |
| MFGCODE | Manufacturer Code | 0x80 | R | byte | 0x00 | 0x81 |
| SWREV | SW Revision | 0x81 | R | byte | 0x83 | 0x82 |
| SWRESET | SW Reset | 0x81 | W | byte | | 0x82 |
| RSTCTRL | System Reset | 0x82 | R/W | byte | 0x00 | 0x83 |
| RSTINTCLR | Clear No Init/Power On Interrupt | 0x84 | W | byte | | 0x85 |
| CLKMODE | Clock Mode | 0x88 | R/W | byte | 0x01 | 0x89 |
| CLKCFG | Clock Configuration | 0x89 | R/W | byte | 0x40 | 0x8A |
| CLKEN | Clock Enable | 0x8A | R/W | byte | 0x00 | 0x8B |
| AUTOSLP | Auto Sleep Enable | 0x8B | R/W | byte | 0x00 | 0x8C |
| AUTOSLPTI | Auto Sleep Time | 0x8C | R/W | word | 0x00FF | 0x8D |

Global Interrupt Registers

shows the register map for global interrupt functionality. In addition to Global Call Reset (see [Global Call Reset](#)) or Software Reset using SWRESET (see [Table 42](#)), these registers are reset to default values by a module reset using RSTCTRL.IRQRST (see [Table 43](#)).

Register Map for Global Interrupt Functionality

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|-------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| IRQST | Global Interrupt Status | 0x91 | R | byte | 0x80 | 0x92 |

GPIO Registers

shows the register map for GPIO functionality. In addition to Global Call Reset (see [Global Call Reset](#)) or Software Reset using SWRESET (see [Table 42](#)), these registers are reset to 0x00 values by a module reset using RSTCTRL.GPIRST and should be rewritten for desired settings (see [Table 43](#)).

Register Map for GPIO Functionality

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|-------------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| IOCFG | I/O Pin Mapping Configuration | 0xA7 | W | byte | | 0xA8 |
| IOPC0 | Pull Resistor Configuration 0 | 0xAA | R/W | word | 0xAAAA | 0xAB |
| IOPC1 | Pull Resistor Configuration 1 | 0xAC | R/W | word | 0x5555 | 0xAD |
| IOPC2 | Pull Resistor Configuration 2 | 0xAE | R/W | word | 0x5A15 | 0xAF |
| GPIODATA0 | GPIO I/O Data 0 | 0xC0 | R/W | byte | 0x00 | 0xC1 |
| GIOMASK0 | GPIO I/O Mask 0 | 0xC1 | W | byte | | 0xC2 |
| GPIODATA1 | GPIO I/O Data 1 | 0xC2 | R/W | byte | 0x00 | 0xC3 |
| GIOMASK1 | GPIO I/O Mask 1 | 0xC3 | W | byte | | 0xC4 |
| GPIODATA2 | GPIO I/O Data 2 | 0xC4 | R/W | byte | 0x00 | 0xC5 |
| GIOMASK2 | GPIO I/O Mask 2 | 0xC5 | W | byte | | 0xC6 |
| GPIODIR0 | GPIO I/O Direction 0 | 0xC6 | R/W | byte | 0x00 | 0xC7 |
| GPIODIR1 | GPIO I/O Direction 1 | 0xC7 | R/W | byte | 0x00 | 0xC8 |
| GPIODIR2 | GPIO I/O Direction 2 | 0xC8 | R/W | byte | 0x08 | 0xC9 |
| GPIOIS0 | GPIO Int Sense Config 0 | 0xC9 | R/W | byte | 0x00 | 0xCA |
| GPIOIS1 | GPIO Int Sense Config 1 | 0xCA | R/W | byte | 0x00 | 0xCB |
| GPIOIS2 | GPIO Int Sense Config 2 | 0xCB | R/W | byte | 0x00 | 0xCC |
| GPIOIBE0 | GPIO Int Both Edges Config 0 | 0xCC | R/W | byte | 0x00 | 0xCD |
| GPIOIBE1 | GPIO Int Both Edges Config 1 | 0xCD | R/W | byte | 0x00 | 0xCE |
| GPIOIBE2 | GPIO Int Both Edges Config 2 | 0xCE | R/W | byte | 0x00 | 0xCF |
| GPIOIEV0 | GPIO Int Edge Select 0 | 0xCF | R/W | byte | 0xFF | 0xD0 |
| GPIOIEV1 | GPIO Int Edge Select 1 | 0xD0 | R/W | byte | 0xFF | 0xD1 |
| GPIOIEV2 | GPIO Int Edge Select 2 | 0xD1 | R/W | byte | 0xFF | 0xD2 |
| GPIOIE0 | GPIO Interrupt Enable 0 | 0xD2 | R/W | byte | 0x00 | 0xD3 |
| GPIOIE1 | GPIO Interrupt Enable 1 | 0xD3 | R/W | byte | 0x00 | 0xD4 |

