General Description

The DA14581 integrated circuit is an optimized version of the DA14580, offering a reduced boot time and supporting up to 8 connections. It has a fully integrated radio transceiver and baseband processor for *Bluetooth® low energy*. It can be used as a standalone application processor or as a data pump in hosted systems.

The DA14581 supports a flexible memory architecture for storing Bluetooth profiles and custom application code, which can be updated over the air (OTA). The qualified *Bluetooth low energy* protocol stack and the HCI ready software are stored in a dedicated ROM. All software runs on the ARM® Cortex®-M0 processor via a simple scheduler.

The *Bluetooth low energy* firmware includes the L2CAP service layer protocols, Security Manager (SM), Attribute Protocol (ATT), the Generic Attribute Profile (GATT) and the Generic Access Profile (GAP). All profiles published by the Bluetooth SIG as well as custom profiles are supported.

The transceiver interfaces directly to the antenna and is fully compliant with the *Bluetooth 4.2* standard.

The DA14581 has dedicated hardware for the Link Layer implementation of *Bluetooth low energy* and interface controllers for enhanced connectivity capabilities.

Features

- Complies with *Bluetooth V4.2*, ETSI EN 300 328 and EN 300 440 Class 2 (Europe), FCC CFR47 Part 15 (US) and ARIB STD-T66 (Japan)
- Supports up to 8 Bluetooth low energy connections
- Fast cold boot in less than 30 ms
- Processing power
  - 16 MHz 32 bit ARM Cortex-M0 with SWD interface
  - Dedicated Link Layer Processor
  - AES-128 bit encryption Processor
- Memories
  - 32 kB One-Time-Programmable (OTP) memory
  - 42 kB System SRAM
  - 84 kB ROM
  - 8 kB Retention SRAM
- Power management
  - Integrated Buck/Boost DC-DC converter
  - P0, P1 and P2 ports with 3.3 V tolerance
  - Easy decoupling of only 4 supply pins
  - Supports coin (typ. 3.0 V) and alkaline (typ. 1.5 V) battery cells
  - 10-bit ADC for battery voltage measurement
- Digital controlled oscillators
  - 16 MHz crystal (±20 ppm max) and RC oscillator
  - 32 kHz crystal (±50 ppm, ±500 ppm max) and RCX oscillator
- General purpose, Capture and Sleep timers
- Digital interfaces
  - Gen. purpose I/Os: 14 (WLCSP34), 24 (QFN40)
  - 2 UARTs with hardware flow control up to 1 MBd
  - SPI+™ interface
  - I2C bus at 100 kHz, 400 kHz
  - 3-axes capable Quadrature Decoder
- Analog interfaces
  - 4-channel 10-bit ADC
- Radio transceiver
  - Fully integrated 2.4 GHz CMOS transceiver
  - Single wire antenna: no RF matching or RX/TX switching required
  - Supply current at VBAT3V:
    TX: 3.4 mA, RX: 3.7 mA (with ideal DC-DC)
  - 0 dBm transmit output power
  - -20 dBm output power in “Near Field Mode”
  - -93 dBm receiver sensitivity
- Packages:
  - WLCSP 34 pins, 2.436 mm x 2.436 mm
  - QFN 40 pins, 5 mm x 5 mm

System Diagram

Contents

1 Block Diagram

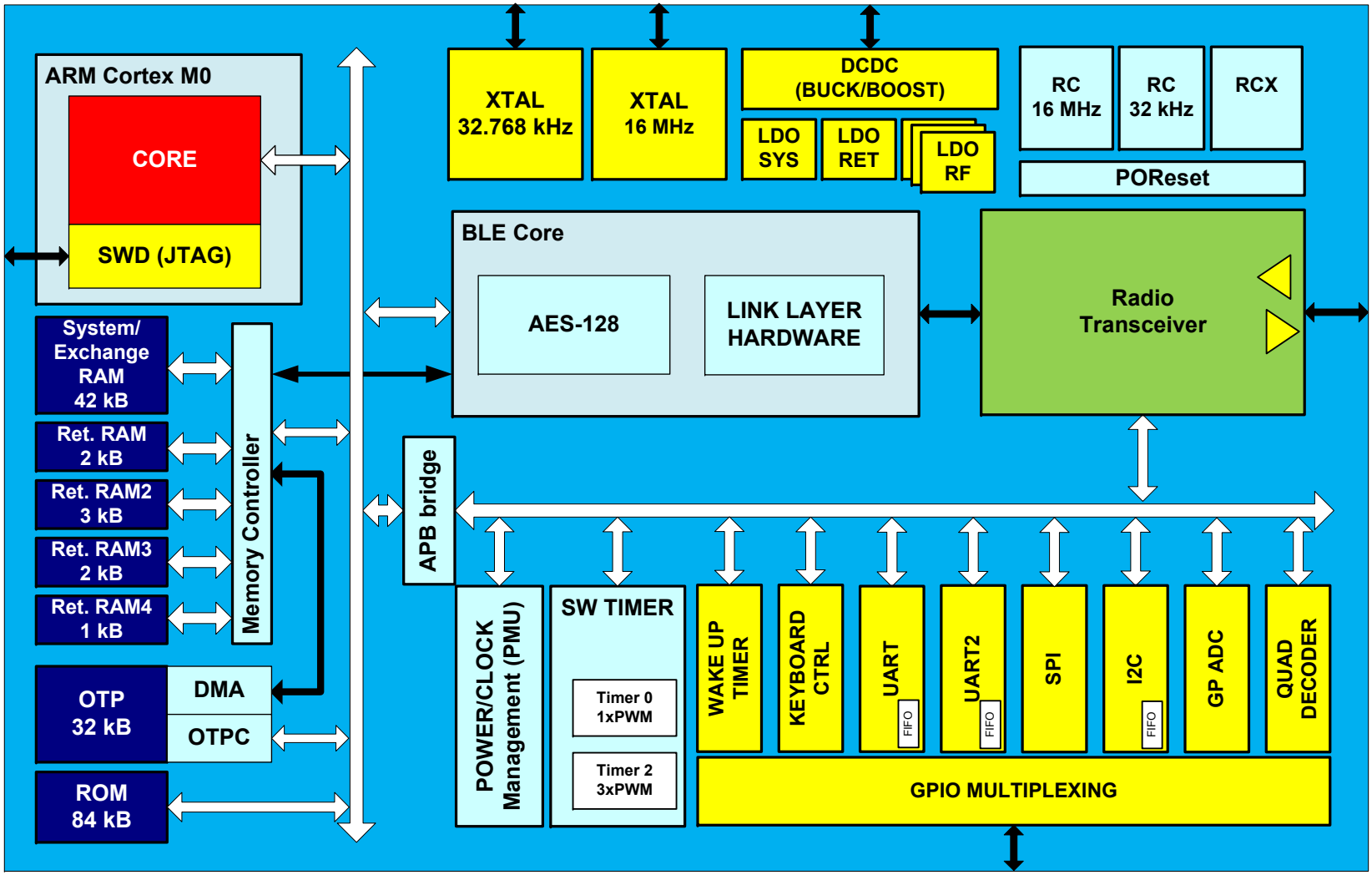


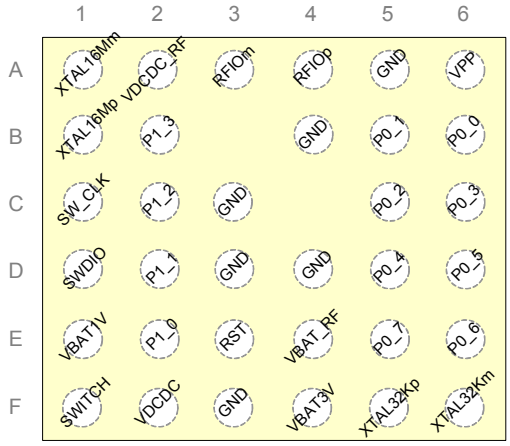
Figure 1: DA14581 Block Diagram

2 Pinout

The DA14581 comes in two packages:

- A Wafer Level Chip Scale Package (WLCSP) with 34 balls
- A Quad Flat Package No Leads (QFN) with 40 pins

The actual pin/ball assignment is depicted in the following figures:



DA14581 (Top View)

Figure 2: WLCSP Ball Assignment

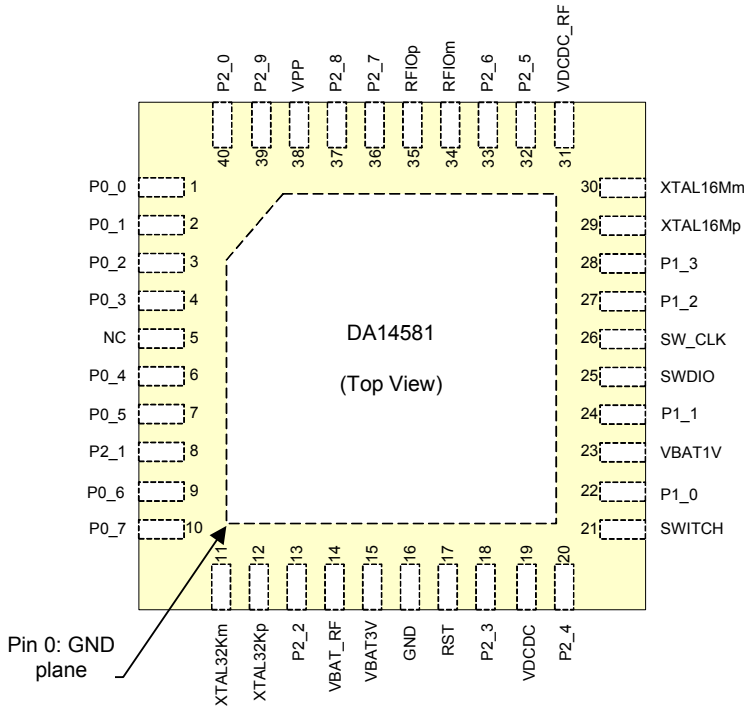


Figure 3: QFN40 Pin Assignment

Table 1: Pin Description

Pin Name	Type	Drive (mA)	Reset State	Description
General Purpose I/Os				
P0_0 P0_1 P0_2 P0_3 P0_4 P0_5 P0_6 P0_7	DIO DIO DIO DIO DIO DIO DIO DIO	4.8	I-PD I-PD I-PD I-PD I-PD I-PD I-PD I-PD	INPUT/OUTPUT with selectable pull up/down resistor. Pull-down enabled during and after reset. General purpose I/O port bit or alternate function nodes. Contains state retention mechanism during power down.
P1_0 P1_1 P1_2 P1_3 P1_4/SWCLK P1_5/SW_DIO	DIO DIO DIO DIO DIO DIO	4.8	I-PD I-PD I-PD I-PD I-PD I-PU	INPUT/OUTPUT with selectable pull up/down resistor. Pull-down enabled during and after reset. General purpose I/O port bit or alternate function nodes. Contains state retention mechanism during power down. This signal is the JTAG clock by default This signal is the JTAG data I/O by default
P2_0 P2_1 P2_2 P2_3 P2_4 P2_5 P2_6 P2_7 P2_8 P2_9	DIO DIO DIO DIO DIO DIO DIO DIO DIO DIO	4.8	I-PD I-PD I-PD I-PD I-PD I-PD I-PD I-PD I-PD I-PD	INPUT/OUTPUT with selectable pull up/down resistor. Pull-down enabled during and after reset. General purpose I/O port bit or alternate function nodes. Contains state retention mechanism during power down. **NOTE: This port is only available on the QFN40 package.**
P3_0 to P3_7	DIO	4.8	I-PD	Not supported.
Debug interface				
SWDIO/P1_5	DIO	4.8	I-PU	INPUT/OUTPUT. JTAG Data input/output. Bidirectional data and control communication. Can also be used as a GPIO
SW_CLK/ P1_4	DIO	4.8	I-PD	INPUT JTAG clock signal. Can also be used as a GPIO
Clocks				
XTAL16Mp	AI			INPUT. Crystal input for the 16 MHz XTAL
XTAL16Mm	AO			OUTPUT. Crystal output for the 16 MHz XTAL
XTAL32kp	AI			INPUT. Crystal input for the 32.768 kHz XTAL
XTAL32km	AO			OUTPUT. Crystal output for the 32.768 kHz XTAL
Quadrature Decoder				
QD_CHA_X	DI			INPUT. Channel A for the X axis. Mapped on Px ports
QD_CHB_X	DI			INPUT. Channel B for the X axis. Mapped on Px ports
QD_CHA_Y	DI			INPUT. Channel A for the Y axis. Mapped on Px ports
QD_CHB_Y	DI			INPUT. Channel B for the Y axis. Mapped on Px ports
QD_CHA_Z	DI			INPUT. Channel A for the Z axis. Mapped on Px ports
QD_CHB_Z	DI			INPUT. Channel B for the Z axis. Mapped on Px ports
SPI Bus Interface				
SPI_CLK	DO			INPUT/OUTPUT. SPI Clock. Mapped on Px ports
SPI_DI	DI			INPUT. SPI Data input. Mapped on Px ports
SPI_DO	DO			OUTPUT. SPI Data output. Mapped on Px ports

Table 1: Pin Description

Pin Name	Type	Drive (mA)	Reset State	Description
SPI_EN	DI			INPUT. SPI Clock enable. Mapped on Px ports
I2C Bus Interface				
SDA	DIO/DIOD			INPUT/OUTPUT. I2C bus Data with open drain port. Mapped on Px ports
SCL	DIO/DIOD			INPUT/OUTPUT. I2C bus Clock with open drain port. In open drain mode, SCL is monitored to support bit stretching by a slave. Mapped on Px ports.
UART Interface				
UTX	DO			OUTPUT. UART transmit data. Mapped on Px ports
URX	DI			INPUT. UART receive data. Mapped on Px ports
URTS	DO			OUTPUT. UART Request to Send. Mapped on Px ports
UCTS	DI			INPUT. UART Clear to Send. Mapped on Px ports
UTX2	DO			OUTPUT. UART 2 transmit data. Mapped on Px ports
URX2	DI			INPUT. UART 2 receive data. Mapped on Px ports
URTS2	DO			OUTPUT. UART 2 Request to Send. Mapped on Px ports
UCTS2	DI			INPUT. UART 2 Clear to Send. Mapped on Px ports
Analog Interface				
ADC[0]	AI			INPUT. Analog to Digital Converter input 0. Mapped on P0[0]
ADC[1]	AI			INPUT. Analog to Digital Converter input 1. Mapped on P0[1]
ADC[2]	AI			INPUT. Analog to Digital Converter input 2. Mapped on P0[2]
ADC[3]	AI			INPUT. Analog to Digital Converter input 3. Mapped on P0[3]
Radio Transceiver				
RFIOp	AIO			RF input/output. Impedance 50 Ω.
RFIOm	AIO			RF ground
Miscellaneous				
RST	DI			INPUT. Reset signal (active high). Must be connected to GND if not used.
VBAT_RF	AIO			Connect to VBAT3V on the PCB
VDCDC_RF	AIO			Connect to VDCDC on the PCB
VPP	AI			INPUT. This pin is used while OTP programming and testing. OTP programming: VPP = 6.7 V ± 0.1 V OTP Normal operation: leave VPP floating
Power Supply				
VBAT3V	AIO			INPUT/OUTPUT. Battery connection. Used for a single coin battery (3 V). If an alkaline or a NiMH battery (1.5 V) is attached to pin VBAT1V, this is the second output of the DC-DC converter.
VBAT1V	AI			INPUT. Battery connection. Used for an alkaline or a NiMh battery (1.5 V). If a single coin battery (3 V) is attached to pin VBAT3V,this pin must be connected to GND.
SWITCH	AIO			INPUT/OUTPUT. Connection for the external DC-DC converter inductor.
VDCDC	AO			Output of the DC-DC converter
GND	AIO	-	-	Ground

3 Ordering Information

Table 2: Ordering Information (Samples)

Part Number	Package	Size (Mm)	Shipment Form	Pack Quantity
DA14581-00UNA	WLCSP34	2.436 x 2.436	Mini-reel	50/100/1000
DA14581-00AT1	QFN40	5 x 5	Tray	50

Table 3: Ordering Information (Production)

Part Number	Package	Size (Mm)	Shipment Form	Pack Quantity
DA14581-00UNA	WLCSP34	2.436 x 2.436	Mini-reel	5000
DA14581-00AT2	QFN40	5 x 5	Reel	5000

Part Number Legend:

DA14581-nnXYZ

nn: chip revision number

XY: package code

Z: packing method

4 System Overview

The DA14581 contains the following internal blocks:

4.1 ARM CORTEX–M0 CPU

The Cortex-M0 processor is a 32-bit Reduced Instruction Set Computing (RISC) processor with a von Neumann architecture (single bus interface). It uses an instruction set called Thumb, which was first supported in the ARM7TDMI processor; however, several newer instructions from the ARMv6 architecture and a few instructions from the Thumb-2 technology are also included. Thumb-2 technology extended the previous Thumb instruction set to allow all operations to be carried out in one CPU state. The instruction set in Thumb-2 includes both 16-bit and 32-bit instructions; most instructions generated by the C compiler use the 16-bit instructions, and the 32-bit instructions are used when the 16-bit version cannot carry out the required operations. This results in high code density and avoids the overhead of switching between two instruction sets.

In total, the Cortex-M0 processor supports only 56 base instructions, although some instructions can have more than one form. Although the instruction set is small, the Cortex-M0 processor is highly capable because the Thumb instruction set is highly optimized.

Academically, the Cortex-M0 processor is classified as load-store architecture, as it has separate instructions for reading and writing to memory, and instructions for arithmetic or logical operations that use registers.

Features

- Thumb instruction set. Highly efficient, high code density and able to execute all Thumb instructions from the ARM7TDMI processor.
- High performance. Up to 0.9 DMIPS/MHz (Dhrystone 2.1) with fast multiplier.
- Built-in Nested Vectored Interrupt Controller (NVIC). This makes interrupt configuration and coding of exception handlers easy. When an interrupt request is taken, the corresponding interrupt handler is executed automatically without the need to determine the exception vector in software.
- Interrupts can have four different programmable priority levels. The NVIC automatically handles nested interrupts.
- The design is configured to respond to exceptions (e.g. interrupts) as soon as possible (minimum 16 clock cycles).
- Non maskable interrupt (NMI) input for safety critical systems.
- Easy to use and C friendly. There are only two modes (Thread mode and Handler mode). The whole application, including exception handlers, can be written in C without any assembler.
- Built-in System Tick timer for OS support. A 24-bit timer with a dedicated exception type is included in the architecture, which the OS can use as a tick timer or as a general timer in other applications without an OS.
- SuperVisor Call (SVC) instruction with a dedicated SVC exception and PendSV (Pendable SuperVisor service) to support various operations in an embedded OS.
- Architecturally defined sleep modes and instructions to enter sleep. The sleep features allow power consumption to be reduced dramatically. Defining sleep modes as an architectural feature makes porting of software easier because sleep is entered by a specific instruction rather than implementation defined control registers.
- Fault handling exception to catch various sources of errors in the system.
- Support for 24 interrupts.
- Little endian memory support.
- Wake up Interrupt Controller (WIC) to allow the processor to be powered down during sleep, while still allowing interrupt sources to wake up the system.
- Halt mode debug. Allows the processor activity to stop completely so that register values can be accessed and modified. No overhead in code size and stack memory size.
- CoreSight technology. Allows memories and peripherals to be accessed from the debugger without halting the processor.
- Supports Serial Wire Debug (SWD) connections. The serial wire debug protocol can handle the same debug features as the JTAG, but it only requires two wires and is already supported by a number of debug solutions from various tools vendors.
- Four (4) hardware breakpoints and two (2) watch points.
- Breakpoint instruction support for an unlimited number of software breakpoints.
- Programmer's model similar to the ARM7TDMI processor. Most existing Thumb code for the ARM7TDMI processor can be reused. This also makes it easy for ARM7TDMI users, as there is no need to learn a new instruction set.

4.2 BLUETOOTH LOW ENERGY

4.2.1 BLE Core

The BLE (Bluetooth low energy) core is a qualified Bluetooth baseband controller compatible with the Bluetooth low energy 4.2 specification and it is in charge of packet encoding/decoding and frame scheduling.

Features

- All device classes support (Broadcaster, Central, Observer, Peripheral)

- All packet types (Advertising / Data / Control)
- Encryption (AES / CCM)
- Bit stream processing (CRC, Whitening)
- FDMA/TDMA/events formatting and synchronization
- Frequency hopping calculation
- Operating clock 16 MHz or 8 MHz
- Low power modes supporting 32.0 kHz or 32.768 kHz
- Supports power down of the baseband during the protocol's idle periods
- AHB Slave interface for register file access
- AHB Slave interface for Exchange Memory access of CPU via BLE core
- AHB Master interface for direct access of BLE core to Exchange Memory space

4.2.2 Radio Transceiver

The Radio Transceiver implements the RF part of the Bluetooth low energy protocol. Together with the Bluetooth 4.2 PHY layer, this provides a 93 dB RF link budget for reliable wireless communication.

All RF blocks are supplied by on-chip low-drop out-regulators (LDOs). The bias scheme is programmable per block and optimized for minimum power consumption.

The Bluetooth LE radio comprises the Receiver, Transmitter, Synthesizer, Rx/Tx combiner block, and Biasing LDOs.

Features

- Single ended RFIO interface, 50 Ω matched
- Alignment free operation
- -93 dBm receiver sensitivity
- 0 dBm transmit output power
- Ultra low power consumption
- Fast frequency tuning minimizes overhead

4.2.3 SmartSnippets™

The DA14580 comes complete with Dialog's SmartSnippets™ Bluetooth Software platform which includes a qualified Bluetooth Smart single-mode stack on chip. Numerous Bluetooth Smart profiles for consumer wellness, sport, fitness, security and proximity applications are supplied as standard, while additional customer profiles can be developed and added as needed.

The SmartSnippets™ software development environment is based on Keil™'s uVision mature tools and contains example application code for both embedded and hosted modes.

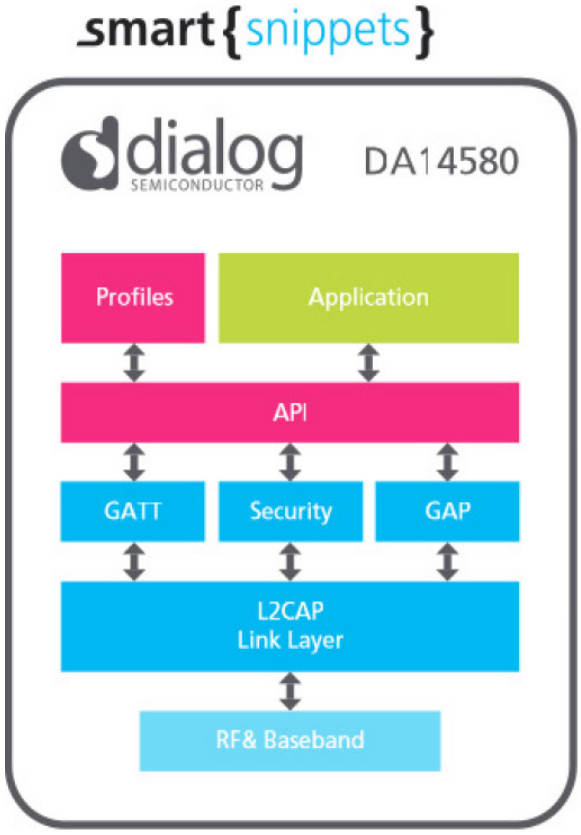


Figure 4: SmartSnippets Stack

Apart from the protocol stack, the Software platform supports a Hardware Abstraction Layer (HAL) which enables easy access to peripheral's features from a programmer's point of view, as presented in the following figure.

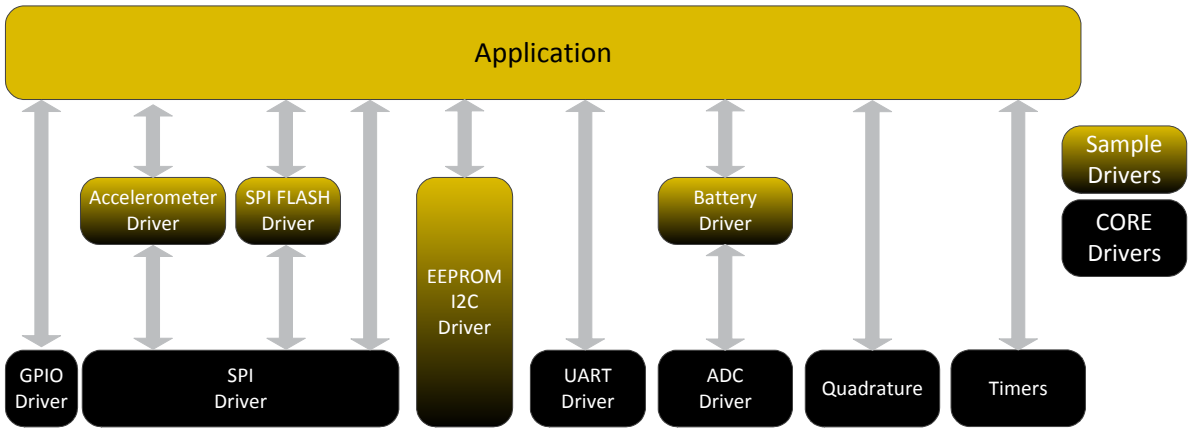


Figure 5: Hardware Abstraction Layer

Core drivers are provided for each interface of the DA14580 enabling optimized usage of the hardware's capabilities. These drivers provide an easy-to-use interface towards the hardware engines without having to interfere with the register programming directly.

On top of the core drivers, a number of sample drivers is also provided enabling communication with basic Bluetooth Smart application components: accelerometers, FLASH/EEPROM non-volatile memories, etc.

4.3 MEMORIES

The following memories are part of the DA14580's internal blocks:

ROM. This is a 84 kB ROM containing the Bluetooth low energy protocol stack as well as the boot code sequence.

OTP. This is a 32 kB One-Time Programmable memory array, used to store the application code as well as Bluetooth low energy profiles. It also contains the system configuration and calibration data.

System SRAM. This is a 42 kB system SRAM (SysRAM) which is primarily used for mirroring the program code from the OTP when the system wakes/powers up. It also serves as Data RAM for intermediate variables and various data that the protocol requires. Optionally, it can be used as extra memory space for the BLE TX and RX data structures.

Retention RAMs. These are 4 special low leakage SRAM cells (2 kB + 2 kB + 3 kB + 1 kB) used to store various data of the Bluetooth low energy protocol as well as the system's global variables and processor stack when the system goes into Deep Sleep mode. Storage of this data ensures secure and quick configuration of the BLE Core after the system wakes up. Every cell can be powered on or off according to the application needs for retention area when in Deep Sleep mode.

4.4 FUNCTIONAL MODES

The DA14581 is optimized for deeply embedded applications such as health monitoring, sports measuring, human interaction devices etc. Customers are able to develop and test their own applications. Upon completion of the development, the application code can be programmed into the OTP. In general, the system has three functional modes of operation:

A. Development Mode: During this phase application code is developed using the ARM Cortex-M0 SW environment. The compiled code is then downloaded into the System RAM or any Retention RAMs by means of SWD (JTAG) or any serial interface (e.g. UART). Address 0x00 is remapped to the physical memory that contains the code and the CPU is configured to reset and execute code from the remapped device. This mode is enabling application development, debugging and on-the-fly testing.

B. Normal Mode: After the application is ready and verified, the code can be burned into the OTP. When the system boots/wakes up, the DMA of the OTP controller will automatically copy the program code from the OTP into the system RAM. Next, a SW reset or a jump to the System RAM occurs and code execution is started. Hence, in this mode, the system is autonomous, contains the required SW in OTP and is ready for integration into the final product.

C. Calibration Mode: Between Development and Normal mode, there is an intermediate stage where the chip needs to be calibrated with respect to two important features:

- Programming of the Bluetooth device address
- Programming of the trimming value for the external 16 MHz crystal.

This mode of operation applies to the final product and is performed by the customer. During this phase, certain fields in the OTP should be programmed

4.5 POWER MODES

There are four different power modes in the DA14580:

- *Active mode*: System is active and operates at full speed.
- *Sleep mode*: No power gating has been programmed, the ARM CPU is idle, waiting for an interrupt. PD_SYS is on. PD_PER and PED_RAD depending on the programmed enabled value.
- *Extended Sleep mode*: All power domains are off except for the PD_AON, the programmed PD_RRx and the PD_SR. Since the SysRAM retains its data, no OTP mirroring is required upon waking up the system.
- *Deep Sleep mode*: All power domains are off except for the PD_AON and the programmed PD_RRx. This mode dissipates the minimum leakage power. However, since the SysRAM has not retained its data, an OTP mirror action is required upon waking up the system.

4.6 INTERFACES

4.6.1 UARTs

The UART is compliant to the industry-standard 16550 and is used for serial communication with a peripheral, modem (data carrier equipment, DCE) or data set. Data is written from a master (CPU) over the APB bus to the UART and it is converted to serial form and transmitted to the destination device. Serial data is also received by the UART and stored for the master (CPU) to read back.

There is no DMA support on the UART block since its contains internal FIFOs. Both UARTs support hardware flow control signals (RTS, CTS, DTR, DSR).

Features

- 16 bytes Transmit and receive FIFOs
- Hardware flow control support (CTS/RTS)
- Shadow registers to reduce software overhead and also include a software programmable reset
- Transmitter Holding Register Empty (THRE) interrupt mode
- IrDA 1.0 SIR mode supporting low power mode.
- Functionality based on the 16550 industry standard:
- Programmable character properties, such as number of data bits per character (5-8), optional
- parity bit (with odd or even select) and number of stop bits (1, 1.5 or 2)
- Line break generation and detection
- Prioritized interrupt identification
- Programmable serial data baud rate as calculated by the following: baud rate = (serial clock frequency)/(divisor).

4.6.2 SPI+

This interface supports a subset of the Serial Peripheral Interface (SPI™). The serial interface can transmit and receive 8, 16 or 32 bits in master/slave mode and transmit 9 bits in master mode. The SPI+ interface has enhanced functionality with bidirectional 2x16-bit word FIFOs.

SPI is a trademark of Motorola, Inc.

Features

- Slave and Master mode
- 8 bit, 9 bit, 16 bit or 32 bit operation
- Clock speeds up to 16 MHz for the SPI controller. Programmable output frequencies of SPI source clock divided by 1, 2, 4, 8
- SPI clock line speed up to 8 MHz
- SPI mode 0, 1, 2, 3 support (clock edge and phase)
- Programmable SPI_DO idle level
- Maskable Interrupt generation
- Bus load reduction by unidirectional writes-only and reads-only modes.

Built-in RX/TX FIFOs for continuous SPI bursts.

4.6.3 I2C Interface

The I2C interface is a programmable control bus that provides support for the communications link between Integrated Circuits in a system. It is a simple two-wire bus with a software-defined protocol for system control, which is used in temperature sensors and voltage level translators to EEPROMs, general-purpose I/O, A/D and D/A converters.

Features

- Two-wire I2C serial interface consists of a serial data line (SDA) and a serial clock (SCL)
- Two speeds are supported:
- Standard mode (0 to 100 kbit/s)
- Fast mode (<= 400 kbit/s)
- Clock synchronization
- 32 deep transmit/receive FIFOs
- Master transmit, Master receive operation
- 7 or 10-bit addressing
- 7 or 10-bit combined format transfers
- Bulk transmit mode
- Default slave address of 0x055

- Interrupt or polled-mode operation
- Handles Bit and Byte waiting at both bus speeds
- Programmable SDA hold time

4.6.4 General Purpose ADC

The DA14581 is equipped with a high-speed ultra low power 10-bit general purpose Analog-to-Digital Converter (GPADC). It can operate in unipolar (single ended) mode as well as in bipolar (differential) mode. The ADC has its own voltage regulator (LDO) of 1.2 V, which represents the full scale reference voltage.

Features

- 10-bit dynamic ADC with 65 ns conversion time
- Maximum sampling rate 3.3 Msample/s
- Ultra low power (5 μA typical supply current at 100 ksample/s)
- Single-ended as well as differential input with two input scales
- Four single-ended or two differential external input channels
- Battery monitoring function
- Chopper function
- Offset and zero scale adjust
- Common-mode input level adjust

4.6.5 Quadrature Decoder

This block decodes the pulse trains from a rotary encoder to provide the step and the direction of the movement of an external device. Three axes (X, Y, Z) are supported.

The integrated quadrature decoder can automatically decode the signals for the X, Y and Z axes of a HID input device, reporting step count and direction: the channels are expected to provide a pulse train with 90 degrees phase difference; depending on whether the reference channel is leading or lagging, the direction can be determined.

This block can be used for waking up the chip as soon as there is any kind of movement from the external device connected to it.

Features

- Three 16-bit signed counters that provide the step count and direction on each of the axes (X, Y and Z)
- Programmable system clock sampling at maximum 16 MHz.
- APB interface for control and programming
- Programmable source from P0, P1 and P2 ports
- Digital filter on the channel inputs to avoid spikes

4.6.6 Keyboard Controller

The Keyboard controller can be used for debouncing the incoming GPIO signals when implementing a keyboard scanning engine. It generates an interrupt to the CPU (KEYBR_IRQ).

In parallel, five extra interrupt lines can be triggered by a state change on 32 selectable GPIOs (GPIOx_IRQ).

Features

- Monitors any of the 32 available GPIOs (12 in the WLCSP package, 22 in the QFN40 and 32 in the QFN48)
- Generates a keyboard interrupt on key press or key release
- Implements debouncing time from 0 up to 63 ms

Supports five separate interrupt generation lines from GPIO toggling

4.6.7 Input/Output Ports

The DA14581 has software-configurable I/O pin assignment, organized into ports Port 0, Port 1 and Port 2. Port 2 is only available in the QFN40 package.

Note: Port 3 is not supported in the DA14581.

Features

- Port 0: 8 pins, Port 1: 6 pins (including SW_CLK and SWDIO), Port 2: 10 pins
- Fully programmable pin assignment
- Selectable 25 kΩ pull-up, pull-down resistors per pin
- Pull-up voltage either VBAT3V (BUCK mode) or VBAT1V (BOOST mode) configurable per pin
- Fixed assignment for analog pin ADC[3:0]
- Pins retain their last state when system enters the Extended or Deep Sleep mode.

4.7 TIMERS

4.7.1 General Purpose Timers

The Timer block contains 2 timer modules that are software controlled, programmable and can be used for various tasks.

Timer 0

- 16-bit general purpose timer
- Ability to generate 2 Pulse Width Modulated signals (PWM0 and PWM1, with common programming)
- Programmable output frequency:

$$f = \frac{(16,\ 8,\ 4,\ 2\ MHz\ or\ 32\ kHz)}{(M+1)+(N+1)}$$

with N = 0 to (2^{16}-1), M = 0 to (2^{16}-1)

- Programmable duty cycle:

$$\delta = \frac{M+1}{(M+1)+(N+1)} \times 100\ \%$$

- Separately programmable interrupt timer:

$$T = \frac{(16,\ 8,\ 4,\ 2\ MHz\ or\ 32\ kHz)}{(ON+1)}$$

Timer 2

- 14-bit general purpose timer
- Ability to generate 3 Pulse Width Modulated signals (PWM2, PWM3 and PWM4)
- Input clock frequency:

$$f_{IN} = \frac{sys\_clk}{N}$$ with N = 1, 2, 4 or 8

and sys_clk = 16 MHz or 32 kHz

- Programmable output frequency:

$$f_{OUT} = \left(\frac{f_{IN}}{2}\right) to \left(\frac{f_{IN}}{2^{14}-1}\right)$$

- Three outputs with Programmable duty cycle from 0 % to 100 %
- Used for white LED intensity (on/off) control

4.7.2 Wake-Up Timer

The Wake-up timer can be programmed to wake up the DA14581 from power down mode after a preprogrammed number of GPIO events.

Features

- Monitors any GPIO state change
- Implements debouncing time from 0 up to 63 ms
- Accumulates external events and compares the number to a programmed value
- Generates an interrupt to the CPU

A minimum pulse duration of 2 sleep clock cycles must be applied to the GPIO to ensure a successful system wake-up.

4.7.3 Watchdog Timer

The Watchdog timer is an 8-bit timer with sign bit that can be used to detect an unexpected execution sequence caused by a software run-away and can generate a full system reset or a Non-Maskable Interrupt (NMI).

Features

- 8 bits down counter with sign bit, clocked with a 10.24 ms clock for a maximum 2.6 s time-out.
- Non-Maskable Interrupt (NMI) or WDOG reset.
- Optional automatic WDOG reset if NMI handler fails to update the Watchdog register.
- Non-maskable Watchdog freeze of the Cortex-M0 Debug module when the Cortex-M0 is halted in Debug state.
- Maskable Watchdog freeze by user program.

Note that if the system is not remapped, i.e. SysRAM is at address 0x20000000, then a watchdog fire will trigger the BootROM code to be executed again.

4.8 CLOCK/RESET

4.8.1 Clocks

The Digital Controlled Xtal Oscillator (DXCO) is a Pierce configured type of oscillator designed for low power consumption and high stability. There are two such crystal oscillators in the system, one at 16 MHz(XTAL16M) and a second at 32.768 kHz (XTAL32K). The 32.768 kHz oscillator has no trimming capabilities and is used as the clock of the Extended/ Deep Sleep modes. The 16 MHz oscillator can be trimmed.

The principle schematic of the two oscillators is shown in Figure 6 below. No external components to the DA14581 are required other than the crystal itself. If the crystal has a case connection, it is advised to connect the case to ground.

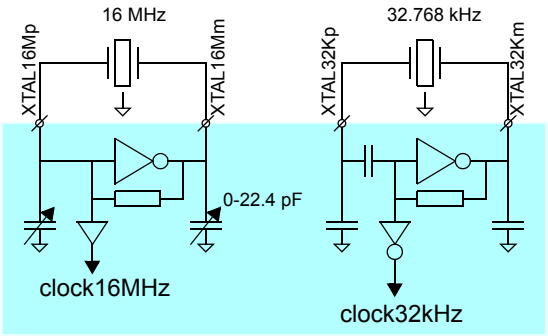


Figure 6: Crystal Oscillator Circuits

There are 3 RC oscillators in the DA14581: one providing 16 MHz (RC16M), one providing 32 kHz (RC32K) and one providing a frequency in the range of 10.5 kHz (RCX).

4.8.2 Reset

The DA14581 comprises an RST pad which is active high. It contains an RC filter for spikes suppression with 400 kΩ and 2.8 pF for the resistor and the capacitor respectively. It also contains a 25 kΩ pull-down resistor. This pad should be connected to ground if not needed by the application. The typical latency of the

RST pad is in the range of 2 µs.

4.9 POWER MANAGEMENT

The DA14581 has a complete power management function integrated with Buck or Boost DC-DC converter and separate LDOs for the different power domains of the system.

Features

- On-chip LDOs, without external capacitors
- Synchronous DC-DC converter which can be configured as either:
  - Boost (step-up) converter, starting from 0.9 V, when running from an Alkaline/NiMH cell.
  - Buck (step-down) converter for increased efficiency when running from a Lithium coin-cell or 2 Alkaline batteries down to 2.35 V.
- Battery voltage measurement ADC (multiplexed input from general purpose ADC)
- Use of small external components (2.2 µH inductor and 1 µF capacitor)

The Power Block contains a DC-DC converter which can be configured to operate as a Step-Up or a Step-Down converter. The converter provides power to four LDO groups in the system:

1. LDO RET: This is the LDO providing power to the Retention domain (PD_AON). It powers the Retention RAMs and the digital part which is always on.

2. LDO OTP: This is the LDO powering the OTP macro cell. This is the reason for using the step-up DC-DC converter when running from an Alkaline battery.

3. LDO SYS: This is the LDO providing the system with the actual VDD power required for the digital part to operate. Note that the Power Block implements seamless switching from the LDO SYS to the LDO RET when the system enters Deep Sleep mode. In the latter case, a low voltage is applied to the PD_AON power domain to further reduce leakage.

4. LDO (various): This a group of LDOs used for the elaborate control of the powering up/down of the Radio, the GP ADC and the XTAL16M oscillator.

There are two ways of connecting external batteries to the Power Block of the DA14581. They depend on the specific battery cell used and its voltage range. Battery cells are distinguished into Lithium coin cells (2.35 V to 3.3 V) and Alkaline cells (1.0 V to 1.8 V). The connection diagrams are presented in Figure 8 and Figure 7 respectively:

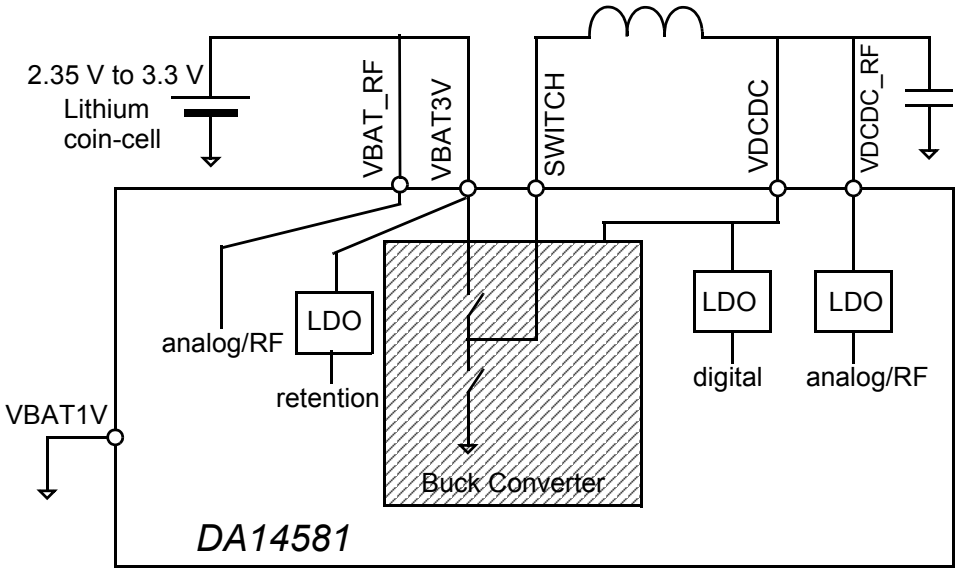


Figure 7: Supply Overview, Coin-Cell Application

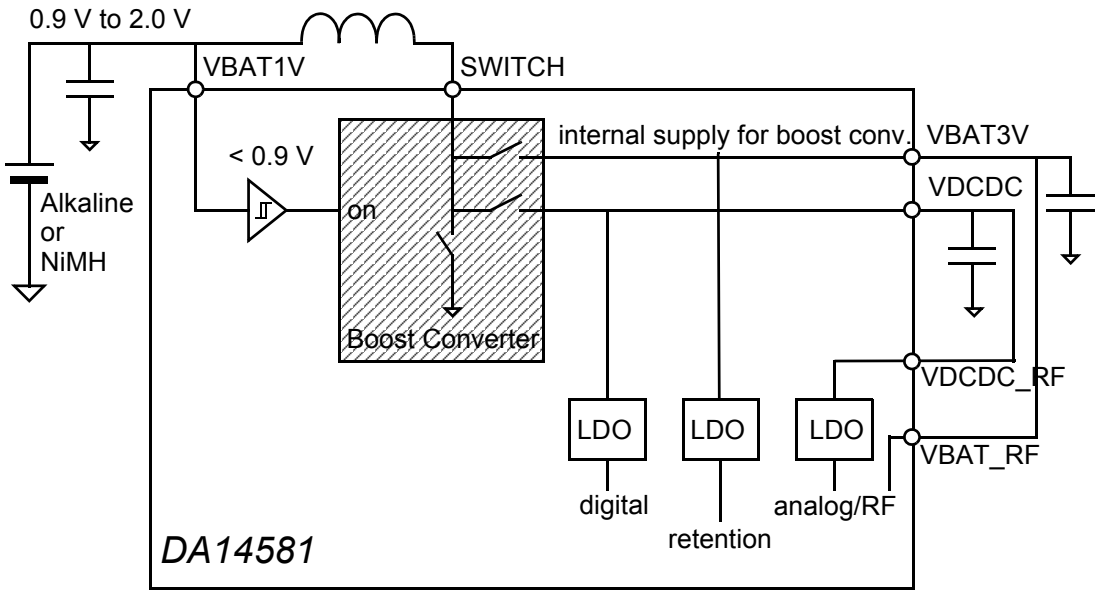


Figure 8: Supply Overview, Alkaline-Cell Application

The usage of Boost or Buck mode with respect to the provided voltage ranges is illustrated in the following figure which also illustrates the efficiency of the engine assuming a 10 mA constant load.

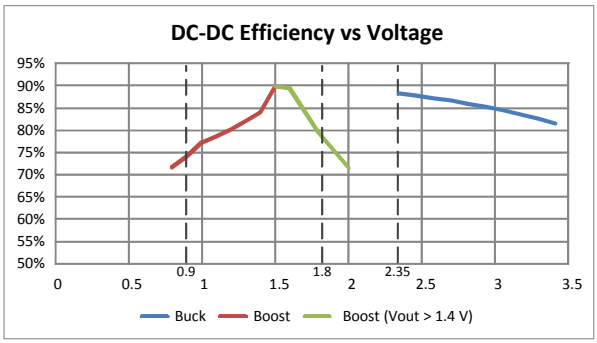


Figure 9: DC-DC Efficiency in Buck/Boost Mode at Various Voltage Levels

The X axis represents the supply voltage. BOOST mode should be used when voltage ranges from 0.9 V to 2.0 V to sustain a decent efficiency over 70 %. From that point on, the power dissipation becomes quite large.

BUCK mode can operate correctly with voltages in the range of 2.35 V to 3.3 V.

There are two voltage areas in Figure 9 designated by dashed lines. The first one (0 V to 0.9 V) indicates that the DA14581 is not operational when the voltage is below 0.9 V. This is the absolute threshold for the DC-DC converter Boost mode.

The second area (1.8 V to 2.35 V) indicates that Deep Sleep mode is not allowed when the DC-DC converter is configured in BUCK mode and the voltage is within this range, because the OTP will not be readable any more. However, this part of the voltage range can be covered by the BOOST mode. Furthermore, when BUCK mode is mandatory, Extended Sleep mode can be activated instead of Deep Sleep mode, thus not using the OTP for the code mirroring but retain the code in SysRAM.

Note: The system should never be cold booted when the supply voltage is less than 2.5 V. A manual power up with a power supply less than 2.5 V in buck mode might create instability.

5 Registers

This section contains a detailed view of the DA14581 registers. It is organized as follows: An overview table is presented initially, which depicts all register names, addresses and descriptions. A detailed bit level description of each register follows.

Note: The registers related to port P3 are not supported in the DA14581.

The register file of the ARM Cortex-M0 can be found in the following documents, available on the ARM website:

Devices Generic User Guide:
DUI0497A_cortex_m0_r0p0_generic_ug.pdf

Technical Reference Manual:
DDI0432C_cortex_m0_r0p0_trm.pdf

These documents contain the register descriptions for the Nested Vectored Interrupt Controller (NVIC), the System Control Block (SCB) and the System Timer (SysTick).