Register Map for GPIO Functionality (continued)

| Register Name | Description | Register File Address | Register Type | ACCESS Size | Default value | Next RF Address |
|---------------|-------------------------------|-----------------------|---------------|-------------|---------------|-----------------|
| GPIIOE2 | GPIO Interrupt Enable 2 | 0xD4 | R/W | byte | 0x00 | 0xD5 |
| GPIORIS0 | GPIO Raw Int Status 0 | 0xD6 | R | byte | 0x00 | 0xD7 |
| GPIORIS1 | GPIO Raw Int Status 1 | 0xD7 | R | byte | 0x00 | 0xD8 |
| GPIORIS2 | GPIO Raw Int Status 2 | 0xD8 | R | byte | 0x00 | 0xD9 |
| GPIOMIS0 | GPIO Masked Int Status 0 | 0xD9 | R | byte | 0x00 | 0xDA |
| GPIOMIS1 | GPIO Masked Int Status 1 | 0xDA | R | byte | 0x00 | 0xDB |
| GPIOMIS2 | GPIO Masked Int Status 2 | 0xDB | R | byte | 0x00 | 0xDC |
| GPIOIC0 | GPIO Interrupt Clear 0 | 0xDC | W | byte | | 0xDD |
| GPIOIC1 | GPIO Interrupt Clear 1 | 0xDD | W | byte | | 0xDE |
| GPIOIC2 | GPIO Interrupt Clear 2 | 0xDE | W | byte | | 0xDF |
| GPIOOME0 | GPIO Open Drain Mode Enable 0 | 0xE0 | R/W | byte | 0x00 | 0xE1 |
| GPIOOMS0 | GPIO Open Drain Mode Select 0 | 0xE1 | R/W | byte | 0x00 | 0xE2 |
| GPIOOME1 | GPIO Open Drain Mode Enable 1 | 0xE2 | R/W | byte | 0x00 | 0xE3 |
| GPIOOMS1 | GPIO Open Drain Mode Select 1 | 0xE3 | R/W | byte | 0x00 | 0xE4 |
| GPIOOME2 | GPIO Open Drain Mode Enable 2 | 0xE4 | R/W | byte | 0x08 | 0xE5 |
| GPIOOMS2 | GPIO Open Drain Mode Select 2 | 0xE5 | R/W | byte | 0x00 | 0xE6 |
| GPIOWAKE0 | GPIO Wakeup Enable 0 | 0xE9 | R/W | byte | 0x00 | 0xEA |
| GPIOWAKE1 | GPIO Wakeup Enable 1 | 0xEA | R/W | byte | 0x00 | 0xEB |
| GPIOWAKE2 | GPIO Wakeup Enable 2 | 0xEB | R/W | byte | 0x00 | 0xEC |