Table 4: Register Map

Address	Port	Description
0x40008000	OTPC_MODE_REG	Mode register
0x40008004	OTPC_PCTRL_REG	Bit-programming control register
0x40008008	OTPC_STAT_REG	Status register
0x4000800C	OTPC_AHBADR_REG	AHB master start address
0x40008010	OTPC_CELADR_REG	Macrocell start address
0x40008014	OTPC_NWORDS_REG	Number of words
0x40008018	OTPC_FFPRT_REG	Ports access to fifo logic
0x4000801C	OTPC_FFRD_REG	Latest read data from the OTPC_FFPRT_REG
0x50000000	CLK_AMBA_REG	HCLK, PCLK, divider and clock gates
0x50000002	CLK_FREQ_TRIM_REG	Xtal frequency trimming register
0x50000004	CLK_PER_REG	Peripheral divider register
0x50000008	CLK_RADIO_REG	Radio PLL control register
0x5000000A	CLK_CTRL_REG	Clock control register
0x50000010	PMU_CTRL_REG	Power Management Unit control register
0x50000012	SYS_CTRL_REG	System Control register
0x50000014	SYS_STAT_REG	System status register
0x50000016	TRIM_CTRL_REG	Control trimming of the XTAL16M
0x50000020	CLK_32K_REG	32 kHz oscillator register
0x50000022	CLK_16M_REG	16 MHz RC-oscillator register
0x50000024	CLK_RCX20K_REG	20 kHz RXC-oscillator control register
0x50000028	BANDGAP_REG	Bandgap trimming
0x5000002A	ANA_STATUS_REG	Status bit of analog (power management) circuits
0x50000100	WKUP_CTRL_REG	Control register for the wakeup counter
0x50000102	WKUP_COMPARE_REG	Number of events before wakeup interrupt
0x50000104	WKUP_RESET_IRQ_REG	Reset wakeup interrupt
0x50000106	WKUP_COUNTER_REG	Actual number of events of the wakeup counter
0x50000108	WKUP_RESET_CNTR_REG	Reset the event counter
0x5000010A	WKUP_SELECT_P0_REG	Select which inputs from P0 port can trigger wkup counter
0x5000010C	WKUP_SELECT_P1_REG	Select which inputs from P1 port can trigger wkup counter
0x5000010E	WKUP_SELECT_P2_REG	Select which inputs from P2 port can trigger wkup counter
0x50000110	WKUP_SELECT_P3_REG	Select which inputs from P3 port can trigger wkup counter
0x50000112	WKUP_POL_P0_REG	Select the sensitivity polarity for each P0 input
0x50000114	WKUP_POL_P1_REG	Select the sensitivity polarity for each P1 input
0x50000116	WKUP_POL_P2_REG	Select the sensitivity polarity for each P2 input
0x50000118	WKUP_POL_P3_REG	Select the sensitivity polarity for each P3 input
0x50000200	QDEC_CTRL_REG	Quad Decoder control register
0x50000202	QDEC_XCNT_REG	Counter value of the X Axis
0x50000204	QDEC_YCNT_REG	Counter value of the Y Axis
0x50000206	QDEC_CLOCKDIV_REG	Clock divider register
0x50000208	QDEC_CTRL2_REG	Quad Decoder control register

Table 4: Register Map

Address	Port	Description
0x5000020A	QDEC_ZCNT_REG	Z_counter
0x50001000	UART_RBR_THR_DLL_REG	Receive Buffer Register
0x50001004	UART_IER_DLH_REG	Interrupt Enable Register
0x50001008	UART_IIR_FCR_REG	Interrupt Identification Register/FIFO Control Register
0x5000100C	UART_LCR_REG	Line Control Register
0x50001010	UART_MCR_REG	Modem Control Register
0x50001014	UART_LSR_REG	Line Status Register
0x50001018	UART_MSR_REG	Modem Status Register
0x5000101C	UART_SCR_REG	Scratchpad Register
0x50001020	UART_LPDLL_REG	Low Power Divisor Latch Low
0x50001024	UART_LPDLH_REG	Low Power Divisor Latch High
0x50001030	UART_SRBR_STHR0_REG	Shadow Receive/Transmit Buffer Register
0x50001034	UART_SRBR_STHR1_REG	Shadow Receive/Transmit Buffer Register
0x50001038	UART_SRBR_STHR2_REG	Shadow Receive/Transmit Buffer Register
0x5000103C	UART_SRBR_STHR3_REG	Shadow Receive/Transmit Buffer Register
0x50001040	UART_SRBR_STHR4_REG	Shadow Receive/Transmit Buffer Register
0x50001044	UART_SRBR_STHR5_REG	Shadow Receive/Transmit Buffer Register
0x50001048	UART_SRBR_STHR6_REG	Shadow Receive/Transmit Buffer Register
0x5000104C	UART_SRBR_STHR7_REG	Shadow Receive/Transmit Buffer Register
0x50001050	UART_SRBR_STHR8_REG	Shadow Receive/Transmit Buffer Register
0x50001054	UART_SRBR_STHR9_REG	Shadow Receive/Transmit Buffer Register
0x50001058	UART_SRBR_STHR10_REG	Shadow Receive/Transmit Buffer Register
0x5000105C	UART_SRBR_STHR11_REG	Shadow Receive/Transmit Buffer Register
0x50001060	UART_SRBR_STHR12_REG	Shadow Receive/Transmit Buffer Register
0x50001064	UART_SRBR_STHR13_REG	Shadow Receive/Transmit Buffer Register
0x50001068	UART_SRBR_STHR14_REG	Shadow Receive/Transmit Buffer Register
0x5000106C	UART_SRBR_STHR15_REG	Shadow Receive/Transmit Buffer Register
0x5000107C	UART_USR_REG	UART Status register.
0x50001080	UART_TFL_REG	Transmit FIFO Level
0x50001084	UART_RFL_REG	Receive FIFO Level.
0x50001088	UART_SRR_REG	Software Reset Register.
0x5000108C	UART_SRTS_REG	Shadow Request to Send
0x50001090	UART_SBCR_REG	Shadow Break Control Register
0x50001094	UART_SDMAM_REG	Shadow DMA Mode
0x50001098	UART_SFE_REG	Shadow FIFO Enable
0x5000109C	UART_SRT_REG	Shadow RCVR Trigger
0x500010A0	UART_STET_REG	Shadow TX Empty Trigger
0x500010A4	UART_HTX_REG	Halt TX
0x500010F4	UART_CPR_REG	Component Parameter Register
0x500010F8	UART_UCV_REG	Component Version
0x500010FC	UART_CTR_REG	Component Type Register
0x50001100	UART2_RBR_THR_DLL_REG	Receive Buffer Register
0x50001104	UART2_IER_DLH_REG	Interrupt Enable Register
0x50001108	UART2_IIR_FCR_REG	Interrupt Identification Register/FIFO Control Register

Table 4: Register Map

Address	Port	Description
0x5000110C	UART2_LCR_REG	Line Control Register
0x50001110	UART2_MCR_REG	Modem Control Register
0x50001114	UART2_LSR_REG	Line Status Register
0x50001118	UART2_MSR_REG	Modem Status Register
0x5000111C	UART2_SCR_REG	Scratchpad Register
0x50001120	UART2_LPDLL_REG	Low Power Divisor Latch Low
0x50001124	UART2_LPDLH_REG	Low Power Divisor Latch High
0x50001130	UART2_SRBR_STHR0_REG	Shadow Receive/Transmit Buffer Register
0x50001134	UART2_SRBR_STHR1_REG	Shadow Receive/Transmit Buffer Register
0x50001138	UART2_SRBR_STHR2_REG	Shadow Receive/Transmit Buffer Register
0x5000113C	UART2_SRBR_STHR3_REG	Shadow Receive/Transmit Buffer Register
0x50001140	UART2_SRBR_STHR4_REG	Shadow Receive/Transmit Buffer Register
0x50001144	UART2_SRBR_STHR5_REG	Shadow Receive/Transmit Buffer Register
0x50001148	UART2_SRBR_STHR6_REG	Shadow Receive/Transmit Buffer Register
0x5000114C	UART2_SRBR_STHR7_REG	Shadow Receive/Transmit Buffer Register
0x50001150	UART2_SRBR_STHR8_REG	Shadow Receive/Transmit Buffer Register
0x50001154	UART2_SRBR_STHR9_REG	Shadow Receive/Transmit Buffer Register
0x50001158	UART2_SRBR_STHR10_REG	Shadow Receive/Transmit Buffer Register
0x5000115C	UART2_SRBR_STHR11_REG	Shadow Receive/Transmit Buffer Register
0x50001160	UART2_SRBR_STHR12_REG	Shadow Receive/Transmit Buffer Register
0x50001164	UART2_SRBR_STHR13_REG	Shadow Receive/Transmit Buffer Register
0x50001168	UART2_SRBR_STHR14_REG	Shadow Receive/Transmit Buffer Register
0x5000116C	UART2_SRBR_STHR15_REG	Shadow Receive/Transmit Buffer Register
0x5000117C	UART2_USR_REG	UART Status register.
0x50001180	UART2_TFL_REG	Transmit FIFO Level
0x50001184	UART2_RFL_REG	Receive FIFO Level.
0x50001188	UART2_SRR_REG	Software Reset Register.
0x5000118C	UART2_SRTS_REG	Shadow Request to Send
0x50001190	UART2_SBCR_REG	Shadow Break Control Register
0x50001194	UART2_SDMAM_REG	Shadow DMA Mode
0x50001198	UART2_SFE_REG	Shadow FIFO Enable
0x5000119C	UART2_SRT_REG	Shadow RCVR Trigger
0x500011A0	UART2_STET_REG	Shadow TX Empty Trigger
0x500011A4	UART2_HTX_REG	Halt TX
0x500011F4	UART2_CPR_REG	Component Parameter Register
0x500011F8	UART2_UCV_REG	Component Version
0x500011FC	UART2_CTR_REG	Component Type Register
0x50001200	SPI_CTRL_REG	SPI control register 0
0x50001202	SPI_RX_TX_REG0	SPI RX/TX register0
0x50001204	SPI_RX_TX_REG1	SPI RX/TX register1
0x50001206	SPI_CLEAR_INT_REG	SPI clear interrupt register
0x50001208	SPI_CTRL_REG1	SPI control register 1
0x50001300	I2C_CON_REG	I2C Control Register
0x50001304	I2C_TAR_REG	I2C Target Address Register

Table 4: Register Map

Address	Port	Description
0x50001308	I2C_SAR_REG	I2C Slave Address Register
0x50001310	I2C_DATA_CMD_REG	I2C Rx/Tx Data Buffer and Command Register
0x50001314	I2C_SS_SCL_HCNT_REG	Standard Speed I2C Clock SCL High Count Register
0x50001318	I2C_SS_SCL_LCNT_REG	Standard Speed I2C Clock SCL Low Count Register
0x5000131C	I2C_FS_SCL_HCNT_REG	Fast Speed I2C Clock SCL High Count Register
0x50001320	I2C_FS_SCL_LCNT_REG	Fast Speed I2C Clock SCL Low Count Register
0x5000132C	I2C_INTR_STAT_REG	I2C Interrupt Status Register
0x50001330	I2C_INTR_MASK_REG	I2C Interrupt Mask Register
0x50001334	I2C_RAW_INTR_STAT_REG	I2C Raw Interrupt Status Register
0x50001338	I2C_RX_TL_REG	I2C Receive FIFO Threshold Register
0x5000133C	I2C_TX_TL_REG	I2C Transmit FIFO Threshold Register
0x50001340	I2C_CLR_INTR_REG	Clear Combined and Individual Interrupt Register
0x50001344	I2C_CLR_RX_UNDER_REG	Clear RX_UNDER Interrupt Register
0x50001348	I2C_CLR_RX_OVER_REG	Clear RX_OVER Interrupt Register
0x5000134C	I2C_CLR_TX_OVER_REG	Clear TX_OVER Interrupt Register
0x50001350	I2C_CLR_RD_REQ_REG	Clear RD_REQ Interrupt Register
0x50001354	I2C_CLR_TX_ABRT_REG	Clear TX_ABRT Interrupt Register
0x50001358	I2C_CLR_RX_DONE_REG	Clear RX_DONE Interrupt Register
0x5000135C	I2C_CLR_ACTIVITY_REG	Clear ACTIVITY Interrupt Register
0x50001360	I2C_CLR_STOP_DET_REG	Clear STOP_DET Interrupt Register
0x50001364	I2C_CLR_START_DET_REG	Clear START_DET Interrupt Register
0x50001368	I2C_CLR_GEN_CALL_REG	Clear GEN_CALL Interrupt Register
0x5000136C	I2C_ENABLE_REG	I2C Enable Register
0x50001370	I2C_STATUS_REG	I2C Status Register
0x50001374	I2C_TXFLR_REG	I2C Transmit FIFO Level Register
0x50001378	I2C_RXFLR_REG	I2C Receive FIFO Level Register
0x5000137C	I2C_SDA_HOLD_REG	I2C SDA Hold Time Length Register
0x50001380	I2C_TX_ABRT_SOURCE_REG	I2C Transmit Abort Source Register
0x50001394	I2C_SDA_SETUP_REG	I2C SDA Setup Register
0x50001398	I2C_ACK_GENERAL_CALL_REG	I2C ACK General Call Register
0x5000139C	I2C_ENABLE_STATUS_REG	I2C Enable Status Register
0x500013A0	I2C_IC_FS_SPKLEN_REG	I2C SS and FS spike suppression limit Size
0x50001400	GPIO_IRQ0_IN_SEL_REG	GPIO interrupt selection for GPIO_IRQ0
0x50001402	GPIO_IRQ1_IN_SEL_REG	GPIO interrupt selection for GPIO_IRQ1
0x50001404	GPIO_IRQ2_IN_SEL_REG	GPIO interrupt selection for GPIO_IRQ2
0x50001406	GPIO_IRQ3_IN_SEL_REG	GPIO interrupt selection for GPIO_IRQ3
0x50001408	GPIO_IRQ4_IN_SEL_REG	GPIO interrupt selection for GPIO_IRQ4
0x5000140C	GPIO_DEBOUNCE_REG	debounce counter value for GPIO inputs
0x5000140E	GPIO_RESET_IRQ_REG	GPIO interrupt reset register
0x50001410	GPIO_INT_LEVEL_CTRL_REG	high or low level select for GPIO interrupts
0x50001412	KBRD_IRQ_IN_SEL0_REG	GPIO interrupt selection for KBRD_IRQ for P0
0x50001414	KBRD_IRQ_IN_SEL1_REG	GPIO interrupt selection for KBRD_IRQ for P1 and P2
0x50001416	KBRD_IRQ_IN_SEL2_REG	GPIO interrupt selection for KBRD_IRQ for P3
0x50001500	GP_ADC_CTRL_REG	General Purpose ADC Control Register

Table 4: Register Map

Address	Port	Description
0x50001502	GP_ADC_CTRL2_REG	General Purpose ADC Second Control Register
0x50001504	GP_ADC_OFFP_REG	General Purpose ADC Positive Offset Register
0x50001506	GP_ADC_OFFN_REG	General Purpose ADC Negative Offset Register
0x50001508	GP_ADC_CLEAR_INT_REG	General Purpose ADC Clear Interrupt Register
0x5000150A	GP_ADC_RESULT_REG	General Purpose ADC Result Register
0x5000150C	GP_ADC_DELAY_REG	General Purpose ADC Delay Register
0x5000150E	GP_ADC_DELAY2_REG	General Purpose ADC Second Delay Register
0x50001600	CLK_REF_SEL_REG	Select clock for oscillator calibration
0x50001602	CLK_REF_CNT_REG	Count value for oscillator calibration
0x50001604	CLK_REF_VAL_L_REG	XTAL16M reference cycles, lower 16 bits
0x50001606	CLK_REF_VAL_H_REG	XTAL16M reference cycles, upper 16 bits
0x50003000	P0_DATA_REG	P0 Data input / output register
0x50003002	P0_SET_DATA_REG	P0 Set port pins register
0x50003004	P0_RESET_DATA_REG	P0 Reset port pins register
0x50003006	P00_MODE_REG	P00 Mode Register
0x50003008	P01_MODE_REG	P01 Mode Register
0x5000300A	P02_MODE_REG	P02 Mode Register
0x5000300C	P03_MODE_REG	P03 Mode Register
0x5000300E	P04_MODE_REG	P04 Mode Register
0x50003010	P05_MODE_REG	P05 Mode Register
0x50003012	P06_MODE_REG	P06 Mode Register
0x50003014	P07_MODE_REG	P07 Mode Register
0x50003020	P1_DATA_REG	P1 Data input / output register
0x50003022	P1_SET_DATA_REG	P1 Set port pins register
0x50003024	P1_RESET_DATA_REG	P1 Reset port pins register
0x50003026	P10_MODE_REG	P10 Mode Register
0x50003028	P11_MODE_REG	P11 Mode Register
0x5000302A	P12_MODE_REG	P12 Mode Register
0x5000302C	P13_MODE_REG	P13 Mode Register
0x5000302E	P14_MODE_REG	P14 Mode Register
0x50003030	P15_MODE_REG	P15 Mode Register
0x50003040	P2_DATA_REG	P2 Data input / output register
0x50003042	P2_SET_DATA_REG	P2 Set port pins register
0x50003044	P2_RESET_DATA_REG	P2 Reset port pins register
0x50003046	P20_MODE_REG	P20 Mode Register
0x50003048	P21_MODE_REG	P21 Mode Register
0x5000304A	P22_MODE_REG	P22 Mode Register
0x5000304C	P23_MODE_REG	P23 Mode Register
0x5000304E	P24_MODE_REG	P24 Mode Register
0x50003050	P25_MODE_REG	P25 Mode Register
0x50003052	P26_MODE_REG	P26 Mode Register
0x50003054	P27_MODE_REG	P27 Mode Register
0x50003056	P28_MODE_REG	P28 Mode Register
0x50003058	P29_MODE_REG	P29 Mode Register

Table 4: Register Map

Address	Port	Description
0x50003070	P01_PADPWR_CTRL_REG	Ports 0 and 1 Output Power Control Register
0x50003072	P2_PADPWR_CTRL_REG	Port 2 Output Power Control Register
0x50003074	P3_PADPWR_CTRL_REG	Port 3 Output Power Control Register
0x50003080	P3_DATA_REG	P3 Data input / output register
0x50003082	P3_SET_DATA_REG	P3 Set port pins register
0x50003084	P3_RESET_DATA_REG	P3 Reset port pins register
0x50003086	P30_MODE_REG	P30 Mode Register
0x50003088	P31_MODE_REG	P31 Mode Register
0x5000308A	P32_MODE_REG	P32 Mode Register
0x5000308C	P33_MODE_REG	P33 Mode Register
0x5000308E	P34_MODE_REG	P34 Mode Register
0x50003090	P35_MODE_REG	P35 Mode Register
0x50003092	P36_MODE_REG	P36 Mode Register
0x50003094	P37_MODE_REG	P37 Mode Register
0x50003100	WATCHDOG_REG	Watchdog timer register.
0x50003102	WATCHDOG_CTRL_REG	Watchdog control register.
0x50003200	CHIP_ID1_REG	Chip identification register 1.
0x50003201	CHIP_ID2_REG	Chip identification register 2.
0x50003202	CHIP_ID3_REG	Chip identification register 3.
0x50003203	CHIP_SWC_REG	Software compatibility register.
0x50003204	CHIP_REVISION_REG	Chip revision register.
0x50003300	SET_FREEZE_REG	Controls freezing of various timers/counters.
0x50003302	RESET_FREEZE_REG	Controls unfreezing of various timers/counters.
0x50003304	DEBUG_REG	Various debug information register.
0x50003306	GP_STATUS_REG	General purpose system status register.
0x50003308	GP_CONTROL_REG	General purpose system control register.
0x50003400	TIMER0_CTRL_REG	Timer0 control register
0x50003402	TIMER0_ON_REG	Timer0 on control register
0x50003404	TIMER0_RELOAD_M_REG	16 bits reload value for Timer0
0x50003406	TIMER0_RELOAD_N_REG	16 bits reload value for Timer0
0x50003408	PWM2_DUTY_CYCLE	Duty Cycle for PWM2
0x5000340A	PWM3_DUTY_CYCLE	Duty Cycle for PWM3
0x5000340C	PWM4_DUTY_CYCLE	Duty Cycle for PWM4
0x5000340E	TRIPLE_PWM_FREQUENCY	Frequency for PWM 2,3 and 4
0x50003410	TRIPLE_PWM_CTRL_REG	PWM 2 3 4 Control

Table 5: OTPC_MODE_REG (0x40008000)

Bit	Mode	Symbol	Description	Reset
31:30	-	-	Reserved	0x0

Table 5: OTPC_MODE_REG (0x40008000)

Bit	Mode	Symbol	Description	Reset
29:28	r/w	OTPC_MODE_PRG_PORT_MUX	Selects the source that is connected to the prg_port port of the controller. 00 - {16'd0, BANDGAP_REG[15:0]} 01 - {RF_RSSI_COMP_CTRL_REG[15:0], 8'd0, RFIO_CTRL1_REG{7:0]} 10 - {3'd0, RF_LNA_CTRL3_REG[4:0], RF_LNA_CTRL2_REG[11:0], RF_LNA_CTRL1_REG[11:0]} 11 - {28'd0, RF_VCO_CTRL_REG[3:0]} See OTPC_MODE_PRG_PORT_SEL about the use of the prg_port	0x0
27:9	-	-	Reserved	0x0
8	r/w	OPTC_MODE_PRG_FAST	Defines the timing that will be used for all the programming activities (APROG, MPROG and TWR) 0 - Selects the normal timing 1 - Selects the fast timing	0
7	r/w	OTPC_MODE_PRG_PORT_SEL	Selects an alternative data source for the programming of the OTP macrocells, when the controller is configured in APROG mode. 0 - The fifo will be used as the data source. The fifo will be filled with a way defined by the register OTPC_MODE_USE_DMA. The number of words that will be programmed is defined by OTPC_NWORDS. 1 - Only one word will programmed. The value of the word is contained in the prg_port port of the controller. The values of the registers OTPC_MODE_USE_DMA, OTPC_NWORDS and the contents of the FIFO will not be used.	0x0
6	r/w	OTPC_MODE_TWO_CC_ACC	Defines the duration of each read from the OTP macrocells. 0 - Reads 16 bits of data every one clock cycle. 1 - Reads 16 bits of data every two clock cycles.	0x0
5	r/w	OTPC_MODE_FIFO_FLUSH	Writing 1, removes any content from the FIFO. This bit returns automatically to 0.	0x0
4	r/w	OTPC_MODE_USE_DMA	Selects the use of the dma, when the controller is configured in one of the modes: AREAD or APROG. 0 - DMAis not used. The data should be transfered from/to controller through OTPC_FFPRT_REG 1 - DMA is used. Data transfers from/to controller are performed automatically. The AHB base address should be configured in OTPC_AHBADR_REG before the selection of the mode. If programming of the OTPC_MODE_REG is performed through the serial interface,the OTPC_MODE_USE_DMA will be set to 0 automatically. If the controller is in APROG mode and the OTPC_MODE_PRG_PORT_SEL is enabled, the dma will stay inactive.	0x0
3	-	-	Reserved	0x0

Table 5: OTPC_MODE_REG (0x40008000)

Bit	Mode	Symbol	Description	Reset
2:0	r/w	OTPC_MODE_MODE	Defines the mode of operation of the OTPC controller. The encoding of the modes is as follows: 000 - STBY mode 001 - MREAD mode 010 - MPROG mode 011 - AREAD mode 100 - APROG mode 101 - Test mode. Reserved 110 - Test mode. Reserved 111 - Test mode. Reserved To manually move between modes, always return to STBY mode first.	0x0

Table 6: OTPC_PCTRL_REG (0x40008004)

Bit	Mode	Symbol	Description	Reset
31:28	-	-	Reserved	0x0
27	r/w	OTPC_PCTRL_ENU	Enables the programming in the upper bank of the OTP. 0 - Programming sequence is not applied in the upper bank. 1 - Programming sequence is applied in the upper bank.	0x0
26	r/w	OTPC_PCTRL_BITU	Defines the value of the selected bit in the upper bank, after the programming sequence.	0x0
25	r/w	OTPC_PCTRL_ENL	Enables the programming in the lower bank. 0 - The programming sequence is not applied in the lower bank. 1 -The programming sequence is applied in the lower bank.	0x0
24	r/w	OTPC_PCTRL_BITL	Defines the value of the selected bit in the lower bank, after the programming sequence.	0x0
23	r/w	OTPC_PCTRL_BSELU	Selects between the U1 and U0 byte for the programming sequence in the upper bank. 0 - Program the U0 byte 1 - Program the U1 byte	0x0
22:20	r/w	OTPC_PCTRL_BADRU	Selects the bit inside the Ux (x=0,1) byte, which will be programmed in the upper bank.	0x0
19	r/w	OTPC_PCTRL_BSELL	Selects between the L1 and L0 byte for the programming sequence in the lower bank. 0 - Program the L0 byte 1 - Program the L1 byte	0x0
18:16	r/w	OTPC_PCTRL_BADRL	Selects the bit inside the Lx (x=0,1) byte, which will be programmed in the lower bank.	0x0
15:13	-	-	Reserved	0x0
12:0	r/w	OTPC_PCTRL_WADDR	Defines the address of a 32 bits word {U1,L1,U0,L0} in the macrocells, where one or two bits will be programmed. There are two macrocell banks, with 8 bits each. Each bank contribute with two memory positions for each 32 bits word. The Ux, Lx represent the bytes of the upper and lower bank respectively.	0x0

Table 7: OTPC_STAT_REG (0x40008008)

Bit	Mode	Symbol	Description	Reset
31:29	-	-	Reserved	0x0

Table 7: OTPC_STAT_REG (0x40008008)

Bit	Mode	Symbol	Description	Reset
28:16	r	OTPC_STAT_NWORDS	Contains the current value of the words to be processed.	0
15	r	OTPC_STAT_TERR_U	Indicates the upper bank as the source of a test error. This value is valid when OTPC_STAT_TERROR is valid. 0 - There is no test error in the upper bank 1 - A test error has occured in the upper bank	0x0
14	r	OTPC_STAT_TERR_L	Indicates the lower bank as the source of a test error. The value is valid when OTPC_STAT_TERROR is valid. 0 - There is no test error in the lower bank 1 - A test error has occured in the lower bank	0x0
13	r	OTPC_STAT_PERR_U	Indicates the upper bank as the source of a programming error. The value is valid when OTPC_STAT_PERROR is valid. 0 - There is no programming error in the upper bank 1 - A programming error has occured in the upper bank	0x0
12	r	OTPC_STAT_PERR_L	Indicates the lower bank as the source of a programming error. The value is valid when OTPC_STAT_PERROR is valid. 0 - There is no programming error in the lower bank 1 - A programming error has occured in the lower bank	0x0
11:8	r	OTPC_STAT_FWORDS	Indicates the number of words which contained in the fifo of the controller.	0x0
7:5	-	-	Reserved	0x0
4	r	OTPC_STAT_ARDY	Monitors the progress of read or programming operations while in the AREAD or APROG modes. 0 - The controller is busy while reading or programming (AREAD or APROG modes). 1 - The controller is not busy in AREAD or APROG mode.	0x1
3	r	OTPC_STAT_TERROR	Indicates the result of a test sequence. Should be checked after the end of a TBLANK, TDEC and TWR mode (OTPC_STAT_TRDY= 1). 0 - The test sequence ends with no error. 1 - The test sequence has failed.	0x0
2	r	OTPC_STAT_TRDY	Indicates the state of a test mode. Should be used to monitor the progress of the TBLANK, TDEC and TWR modes. 0 - The controller is busy. A test mode is in progress. 1 - There is no active test mode.	0x1
1	r	OTPC_STAT_PERROR	Indicates that an error has occurred during the bit-programming process. 0 - No error during the bit-programming process. 1 - The process of bit-programming failed. When the controller is in MPROG mode, this bit should be checked after the end of the programming process (OTPC_STAT_PRDY= 1). During APROG mode, the value of this field is normal to change periodically. Upon finishing the operation in the APROG mode (OTPC_STAT_ARDY= 1), this field indicates if the programming has failed or ended succesfully.	0x0
0	r	OTPC_STAT_PRDY	Indicates the state of a bit-programming process. 0 - The controller is busy. A bit-programming is in progress 1 - The logic which performs bit-programming is idle. When the controller is in MPROG mode, this bit should be used to monitor the progress of a programming request. During APROG mode, the value of this field it is normal to changing periodically.	0x1

Table 8: OTPC_AHBADR_REG (0x4000800C)

Bit	Mode	Symbol	Description	Reset
31:2	r/w	OTPC_AHBADR	Tthe AHB address used by the AHB master interface of the controller (bits [31:2]).	0x0
1:0	-	-	Reserved	0x0

Table 9: OTPC_CELADR_REG (0x40008010)

Bit	Mode	Symbol	Description	Reset
31:13	-	-	Reserved	0x0
12:0	r/w	OTPC_CELADR	Defines a word address inside the macrocell. Used in modes AREAD and APROG and is automatically updated.	0x0

Table 10: OTPC_NWORDS_REG (0x40008014)

Bit	Mode	Symbol	Description	Reset
31:13	-	-	Reserved	0x0
12:0	r/w	OTPC_NWORDS	The number of words (minus one) for reading/programming during the AREAD/APROG mode. If in APROG mode, and the OTPC_MODE_PRG_PORT_SEL is enabled (=1), this register will not be used and will stay unchanged. During mirroring, this register reflects the current amount of data that will be copied. It keeps its value until be written by the software with a new value. The number of the words that remaining to be processed by the controller is contained in the field OTPC_STAT_NWORDS.	0x0

Table 11: OTPC_FFPRT_REG (0x40008018)

Bit	Mode	Symbol	Description	Reset
31:0	r/w	OTPC_FFPRT	Provides access to the fifo through an access port. Write this register with the corresponding data, when the APROG mode is selected and the DMA is disabled. Read from this register the corresponding data, when the AREAD mode is selected and the DMA is disabled. Check OTPC_STAT_FWORDS register for data/space availability, before accessing the fifo.	0x0

Table 12: OTPC_FFRD_REG (0x4000801C)

Bit	Mode	Symbol	Description	Reset
31:0	r	OTPC_FFRD	Contains the value read from the fifo, after a read of the OTPC_FFPRT_REG register.	0x0

Table 13: CLK_AMBA_REG (0x50000000)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	OTP_ENABLE	Clock enable for OTP controller	0x0
6	-	-	Reserved	0x0

Table 13: CLK_AMBA_REG (0x50000000)

Bit	Mode	Symbol	Description	Reset
5:4	r/w	PCLK_DIV	APB interface clock (PCLK). Divider is cascaded with HCLK_DIV. PCLK is HCLK divided by: 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8	0x2
3:2	-	-	Reserved	0x0
1:0	r/w	HCLK_DIV	AHB interface and microprocessor clock (HCLK). HCLK is source clock divided by: 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8	0x2

Table 14: CLK_FREQ_TRIM_REG (0x50000002)

Bit	Mode	Symbol	Description	Reset
15:11	-	-	Reserved	0x0
10:8	r/w	COARSE_ADJ	Xtal frequency course trimming register. 0x0: lowest frequency 0x7: highest frequencyIncrement or decrement the binary value with 1. Wait approximately 200 us to allow the adjustment to settle.	0x0
7:0	r/w	FINE_ADJ	Xtal frequency fine trimming register. 0x00: lowest frequency 0xFF: highest frequency	0x0

Table 15: CLK_PER_REG (0x50000004)

Bit	Mode	Symbol	Description	Reset
15	r/w	QUAD_ENABLE	Enable the Quadrature clock	0x0
14:12	-	-	Reserved	0x0
11	r/w	SPI_ENABLE	Enable SPI clock	0x0
10	-	-	Reserved	0x0
9:8	r/w	SPI_DIV	Division factor for SPI 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8	0x0
7	r/w	UART1_ENABLE	Enable UART1 clock	0x0
6	r/w	UART2_ENABLE	Enable UART2 clock	0x0
5	r/w	I2C_ENABLE	Enable I2C clock	0x0
4	r/w	WAKEUPCT_ENABLE	Enable Wakeup CaptureTimer clock	0x0
3	r/w	TMR_ENABLE	Enable TIMER0 and TIMER2 clock	0x0
2	-	-	Reserved	0x0
1:0	r/w	TMR_DIV	Division factor for TIMER0 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8	0x0

Table 16: CLK_RADIO_REG (0x50000008)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	BLE_ENABLE	Enable the BLE core clocks	0x0
6	r/w	BLE_LP_RESET	Reset for the BLE LP timer	0x1
5:4	r/w	BLE_DIV	Division factor for BLE core blocks 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8 The programmed frequency should not be lower than 8 MHz and not faster than the programmed CPU clock frequency. Refer also to BLE_CNTL2_REG[BLE_CLK_SEL].	0x0
3	r/w	RFCU_ENABLE	Enable the RF control Unit clock	0x0
2	-	-	Reserved	0x0
1:0	r/w	RFCU_DIV	Division factor for RF Control Unit 0x0: divide by 1 0x1: divide by 2 0x2: divide by 4 0x3: divide by 8 The programmed frequency must be exactly 8 MHz.	0x0

Table 17: CLK_CTRL_REG (0x5000000A)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r	RUNNING_AT_XTAL16M	Indicates that the XTAL16M clock is used as clock, and may not be switched off	0x1
6	r	RUNNING_AT_RC16M	Indicates that the RC16M clock is used as clock	0x0
5	r	RUNNING_AT_32K	Indicates that either the RC32k or XTAL32k is being used as clock	0x0
4	-	-	Reserved	0x0
3	r/w	XTAL16M_SPIKE_FLT_DISABLE	Disable spikefilter in digital clock	0x0
2	r/w	XTAL16M_DISABLE	Setting this bit instantaneously disables the 16 MHz crystal oscillator. Also, after sleep/wakeup cycle, the oscillator will not be enabled. This bit may not be set to '1'when "RUNNING_AT_XTAL16M is '1' to prevent deadlock. After resetting this bit, wait for XTAL16_SETTLED or XTAL16_TRIM_READY to become '1' before switching to XTAL16 clock source.	0x0
1:0	r/w	SYS_CLK_SEL	Selects the clock source. 0x0: XTAL16M (check the XTAL16_SETTLED and XTAL16_TRIM_READY bits!!) 0x1: RC16M 0x2/0x3: either RC32k or XTAL32k is used	0x0

Table 18: PMU_CTRL_REG (0x50000010)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0x0

Table 18: PMU_CTRL_REG (0x50000010)

Bit	Mode	Symbol	Description	Reset
11:8	r/w	RETENTION_MODE	Select the retainability of the 4 retention RAM macros. '1' is retainable, '0' is power gated. (3) is RETRAM4 (2) is RETRAM3 (1) is RETRAM2 (0) is RETRAM1	0x0
7	r/w	FORCE_BOOST	Force the DC-DC into boost mode at next wakeup. Setting this bit reduces the deepsleep current. FORCE_BOOST has highest priority. When either FORCE_BOOST or FORCE_BUCK have been written, these bits cannot be changed.	0x0
6	r/w	FORCE_BUCK	Force the DC-DC into buck mode at next wakeup. Setting this bit reduces the deepsleep current. FORCE_BOOST has highest priority. When either FORCE_BOOST or FORCE_BUCK have been written, these bits cannot be changed.	0x0
5:4	r/w	OTP_COPY_DIV	Sets the HCLK division during OTP mirroring	0x0
2	r/w	RADIO_SLEEP	Put the digital part of the radio in powerdown	0x1
1	r/w	PERIPH_SLEEP	Put all peripherals (I2C, UART, SPI, ADC) in powerdown	0x1
0	r/w	RESET_ON_WAKEUP	Perform a Hardware Reset after waking up. Booter will be started.	0x0

Table 19: SYS_CTRL_REG (0x50000012)

Bit	Mode	Symbol	Description	Reset
15	w	SW_RESET	Writing a '1' to this bit will reset the device, except for: SYS_CTRL_REG CLK_FREQ_TRIM_REG ...	0x0
9	r/w	TIMEOUT_DISABLE	Disables timeout in Power statemachine. By default, the statemachine continues if after 2 ms the blocks are not started up. This can be read back from ANA_STATUS_REG.	0x0
8	-	-	Reserved	0x0
7	r/w	DEBUGGER_ENABLE	Enable the debugger. This bit is set by the booter according to the OTP header. If not set, the SWDIO and SW_CLK can be used as gpio ports.	0x0
6	r/w	OTPC_RESET_REQ	Reset request for the OTP controller.	0x0
5	r/w	PAD_LATCH_EN	Latches the control signals of the pads for state retention in powerdown mode. 0: Control signals are retained 1: Latch is transparant, pad can be recontrolled	0x1
4	r/w	OTP_COPY	Enables OTP to SysRAM copy action after waking up PD_SYS	0x0
3	r/w	CLK32_SOURCE	Sets the clock source of the 32 kHz clock 0 = RC-oscillator 1 = 32 kHz crystal oscillator	0x0

Table 19: SYS_CTRL_REG (0x50000012)

Bit	Mode	Symbol	Description	Reset
2	r/w	RET_SYSRAM	Sets the development phase mode. The PD_SYS is not actually power gated (SysRAM is retained). No copy action to SysRAM is done when the system wakes up. For emulating startup time, the OTP_COPY bit still needs to be set.	0x0
1:0	r/w	REMAP_ADR0	Controls which memory is located at address 0x0000 for execution. 0x0: ROM 0x1: OTP 0x2: SysRAM 0x3: RetRAM	0x0

Table 20: SYS_STAT_REG (0x50000014)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r	XTAL16_SETTLED	Indicates that XTAL16 has had > 2 ms of settle time	0x0
6	r	XTAL16_TRIM_READY	Indicates that XTAL trimming mechanism is ready, i.e. the trimming equals CLK_FREQ_TRIM_REG.	0x1
5	r	DBG_IS_UP	Indicates that PD_DBG is functional	0x0
4	r	DBG_IS_DOWN	Indicates that PD_DBG is in power down	0x1
3	r	PER_IS_UP	Indicates that PD_PER is functional	0x0
2	r	PER_IS_DOWN	Indicates that PD_PER is in power down	0x1
1	r	RAD_IS_UP	Indicates that PD_RAD is functional	0x0
0	r	RAD_IS_DOWN	Indicates that PD_RAD is in power down	0x1

Table 21: TRIM_CTRL_REG (0x50000016)

Bit	Mode	Symbol	Description	Reset
7:4	r/w	TRIM_TIME	Defines the delay between XTAL16M enable and applying the CLK_FREQ_TRIM_REG in steps of 250 us. 0x0: apply directly 0x1: wait between 0 and 250 us 0x2: wait between 250 us and 500 us etc. **(Note 1)**	0xA
3:0	r/w	SETTLE_TIME	Defines the delay between applying CLK_FREQ_TRIM_REG and XTAL16_SETTLED in steps of 250 us. 0x0: XTAL16_SETTLED is set direcly 0x1: wait between 0 and 250 us 0x2: wait between 250 us and 500 us etc.	0x2

Note 1: The period duration of 250 us is derived by dividing the RC16M clock signal by 4000. Consequently, the period duration may vary over temperature.

Table 22: CLK_32K_REG (0x50000020)

Bit	Mode	Symbol	Description	Reset
15:13	-	-	Reserved	0x0

Table 22: CLK_32K_REG (0x50000020)

Bit	Mode	Symbol	Description	Reset
12	r/w	XTAL32K_DISABLE_AMPREG	Setting this bit disables the amplitude regulation of the XTAL32kHz oscillator. Set this bit to '1' for an external clock applied at XTAL32Kp. Keep this bit '0' with a crystal between XTAL32Kp and XTAL32Km.	0x0
11:8	r/w	RC32K_TRIM	Controls the frequency of the RC32K oscillator. 0x0: lowest frequency 0x7: default 0xF: highest frequency	0x7
7	r/w	RC32K_ENABLE	Enables the 32 kHz RC oscillator	0x1
6:3	r/w	XTAL32K_CUR	Bias current for the 32kHz XTAL oscillator. 0x0: minimum 0x3: default 0xF: maximum For each application there is an optimal setting for which the startup behavior is optimal.	0x3
2:1	r/w	XTAL32K_RBIAS	Setting for the bias resistor of the 32 kHz XTAL oscillator. 0x0: maximum 0x3: minimum Prefered setting will be provided by Dialog.	0x2
0	r/w	XTAL32K_ENABLE	Enables the 32 kHz XTAL oscillator	0x0

Table 23: CLK_16M_REG (0x50000022)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9	r/w	XTAL16_NOISE_FILT_ENABLE	Enables noise flter in 16 MHz crystal oscillator	0x0
8	r/w	XTAL16_BIAS_SH_ENABLE	Enables Ibias sample/hold function in 16 MHz crystal oscillator. This bit should be set when the system wake up and reset before entering deep or extended sleep mode.	0x0
7:5	r/w	XTAL16_CUR_SET	Bias current for the 16 MHz XTAL oscillator. 0x0: minimum 0x7: maximum	0x5
4:1	r/w	RC16M_TRIM	Controls the frequency of the RC16M oscillator. 0x0: lowest frequency 0xF: highest frequency	0x0
0	r/w	RC16M_ENABLE	Enables the 16 MHz RC oscillator	0x0

Table 24: CLK_RCX20K_REG (0x50000024)

Bit	Mode	Symbol	Description	Reset
12	r/w	RCX20K_SELECT	Selects RCX oscillator. 0 : RC32K oscillator 1: RCX oscillator	0
11	r/w	RCX20K_ENABLE	Enable the RCX oscillator	0
10	r/w	RCX20K_LOWF	Extra low frequency	0
9:8	r/w	RCX20K_BIAS	Bias control	1
7:4	r/w	RCX20K_NTC	Temperature control	7

Table 24: CLK_RCX20K_REG (0x50000024)

Bit	Mode	Symbol	Description	Reset
3:0	r/w	RCX20K_TRIM	Controls the frequency of the RCX oscillator. 0x0: lowest frequency 0x7: default 0xF: highest frequency	8

Table 25: BANDGAP_REG (0x50000028)

Bit	Mode	Symbol	Description	Reset
15	-	-	Reserved	0x0
14	r/w	BGR_LOWPOWER	Test-mode, do not use. It disables the bandgap core (voltages will continue for some time, but will slowely drift away)	0x0
13:10	r/w	LDO_RET_TRIM	**(Note 2)**	0x0
9:5	r/w	BGR_ITRIM	Current trimming for bias	0x0
4:0	r/w	BGR_TRIM	Trim register for bandgap	0x0

Note 2: 0xF is the lowest voltage, but is too low for reliable startup at high temperature in combination with extended sleep. 0xA is 100 mV higher and considered to be the lowest value which is safe to use. 0x0 or 0x1 is again 100 mV higher and 0x0 is the reset value. 0x4 is the maximum voltage.

Table 26: ANA_STATUS_REG (0x5000002A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9	r	BOOST_SELECTED	Indicates that DC-DC is in boost mode	0x0
8	-	-	Reserved	0x0
7	r	BANDGAP_OK	Indicates that BANDGAP is OK	0x1
6	r	BOOST_VBAT_OK	Indicates that VBAT is above threshold while in BOOST converter mode.	0x0
5	r	LDO_ANA_OK	Indicates that LDO_ANA is in regulation. This LDO is used for the general-purpose ADC only	0x0
4	r	LDO_VDD_OK	Indicates that LDO_VDD is in regulation	0x1
3	r	LDO_OTP_OK	Indicates that LDO_OTP is in regulation	0x0
2	r	VDCDC_OK	Indicates that VDCDC is above threshold.	0x0
1	r	VBAT1V_OK	Indicates that VBAT1V is above threshold.	0x0
0	r	VBAT1V_AVAILABLE	Indicates that VBAT1V is available.	0x0

Table 27: WKUP_CTRL_REG (0x50000100)

Bit	Mode	Symbol	Description	Reset
15:14	-	-	Reserved	0x0
7	r/w	WKUP_ENABLE_IRQ	0: no interrupt will be enabled 1: if the event counter reaches the value set by WKUP_COMPARE_REG an IRQ will be generated	0x0
6	r/w	WKUP_SFT_KEYHIT	0: no effect 1: emulate key hit. The event counter will increment by 1 (after debouncing if enabled). First make this bit 0 before any new key hit can be sensed.	0x0
5:0	r/w	WKUP_DEB_VALUE	Keyboard debounce time (N*1 ms with N = 1 to 63). 0x0: no debouncing 0x1 to 0x3F: 1 ms to 63 ms debounce time	0x0

Table 28: WKUP_COMPARE_REG (0x50000102)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	COMPARE	The number of events that have to be counted before the wakeup interrupt will be given	0x0

Table 29: WKUP_RESET_IRQ_REG (0x50000104)

Bit	Mode	Symbol	Description	Reset
15:0	w	WKUP_IRQ_RST	writing any value to this register will reset the interrupt. reading always returns 0.	0x0

Table 30: WKUP_COUNTER_REG (0x50000106)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r	EVENT_VALUE	This value represents the number of events that have been counted so far. It will be reset by resetting the interrupt.	0x0

Table 31: WKUP_RESET_CNTR_REG (0x50000108)

Bit	Mode	Symbol	Description	Reset
15:0	w	WKUP_CNTR_RST	writing any value to this register will reset the event counter	0x0

Table 32: WKUP_SELECT_P0_REG (0x5000010A)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	WKUP_SELECT_P0	0: input P0x is not enabled for wakeup event counter 1: input P0x is enabled for wakeup event counter	0x0

Table 33: WKUP_SELECT_P1_REG (0x5000010C)

Bit	Mode	Symbol	Description	Reset
5:0	r/w	WKUP_SELECT_P1	0: input P1x is not enabled for wakeup event counter 1: input P1x is enabled for wakeup event counter	0x0

Table 34: WKUP_SELECT_P2_REG (0x5000010E)

Bit	Mode	Symbol	Description	Reset
9:0	r/w	WKUP_SELECT_P2	0: input P2x is not enabled for wakeup event counter 1: input P2x is enabled for wakeup event counter	0x0

Table 35: WKUP_SELECT_P3_REG (0x50000110)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	WKUP_SELECT_P3	0: input P3x is not enabled for wakeup event counter 1: input P3x is enabled for wakeup event counter	0x0

Table 36: WKUP_POL_P0_REG (0x50000112)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	WKUP_POL_P0	0: enabled input P0x will increment the event counter if that input goes high 1: enabled input P0x will increment the event counter if that input goes low	0x0

Table 37: WKUP_POL_P1_REG (0x50000114)

Bit	Mode	Symbol	Description	Reset
5:0	r/w	WKUP_POL_P1	0: enabled input P1x will increment the event counter if that input goes high 1: enabled input P1x will increment the event counter if that input goes low	0x0

Table 38: WKUP_POL_P2_REG (0x50000116)

Bit	Mode	Symbol	Description	Reset
9:0	r/w	WKUP_POL_P2	0: enabled input P2x will increment the event counter if that input goes high 1: enabled input P2x will increment the event counter if that input goes low	0x0

Table 39: WKUP_POL_P3_REG (0x50000118)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	WKUP_POL_P3	0: enabled input P3x will increment the event counter if that input goes high 1: enabled input P3x will increment the event counter if that input goes low	0x0

Table 40: QDEC_CTRL_REG (0x50000200)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:3	r/w	QD_IRQ_THRES	The number of events on either counter (X or Y) that need to be reached before an interrupt is generated. If 0 is written, then threshold is considered to be 1.	0x2
2	r	QD_IRQ_STATUS	Interrupt Status. If 1 an interrupt has occured.	0x0
1	r/w	QD_IRQ_CLR	Writing 1 to this bit clears the interrupt. This bit is auto-cleared	0x0
0	r/w	QD_IRQ_MASK	0: interrupt is masked 1: interrupt is enabled	0x0

Table 41: QDEC_XCNT_REG (0x50000202)

Bit	Mode	Symbol	Description	Reset
15:0	r	X_COUNTER	Contains a signed value of the events. Zero when channel is disabled	0x0

Table 42: QDEC_YCNT_REG (0x50000204)

Bit	Mode	Symbol	Description	Reset
15:0	r	Y_COUNTER	Contains a signed value of the events. Zero when channel is disabled	0x0

Table 43: QDEC_CLOCKDIV_REG (0x50000206)

Bit	Mode	Symbol	Description	Reset
9:0	r/w	CLOCK_DIVIDER	Contains the number of the input clock cycles minus one, that are required to generate one logic clock cycle.	0x0

Table 44: QDEC_CTRL2_REG (0x50000208)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0
11:8	r/w	CHZ_PORT_SEL	Defines which GPIOs are mapped on Channel Z 0: none 1: P0[0] -> CHZ_A, P0[1] -> CHZ_B 2: P0[2] -> CHZ_A, P0[3] -> CHZ_B 3: P0[4] -> CHZ_A, P0[5] -> CHZ_B 4: P0[6] -> CHZ_A, P0[7] -> CHZ_B 5: P1[0] -> CHZ_A, P1[1] -> CHZ_B 6: P1[2] -> CHZ_A, P1[3] -> CHZ_B 7: P2[3] -> CHZ_A, P2[4] -> CHZ_B 8: P2[5] -> CHZ_A, P2[6] -> CHZ_B 9: P2[7] -> CHZ_A, P2[8] -> CHZ_B 10: P2[9] -> CHZ_A, P2[0] -> CHZ_B 11..15: None	0
7:4	r/w	CHY_PORT_SEL	Defines which GPIOs are mapped on Channel Y 0: none 1: P0[0] -> CHY_A, P0[1] -> CHY_B 2: P0[2] -> CHY_A, P0[3] -> CHY_B 3: P0[4] -> CHY_A, P0[5] -> CHY_B 4: P0[6] -> CHY_A, P0[7] -> CHY_B 5: P1[0] -> CHY_A, P1[1] -> CHY_B 6: P1[2] -> CHY_A, P1[3] -> CHY_B 7: P2[3] -> CHY_A, P2[4] -> CHY_B 8: P2[5] -> CHY_A, P2[6] -> CHY_B 9: P2[7] -> CHY_A, P2[8] -> CHY_B 10: P2[9] -> CHY_A, P2[0] -> CHY_B 11..15: None	0
3:0	r/w	CHX_PORT_SEL	Defines which GPIOs are mapped on Channel X 0: none 1: P0[0] -> CHX_A, P0[1] -> CHX_B 2: P0[2] -> CHX_A, P0[3] -> CHX_B 3: P0[4] -> CHX_A, P0[5] -> CHX_B 4: P0[6] -> CHX_A, P0[7] -> CHX_B 5: P1[0] -> CHX_A, P1[1] -> CHX_B 6: P1[2] -> CHX_A, P1[3] -> CHX_B 7: P2[3] -> CHX_A, P2[4] -> CHX_B 8: P2[5] -> CHX_A, P2[6] -> CHX_B 9: P2[7] -> CHX_A, P2[8] -> CHX_B 10: P2[9] -> CHX_A, P2[0] -> CHX_B 11..15: None	0

Table 45: QDEC_ZCNT_REG (0x5000020A)

Bit	Mode	Symbol	Description	Reset
15:0	r	Z_COUNTER	Contains a signed value of the events. Zero when channel is disabled	0

Table 46: UART_RBR_THR_DLL_REG (0x50001000)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	RBR_THR_DLL	Receive Buffer Register: This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Transmit Holding Register: This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost. Divisor Latch (Low): This register makes up the lower 8-bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set. The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor) Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications will occur. Also, once the DLL is set, at least 8 clock cycles of the slowest DW_apb_uart clock should be allowed to pass before transmitting or receiving data.	0x0

Table 47: UART_IER_DLH_REG (0x50001004)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	PTIME_DLH7	Interrupt Enable Register: PTIME, Programmable THRE Interrupt Mode Enable. This is used to enable/disable the generation of THRE Interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[7] of the 8 bit DLH register.	0x0
6:4	-	-	Reserved	0x0

Table 47: UART_IER_DLH_REG (0x50001004)