REGISTER LAYOUT - Control Bits in LM8325-1 Registers

| Register | Addr. | BIT 7 | BIT 6 | BIT 5 | BIT 4 | BIT 3 | BIT 2 | BIT 1 | BIT 0 |
|------------------------|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| KBDSETTLE | 0x01 | Wait[7:0] | | | | | | | |
| KBDBOUNCE | 0x02 | Wait[7:0] | | | | | | | |
| KBDSIZE | 0x03 | ROW-SIZE3 | ROW-SIZE2 | ROW-SIZE1 | ROW-SIZE0 | COL-SIZE3 | COL-SIZE2 | COL-SIZE1 | COL-SIZE0 |
| KBDDCDCFG ₀ | 0x04 | COL9 | COL8 | COL7 | COL6 | COL5 | COL4 | COL3 | COL2 |
| KBDDCDCFG ₁ | 0x05 | ROW7 | ROW6 | ROW5 | ROW4 | ROW3 | ROW2 | COL11 | COL10 |
| KBDRIS | 0x06 | | | | | RELINT | REVTINT | RKLINT | RSINT |
| KBDMIS | 0x07 | | | | | MELINT | MEVTINT | MKLINT | MSINT |
| KBDIC | 0x08 | SFOFF | | | | | | EVTIC | KBDIC |
| KBDMSK | 0x09 | | | | | MSKELINT | MSKEINT | MSKLINT | MSKSINT |
| KBDCODE0 | 0x0B | MULTIKEY | KEYROW2 | KEYROW1 | KEYROW0 | KEYCOL3 | KEYCOL2 | KEYCOL1 | KEYCOL0 |

REGISTER LAYOUT - Control Bits in LM8325-1 Registers (continued)

| Register | Addr. | BIT 7 | BIT 6 | BIT 5 | BIT 4 | BIT 3 | BIT 2 | BIT 1 | BIT 0 |
|------------------|-------|----------------|------------|--------------|--------------|-------------|----------------|--------------|---------|
| KBDCODE1 | 0x0C | MULTIKEY | KEYROW2 | KEYROW1 | KEYROW0 | KEYCOL3 | KEYCOL2 | KEYCOL1 | KEYCOL0 |
| KBDCODE2 | 0x0D | MULTIKEY | KEYROW2 | KEYROW1 | KEYROW0 | KEYCOL3 | KEYCOL2 | KEYCOL1 | KEYCOL0 |
| KBDCODE3 | 0x0E | MULTIKEY | KEYROW2 | KEYROW1 | KEYROW0 | KEYCOL3 | KEYCOL2 | KEYCOL1 | KEYCOL0 |
| EVTCODE | 0x10 | RELEASE | KEYROW2 | KEYROW1 | KEYROW0 | KEYCOL3 | KEYCOL2 | KEYCOL1 | KEYCOL0 |
| TIMCFG0 | 0x60 | | | | CICIRQ0-MASK | | | | START |
| PWMCFG0 | 0x61 | | | | | CDIRQ0-MASK | PGE | PWMEN | PWMPOL |
| TIMSCAL0 | 0x62 | SCAL[7:0] | | | | | | | |
| TIMCFG1 | 0x68 | | | | CYIRQ1 MASK | | | | START |
| PWMCFG1 | 0x69 | | | | | CDIRQ1-MASK | PGE | PWMEN | PWMPOL |
| TIMSCAL1 | 0x6A | SCAL7 | SCAL6 | SCAL5 | SCAL4 | SCAL3 | SCAL2 | SCAL1 | SCAL0 |
| TIMCFG2 | 0x70 | | | | IRQMASK | | | | START |
| PWMCFG2 | 0x71 | | | | | CDIRQ2-MASK | PGE | PWMEN | PWMPOL |
| TIMSCAL2 | 0x72 | SCAL7 | SCAL6 | SCAL5 | SCAL4 | SCAL3 | SCAL2 | SCAL1 | SCAL0 |
| TIMSWRES | 0x78 | | | | | | SWRES2 | SWRES1 | SWRES0 |
| TIMRIS | 0x7A | | | CDIRQ2 | CDIRQ1 | CDIRQ0 | CICIRQ2 | CICIRQ1 | CICIRQ0 |
| TIMMIS | 0x7B | | | CDIRQ2 | CDIRQ1 | CDIRQ0 | CICIRQ2 | CICIRQ1 | CICIRQ0 |
| TIMIC | 0x7C | | | CDIRQ2 | CDIRQ1 | CDIRQ0 | CICIRQ2 | CICIRQ1 | CICIRQ0 |
| PWMWP | 0x7D | 0 | PWMWP[6:0] | | | | | | |
| PWMCFG (Low) | 0x7E | CMD[7:0] | | | | | | | |
| PWMCFG (High) | 0x7F | CMD[15:8] | | | | | | | |
| I2CSA | 0x80 | SLAVEADDR[7:1] | | | | | | | 0 |
| MFGCODE | 0x80 | MFGBIT[7:0] | | | | | | | |
| SWREV | 0x81 | SWBIT[7:0] | | | | | | | |
| SWRESET | 0x81 | SWBIT[7:0] | | | | | | | |
| RSTCTRL | 0x82 | | | | IRQRST | TIMRST | | KBDRST | GPIRST |
| RSTINTCLR | 0x84 | | | | | | | | IRQCLR |
| CLKMODE | 0x88 | | | | | | | MOD-CTL[1:0] | |
| CLKCFG | 0x89 | CLKSRCSEL[1:0] | | | | | | | |
| CLKEN | 0x8A | CLKOUTEN[1:0] | | | | | TIMEN | | KBDEN |
| AUTOSLP | 0x8B | | | | | | | | ENABLE |
| AUTOSLPTI (Low) | 0x8C | UP-TIME [7:0] | | | | | | | |
| AUTOSLPTI (High) | 0x8D | | | | | | UP-TIME [10:8] | | |
| IRQST | 0x91 | PORIRQ | KBD1RQ | | | TIM2IRQ | TIM1IRQ | TIM0IRQ | GPIIRQ |
| IOCFG | 0xA7 | IOCFGPM [7:0] | | | | | | | |
| IOPC0 (Low) | 0xAA | KPX3PR[1:0] | | KPX2PR[1:0] | | KPX1PR[1:0] | | KPX0PR[1:0] | |
| IOPC0 (High) | 0xAB | KPX7PR[1:0] | | KPX6PR[1:0] | | KPX5PR[1:0] | | KPX4PR[1:0] | |
| IOPC1 (Low) | 0xAC | KPY3PR[1:0] | | KPY2PR[1:0] | | KPY1PR[1:0] | | KPY0PR[1:0] | |
| IOPC1 (High) | 0xAD | KPY7PR[1:0] | | KPY6PR[1:0] | | KPY5PR[1:0] | | KPY4PR[1:0] | |
| IOPC2 (Low) | 0xAE | KPY11PR[1:0] | | KPY10PR[1:0] | | KPY9PR[1:0] | | KPY8PR[1:0] | |
| IOPC2 (High) | 0xAF | | | | | | | | |
| GPIODATA0 | 0xC0 | DATA7 | DATA6 | DATA5 | DATA4 | DATA3 | DATA2 | DATA1 | DATA0 |