Bit	Mode	Symbol	Description	Reset
3	r/w	EDSSI_DLH3	Interrupt Enable Register: EDSSI, Enable Modem Status Interrupt. This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[3] of the 8 bit DLH register	0x0
2	r/w	ELSI_DHL2	Interrupt Enable Register: ELSI, Enable Receiver Line Status Interrupt. This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[2] of the 8 bit DLH register.	0x0
1	r/w	ETBEI_DLH1	Interrupt Enable Register: ETBEI, Enable Transmit Holding Register Empty Interrupt. This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[1] of the 8 bit DLH register.	0x0
0	r/w	ERBFI_DLH0	Interrupt Enable Register: ERBFI, Enable Received Data Available Interrupt. This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFO's enabled). These are the second highest priority interrupts. 0 = disabled 1 = enabled Divisor Latch (High): Bit[0] of the 8 bit DLH register.	0x0

Table 48: UART_IIR_FCR_REG (0x50001008)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IIR_FCR	Interrupt Identification Register, reading this register; FIFO Control Register, writing to this register. Interrupt Identification Register: Bits[7:6], FIFO's Enabled (or FIFOSE): This is used to indicate whether the FIFO's are enabled or disabled. 00 = disabled. 11 = enabled. Bits[3:0], Interrupt ID (or IID): This indicates the highest priority pending interrupt which can be one of the following types: 0000 = modem status. 0001 = no interrupt pending. 0010 = THR empty. 0100 = received data available. 0110 = receiver line status. 0111 = busy detect. 1100 = character timeout. Bits[7:6], RCVR Trigger (or RT):. This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt will be generated. In auto flow control mode it is used to determine when the rts_n signal will be de-asserted. It also determines when the dma_rx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO 1/4 full 10 = FIFO 1/2 full 11 = FIFO 2 less than full Bits[5:4], TX Empty Trigger (or TET): This is used to select the empty threshold level at which the THRE Interrupts will be generated when the mode is active. It also determines when the dma_tx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO 1/4 full 11 = FIFO 1/2 full Bit[3], DMA Mode (or DMAM): This determines the DMA signalling mode used for the dma_tx_req_n and dma_rx_req_n output signals. 0 = mode 0 1 = mode 1 Bit[2], XMIT FIFO Reset (or XFIFOR): This resets the control portion of the transmit FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing' and it is not necessary to clear this bit. Bit[1], RCVR FIFO Reset (or RFIFOR): This resets the control portion of the receive FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing' and it is not necessary to clear this bit. Bit[0], FIFO Enable (or FIFOE): This enables/disables the transmit (XMIT) and receive (RCVR) FIFO's. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFO's will be reset.	0x0

Table 49: UART_LCR_REG (0x5000100C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	UART_DLAB	Divisor Latch Access Bit. This bit is used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.	0x0

Table 49: UART_LCR_REG (0x5000100C)

Bit	Mode	Symbol	Description	Reset
6	r/w	UART_BC	Break Control Bit. This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If active (MCR[6] set to one) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver and the sir_out_n line is forced low.	0x0
5	-	-	Reserved	0x0
4	r/w	UART_EPS	Even Parity Select. This is used to select between even and odd parity, when parity is enabled (PEN set to one). If set to one, an even number of logic 1s is transmitted or checked. If set to zero, an odd number of logic 1s is transmitted or checked.	0x0
3	r/w	UART_PEN	Parity Enable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0 = parity disabled 1 = parity enabled	0x0
2	r/w	UART_STOP	Number of stop bits. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits is transmitted. Otherwise, two stop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit. 0 = 1 stop bit 1 = 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit	0x0
1:0	r/w	UART_DLS	Data Length Select. This is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected areas follows: 00 = 5 bits 01 = 6 bits 10 = 7 bits 11 = 8 bits	0x0

Table 50: UART_MCR_REG (0x50001010)

Bit	Mode	Symbol	Description	Reset
15:7	-	-	Reserved	0x0
6	r/w	UART_SIRE	SIR Mode Enable. This is used to enable/disable the IrDA SIR Mode features as described in "IrDA 1.0 SIR Protocol" on page 53. 0 = IrDA SIR Mode disabled 1 = IrDA SIR Mode enabled	0x0
5	r/w	UART_AFCE	Auto Flow Control Enable. When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, hardware Auto Flow Control is enabled via CTS and RTS. 0 = Auto Flow Control Mode disabled 1 = Auto Flow Control Mode enabled	0x0

Table 50: UART_MCR_REG (0x50001010)

Bit	Mode	Symbol	Description	Reset
4	r/w	UART_LB	LoopBack Bit. This is used to put the UART into a diagnostic mode for test purposes. If operating in UART mode (SIR_MODE not active, MCR[6] set to zero), data on the sout line is held high, while serial data output is looped back to the sin line, internally. In this mode all the interrupts are fully functional. Also, in loopback mode, the modem control inputs (dsr_n, cts_n, ri_n, dcd_n) are disconnected and the modem control outputs (dtr_n, rts_n, out1_n, out2_n) are looped back to the inputs, internally. If operating in infrared mode (SIR_MODE active, MCR[6] set to one), data on the sir_out_n line is held low, while serial data output is inverted and looped back to the sir_in line.	0x0
3	r/w	UART_OUT2	OUT2. This is used to directly control the user-designated Output2 (out2_n) output. The value written to this location is inverted and driven out on out2_n, that is: 0 = out2_n de-asserted (logic 1) 1 = out2_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out2_n output is held inactive high while the value of this location is internally looped back to an input.	0x0
2	r/w	UART_OUT1	OUT1. This is used to directly control the user-designated Output1 (out1_n) output. The value written to this location is inverted and driven out on out1_n, that is: 0 = out1_n de-asserted (logic 1) 1 = out1_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out1_n output is held inactive high while the value of this location is internally looped back to an input.	0x0
1	r/w	UART_RTS	Request to Send. This is used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto Flow Control is disabled (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. When Auto Flow Control is enabled (MCR[5] set to one) and FIFOs are enabled (FCR[0] set to one), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal is de-asserted when MCR[1] is set low. Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive (high) while the value of this location is internally looped back to an input.	0x0
0	-	-	Reserved	0x0

Table 51: UART_LSR_REG (0x50001014)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 51: UART_LSR_REG (0x50001014)

Bit	Mode	Symbol	Description	Reset
7	r	UART_RFE	Receiver FIFO Error bit. This bit is only relevant when FIFOs are enabled (FCR[0] set to one). This is used to indicate if there is at least one parity error, framing error, or break indication in the FIFO. 0 = no error in RX FIFO 1 = error in RX FIFO This bit is cleared when the LSR is read and the character with the error is at the top of the receiver FIFO and there are no subsequent errors in the FIFO.	0x0
6	r	UART_TEMT	Transmitter Empty bit. If FIFOs enabled (FCR[0] set to one), this bit is set whenever the Transmitter Shift Register and the FIFO are both empty. If FIFOs are disabled, this bit is set whenever the Transmitter Holding Register and the Transmitter Shift Register are both empty.	0x1
5	r	UART_THRE	Transmit Holding Register Empty bit. If THRE mode is disabled (IER[7] set to zero) and regardless of FIFO's being implemented/enabled or not, this bit indicates that the THR or TX FIFO is empty. This bit is set whenever data is transferred from the THR or TX FIFO to the transmitter shift register and no new data has been written to the THR or TX FIFO. This also causes a THRE Interrupt to occur, if the THRE Interrupt is enabled. If both modes are active (IER[7] set to one and FCR[0] set to one respectively), the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the FCR[5:4] threshold setting.	0x1
4	r	UART_B1	Break Interrupt bit. This is used to indicate the detection of a break sequence on the serial input data. If in UART mode (SIR_MODE == Disabled), it is set whenever the serial input, sin, is held in a logic '0' state for longer than the sum of start time + data bits + parity + stop bits. If in infrared mode (SIR_MODE == Enabled), it is set whenever the serial input, sir_in, is continuously pulsed to logic '0' for longer than the sum of start time + data bits + parity + stop bits. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.	0x0

Table 51: UART_LSR_REG (0x50001014)

Bit	Mode	Symbol	Description	Reset
3	r	UART_FE	Framing Error bit. This is used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data. In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs, the UART tries to resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no framing error 1 = framing error Reading the LSR clears the FE bit.	0x0
2	r	UART_PE	Parity Error bit. This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no parity error 1 = parity error Reading the LSR clears the PE bit.	0x0
1	r	UART_OE	Overrun error bit. This is used to indicate the occurrence of an overrun error. This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost. 0 = no overrun error 1 = overrun error Reading the LSR clears the OE bit.	0x0
0	r	UART_DR	Data Ready bit. This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0 = no data ready 1 = data ready This bit is cleared when the RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.	0x0

Table 52: UART_MSR_REG (0x50001018)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 52: UART_MSR_REG (0x50001018)

Bit	Mode	Symbol	Description	Reset
7	r	UART_DCD	Data Carrier Detect. This is used to indicate the current state of the modem control line dcd_n. This bit is the complement of dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set. 0 = dcd_n input is de-asserted (logic 1) 1 = dcd_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DCD is the same as MCR[3] (Out2).	0x0
6	r	UART_R1	Ring Indicator. This is used to indicate the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0 = ri_n input is de-asserted (logic 1) 1 = ri_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), RI is the same as MCR[2] (Out1).	0x0
5	-	-	Reserved	0x0
4	r	UART_CTS	Clear to Send. This is used to indicate the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with the UART Ctrl. 0 = cts_n input is de-asserted (logic 1) 1 = cts_n input is asserted (logic 0) In Loopback Mode (MCR[4] = 1), CTS is the same as MCR[1] (RTS).	0x0
3	r	UART_DDCD	Delta Data Carrier Detect. This is used to indicate that the modem control line dcd_n has changed since the last time the MSR was read. 0 = no change on dcd_n since last read of MSR 1 = change on dcd_n since last read of MSR Reading the MSR clears the DDCD bit. In Loopback Mode (MCR[4] = 1), DDCD reflects changes on MCR[3] (Out2). Note, if the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.	0x0
2	r	UART_TERI	Trailing Edge of Ring Indicator. This is used to indicate that a change on the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the MSR was read. 0 = no change on ri_n since last read of MSR 1 = change on ri_n since last read of MSR Reading the MSR clears the TERI bit. In Loopback Mode (MCR[4] = 1), TERI reflects when MCR[2] (Out1) has changed state from a high to a low.	0x0
1	-	-	Reserved	0x0

Table 52: UART_MSR_REG (0x50001018)

Bit	Mode	Symbol	Description	Reset
0	r	UART_DCTS	Delta Clear to Send. This is used to indicate that the modem control line cts_n has changed since the last time the MSR was read. 0 = no change on cts_n since last read of MSR 1 = change on cts_n since last read of MSR Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] = 1), DCTS reflects changes on MCR[1] (RTS). Note, if the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.	0x0

Table 53: UART_SCR_REG (0x5000101C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	UART_SCRATCH_PAD	This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART Ctrl.	0x0

Table 54: UART_LPDLL_REG (0x50001020)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	UART_LPDLL	This register makes up the lower 8-bits of a 16-bit, read/write, Low Power Divisor Latch register that contains the baud rate divisor for the UART, which must give a baud rate of 115.2K. This is required for SIR Low Power (minimum pulse width) detection at the receiver. This register may be accessed only when the DLAB bit (LCR[7]) is set. The output low-power baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: Low power baud rate = (serial clock frequency)/(16* divisor) Therefore, a divisor must be selected to give a baud rate of 115.2K. NOTE: When the Low Power Divisor Latch registers (LPDLL and LPDLH) are set to 0, the low-power baud clock is disabled and no low-power pulse detection (or any pulse detection) occurs at the receiver. Also, once the LPDLL is set, at least eight clock cycles of the slowest UART Ctrl clock should be allowed to pass before transmitting or receiving data.	0x0

Table 55: UART_LPDLH_REG (0x50001024)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 55: UART_LPDLH_REG (0x50001024)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	UART_LPDLH	This register makes up the upper 8-bits of a 16-bit, read/ write, Low Power Divisor Latch register that contains the baud rate divisor for the UART, which must give a baud rate of 115.2K. This is required for SIR Low Power (minimum pulse width) detection at the receiver. This register may be accessed only when the DLAB bit (LCR[7]) is set. The output low-power baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: Low power baud rate = (serial clock frequency)/(16* divisor) Therefore, a divisor must be selected to give a baud rate of 115.2K. NOTE: When the Low Power Divisor Latch registers (LPDLL and LPDLH) are set to 0, the low-power baud clock is disabled and no low-power pulse detection (or any pulse detection) occurs at the receiver. Also, once the LPDLH is set, at least eight clock cycles of the slowest UART Ctrl clock should be allowed to pass before transmitting or receiving data.	0x0

Table 56: UART_SRBR_STHR0_REG (0x50001030)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 57: UART_SRBR_STHR1_REG (0x50001034)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 58: UART_SRBR_STHR2_REG (0x50001038)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 58: UART_SRBR_STHR2_REG (0x50001038)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 59: UART_SRBR_STHR3_REG (0x5000103C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 59: UART_SRBR_STHR3_REG (0x5000103C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 60: UART_SRBR_STHR4_REG (0x50001040)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 60: UART_SRBR_STHR4_REG (0x50001040)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 61: UART_SRBR_STHR5_REG (0x50001044)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 61: UART_SRBR_STHR5_REG (0x50001044)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 62: UART_SRBR_STHR6_REG (0x50001048)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 62: UART_SRBR_STHR6_REG (0x50001048)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 63: UART_SRBR_STHR7_REG (0x5000104C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 63: UART_SRBR_STHR7_REG (0x5000104C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 64: UART_SRBR_STHR8_REG (0x50001050)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 64: UART_SRBR_STHR8_REG (0x50001050)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 65: UART_SRBR_STHR9_REG (0x50001054)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 65: UART_SRBR_STHR9_REG (0x50001054)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 66: UART_SRBR_STHR10_REG (0x50001058)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 66: UART_SRBR_STHR10_REG (0x50001058)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 67: UART_SRBR_STHR11_REG (0x5000105C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 67: UART_SRBR_STHR11_REG (0x5000105C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 68: UART_SRBR_STHR12_REG (0x50001060)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 68: UART_SRBR_STHR12_REG (0x50001060)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 69: UART_SRBR_STHR13_REG (0x50001064)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 69: UART_SRBR_STHR13_REG (0x50001064)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 70: UART_SRBR_STHR14_REG (0x50001068)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 70: UART_SRBR_STHR14_REG (0x50001068)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 71: UART_SRBR_STHR15_REG (0x5000106C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 71: UART_SRBR_STHR15_REG (0x5000106C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 72: UART_USR_REG (0x5000107C)

Bit	Mode	Symbol	Description	Reset
15:5	-	-	Reserved	0x0
4	r	UART_RFF	Receive FIFO Full. This is used to indicate that the receive FIFO is completely full. 0 = Receive FIFO not full 1 = Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.	0x0
3	r	UART_RFNE	Receive FIFO Not Empty. This is used to indicate that the receive FIFO contains one or more entries. 0 = Receive FIFO is empty 1 = Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.	0x0
2	r	UART_TFE	Transmit FIFO Empty. This is used to indicate that the transmit FIFO is completely empty. 0 = Transmit FIFO is not empty 1 = Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty.	0x1

Table 72: UART_USR_REG (0x5000107C)

Bit	Mode	Symbol	Description	Reset
1	r	UART_TFNF	Transmit FIFO Not Full. This is used to indicate that the transmit FIFO in not full. 0 = Transmit FIFO is full 1 = Transmit FIFO is not full This bit is cleared when the TX FIFO is full.	0x1
0	-	-	Reserved	0x0

Table 73: UART_TFL_REG (0x50001080)

Bit	Mode	Symbol	Description	Reset
15:0	r	UART_TRANSMIT_FIFO_LEVEL	Transmit FIFO Level. This is indicates the number of data entries in the transmit FIFO.	0x0

Table 74: UART_RFL_REG (0x50001084)

Bit	Mode	Symbol	Description	Reset
15:0	r	UART_RECEIVE_FIFO_LEVEL	Receive FIFO Level. This is indicates the number of data entries in the receive FIFO.	0x0

Table 75: UART_SRR_REG (0x50001088)

Bit	Mode	Symbol	Description	Reset
15:3	-	-	Reserved	0x0
2	w	UART_XFR	XMIT FIFO Reset. This is a shadow register for the XMIT FIFO Reset bit (FCR[2]). This can be used to remove the burden on software having to store previously written FCR values (which are pretty static) just to reset the transmit FIFO. This resets the control portion of the transmit FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing'. It is not necessary to clear this bit.	0x0
1	w	UART_RFR	RCVR FIFO Reset. This is a shadow register for the RCVR FIFO Reset bit (FCR[1]). This can be used to remove the burden on software having to store previously written FCR values (which are pretty static) just to reset the receive FIFO This resets the control portion of the receive FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing'. It is not necessary to clear this bit.	0x0
0	w	UART_UR	UART Reset. This asynchronously resets the UART Ctrl and synchronously removes the reset assertion. For a two clock implementation both pclk and sclk domains are reset.	0x0

Table 76: UART_SRTS_REG (0x5000108C)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 76: UART_SRTS_REG (0x5000108C)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_SHADOW_REQUEST_TO_SEND	Shadow Request to Send. This is a shadow register for the RTS bit (MCR[1]), this can be used to remove the burden of having to perform a read-modify-write on the MCR. This is used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART Ctrl is ready to exchange data. When Auto Flow Control is disabled (MCR[5] = 0), the rts_n signal is set low by programming MCR[1] (RTS) to a high. When Auto Flow Control is enabled (MCR[5] = 1) and FIFOs are enabled (FCR[0] = 1), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). Note that in Loopback mode (MCR[4] = 1), the rts_n output is held inactive-high while the value of this location is internally looped back to an input.	0x0

Table 77: UART_SBCR_REG (0x50001090)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	UART_SHADOW_BREAK_CONTROL	Shadow Break Control Bit. This is a shadow register for the Break bit (LCR[6]), this can be used to remove the burden of having to performing a read modify write on the LCR. This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If SIR_MODE active (MCR[6] = 1) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver.	0x0

Table 78: UART_SDMAM_REG (0x50001094)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	UART_SHADOW_DMA_MODE	Shadow DMA Mode. This is a shadow register for the DMA mode bit (FCR[3]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the DMA Mode bit gets updated. This determines the DMA signalling mode used for the dma_tx_req_n and dma_rx_req_n output signals. 0 = mode 0 1 = mode 1	0x0

Table 79: UART_SFE_REG (0x50001098)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 79: UART_SFE_REG (0x50001098)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_SHADOW_FIFO_ENABLE	Shadow FIFO Enable. This is a shadow register for the FIFO enable bit (FCR[0]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the FIFO enable bit gets updated.This enables/disables the transmit (XMIT) and receive (RCVR) FIFOs. If this bit is set to zero (disabled) after being enabled then both the XMIT and RCVR controller portion of FIFOs are reset.	0x0

Table 80: UART_SRT_REG (0x5000109C)

Bit	Mode	Symbol	Description	Reset
15:2	-	-	Reserved	0x0
1:0	r/w	UART_SHADOW_RCVR_TRIGGER	Shadow RCVR Trigger. This is a shadow register for the RCVR trigger bits (FCR[7:6]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the RCVR trigger bit gets updated. This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. It also determines when the dma_rx_req_n signal is asserted when DMA Mode (FCR[3]) = 1. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO Â¼ full 10 = FIFO Â½ full 11 = FIFO 2 less than full	0x0

Table 81: UART_STET_REG (0x500010A0)

Bit	Mode	Symbol	Description	Reset
15:2	-	-	Reserved	0x0
1:0	r/w	UART_SHADOW_TX_EMPTY_TRIGGER	Shadow TX Empty Trigger. This is a shadow register for the TX empty trigger bits (FCR[5:4]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the TX empty trigger bit gets updated. This is used to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO Â¼ full 11 = FIFO Â½ full	0x0

Table 82: UART_HTX_REG (0x500010A4)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 82: UART_HTX_REG (0x500010A4)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_HALT_TX	This register is use to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled. 0 = Halt TX disabled 1 = Halt TX enabled Note, if FIFOs are implemented and not enabled, the setting of the halt TX register has no effect on operation.	0x0

Table 83: UART_CPR_REG (0x500010F4)

Bit	Mode	Symbol	Description	Reset
15:0	r	CPR	Component Parameter Register	0x0

Table 84: UART_UCV_REG (0x500010F8)

Bit	Mode	Symbol	Description	Reset
15:0	r	UCV	Component Version	0x3330382A

Table 85: UART_CTR_REG (0x500010FC)

Bit	Mode	Symbol	Description	Reset
15:0	r	CTR	Component Type Register	0x44570110

Table 86: UART2_RBR_THR_DLL_REG (0x50001100)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 86: UART2_RBR_THR_DLL_REG (0x50001100)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	RBR_THR_DLL	Receive Buffer Register: This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Transmit Holding Register: This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost. Divisor Latch (Low): This register makes up the lower 8-bits of a 16-bit, read/write, Divisor Latch register that contains the baud rate divisor for the UART. This register may only be accessed when the DLAB bit (LCR[7]) is set. The output baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: baud rate = (serial clock freq) / (16 * divisor) Note that with the Divisor Latch Registers (DLL and DLH) set to zero, the baud clock is disabled and no serial communications will occur. Also, once the DLL is set, at least 8 clock cycles of the slowest DW_apb_uart clock should be allowed to pass before transmitting or receiving data.	0x0

Table 87: UART2_IER_DLH_REG (0x50001104)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	PTIME_DLH7	Interrupt Enable Register: PTIME, Programmable THRE Interrupt Mode Enable. This is used to enable/disable the generation of THRE Interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[7] of the 8 bit DLH register.	0x0
6:4	-	-	Reserved	0x0
3	r/w	EDSSI_DLH3	Interrupt Enable Register: EDSSI, Enable Modem Status Interrupt. This is used to enable/disable the generation of Modem Status Interrupt. This is the fourth highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[3] of the 8 bit DLH register	0x0

Table 87: UART2_IER_DLH_REG (0x50001104)

Bit	Mode	Symbol	Description	Reset
2	r/w	ELSI_DHL2	Interrupt Enable Register: ELSI, Enable Receiver Line Status Interrupt. This is used to enable/disable the generation of Receiver Line Status Interrupt. This is the highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[2] of the 8 bit DLH register.	0x0
1	r/w	ETBEI_DLH1	Interrupt Enable Register: ETBEI, Enable Transmit Holding Register Empty Interrupt. This is used to enable/disable the generation of Transmitter Holding Register Empty Interrupt. This is the third highest priority interrupt. 0 = disabled 1 = enabled Divisor Latch (High): Bit[1] of the 8 bit DLH register.	0x0
0	r/w	ERBFI_DLH0	Interrupt Enable Register: ERBFI, Enable Received Data Available Interrupt. This is used to enable/disable the generation of Received Data Available Interrupt and the Character Timeout Interrupt (if in FIFO mode and FIFO's enabled). These are the second highest priority interrupts. 0 = disabled 1 = enabled Divisor Latch (High): Bit[0] of the 8 bit DLH register.	0x0

Table 88: UART2_IIR_FCR_REG (0x50001108)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IIR_FCR	Interrupt Identification Register, reading this register; FIFO Control Register, writing to this register. Interrupt Identification Register: Bits[7:6], FIFO's Enabled (or FIFOSE): This is used to indicate whether the FIFO's are enabled or disabled. 00 = disabled. 11 = enabled. Bits[3:0], Interrupt ID (or IID): This indicates the highest priority pending interrupt which can be one of the following types: 0000 = modem status. 0001 = no interrupt pending. 0010 = THR empty. 0100 = received data available. 0110 = receiver line status. 0111 = busy detect. 1100 = character timeout. Bits[7:6], RCVR Trigger (or RT):. This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt will be generated. In auto flow control mode it is used to determine when the rts_n signal will be de-asserted. It also determines when the dma_rx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO 1/4 full 10 = FIFO 1/2 full 11 = FIFO 2 less than full Bits[5:4], TX Empty Trigger (or TET): This is used to select the empty threshold level at which the THRE Interrupts will be generated when the mode is active. It also determines when the dma_tx_req_n signal will be asserted when in certain modes of operation. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO 1/4 full 11 = FIFO 1/2 full Bit[3], DMA Mode (or DMAM): This determines the DMA signalling mode used for the dma_tx_req_n and dma_rx_req_n output signals. 0 = mode 0 1 = mode 1 Bit[2], XMIT FIFO Reset (or XFIFOR): This resets the control portion of the transmit FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing' and it is not necessary to clear this bit. Bit[1], RCVR FIFO Reset (or RFIFOR): This resets the control portion of the receive FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing' and it is not necessary to clear this bit. Bit[0], FIFO Enable (or FIFOE): This enables/disables the transmit (XMIT) and receive (RCVR) FIFO's. Whenever the value of this bit is changed both the XMIT and RCVR controller portion of FIFO's will be reset.	0x0

Table 89: UART2_LCR_REG (0x5000110C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7	r/w	UART_DLAB	Divisor Latch Access Bit. This bit is used to enable reading and writing of the Divisor Latch register (DLL and DLH) to set the baud rate of the UART. This bit must be cleared after initial baud rate setup in order to access other registers.	0x0

Table 89: UART2_LCR_REG (0x5000110C)

Bit	Mode	Symbol	Description	Reset
6	r/w	UART_BC	Break Control Bit. This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If active (MCR[6] set to one) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver and the sir_out_n line is forced low.	0x0
5	-	-	Reserved	0x0
4	r/w	UART_EPS	Even Parity Select. This is used to select between even and odd parity, when parity is enabled (PEN set to one). If set to one, an even number of logic 1s is transmitted or checked. If set to zero, an odd number of logic 1s is transmitted or checked.	0x0
3	r/w	UART_PEN	Parity Enable. This bit is used to enable and disable parity generation and detection in transmitted and received serial character respectively. 0 = parity disabled 1 = parity enabled	0x0
2	r/w	UART_STOP	Number of stop bits. This is used to select the number of stop bits per character that the peripheral transmits and receives. If set to zero, one stop bit is transmitted in the serial data. If set to one and the data bits are set to 5 (LCR[1:0] set to zero) one and a half stop bits is transmitted. Otherwise, two stop bits are transmitted. Note that regardless of the number of stop bits selected, the receiver checks only the first stop bit. 0 = 1 stop bit 1 = 1.5 stop bits when DLS (LCR[1:0]) is zero, else 2 stop bit	0x0
1:0	r/w	UART_DLS	Data Length Select. This is used to select the number of data bits per character that the peripheral transmits and receives. The number of bit that may be selected areas follows: 00 = 5 bits 01 = 6 bits 10 = 7 bits 11 = 8 bits	0x0

Table 90: UART2_MCR_REG (0x50001110)

Bit	Mode	Symbol	Description	Reset
15:7	-	-	Reserved	0x0
6	r/w	UART_SIRE	SIR Mode Enable. This is used to enable/disable the IrDA SIR Mode features as described in "IrDA 1.0 SIR Protocol" on page 53. 0 = IrDA SIR Mode disabled 1 = IrDA SIR Mode enabled	0x0
5	r/w	UART_AFCE	Auto Flow Control Enable. When FIFOs are enabled and the Auto Flow Control Enable (AFCE) bit is set, hardware Auto Flow Control is enabled via CTS and RTS. 0 = Auto Flow Control Mode disabled 1 = Auto Flow Control Mode enabled	0x0

Table 90: UART2_MCR_REG (0x50001110)

Bit	Mode	Symbol	Description	Reset
4	r/w	UART_LB	LoopBack Bit. This is used to put the UART into a diagnostic mode for test purposes. If operating in UART mode (SIR_MODE not active, MCR[6] set to zero), data on the sout line is held high, while serial data output is looped back to the sin line, internally. In this mode all the interrupts are fully functional. Also, in loopback mode, the modem control inputs (dsr_n, cts_n, ri_n, dcd_n) are disconnected and the modem control outputs (dtr_n, rts_n, out1_n, out2_n) are looped back to the inputs, internally. If operating in infrared mode (SIR_MODE active, MCR[6] set to one), data on the sir_out_n line is held low, while serial data output is inverted and looped back to the sir_in line.	0x0
3	r/w	UART_OUT2	OUT2. This is used to directly control the user-designated Output2 (out2_n) output. The value written to this location is inverted and driven out on out2_n, that is: 0 = out2_n de-asserted (logic 1) 1 = out2_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out2_n output is held inactive high while the value of this location is internally looped back to an input.	0x0
2	r/w	UART_OUT1	OUT1. This is used to directly control the user-designated Output1 (out1_n) output. The value written to this location is inverted and driven out on out1_n, that is: 0 = out1_n de-asserted (logic 1) 1 = out1_n asserted (logic 0) Note that in Loopback mode (MCR[4] set to one), the out1_n output is held inactive high while the value of this location is internally looped back to an input.	0x0
1	r/w	UART_RTS	Request to Send. This is used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART is ready to exchange data. When Auto Flow Control is disabled (MCR[5] set to zero), the rts_n signal is set low by programming MCR[1] (RTS) to a high. When Auto Flow Control is enabled (MCR[5] set to one) and FIFOs are enabled (FCR[0] set to one), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). The rts_n signal is de-asserted when MCR[1] is set low. Note that in Loopback mode (MCR[4] set to one), the rts_n output is held inactive (high) while the value of this location is internally looped back to an input.	0x0
0	-	-	Reserved	0x0

Table 91: UART2_LSR_REG (0x50001114)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 91: UART2_LSR_REG (0x50001114)

Bit	Mode	Symbol	Description	Reset
7	r	UART_RFE	Receiver FIFO Error bit. This bit is only relevant when FIFOs are enabled (FCR[0] set to one). This is used to indicate if there is at least one parity error, framing error, or break indication in the FIFO. 0 = no error in RX FIFO 1 = error in RX FIFO This bit is cleared when the LSR is read and the character with the error is at the top of the receiver FIFO and there are no subsequent errors in the FIFO.	0x0
6	r	UART_TEMT	Transmitter Empty bit. If FIFOs enabled (FCR[0] set to one), this bit is set whenever the Transmitter Shift Register and the FIFO are both empty. If FIFOs are disabled, this bit is set whenever the Transmitter Holding Register and the Transmitter Shift Register are both empty.	0x1
5	r	UART_THRE	Transmit Holding Register Empty bit. If THRE mode is disabled (IER[7] set to zero) and regardless of FIFO's being implemented/enabled or not, this bit indicates that the THR or TX FIFO is empty. This bit is set whenever data is transferred from the THR or TX FIFO to the transmitter shift register and no new data has been written to the THR or TX FIFO. This also causes a THRE Interrupt to occur, if the THRE Interrupt is enabled. If both modes are active (IER[7] set to one and FCR[0] set to one respectively), the functionality is switched to indicate the transmitter FIFO is full, and no longer controls THRE interrupts, which are then controlled by the FCR[5:4] threshold setting.	0x1
4	r	UART_B1	Break Interrupt bit. This is used to indicate the detection of a break sequence on the serial input data. If in UART mode (SIR_MODE == Disabled), it is set whenever the serial input, sin, is held in a logic '0' state for longer than the sum of start time + data bits + parity + stop bits. If in infrared mode (SIR_MODE == Enabled), it is set whenever the serial input, sir_in, is continuously pulsed to logic '0' for longer than the sum of start time + data bits + parity + stop bits. A break condition on serial input causes one and only one character, consisting of all zeros, to be received by the UART. In the FIFO mode, the character associated with the break condition is carried through the FIFO and is revealed when the character is at the top of the FIFO. Reading the LSR clears the BI bit. In the non-FIFO mode, the BI indication occurs immediately and persists until the LSR is read.	0x0

Table 91: UART2_LSR_REG (0x50001114)

Bit	Mode	Symbol	Description	Reset
3	r	UART_FE	Framing Error bit. This is used to indicate the occurrence of a framing error in the receiver. A framing error occurs when the receiver does not detect a valid STOP bit in the received data. In the FIFO mode, since the framing error is associated with a character received, it is revealed when the character with the framing error is at the top of the FIFO. When a framing error occurs, the UART tries to resynchronize. It does this by assuming that the error was due to the start bit of the next character and then continues receiving the other bit i.e. data, and/or parity and stop. It should be noted that the Framing Error (FE) bit (LSR[3]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no framing error 1 = framing error Reading the LSR clears the FE bit.	0x0
2	r	UART_PE	Parity Error bit. This is used to indicate the occurrence of a parity error in the receiver if the Parity Enable (PEN) bit (LCR[3]) is set. In the FIFO mode, since the parity error is associated with a character received, it is revealed when the character with the parity error arrives at the top of the FIFO. It should be noted that the Parity Error (PE) bit (LSR[2]) is set if a break interrupt has occurred, as indicated by Break Interrupt (BI) bit (LSR[4]). 0 = no parity error 1 = parity error Reading the LSR clears the PE bit.	0x0
1	r	UART_OE	Overrun error bit. This is used to indicate the occurrence of an overrun error. This occurs if a new data character was received before the previous data was read. In the non-FIFO mode, the OE bit is set when a new character arrives in the receiver before the previous character was read from the RBR. When this happens, the data in the RBR is overwritten. In the FIFO mode, an overrun error occurs when the FIFO is full and a new character arrives at the receiver. The data in the FIFO is retained and the data in the receive shift register is lost. 0 = no overrun error 1 = overrun error Reading the LSR clears the OE bit.	0x0
0	r	UART_DR	Data Ready bit. This is used to indicate that the receiver contains at least one character in the RBR or the receiver FIFO. 0 = no data ready 1 = data ready This bit is cleared when the RBR is read in non-FIFO mode, or when the receiver FIFO is empty, in FIFO mode.	0x0

Table 92: UART2_MSR_REG (0x50001118)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 92: UART2_MSR_REG (0x50001118)

Bit	Mode	Symbol	Description	Reset
7	r	UART_DCD	Data Carrier Detect. This is used to indicate the current state of the modem control line dcd_n. This bit is the complement of dcd_n. When the Data Carrier Detect input (dcd_n) is asserted it is an indication that the carrier has been detected by the modem or data set. 0 = dcd_n input is de-asserted (logic 1) 1 = dcd_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), DCD is the same as MCR[3] (Out2).	0x0
6	r	UART_R1	Ring Indicator. This is used to indicate the current state of the modem control line ri_n. This bit is the complement of ri_n. When the Ring Indicator input (ri_n) is asserted it is an indication that a telephone ringing signal has been received by the modem or data set. 0 = ri_n input is de-asserted (logic 1) 1 = ri_n input is asserted (logic 0) In Loopback Mode (MCR[4] set to one), RI is the same as MCR[2] (Out1).	0x0
5	-	-	Reserved	0x0
4	r	UART_CTS	Clear to Send. This is used to indicate the current state of the modem control line cts_n. This bit is the complement of cts_n. When the Clear to Send input (cts_n) is asserted it is an indication that the modem or data set is ready to exchange data with the UART Ctrl. 0 = cts_n input is de-asserted (logic 1) 1 = cts_n input is asserted (logic 0) In Loopback Mode (MCR[4] = 1), CTS is the same as MCR[1] (RTS).	0x0
3	r	UART_DDCD	Delta Data Carrier Detect. This is used to indicate that the modem control line dcd_n has changed since the last time the MSR was read. 0 = no change on dcd_n since last read of MSR 1 = change on dcd_n since last read of MSR Reading the MSR clears the DDCD bit. In Loopback Mode (MCR[4] = 1), DDCD reflects changes on MCR[3] (Out2). Note, if the DDCD bit is not set and the dcd_n signal is asserted (low) and a reset occurs (software or otherwise), then the DDCD bit is set when the reset is removed if the dcd_n signal remains asserted.	0x0
2	r	UART_TERI	Trailing Edge of Ring Indicator. This is used to indicate that a change on the input ri_n (from an active-low to an inactive-high state) has occurred since the last time the MSR was read. 0 = no change on ri_n since last read of MSR 1 = change on ri_n since last read of MSR Reading the MSR clears the TERI bit. In Loopback Mode (MCR[4] = 1), TERI reflects when MCR[2] (Out1) has changed state from a high to a low.	0x0
1	-	-	Reserved	0x0

Table 92: UART2_MSR_REG (0x50001118)

Bit	Mode	Symbol	Description	Reset
0	r	UART_DCTS	Delta Clear to Send. This is used to indicate that the modem control line cts_n has changed since the last time the MSR was read. 0 = no change on cts_n since last read of MSR 1 = change on cts_n since last read of MSR Reading the MSR clears the DCTS bit. In Loopback Mode (MCR[4] = 1), DCTS reflects changes on MCR[1] (RTS). Note, if the DCTS bit is not set and the cts_n signal is asserted (low) and a reset occurs (software or otherwise), then the DCTS bit is set when the reset is removed if the cts_n signal remains asserted.	0x0

Table 93: UART2_SCR_REG (0x5000111C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	UART_SCRATCH_PAD	This register is for programmers to use as a temporary storage space. It has no defined purpose in the UART Ctrl.	0x0

Table 94: UART2_LPDLL_REG (0x50001120)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	UART_LPDLL	This register makes up the lower 8-bits of a 16-bit, read/write, Low Power Divisor Latch register that contains the baud rate divisor for the UART, which must give a baud rate of 115.2K. This is required for SIR Low Power (minimum pulse width) detection at the receiver. This register may be accessed only when the DLAB bit (LCR[7]) is set. The output low-power baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: Low power baud rate = (serial clock frequency)/(16* divisor) Therefore, a divisor must be selected to give a baud rate of 115.2K. NOTE: When the Low Power Divisor Latch registers (LPDLL and LPDLH) are set to 0, the low-power baud clock is disabled and no low-power pulse detection (or any pulse detection) occurs at the receiver. Also, once the LPDLL is set, at least eight clock cycles of the slowest UART Ctrl clock should be allowed to pass before transmitting or receiving data.	0x0

Table 95: UART2_LPDLH_REG (0x50001124)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 95: UART2_LPDLH_REG (0x50001124)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	UART_LPDLH	This register makes up the upper 8-bits of a 16-bit, read/ write, Low Power Divisor Latch register that contains the baud rate divisor for the UART, which must give a baud rate of 115.2K. This is required for SIR Low Power (minimum pulse width) detection at the receiver. This register may be accessed only when the DLAB bit (LCR[7]) is set. The output low-power baud rate is equal to the serial clock (sclk) frequency divided by sixteen times the value of the baud rate divisor, as follows: Low power baud rate = (serial clock frequency)/(16* divisor) Therefore, a divisor must be selected to give a baud rate of 115.2K. NOTE: When the Low Power Divisor Latch registers (LPDLL and LPDLH) are set to 0, the low-power baud clock is disabled and no low-power pulse detection (or any pulse detection) occurs at the receiver. Also, once the LPDLH is set, at least eight clock cycles of the slowest UART Ctrl clock should be allowed to pass before transmitting or receiving data.	0x0

Table 96: UART2_SRBR_STHR0_REG (0x50001130)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 97: UART2_SRBR_STHR1_REG (0x50001134)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 98: UART2_SRBR_STHR2_REG (0x50001138)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 98: UART2_SRBR_STHR2_REG (0x50001138)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 99: UART2_SRBR_STHR3_REG (0x5000113C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 99: UART2_SRBR_STHR3_REG (0x5000113C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 100: UART2_SRBR_STHR4_REG (0x50001140)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 100: UART2_SRBR_STHR4_REG (0x50001140)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 101: UART2_SRBR_STHR5_REG (0x50001144)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 101: UART2_SRBR_STHR5_REG (0x50001144)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 102: UART2_SRBR_STHR6_REG (0x50001148)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 102: UART2_SRBR_STHR6_REG (0x50001148)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 103: UART2_SRBR_STHR7_REG (0x5000114C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 103: UART2_SRBR_STHR7_REG (0x5000114C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 104: UART2_SRBR_STHR8_REG (0x50001150)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 104: UART2_SRBR_STHR8_REG (0x50001150)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 105: UART2_SRBR_STHR9_REG (0x50001154)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 105: UART2_SRBR_STHR9_REG (0x50001154)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 106: UART2_SRBR_STHR10_REG (0x50001158)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 106: UART2_SRBR_STHR10_REG (0x50001158)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 107: UART2_SRBR_STHR11_REG (0x5000115C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 107: UART2_SRBR_STHR11_REG (0x5000115C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 108: UART2_SRBR_STHR12_REG (0x50001160)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 108: UART2_SRBR_STHR12_REG (0x50001160)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 109: UART2_SRBR_STHR13_REG (0x50001164)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 109: UART2_SRBR_STHR13_REG (0x50001164)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 110: UART2_SRBR_STHR14_REG (0x50001168)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 110: UART2_SRBR_STHR14_REG (0x50001168)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 111: UART2_SRBR_STHR15_REG (0x5000116C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0

Table 111: UART2_SRBR_STHR15_REG (0x5000116C)

Bit	Mode	Symbol	Description	Reset
7:0	r/w	SRBR_STHRX	Shadow Receive Buffer Register x: This is a shadow register for the RBR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains the data byte received on the serial input port (sin) in UART mode or the serial infrared input (sir_in) in infrared mode. The data in this register is valid only if the Data Ready (DR) bit in the Line status Register (LSR) is set. If FIFOs are disabled (FCR[0] set to zero), the data in the RBR must be read before the next data arrives, otherwise it will be overwritten, resulting in an overrun error. If FIFOs are enabled (FCR[0] set to one), this register accesses the head of the receive FIFO. If the receive FIFO is full and this register is not read before the next data character arrives, then the data already in the FIFO will be preserved but any incoming data will be lost. An overrun error will also occur. Shadow Transmit Holding Register 0: This is a shadow register for the THR and has been allocated sixteen 32-bit locations so as to accommodate burst accesses from the master. This register contains data to be transmitted on the serial output port (sout) in UART mode or the serial infrared output (sir_out_n) in infrared mode. Data should only be written to the THR when the THR Empty (THRE) bit (LSR[5]) is set. If FIFO's are disabled (FCR[0] set to zero) and THRE is set, writing a single character to the THR clears the THRE. Any additional writes to the THR before the THRE is set again causes the THR data to be overwritten. If FIFO's are enabled (FCR[0] set to one) and THRE is set, x number of characters of data may be written to the THR before the FIFO is full. The number x (default=16) is determined by the value of FIFO Depth that you set during configuration. Any attempt to write data when the FIFO is full results in the write data being lost.	0x0

Table 112: UART2_USR_REG (0x5000117C)

Bit	Mode	Symbol	Description	Reset
15:5	-	-	Reserved	0x0
4	r	UART_RFF	Receive FIFO Full. This is used to indicate that the receive FIFO is completely full. 0 = Receive FIFO not full 1 = Receive FIFO Full This bit is cleared when the RX FIFO is no longer full.	0x0
3	r	UART_RFNE	Receive FIFO Not Empty. This is used to indicate that the receive FIFO contains one or more entries. 0 = Receive FIFO is empty 1 = Receive FIFO is not empty This bit is cleared when the RX FIFO is empty.	0x0
2	r	UART_TFE	Transmit FIFO Empty. This is used to indicate that the transmit FIFO is completely empty. 0 = Transmit FIFO is not empty 1 = Transmit FIFO is empty This bit is cleared when the TX FIFO is no longer empty.	0x1

Table 112: UART2_USR_REG (0x5000117C)

Bit	Mode	Symbol	Description	Reset
1	r	UART_TFNF	Transmit FIFO Not Full. This is used to indicate that the transmit FIFO in not full. 0 = Transmit FIFO is full 1 = Transmit FIFO is not full This bit is cleared when the TX FIFO is full.	0x1
0	-	-	Reserved	0x0

Table 113: UART2_TFL_REG (0x50001180)

Bit	Mode	Symbol	Description	Reset
15:0	r	UART_TRANSMIT_FIFO_LEVEL	Transmit FIFO Level. This is indicates the number of data entries in the transmit FIFO.	0x0

Table 114: UART2_RFL_REG (0x50001184)

Bit	Mode	Symbol	Description	Reset
15:0	r	UART_RECEIVE_FIFO_LEVEL	Receive FIFO Level. This is indicates the number of data entries in the receive FIFO.	0x0

Table 115: UART2_SRR_REG (0x50001188)

Bit	Mode	Symbol	Description	Reset
15:3	-	-	Reserved	0x0
2	w	UART_XFR	XMIT FIFO Reset. This is a shadow register for the XMIT FIFO Reset bit (FCR[2]). This can be used to remove the burden on software having to store previously written FCR values (which are pretty static) just to reset the transmit FIFO. This resets the control portion of the transmit FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing'. It is not necessary to clear this bit.	0x0
1	w	UART_RFR	RCVR FIFO Reset. This is a shadow register for the RCVR FIFO Reset bit (FCR[1]). This can be used to remove the burden on software having to store previously written FCR values (which are pretty static) just to reset the receive FIFO This resets the control portion of the receive FIFO and treats the FIFO as empty. Note that this bit is 'self-clearing'. It is not necessary to clear this bit.	0x0
0	w	UART_UR	UART Reset. This asynchronously resets the UART Ctrl and synchronously removes the reset assertion. For a two clock implementation both pclk and sclk domains are reset.	0x0

Table 116: UART2_SRTS_REG (0x5000118C)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 116: UART2_SRTS_REG (0x5000118C)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_SHADOW_REQUEST_TO_SEND	Shadow Request to Send. This is a shadow register for the RTS bit (MCR[1]), this can be used to remove the burden of having to perform a read-modify-write on the MCR. This is used to directly control the Request to Send (rts_n) output. The Request To Send (rts_n) output is used to inform the modem or data set that the UART Ctrl is ready to exchange data. When Auto Flow Control is disabled (MCR[5] = 0), the rts_n signal is set low by programming MCR[1] (RTS) to a high. When Auto Flow Control is enabled (MCR[5] = 1) and FIFOs are enabled (FCR[0] = 1), the rts_n output is controlled in the same way, but is also gated with the receiver FIFO threshold trigger (rts_n is inactive high when above the threshold). Note that in Loopback mode (MCR[4] = 1), the rts_n output is held inactive-high while the value of this location is internally looped back to an input.	0x0

Table 117: UART2_SBCR_REG (0x50001190)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	UART_SHADOW_BREAK_CONTROL	Shadow Break Control Bit. This is a shadow register for the Break bit (LCR[6]), this can be used to remove the burden of having to performing a read modify write on the LCR. This is used to cause a break condition to be transmitted to the receiving device. If set to one the serial output is forced to the spacing (logic 0) state. When not in Loopback Mode, as determined by MCR[4], the sout line is forced low until the Break bit is cleared. If SIR_MODE active (MCR[6] = 1) the sir_out_n line is continuously pulsed. When in Loopback Mode, the break condition is internally looped back to the receiver.	0x0

Table 118: UART2_SDMAM_REG (0x50001194)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	UART_SHADOW_DMA_MODE	Shadow DMA Mode. This is a shadow register for the DMA mode bit (FCR[3]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the DMA Mode bit gets updated. This determines the DMA signalling mode used for the dma_tx_req_n and dma_rx_req_n output signals. 0 = mode 0 1 = mode 1	0x0

Table 119: UART2_SFE_REG (0x50001198)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 119: UART2_SFE_REG (0x50001198)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_SHADOW_FIFO_ENABLE	Shadow FIFO Enable. This is a shadow register for the FIFO enable bit (FCR[0]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the FIFO enable bit gets updated.This enables/disables the transmit (XMIT) and receive (RCVR) FIFOs. If this bit is set to zero (disabled) after being enabled then both the XMIT and RCVR controller portion of FIFOs are reset.	0x0

Table 120: UART2_SRT_REG (0x5000119C)

Bit	Mode	Symbol	Description	Reset
15:2	-	-	Reserved	0x0
1:0	r/w	UART_SHADOW_RCVR_TRIGGER	Shadow RCVR Trigger. This is a shadow register for the RCVR trigger bits (FCR[7:6]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the RCVR trigger bit gets updated. This is used to select the trigger level in the receiver FIFO at which the Received Data Available Interrupt is generated. It also determines when the dma_rx_req_n signal is asserted when DMA Mode (FCR[3]) = 1. The following trigger levels are supported: 00 = 1 character in the FIFO 01 = FIFO Â¼ full 10 = FIFO Â½ full 11 = FIFO 2 less than full	0x0

Table 121: UART2_STET_REG (0x500011A0)

Bit	Mode	Symbol	Description	Reset
15:2	-	-	Reserved	0x0
1:0	r/w	UART_SHADOW_TX_EMPTY_TRIGGER	Shadow TX Empty Trigger. This is a shadow register for the TX empty trigger bits (FCR[5:4]). This can be used to remove the burden of having to store the previously written value to the FCR in memory and having to mask this value so that only the TX empty trigger bit gets updated. This is used to select the empty threshold level at which the THRE Interrupts are generated when the mode is active. The following trigger levels are supported: 00 = FIFO empty 01 = 2 characters in the FIFO 10 = FIFO Â¼ full 11 = FIFO Â½ full	0x0

Table 122: UART2_HTX_REG (0x500011A4)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 122: UART2_HTX_REG (0x500011A4)

Bit	Mode	Symbol	Description	Reset
0	r/w	UART_HALT_TX	This register is use to halt transmissions for testing, so that the transmit FIFO can be filled by the master when FIFOs are implemented and enabled. 0 = Halt TX disabled 1 = Halt TX enabled Note, if FIFOs are implemented and not enabled, the setting of the halt TX register has no effect on operation.	0x0

Table 123: UART2_CPR_REG (0x500011F4)

Bit	Mode	Symbol	Description	Reset
15:0	r	CPR	Component Parameter Register	0x0

Table 124: UART2_UCV_REG (0x500011F8)