REGISTER LAYOUT - Control Bits in LM8325-1 Registers (continued)

| Register | Addr. | BIT 7 | BIT 6 | BIT 5 | BIT 4 | BIT 3 | BIT 2 | BIT 1 | BIT 0 |
|-----------|-------|-----------|-----------|-----------|-----------|------------|------------|-----------|-----------|
| GPIOMASK0 | 0xC1 | MASK7 | MASK6 | MASK5 | MASK4 | MASK3 | MASK2 | MASK1 | MASK0 |
| GPIODATA1 | 0xC2 | DATA15 | DATA14 | DATA13 | DATA12 | DATA11 | DATA10 | DATA9 | DATA8 |
| GPIOMASK1 | 0xC3 | MASK15 | MASK14 | MASK13 | MASK12 | MASK11 | DATA10 | DATA9 | DATA8 |
| GPIODATA2 | 0xC4 | | | | | DATA19 | DATA18 | DATA17 | DATA16 |
| GPIOMASK2 | 0xC5 | | | | | MASK19 | MASK18 | MASK17 | MASK16 |
| GPIODIR0 | 0xC6 | KPX7DIR | KPX6DIR | KPX5DIR | KPX4DIR | KPX3DIR | KPX2DIR | KPX1DIR | KPX0DIR |
| GPIODIR1 | 0xC7 | KPY7DIR | KPY6DIR | KPY5DIR | KPY4DIR | KPY3DIR | KPY2DIR | KPY1DIR | KPY0DIR |
| GPIODIR2 | 0xC8 | | | | | KP11DIR | KPY10DIR | KPY9DIR | KPY8DIR |
| GPIOIS0 | 0xC9 | KPX7IS | KPX6IS | KPX5IS | KPX4IS | KPX3IS | KPX2IS | KPX1IS | KPX0IS |
| GPIOIS1 | 0xCA | KPY7IS | KPY6IS | KPY5IS | KPY4IS | KPY3IS | KPY2IS | KPY1IS | KPY0IS |
| GPIOIS2 | 0xCB | | | | | KPY11IS | KPY10IS | KPY9IS | KPY8IS |
| GPIOIBE0 | 0xCC | KPX7IBE | KPX6IBE | KPX5IBE | KPX4IBE | KPX3IBE | KPX2IBE | KPX1IBE | KPX0IBE |
| GPIOIBE1 | 0xCD | KPY7IBE | KPY6IBE | KPY5IBE | KPY4IBE | KPY3IBE | KPY2IBE | KPY1IBE | KPY0IBE |
| GPIOIBE2 | 0xCE | | | | | KPY11IBE | KPY10IBE | KPY9IBE | KPY8IBE |
| GPIOIEV0 | 0xCF | KPX7EV | KPX6EV | KPX5EV | KPX4EV | KPX3EV | KPX2EV | KPX1EV | KPX0EV |
| GPIOIEV1 | 0xD0 | KPY7EV | KPY6EV | KPY5EV | KPY4EV | KPY3EV | KPY2EV | KPY1EV | KPY0EV |
| GPIOIEV2 | 0xD1 | | | | | KPY11IEV | KPY10IEV | KPY9IEV | KPY8IEV |
| GPIOIE0 | 0xD2 | KPX7IE | KPX6IE | KPX5IE | KPX4IE | KPX3IE | KPX2IE | KPX1IE | KPX0IE |
| GPIOIE1 | 0xD3 | KPY7IE | KPY6IE | KPY5IE | KPY4IE | KPY3IE | KPY2IE | KPY1IE | KPY0IE |
| GPIOIE2 | 0xD4 | | | | | KPY11IE | KPY10IE | KPY9IE | KPY8IE |
| GPIORIS0 | 0xD6 | KPX7RIS | KPX6RIS | KPX5RIS | KPX4RIS | KPX3RIS | KPX2RIS | KPX1RIS | KPX0RIS |
| GPIORIS1 | 0xD7 | KPY7RIS | KPY6RIS | KPY5RIS | KPY4RIS | KPY3RIS | KPY2RIS | KPY1RIS | KPY0RIS |
| GPIORIS2 | 0xD8 | | | | | KPY11RIS | KPY10RIS | KPY9RIS | KPY8RIS |
| GPIOMIS0 | 0xD9 | KPX7MIS | KPX6MIS | KPX5MIS | KPX4MIS | KPX3MIS | KPX2MIS | KPX1MIS | KPX0MIS |
| GPIOMIS1 | 0xDA | KPY7MIS | KPY6MIS | KPY5MIS | KPY4MIS | KPY3MIS | KPY2MIS | KPY1MIS | KPY0MIS |
| GPIOMIS2 | 0xDB | | | | | KPY11MIS | KPY10MIS | KPY9MIS | KPY8MIS |
| GPIOIC0 | 0xDC | KPX7IC | KPX6IC | KPX5IC | KPX4IC | KPX3IC | KPX2IC | KPX1IC | KPX0IC |
| GPIOIC1 | 0xDD | KPY7IC | KPY6IC | KPY5IC | KPY4IC | KPY3IC | KPY2IC | KPY1IC | KPY0IC |
| GPIOIC2 | 0xDE | | | | | KPY11IC | KPY10IC | KPY9IC | KPY8IC |
| GPIOOME0 | 0xE0 | KPX7ODE | KPX6ODE | KPX5ODE | KPX4ODE | KPX3ODE | KPX2ODE | KPX1ODE | KPX0ODE |
| GPIOOMS0 | 0xE1 | KPX7ODM | KPX6ODM | KPX5ODM | KPX4ODM | KPX3ODM | KPX2ODM | KPX1ODM | KPX0ODM |
| GPIOOME1 | 0xE2 | KPY7ODE | KPY6ODE | KPY5ODE | KPY4ODE | KPY3ODE | KPY2ODE | KPY1ODE | KPY0ODE |
| GPIOOMS1 | 0xE3 | KPY7ODM | KPY6ODM | KPY5ODM | KPY4ODM | KPY3ODM | KPY2ODM | KPY1ODM | KPY0ODM |
| GPIOOME2 | 0xE4 | | | | | KPY11ODE | KPY10ODE | KPY9ODE | KPY8ODE |
| GPIOOMS2 | 0xE5 | | | | | KPY11ODM | KPY10ODM | KPY9ODM | KPY8ODM |
| GPIOWAKE0 | 0xE9 | KPX7 WAKE | KPX6 WAKE | KPX5 WAKE | KPX4 WAKE | KPX3 WAKE | KPX2 WAKE | KPX1 WAKE | KPX0 WAKE |
| GPIOWAKE1 | 0xEA | KPY7 WAKE | KPY6 WAKE | KPY5 WAKE | KPY4 WAKE | KPY3 WAKE | KPY2 WAKE | KPY1 WAKE | KPY0 WAKE |
| GPIOWAKE2 | 0xEB | | | | | KPY11 WAKE | KPY10 WAKE | KPY9 WAKE | KPY8 WAKE |

REVISION HISTORY

| Changes from Original (March 2013) to Revision A | Page |
|--|--------------------|
| • Changed layout of National Data Sheet to TI format | 49 |

PACKAGING INFORMATION

| Orderable Device | Status (1) | Package Type | Package Drawing | Pins | Package Qty | Eco Plan (2) | Lead finish/ Ball material (6) | MSL Peak Temp (3) | Op Temp (°C) | Device Marking (4/5) | Samples |
|--------------------|---------------|--------------|--------------------|------|----------------|-----------------|--------------------------------------|----------------------|--------------|-------------------------|----------------|
| LM8325DGR8-1/NOPB | ACTIVE | csBGA | NYA | 25 | 1000 | RoHS & Green | SNAGCU | Level-1-260C-UNLIM | -40 to 85 | 25-1 | Samples |
| LM8325DGR8X-1/NOPB | ACTIVE | csBGA | NYA | 25 | 3500 | RoHS & Green | SNAGCU | Level-1-260C-UNLIM | -40 to 85 | 25-1 | Samples |

(1) The marketing status values are defined as follows:

ACTIVE: Product device recommended for new designs.

LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

PREVIEW: Device has been announced but is not in production. Samples may or may not be available.

OBsolete: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".

RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.

Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

Important Information and Disclaimer:The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