Bit	Mode	Symbol	Description	Reset
15:0	r	UCV	Component Version	0x3330382A

Table 125: UART2_CTR_REG (0x500011FC)

Bit	Mode	Symbol	Description	Reset
15:0	r	CTR	Component Type Register	0x44570110

Table 126: SPI_CTRL_REG (0x50001200)

Bit	Mode	Symbol	Description	Reset
15	r/w	SPI_EN_CTRL	0 = SPI_EN pin disabled in slave mode. Pin SPI_EN is don't care. 1 = SPI_EN pin enabled in slave mode.	0x0
14	r/w	SPI_MINT	0 = Disable SPI_INT_BIT to the Interrupt Controller 1 = Enable SPI_INT_BIT to the Interrupt Controller	0x0
13	r	SPI_INT_BIT	0 = RX Register or FIFO is empty. 1 = SPI interrupt. Data has been transmitted and received-Must be reset by SW by writing to SPI_CLEAR_INT_REG.	0x0
12	r	SPI_DI	Returns the actual value of pin SPI_DIN (delayed with two internal SPI clock cycles)	0x0
11	r	SPI_TXH	0 = TX-FIFO is not full, data can be written. 1 = TX-FIFO is full, data can not be written.	0x0
10	r/w	SPI_FORCE_DO	0 = normal operation 1 = Force SPIDO output level to value of SPI_DO.	0x0
9	r/w	SPI_RST	0 = normal operation 1 = Reset SPI. Same function as SPI_ON except that internal clock remain active.	0x0
8:7	r/w	SPI_WORD	00 = 8 bits mode, only SPI_RX_TX_REG0 used 01 = 16 bit mode, only SPI_RX_TX_REG0 used 10 = 32 bits mode, SPI_RX_TX_REG0 & SPI_RX_TX_REG1 used 11 = 9 bits mode. Only valid in master mode.	0x0

Table 126: SPI_CTRL_REG (0x50001200)

Bit	Mode	Symbol	Description	Reset
6	r/w	SPI_SMN	Master/slave mode 0 = Master, 1 = Slave(SPI1 only)	0x0
5	r/w	SPI_DO	Pin SPI_DO output level when SPI is idle or when SPI_FORCE_DO=1	0x0
4:3	r/w	SPI_CLK	Select SPI_CLK clock frequency in master mode:00 = (XTAL) / (CLK_PER_REG *8) 01 = (XTAL) / (CLK_PER_REG *4) 10 = (XTAL) / (CLK_PER_REG *2) 11 = (XTAL) / (CLK_PER_REG *14)	0x0
2	r/w	SPI_POL	Select SPI_CLK polarity. 0 = SPI_CLK is initially low. 1 = SPI_CLK is initially high.	0x0
1	r/w	SPI_PHA	Select SPI_CLK phase. See functional timing diagrams in SPI chapter	0x0
0	r/w	SPI_ON	0 = SPI Module switched off (power saving). Everything is reset except SPI_CTRL_REG0 and SPI_CTRL_REG1. When this bit is cleared the SPI will remain active in master mode until the shift register and holding register are both empty. 1 = SPI Module switched on. Should only be set after all control bits have their desired values. So two writes are needed!	0x0

Table 127: SPI_RX_TX_REG0 (0x50001202)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	SPI_DATA0	Write: SPI_TX_REG0 output register 0 (TX-FIFO) Read: SPI_RX_REG0 input register 0 (RX-FIFO) In 8 or 9 bits mode bits 15 to 8 are not used, they contain old data.	0x0

Table 128: SPI_RX_TX_REG1 (0x50001204)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	SPI_DATA1	Write: SPI_TX_REG1 output register 1 (MSB's of TX-FIFO) Read: SPI_RX_REG1 input register 1 (MSB's of RX-FIFO) In 8 or 9 or 16 bits mode bits this register is not used.	0x0

Table 129: SPI_CLEAR_INT_REG (0x50001206)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	SPI_CLEAR_INT	Writing any value to this register will clear the SPI_CTRL_REG[SPI_INT_BIT] Reading returns 0.	0x0

Table 130: SPI_CTRL_REG1 (0x50001208)

Bit	Mode	Symbol	Description	Reset
15:5	-	-	Reserved	0x0
4	r/w	SPI_9BIT_VAL	Determines the value of the first bit in 9 bits SPI mode.	0x0

Table 130: SPI_CTRL_REG1 (0x50001208)

Bit	Mode	Symbol	Description	Reset
3	r	SPI_BUSY	0 = The SPI is not busy with a transfer. This means that either no TX-data is available or that the transfers have been suspended due to a full RX-FIFO. The SPIx_CTRL_REG0[SPI_INT_BIT] can be used to distinguish between these situations. 1 = The SPI is busy with a transfer.	0x0
2	r/w	SPI_PRIORITY	0 = The SPI has low priority, the DMA request signals are reset after the corresponding acknowledge. 1 = The SPI has high priority, DMA request signals remain active until the FIFOS are filled/emptied, so the DMA holds the AHB bus.	0x0
1:0	r/w	SPI_FIFO_MODE	0: TX-FIFO and RX-FIFO used (Bidirectional mode). 1: RX-FIFO used (Read Only Mode) TX-FIFO single depth, no flow control 2: TX-FIFO used (Write Only Mode), RX-FIFO single depth, no flow control 3: No FIFOs used (backwards compatible mode)	0x3

Table 131: I2C_CON_REG (0x50001300)

Bit	Mode	Symbol	Description	Reset
15:7	-	-	Reserved	0x0
6	r/w	I2C_SLAVE_DISABLE	Slave enabled or disabled after reset is applied, which means software does not have to configure the slave. 0=slave is enabled 1=slave is disabled Software should ensure that if this bit is written with '0', then bit 0 should also be written with a '0'.	0x1
5	r/w	I2C_RESTART_EN	Determines whether RESTART conditions may be sent when acting as a master 0= disable 1=enable	0x1
4	r/w	I2C_10BITADDR_MASTER	Controls whether the controller starts its transfers in 7- or 10-bit addressing mode when acting as a master. 0= 7-bit addressing 1= 10-bit addressing	0x1
3	r/w	I2C_10BITADDR_SLAVE	When acting as a slave, this bit controls whether the controller responds to 7- or 10-bit addresses. 0= 7-bit addressing 1= 10-bit addressing	0x1
2:1	r/w	I2C_SPEED	These bits control at which speed the controller operates. 1= standard mode (100 kbit/s) 2= fast mode (400 kbit/s)	0x2
0	r/w	I2C_MASTER_MODE	This bit controls whether the controller master is enabled. 0= master disabled 1= master enabled Software should ensure that if this bit is written with '1' then bit 6 should also be written with a '1'.	0x1

Table 132: I2C_TAR_REG (0x50001304)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0x0

Table 132: I2C_TAR_REG (0x50001304)

Bit	Mode	Symbol	Description	Reset
11	r/w	SPECIAL	This bit indicates whether software performs a General Call or START BYTE command. 0: ignore bit 10 GC_OR_START and use IC_TAR normally 1: perform special I2C command as specified in GC_OR_START bit	0x0
10	r/w	GC_OR_START	If bit 11 (SPECIAL) is set to 1, then this bit indicates whether a General Call or START byte command is to be performed by the controller. 0: General Call Address - after issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TX_ABRT) of the IC_RAW_INTR_STAT register. The controller remains in General Call mode until the SPECIAL bit value (bit 11) is cleared. 1: START BYTE	0x0
9:0	r/w	IC_TAR	This is the target address for any master transaction. When transmitting a General Call, these bits are ignored. To generate a START BYTE, the CPU needs to write only once into these bits. Note: If the IC_TAR and IC_SAR are the same, loopback exists but the FIFOs are shared between master and slave, so full loopback is not feasible. Only one direction loopback mode is supported (simplex), not duplex. A master cannot transmit to itself; it can transmit to only a slave	0x55

Table 133: I2C_SAR_REG (0x50001308)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	IC_SAR	The IC_SAR holds the slave address when the I2C is operating as a slave. For 7-bit addressing, only IC_SAR[6:0] is used. This register can be written only when the I2C interface is disabled, which corresponds to the IC_ENABLE register being set to 0. Writes at other times have no effect.	0x55

Table 134: I2C_DATA_CMD_REG (0x50001310)

Bit	Mode	Symbol	Description	Reset
15:9	-	-	Reserved	0x0

Table 134: I2C_DATA_CMD_REG (0x50001310)

Bit	Mode	Symbol	Description	Reset
8	r/w	CMD	This bit controls whether a read or a write is performed. This bit does not control the direction when the I2C Ctrl acts as a slave. It controls only the direction when it acts as a master. 1 = Read 0 = Write When a command is entered in the TX FIFO, this bit distinguishes the write and read commands. In slave-receiver mode, this bit is a "don't care" because writes to this register are not required. In slave-transmitter mode, a "0" indicates that CPU data is to be transmitted and as DAT or IC_DATA_CMD[7:0]. When programming this bit, you should remember the following: attempting to perform a read operation after a General Call command has been sent results in a TX_ABRT interrupt (bit 6 of the I2C_RAW_INTR_STAT_REG), unless bit 11 (SPECIAL) in the I2C_TAR register has been cleared. If a "1" is written to this bit after receiving a RD_REQ interrupt, then a TX_ABRT interrupt occurs. NOTE: It is possible that while attempting a master I2C read transfer on the controller, a RD_REQ interrupt may have occurred simultaneously due to a remote I2C master addressing the controller. In this type of scenario, it ignores the I2C_DATA_CMD write, generates a TX_ABRT interrupt, and waits to service the RD_REQ interrupt	0x0
7:0	r/w	DAT	This register contains the data to be transmitted or received on the I2C bus. If you are writing to this register and want to perform a read, bits 7:0 (DAT) are ignored by the controller. However, when you read this register, these bits return the value of data received on the controller's interface.	0x0

Table 135: I2C_SS_SCL_HCNT_REG (0x50001314)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IC_SS_SCL_HCNT	This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for standard speed. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set. NOTE: This register must not be programmed to a value higher than 65525, because the controller uses a 16-bit counter to flag an I2C bus idle condition when this counter reaches a value of IC_SS_SCL_HCNT + 10.	0x48

Table 136: I2C_SS_SCL_LCNT_REG (0x50001318)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IC_SS_SCL_LCNT	This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low period count for standard speed. This register can be written only when the I2C interface is disabled which corresponds to the I2C_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted, results in 8 being set.	0x4F

Table 137: I2C_FS_SCL_HCNT_REG (0x5000131C)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IC_FS_SCL_HCNT	This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock high-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 6; hardware prevents values less than this being written, and if attempted results in 6 being set.	0x8

Table 138: I2C_FS_SCL_LCNT_REG (0x50001320)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IC_FS_SCL_LCNT	This register must be set before any I2C bus transaction can take place to ensure proper I/O timing. This register sets the SCL clock low-period count for fast speed. It is used in high-speed mode to send the Master Code and START BYTE or General CALL. This register can be written only when the I2C interface is disabled, which corresponds to the I2C_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 8; hardware prevents values less than this being written, and if attempted results in 8 being set. For designs with APB_DATA_WIDTH = 8 the order of programming is important to ensure the correct operation of the controller. The lower byte must be programmed first. Then the upper byte is programmed.	0x17

Table 139: I2C_INTR_STAT_REG (0x5000132C)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0x0
11	r	R_GEN_CALL	Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. The controller stores the received data in the Rx buffer.	0x0

Table 139: I2C_INTR_STAT_REG (0x5000132C)

Bit	Mode	Symbol	Description	Reset
10	r	R_START_DET	Indicates whether a START or RESTART condition has occurred on the I2C interface regardless of whether controller is operating in slave or master mode.	0x0
9	r	R_STOP_DET	Indicates whether a STOP condition has occurred on the I2C interface regardless of whether controller is operating in slave or master mode.	0x0
8	r	R_ACTIVITY	This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: => Disabling the I2C Ctrl => Reading the IC_CLR_ACTIVITY register => Reading the IC_CLR_INTR register => System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared, indicating that there was activity on the bus.	0x0
7	r	R_RX_DONE	When the controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	r	R_TX_ABRT	This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave, and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	r	R_RD_REQ	This bit is set to 1 when the controller is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register	0x0
4	r	R_TX_EMPTY	This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with ic_en=0, this bit is set to 0.	0x0
3	r	R_TX_OVER	Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared	0x0

Table 139: I2C_INTR_STAT_REG (0x5000132C)

Bit	Mode	Symbol	Description	Reset
2	r	R_RX_FULL	Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the I2C_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.	0x0
1	r	R_RX_OVER	Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	r	R_RX_UNDER	Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 140: I2C_INTR_MASK_REG (0x50001330)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0x0
11	r/w	M_GEN_CALL	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
10	r/w	M_START_DET	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
9	r/w	M_STOP_DET	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
8	r/w	M_ACTIVITY	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x0
7	r/w	M_RX_DONE	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
6	r/w	M_TX_ABRT	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
5	r/w	M_RD_REQ	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
4	r/w	M_TX_EMPTY	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
3	r/w	M_TX_OVER	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
2	r/w	M_RX_FULL	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
1	r/w	M_RX_OVER	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1
0	r/w	M_RX_UNDER	These bits mask their corresponding interrupt status bits in the I2C_INTR_STAT register.	0x1

Table 141: I2C_RAW_INTR_STAT_REG (0x50001334)

Bit	Mode	Symbol	Description	Reset
15:12	-	-	Reserved	0x0
11	r	GEN_CALL	Set only when a General Call address is received and it is acknowledged. It stays set until it is cleared either by disabling controller or when the CPU reads bit 0 of the I2C_CLR_GEN_CALL register. I2C Ctrl stores the received data in the Rx buffer.	0x0
10	r	START_DET	Indicates whether a START or RESTART condition has occurred on the I2C interface regardless of whether controller is operating in slave or master mode.	0x0
9	r	STOP_DET	Indicates whether a STOP condition has occurred on the I2C interface regardless of whether controller is operating in slave or master mode.	0x0
8	r	ACTIVITY	This bit captures I2C Ctrl activity and stays set until it is cleared. There are four ways to clear it: => Disabling the I2C Ctrl => Reading the IC_CLR_ACTIVITY register => Reading the IC_CLR_INTR register => System reset Once this bit is set, it stays set unless one of the four methods is used to clear it. Even if the controller module is idle, this bit remains set until cleared, indicating that there was activity on the bus.	0x0
7	r	RX_DONE	When the controller is acting as a slave-transmitter, this bit is set to 1 if the master does not acknowledge a transmitted byte. This occurs on the last byte of the transmission, indicating that the transmission is done.	0x0
6	r	TX_ABRT	This bit indicates if the controller, as an I2C transmitter, is unable to complete the intended actions on the contents of the transmit FIFO. This situation can occur both as an I2C master or an I2C slave, and is referred to as a "transmit abort". When this bit is set to 1, the I2C_TX_ABRT_SOURCE register indicates the reason why the transmit abort takes places. NOTE: The controller flushes/resets/empties the TX FIFO whenever this bit is set. The TX FIFO remains in this flushed state until the register I2C_CLR_TX_ABRT is read. Once this read is performed, the TX FIFO is then ready to accept more data bytes from the APB interface.	0x0
5	r	RD_REQ	This bit is set to 1 when I2C Ctrl is acting as a slave and another I2C master is attempting to read data from the controller. The controller holds the I2C bus in a wait state (SCL=0) until this interrupt is serviced, which means that the slave has been addressed by a remote master that is asking for data to be transferred. The processor must respond to this interrupt and then write the requested data to the I2C_DATA_CMD register. This bit is set to 0 just after the processor reads the I2C_CLR_RD_REQ register	0x0
4	r	TX_EMPTY	This bit is set to 1 when the transmit buffer is at or below the threshold value set in the I2C_TX_TL register. It is automatically cleared by hardware when the buffer level goes above the threshold. When the IC_ENABLE bit 0 is 0, the TX FIFO is flushed and held in reset. There the TX FIFO looks like it has no data within it, so this bit is set to 1, provided there is activity in the master or slave state machines. When there is no longer activity, then with ic_en=0, this bit is set to 0.	0x0

Table 141: I2C_RAW_INTR_STAT_REG (0x50001334)

Bit	Mode	Symbol	Description	Reset
3	r	TX_OVER	Set during transmit if the transmit buffer is filled to 32 and the processor attempts to issue another I2C command by writing to the IC_DATA_CMD register. When the module is disabled, this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared	0x0
2	r	RX_FULL	Set when the receive buffer reaches or goes above the RX_TL threshold in the I2C_RX_TL register. It is automatically cleared by hardware when buffer level goes below the threshold. If the module is disabled (I2C_ENABLE[0]=0), the RX FIFO is flushed and held in reset; therefore the RX FIFO is not full. So this bit is cleared once the I2C_ENABLE bit 0 is programmed with a 0, regardless of the activity that continues.	0x0
1	r	RX_OVER	Set if the receive buffer is completely filled to 32 and an additional byte is received from an external I2C device. The controller acknowledges this, but any data bytes received after the FIFO is full are lost. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0
0	r	RX_UNDER	Set if the processor attempts to read the receive buffer when it is empty by reading from the IC_DATA_CMD register. If the module is disabled (I2C_ENABLE[0]=0), this bit keeps its level until the master or slave state machines go into idle, and when ic_en goes to 0, this interrupt is cleared.	0x0

Table 142: I2C_RX_TL_REG (0x50001338)

Bit	Mode	Symbol	Description	Reset
15:5	-	-	Reserved	0x0
4:0	r/w	RX_TL	Receive FIFO Threshold Level Controls the level of entries (or above) that triggers the RX_FULL interrupt (bit 2 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that hardware does not allow this value to be set to a value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 1 entry, and a value of 31 sets the threshold for 32 entries.	0x0

Table 143: I2C_TX_TL_REG (0x5000133C)

Bit	Mode	Symbol	Description	Reset
15:5	-	-	Reserved	0x0
4:0	r/w	RX_TL	Transmit FIFO Threshold Level Controls the level of entries (or below) that trigger the TX_EMPTY interrupt (bit 4 in I2C_RAW_INTR_STAT register). The valid range is 0-31, with the additional restriction that it may not be set to value larger than the depth of the buffer. If an attempt is made to do that, the actual value set will be the maximum depth of the buffer. A value of 0 sets the threshold for 0 entries, and a value of 31 sets the threshold for 32 entries..	0x0

Table 144: I2C_CLR_INTR_REG (0x50001340)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_INTR	Read this register to clear the combined interrupt, all individual interrupts, and the I2C_TX_ABRT_SOURCE register. This bit does not clear hardware clearable interrupts but software clearable interrupts. Refer to Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing I2C_TX_ABRT_SOURCE	0x0

Table 145: I2C_CLR_RX_UNDER_REG (0x50001344)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_RX_UNDER	Read this register to clear the RX_UNDER interrupt (bit 0) of the I2C_RAW_INTR_STAT register.	0x0

Table 146: I2C_CLR_RX_OVER_REG (0x50001348)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_RX_OVER	Read this register to clear the RX_OVER interrupt (bit 1) of the I2C_RAW_INTR_STAT register.	0x0

Table 147: I2C_CLR_TX_OVER_REG (0x5000134C)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_TX_OVER	Read this register to clear the TX_OVER interrupt (bit 3) of the I2C_RAW_INTR_STAT register.	0x0

Table 148: I2C_CLR_RD_REQ_REG (0x50001350)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_RD_REQ	Read this register to clear the RD_REQ interrupt (bit 5) of the I2C_RAW_INTR_STAT register.	0x0

Table 149: I2C_CLR_TX_ABRT_REG (0x50001354)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_TX_ABRT	Read this register to clear the TX_ABRT interrupt (bit 6) of the IC_RAW_INTR_STAT register, and the I2C_TX_ABRT_SOURCE register. This also releases the TX FIFO from the flushed/reset state, allowing more writes to the TX FIFO. Refer to Bit 9 of the I2C_TX_ABRT_SOURCE register for an exception to clearing IC_TX_ABRT_SOURCE.	0x0

Table 150: I2C_CLR_RX_DONE_REG (0x50001358)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_RX_DONE	Read this register to clear the RX_DONE interrupt (bit 7) of the I2C_RAW_INTR_STAT register.	0x0

Table 151: I2C_CLR_ACTIVITY_REG (0x5000135C)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_ACTIVITY	Reading this register clears the ACTIVITY interrupt if the I2C is not active anymore. If the I2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT register	0x0

Table 152: I2C_CLR_STOP_DET_REG (0x50001360)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_ACTIVITY	Reading this register clears the ACTIVITY interrupt if the I2C is not active anymore. If the I2C module is still active on the bus, the ACTIVITY interrupt bit continues to be set. It is automatically cleared by hardware if the module is disabled and if there is no further activity on the bus. The value read from this register to get status of the ACTIVITY interrupt (bit 8) of the IC_RAW_INTR_STAT register.	0x0

Table 153: I2C_CLR_START_DET_REG (0x50001364)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_START_DET	Read this register to clear the START_DET interrupt (bit 10) of the IC_RAW_INTR_STAT register.	0x0

Table 154: I2C_CLR_GEN_CALL_REG (0x50001368)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r	CLR_GEN_CALL	Read this register to clear the GEN_CALL interrupt (bit 11) of I2C_RAW_INTR_STAT register.	0x0

Table 155: I2C_ENABLE_REG (0x5000136C)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0

Table 155: I2C_ENABLE_REG (0x5000136C)

Bit	Mode	Symbol	Description	Reset
0	r/w	CTRL_ENABLE	Controls whether the controller is enabled. 0: Disables the controller (TX and RX FIFOs are held in an erased state) 1: Enables the controller Software can disable the controller while it is active. However, it is important that care be taken to ensure that the controller is disabled properly. When the controller is disabled, the following occurs: * The TX FIFO and RX FIFO get flushed. * Status bits in the IC_INTR_STAT register are still active until the controller goes into IDLE state. If the module is transmitting, it stops as well as deletes the contents of the transmit buffer after the current transfer is complete. If the module is receiving, the controller stops the current transfer at the end of the current byte and does not acknowledge the transfer. There is a two ic_clk delay when enabling or disabling the controller	0x0

Table 156: I2C_STATUS_REG (0x50001370)

Bit	Mode	Symbol	Description	Reset
15:7	-	-	Reserved	0x0
6	r	SLV_ACTIVITY	Slave FSM Activity Status. When the Slave Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0: Slave FSM is in IDLE state so the Slave part of the controller is not Active 1: Slave FSM is not in IDLE state so the Slave part of the controller is Active	0x0
5	r	MST_ACTIVITY	Master FSM Activity Status. When the Master Finite State Machine (FSM) is not in the IDLE state, this bit is set. 0: Master FSM is in IDLE state so the Master part of the controller is not Active 1: Master FSM is not in IDLE state so the Master part of the controller is Active	0x0
4	r	RFF	Receive FIFO Completely Full. When the receive FIFO is completely full, this bit is set. When the receive FIFO contains one or more empty location, this bit is cleared. 0: Receive FIFO is not full 1: Receive FIFO is full	0x0
3	r	RFNE	Receive FIFO Not Empty. This bit is set when the receive FIFO contains one or more entries; it is cleared when the receive FIFO is empty. 0: Receive FIFO is empty 1: Receive FIFO is not empty	0x0
2	r	TFE	Transmit FIFO Completely Empty. When the transmit FIFO is completely empty, this bit is set. When it contains one or more valid entries, this bit is cleared. This bit field does not request an interrupt. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty	0x1
1	r	TFNF	Transmit FIFO Not Full. Set when the transmit FIFO contains one or more empty locations, and is cleared when the FIFO is full. 0: Transmit FIFO is full 1: Transmit FIFO is not full	0x1

Table 156: I2C_STATUS_REG (0x50001370)

Bit	Mode	Symbol	Description	Reset
0	r	I2C_ACTIVITY	I2C Activity Status.	0x0

Table 157: I2C_TXFLR_REG (0x50001374)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r	TXFLR	Transmit FIFO Level. Contains the number of valid data entries in the transmit FIFO. Size is constrained by the TXFLR value	0x0

Table 158: I2C_RXFLR_REG (0x50001378)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r	RXFLR	Receive FIFO Level. Contains the number of valid data entries in the receive FIFO. Size is constrained by the RXFLR value	0x0

Table 159: I2C_SDA_HOLD_REG (0x5000137C)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	IC_SDA_HOLD	SDA Hold time	0x1

Table 160: I2C_TX_ABRT_SOURCE_REG (0x50001380)