TAPE AND REEL INFORMATION

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE


*All dimensions are nominal

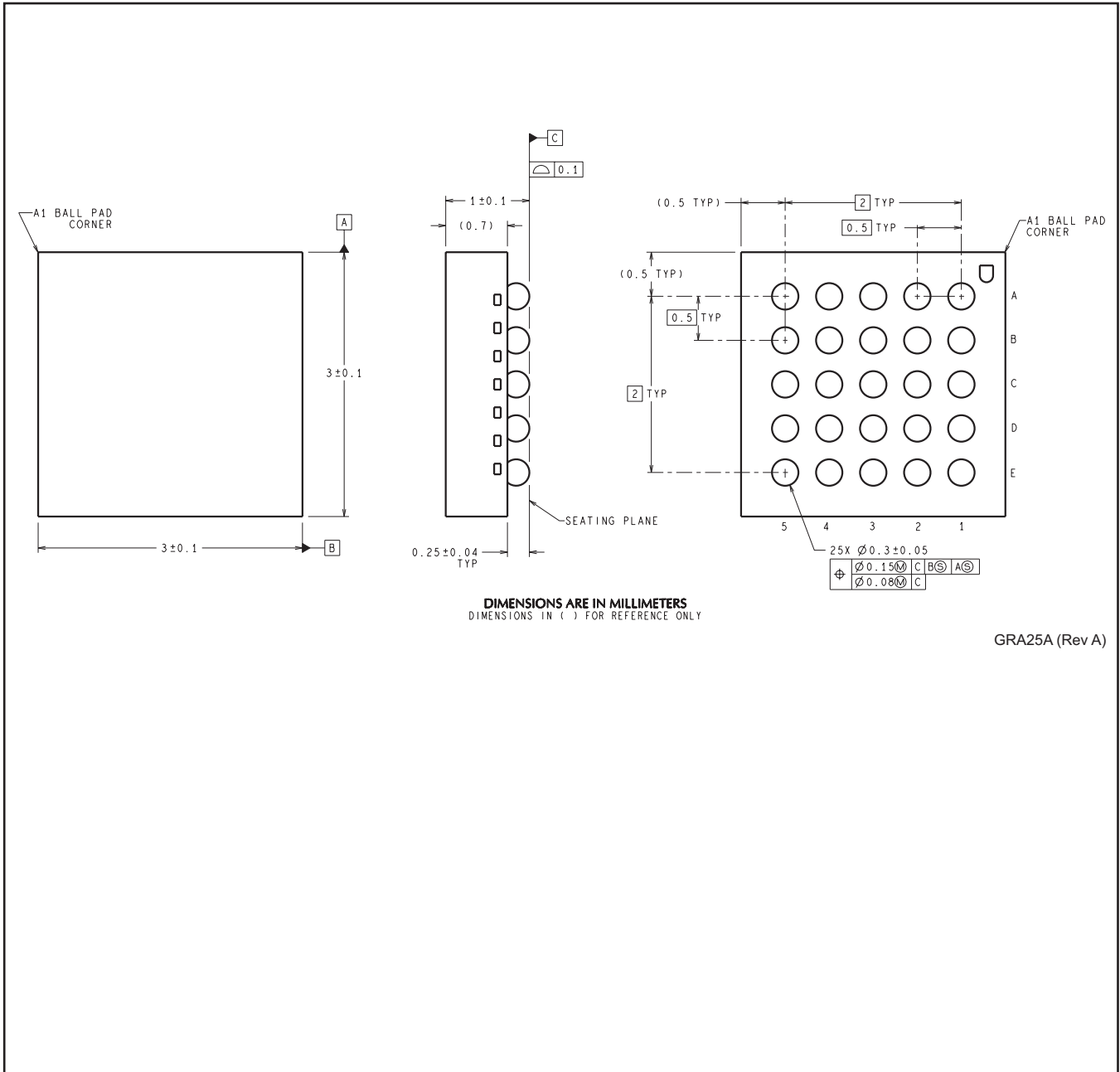
| Device | Package Type | Package Drawing | Pins | SPQ | Reel Diameter (mm) | Reel Width W1 (mm) | A0 (mm) | B0 (mm) | K0 (mm) | P1 (mm) | W (mm) | Pin1 Quadrant |
|--------------------|--------------|-----------------|------|------|--------------------|--------------------|---------|---------|---------|---------|--------|---------------|
| LM8325DGR8-1/NOPB | csBGA | NYA | 25 | 1000 | 178.0 | 12.4 | 3.3 | 3.3 | 1.6 | 8.0 | 12.0 | Q1 |
| LM8325DGR8X-1/NOPB | csBGA | NYA | 25 | 3500 | 330.0 | 12.4 | 3.3 | 3.3 | 1.6 | 8.0 | 12.0 | Q1 |

TAPE AND REEL BOX DIMENSIONS


*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Length (mm) | Width (mm) | Height (mm) |
|--------------------|--------------|-----------------|------|------|-------------|------------|-------------|
| LM8325DGR8-1/NOPB | csBGA | NYA | 25 | 1000 | 208.0 | 191.0 | 35.0 |
| LM8325DGR8X-1/NOPB | csBGA | NYA | 25 | 3500 | 367.0 | 367.0 | 35.0 |

NYA0025A



IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2021, Texas Instruments Incorporated