Bit	Mode	Symbol	Description	Reset
15	r	ABRT_SLVRD_INTX	1: When the processor side responds to a slave mode request for data to be transmitted to a remote master and user writes a 1 in CMD (bit 8) of 2IC_DATA_CMD register	0x0
14	r	ABRT_SLV_ARBLOST	1: Slave lost the bus while transmitting data to a remote master. I2C_TX_ABRT_SOURCE[12] is set at the same time. Note: Even though the slave never "owns" the bus, something could go wrong on the bus. This is a fail safe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then the controller no longer own the bus.	0x0
13	r	ABRT_SLVFLUSH_TXFIFO	1: Slave has received a read command and some data exists in the TX FIFO so the slave issues a TX_ABRT interrupt to flush old data in TX FIFO.	0x0
12	r	ARB_LOST	1: Master has lost arbitration, or if I2C_TX_ABRT_SOURCE[14] is also set, then the slave transmitter has lost arbitration. Note: I2C can be both master and slave at the same time.	0x0
11	r	ABRT_MASTER_DIS	1: User tries to initiate a Master operation with the Master mode disabled.	0x0
10	r	ABRT_10B_RD_NORSTRT	1: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the master sends a read command in 10-bit addressing mode.	0x0

Table 160: I2C_TX_ABRT_SOURCE_REG (0x50001380)

Bit	Mode	Symbol	Description	Reset
9	r	ABRT_SBYTE_NORSTRT	To clear Bit 9, the source of the ABRT_SBYTE_NORSTRT must be fixed first; restart must be enabled (I2C_CON[5]=1), the SPECIAL bit must be cleared (I2C_TAR[11]), or the GC_OR_START bit must be cleared (I2C_TAR[10]). Once the source of the ABRT_SBYTE_NORSTRT is fixed, then this bit can be cleared in the same manner as other bits in this register. If the source of the ABRT_SBYTE_NORSTRT is not fixed before attempting to clear this bit, bit 9 clears for one cycle and then gets re-asserted. 1: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to send a START Byte.	0x0
8	r	ABRT_HS_NORSTRT	1: The restart is disabled (IC_RESTART_EN bit (I2C_CON[5]) = 0) and the user is trying to use the master to transfer data in High Speed mode	0x0
7	r	ABRT_SBYTE_ACKDET	1: Master has sent a START Byte and the START Byte was acknowledged (wrong behavior).	0x0
6	r	ABRT_HS_ACKDET	1: Master is in High Speed mode and the High Speed Master code was acknowledged (wrong behavior).	0x0
5	r	ABRT_GCALL_READ	1: the controller in master mode sent a General Call but the user programmed the byte following the General Call to be a read from the bus (IC_DATA_CMD[9] is set to 1).	0x0
4	r	ABRT_GCALL_NOACK	1: the controller in master mode sent a General Call and no slave on the bus acknowledged the General Call.	0x0
3	r	ABRT_TXDATA_NOACK	1: This is a master-mode only bit. Master has received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s).	0x0
2	r	ABRT_10ADDR2_NOACK	1: Master is in 10-bit address mode and the second address byte of the 10-bit address was not acknowledged by any slave.	0x0
1	r	ABRT_10ADDR1_NOACK	1: Master is in 10-bit address mode and the first 10-bit address byte was not acknowledged by any slave.	0x0
0	r	ABRT_7B_ADDR_NOACK	1: Master is in 7-bit addressing mode and the address sent was not acknowledged by any slave.	0x0

Table 161: I2C_SDA_SETUP_REG (0x50001394)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	SDA_SETUP	SDA Setup. This register controls the amount of time delay (number of I2C clock periods) between the rising edge of SCL and SDA changing by holding SCL low when I2C block services a read request while operating as a slave-transmitter. The relevant I2C requirement is tSU:DAT (note 4) as detailed in the I2C Bus Specification. This register must be programmed with a value equal to or greater than 2. It is recommended that if the required delay is 1000ns, then for an I2C frequency of 10 MHz, IC_SDA_SETUP should be programmed to a value of 11.Writes to this register succeed only when IC_ENABLE[0] = 0.	0x64

Table 162: I2C_ACK_GENERAL_CALL_REG (0x50001398)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	ACK_GEN_CALL	ACK General Call. When set to 1, I2C Ctrl responds with a ACK (by asserting ic_data_oe) when it receives a General Call. When set to 0, the controller does not generate General Call interrupts.	0x0

Table 163: I2C_ENABLE_STATUS_REG (0x5000139C)

Bit	Mode	Symbol	Description	Reset
15:3	-	-	Reserved	0x0
2	r	SLV_RX_DATA_LOST	Slave Received Data Lost. This bit indicates if a Slave-Receiver operation has been aborted with at least one data byte received from an I2C transfer due to the setting of IC_ENABLE from 1 to 0. When read as 1, the controller is deemed to have been actively engaged in an aborted I2C transfer (with matching address) and the data phase of the I2C transfer has been entered, even though a data byte has been responded with a NACK. NOTE: If the remote I2C master terminates the transfer with a STOP condition before the controller has a chance to NACK a transfer, and IC_ENABLE has been set to 0, then this bit is also set to 1. When read as 0, the controller is deemed to have been disabled without being actively involved in the data phase of a Slave-Receiver transfer. NOTE: The CPU can safely read this bit when IC_EN (bit 0) is read as 0.	0x0
1	r	SLV_DISABLED_WHILE_BUSY	Slave Disabled While Busy (Transmit, Receive). This bit indicates if a potential or active Slave operation has been aborted due to the setting of the IC_ENABLE register from 1 to 0. This bit is set when the CPU writes a 0 to the IC_ENABLE register while: (a) I2C Ctrl is receiving the address byte of the Slave-Transmitter operation from a remote master; OR, (b) address and data bytes of the Slave-Receiver operation from a remote master. When read as 1, the controller is deemed to have forced a NACK during any part of an I2C transfer, irrespective of whether the I2C address matches the slave address set in I2C Ctrl (IC_SAR register) OR if the transfer is completed before IC_ENABLE is set to 0 but has not taken effect. NOTE: If the remote I2C master terminates the transfer with a STOP condition before the the controller has a chance to NACK a transfer, and IC_ENABLE has been set to 0, then this bit will also be set to 1. When read as 0, the controller is deemed to have been disabled when there is master activity, or when the I2C bus is idle. NOTE: The CPU can safely read this bit when IC_EN (bit 0) is read as 0.	0x0

Table 163: I2C_ENABLE_STATUS_REG (0x5000139C)

Bit	Mode	Symbol	Description	Reset
0	r	IC_EN	ic_en Status. This bit always reflects the value driven on the output port ic_en. When read as 1, the controller is deemed to be in an enabled state. When read as 0, the controller is deemed completely inactive. NOTE: The CPU can safely read this bit anytime. When this bit is read as 0, the CPU can safely read SLV_RX_DATA_LOST (bit 2) and SLV_DISABLED_WHILE_BUSY (bit 1).	0x0

Table 164: I2C_IC_FS_SPKLEN_REG (0x500013A0)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	IC_FS_SPKLEN	This register must be set before any I2C bus transaction can take place to ensure stable operation. This register sets the duration, measured in ic_clk cycles, of the longest spike in the SCL or SDA lines that will be filtered out by the spike suppression logic. This register can be written only when the I2C interface is disabled which corresponds to the IC_ENABLE register being set to 0. Writes at other times have no effect. The minimum valid value is 2; hardware prevents values less than this being written, and if attempted results in 2 being set.	0x1

Table 165: GPIO_IRQ0_IN_SEL_REG (0x50001400)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0

Table 165: GPIO_IRQ0_IN_SEL_REG (0x50001400)

Bit	Mode	Symbol	Description	Reset
5:0	r/w	KBRD_IRQ0_SEL	input selection that can generate a GPIO interrupt 0: no input selected 1: P0[0] is selected 2: P0[1] is selected 3: P0[2] is selected 4: P0[3] is selected 5: P0[4] is selected 6: P0[5] is selected 7: P0[6] is selected 8: P0[7] is selected 9: P1[0] is selected 10: P1[1] is selected 11: P1[2] is selected 12: P1[3] is selected 13: P1[4] is selected 14: P1[5] is selected 15: P2[0] is selected 16: P2[1] is selected 17: P2[2] is selected 18: P2[3] is selected 19: P2[4] is selected 20: P2[5] is selected 21: P2[6] is selected 22: P2[7] is selected 23: P2[8] is selected 24: P2[9] is selected 25: P3[0] is selected 26: P3[1] is selected 27: P3[2] is selected 28: P3[3] is selected 29: P3[4] is selected 30: P3[5] is selected 31: P3[6] is selected 32: P3[7] is selected all others: no input selected	0x0

Table 166: GPIO_IRQ1_IN_SEL_REG (0x50001402)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r/w	KBRD_IRQ1_SEL	see KBRD_IRQ0_SEL	0x0

Table 167: GPIO_IRQ2_IN_SEL_REG (0x50001404)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r/w	KBRD_IRQ2_SEL	see KBRD_IRQ0_SEL	0x0

Table 168: GPIO_IRQ3_IN_SEL_REG (0x50001406)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r/w	KBRD_IRQ3_SEL	see KBRD_IRQ0_SEL	0x0

Table 169: GPIO_IRQ4_IN_SEL_REG (0x50001408)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5:0	r/w	KBRD_IRQ4_SEL	see KBRD_IRQ0_SEL	0x0

Table 170: GPIO_DEBOUNCE_REG (0x5000140C)

Bit	Mode	Symbol	Description	Reset
15:14	-	-	Reserved	0x0
13	r/w	DEB_ENABLE_KBRD	enables the debounce counter for the KBRD interface	0x0
12	r/w	DEB_ENABLE4	enables the debounce counter for GPIO IRQ4	0x0
11	r/w	DEB_ENABLE3	enables the debounce counter for GPIO IRQ3	0x0
10	r/w	DEB_ENABLE2	enables the debounce counter for GPIO IRQ2	0x0
9	r/w	DEB_ENABLE1	enables the debounce counter for GPIO IRQ1	0x0
8	r/w	DEB_ENABLE0	enables the debounce counter for GPIO IRQ0	0x0
7:6	-	-	Reserved	0x0
5:0	r/w	DEB_VALUE	Keyboard debounce time if enabled. Generate KEYB_INT after specified time. Debounce time: N*1 ms. N =0..63	0x0

Table 171: GPIO_RESET_IRQ_REG (0x5000140E)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0x0
5	r0/w	RESET_KBRD_IRQ	writing a 1 to this bit will reset the KBRD IRQ. Reading returns 0.	0x0
4	r0/w	RESET_GPIO4_IRQ	writing a 1 to this bit will reset the GPIO4 IRQ. Reading returns 0.	0x0
3	r0/w	RESET_GPIO3_IRQ	writing a 1 to this bit will reset the GPIO3 IRQ. Reading returns 0.	0x0
2	r0/w	RESET_GPIO2_IRQ	writing a 1 to this bit will reset the GPIO2 IRQ. Reading returns 0.	0x0
1	r0/w	RESET_GPIO1_IRQ	writing a 1 to this bit will reset the GPIO1 IRQ. Reading returns 0.	0x0
0	r0/w	RESET_GPIO0_IRQ	writing a 1 to this bit will reset the GPIO0 IRQ. Reading returns 0.	0x0

Table 172: GPIO_INT_LEVEL_CTRL_REG (0x50001410)

Bit	Mode	Symbol	Description	Reset
15:14	-	-	Reserved	0x0
12	r/w	EDGE_LEVELN4	see EDGE_LEVELn0, but for GPIO IRQ4	0x0
11	r/w	EDGE_LEVELN3	see EDGE_LEVELn0, but for GPIO IRQ3	0x0
10	r/w	EDGE_LEVELN2	see EDGE_LEVELn0, but for GPIO IRQ2	0x0
9	r/w	EDGE_LEVELN1	see EDGE_LEVELn0, but for GPIO IRQ1	0x0
8	r/w	EDGE_LEVELN0	0: do not wait for key release after interrupt was reset for GPIO IRQ0, so a new interrupt can be initiated immediately 1: wait for key release after interrupt was reset for IRQ0	0x0
7:6	-	-	Reserved	0x0

Table 172: GPIO_INT_LEVEL_CTRL_REG (0x50001410)

Bit	Mode	Symbol	Description	Reset
4	r/w	INPUT_LEVEL4	see INPUT_LEVEL0, but for GPIO IRQ4	0x0
3	r/w	INPUT_LEVEL3	see INPUT_LEVEL0, but for GPIO IRQ3	0x0
2	r/w	INPUT_LEVEL2	see INPUT_LEVEL0, but for GPIO IRQ2	0x0
1	r/w	INPUT_LEVEL1	see INPUT_LEVEL0, but for GPIO IRQ1	0x0
0	r/w	INPUT_LEVEL0	0 = selected input will generate GPIO IRQ0 if that input is high. 1 = selected input will generate GPIO IRQ0 if that input is low.	0x0

Table 173: KBRD_IRQ_IN_SEL0_REG (0x50001412)

Bit	Mode	Symbol	Description	Reset
15	r/w	KBRD_REL	0 = No interrupt on key release 1 = Interrupt also on key release (also debouncing if enabled)	0x0
14	r/w	KBRD_LEVEL	0 = enabled input will generate KBRD IRQ if that input is high. 1 = enabled input will generate KBRD IRQ if that input is low.	0x0
13:8	r/w	KEY_REPEAT	While key is pressed, automatically generate repeating KEYB_INT after specified time unequal to 0. Repeat time: N*1 ms. N =1..63, N=0 disables the timer.	0x0
7	r/w	KBRD_P07_EN	enable P0[7] for the keyboard interrupt	0x0
6	r/w	KBRD_P06_EN	enable P0[6] for the keyboard interrupt	0x0
5	r/w	KBRD_P05_EN	enable P0[5] for the keyboard interrupt	0x0
4	r/w	KBRD_P04_EN	enable P0[4] for the keyboard interrupt	0x0
3	r/w	KBRD_P03_EN	enable P0[3] for the keyboard interrupt	0x0
2	r/w	KBRD_P02_EN	enable P0[2] for the keyboard interrupt	0x0
1	r/w	KBRD_P01_EN	enable P0[1] for the keyboard interrupt	0x0
0	r/w	KBRD_P00_EN	enable P0[0] for the keyboard interrupt	0x0

Table 174: KBRD_IRQ_IN_SEL1_REG (0x50001414)

Bit	Mode	Symbol	Description	Reset
15	r/w	KBRD_P15_EN	enable P1[5] for the keyboard interrupt	0x0
14	r/w	KBRD_P14_EN	enable P1[4] for the keyboard interrupt	0x0
13	r/w	KBRD_P13_EN	enable P1[3] for the keyboard interrupt	0x0
12	r/w	KBRD_P12_EN	enable P1[2] for the keyboard interrupt	0x0
11	r/w	KBRD_P11_EN	enable P1[1] for the keyboard interrupt	0x0
10	r/w	KBRD_P10_EN	enable P1[0] for the keyboard interrupt	0x0
9	r/w	KBRD_P29_EN	enable P2[9] for the keyboard interrupt	0x0
8	r/w	KBRD_P28_EN	enable P2[8] for the keyboard interrupt	0x0
7	r/w	KBRD_P27_EN	enable P2[7] for the keyboard interrupt	0x0
6	r/w	KBRD_P26_EN	enable P2[6] for the keyboard interrupt	0x0
5	r/w	KBRD_P25_EN	enable P2[5] for the keyboard interrupt	0x0
4	r/w	KBRD_P24_EN	enable P2[4] for the keyboard interrupt	0x0
3	r/w	KBRD_P23_EN	enable P2[3] for the keyboard interrupt	0x0
2	r/w	KBRD_P22_EN	enable P2[2] for the keyboard interrupt	0x0

Table 174: KBRD_IRQ_IN_SEL1_REG (0x50001414)

Bit	Mode	Symbol	Description	Reset
1	r/w	KBRD_P21_EN	enable P2[1] for the keyboard interrupt	0x0
0	r/w	KBRD_P20_EN	enable P2[0] for the keyboard interrupt	0x0

Table 175: KBRD_IRQ_IN_SEL2_REG (0x50001416)

Bit	Mode	Symbol	Description	Reset
7	r/w	KBRD_P37_EN	enable P3[7] for the keyboard interrupt	0x0
6	r/w	KBRD_P36_EN	enable P3[6] for the keyboard interrupt	0x0
5	r/w	KBRD_P35_EN	enable P3[5] for the keyboard interrupt	0x0
4	r/w	KBRD_P34_EN	enable P3[4] for the keyboard interrupt	0x0
3	r/w	KBRD_P33_EN	enable P3[3] for the keyboard interrupt	0x0
2	r/w	KBRD_P32_EN	enable P3[2] for the keyboard interrupt	0x0
1	r/w	KBRD_P31_EN	enable P3[1] for the keyboard interrupt	0x0
0	r/w	KBRD_P30_EN	enable P3[0] for the keyboard interrupt	0x0

Table 176: GP_ADC_CTRL_REG (0x50001500)

Bit	Mode	Symbol	Description	Reset
15	r/w	GP_ADC_LDO_ZERO	Forces LDO-output to 0V.	0x0
14	r/w	GP_ADC_LDO_EN	Turns on LDO.	0x0
13	r/w	GP_ADC_CHOP	Takes two samples with opposite GP_ADC_SIGN to cancel the internal offset voltage of the ADC; Highly recommended for DC-measurements.	0x0
12	r/w	GP_ADC_MUTE	Takes sample at mid-scale (to dertermine the internal offset and/or noise of the ADC with regards to VDD_REF which is also sampled by the ADC).	0x0
11	r/w	GP_ADC_SE	0 = Differential mode 1 = Single ended mode	0x0
10	r/w	GP_ADC_SIGN	0 = Default 1 = Conversion with opposite sign at input and output to cancel out the internal offset of the ADC and low-frequency	0x0
9:6	r/w	GP_ADC_SEL	ADC input selection which must be set before the GP_ADC_START bit is enabled. If GP_ADC_SE = 1 (single ended mode): 0000 = P0[0] 0001 = P0[1] 0010 = P0[2] 0011 = P0[3] 0100 = AVS 0101 = VDD_REF 0110 = VDD_RTT 0111 = VBAT3V 1000 = VDCDC 1001 = VBAT1V All other combinations are reserved. If GP_ADC_SE = 0 (differential mode): 0000 = P0[0] vs P0[1] All other combinations are P0[2] vs P0[3].	0x0
5	r/w	GP_ADC_MINT	0 = Disable (mask) GP_ADC_INT. 1 = Enable GP_ADC_INT to ICU.	0x0

Table 176: GP_ADC_CTRL_REG (0x50001500)

Bit	Mode	Symbol	Description	Reset
4	r	GP_ADC_INT	1 = AD conversion ready and has generated an interrupt. Must be cleared by writing any value to GP_ADC_CLEAR_INT_REG.	0x0
3	r/w	GP_ADC_CLK_SEL	0 = Internal high-speed ADC clock used. 1 = Digital clock used.	0x0
2	rsvd	GP_ADC_TEST	Reserved, keep 0.	0x0
1	r/w	GP_ADC_START	0 = ADC conversion ready. 1 = If a 1 is written, the ADC starts a conversion. After the conversion this bit will be set to 0 and the GP_ADC_INT bit will be set.	0x0
0	r/w	GP_ADC_EN	0 = ADC is disabled and in reset. 1 = ADC is enabled and sampling of input is started.	0x0

Table 177: GP_ADC_CTRL2_REG (0x50001502)

Bit	Mode	Symbol	Description	Reset
15:4	-	-	Reserved	0x0
3	r/w	GP_ADC_I20U	Adds 20uA constant load current at the ADC LDO to minimize ripple on the reference voltage of the ADC.	0x0
2	r/w	GP_ADC_IDYN	Enables dynamic load current at the ADC LDO to minimize ripple on the reference voltage of the ADC.	0x0
1	r/w	GP_ADC_ATTN3X	0 = Input voltages up to 1.2V allowed. 1 = Input voltages up to 3.6V allowed by enabling 3x attenuator.	0x0
0	r/w	GP_ADC_DELAY_EN	Enables delay function for several signals. This is not auto-cleared. Toggle this bit before every sampling to enable succesive conversions.	0x0

Table 178: GP_ADC_OFFP_REG (0x50001504)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	GP_ADC_OFFP	Offset adjust of 'positive' array of ADC-network (effective if "GP_ADC_SE=0", or "GP_ADC_SE=1 AND GP_ADC_SIGN=0")	0x200

Table 179: GP_ADC_OFFN_REG (0x50001506)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	GP_ADC_OFFN	Offset adjust of 'negative' array of ADC-network (effective if "GP_ADC_SE=0", or "GP_ADC_SE=1 AND GP_ADC_SIGN=1")	0x200

Table 180: GP_ADC_CLEAR_INT_REG (0x50001508)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	GP_ADC_CLR_INT	Writing any value to this register will clear the ADC_INT interrupt. Reading returns 0.	0x0

Table 181: GP_ADC_RESULT_REG (0x5000150A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r	GP_ADC_VAL	Returns the 10 bits linear value of the last AD conversion.	0x0

Table 182: GP_ADC_DELAY_REG (0x5000150C)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	DEL_LDO_EN	Defines the delay before the LDO enable (GP_ADC_LDO_EN). Reset value is 0 µs since the LDO enable should be the first thing to be programmed in the sequence of bringing the GP ADC up.	0x0

Table 183: GP_ADC_DELAY2_REG (0x5000150E)

Bit	Mode	Symbol	Description	Reset
15:8	r/w	DEL_ADC_START	Defines the delay for the GP_ADC_START bit. Reset value is 17 µs which is the recommended value to wait before starting the GP ADC. This is the third and last step of bringing up the GP ADC	0x88
7:0	r/w	DEL_ADC_EN	Defines the delay for the GP_ADC_EN bit. Reset value is 16 µs which is the recommended value to wait after enabling the LDO. This is the second step in bringing up the GP ADC.	0x80

Table 184: CLK_REF_SEL_REG (0x50001600)

Bit	Mode	Symbol	Description	Reset
15:3	-	-	Reserved	0x0
2	r/w	REF_CAL_START	Writing a '1' starts a calibration. This bit is cleared when calibration is finished, and CLK_REF_VAL is ready.	0x0
1:0	r/w	REF_CLK_SEL	Select clock input for calibration: 0x0 : RC32KHz oscillator 0x1 : RC16MHz oscillator 0x2 : XTAL32KHz oscillator 0x3 : RCX32KHz oscillator	0x0

Table 185: CLK_REF_CNT_REG (0x50001602)

Bit	Mode	Symbol	Description	Reset
15:0	r/w	REF_CNT_VAL	Indicates the calibration time, with a decrement counter to 1.	0x0

Table 186: CLK_REF_VAL_L_REG (0x50001604)

Bit	Mode	Symbol	Description	Reset
15:0	r	XTAL_CNT_VAL	Returns the lower 16 bits of XTAL16 clock cycles during the calibration time, defined with REF_CNT_VAL	0x0

Table 187: CLK_REF_VAL_H_REG (0x50001606)

Bit	Mode	Symbol	Description	Reset
15:0	r	XTAL_CNT_VAL	Returns the upper 16 bits of XTAL16 clock cycles during the calibration time, defined with REF_CNT_VAL	0x0

Table 188: P0_DATA_REG (0x50003000)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P0_DATA	Set P0 output register when written; Returns the value of P0 port when read	0x0

Table 189: P0_SET_DATA_REG (0x50003002)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P0_SET	Writing a 1 to P0[y] sets P0[y] to 1. Writing 0 is discarded; Reading returns 0	0x0

Table 190: P0_RESET_DATA_REG (0x50003004)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P0_RESET	Writing a 1 to P0[y] sets P0[y] to 0. Writing 0 is discarded; Reading returns 0	0x0

Table 191: P00_MODE_REG (0x50003006)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 191: P00_MODE_REG (0x50003006)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 192: P01_MODE_REG (0x50003008)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 192: P01_MODE_REG (0x50003008)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 193: P02_MODE_REG (0x5000300A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 193: P02_MODE_REG (0x5000300A)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 194: P03_MODE_REG (0x5000300C)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 194: P03_MODE_REG (0x5000300C)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 195: P04_MODE_REG (0x5000300E)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 195: P04_MODE_REG (0x5000300E)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 196: P05_MODE_REG (0x50003010)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 196: P05_MODE_REG (0x50003010)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 197: P06_MODE_REG (0x50003012)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 197: P06_MODE_REG (0x50003012)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 198: P07_MODE_REG (0x50003014)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0

Table 198: P07_MODE_REG (0x50003014)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	Function of port 0 = Port function, PUPD as set above 1 = UART1_RX 2 = UART1_TX 3 = UART2_RX 4 = UART2_TX 5 = SPI_DI 6 = SPI_DO 7 = SPI_CLK 8 = SPI_EN 9 = I2C_SCL 10 = I2C_SDA 11 = UART1_IRDA_RX 12 = UART1_IRDA_TX 13 = UART2_IRDA_RX 14 = UART2_IRDA_TX 15 = ADC (only for P0[3:0]) 16 = PWM0 17 = PWM1 18 = BLE_DIAG (only for P0[7:0]) 19 = UART1_CTSN 20 = UART1_RTSN 21 = UART2_CTSN 22 = UART2_RTSN 23 = PWM2 24 = PWM3 25 = PWM4 Note: when a certain input function (like SPI_DI) is selected on more than 1 port pin, the port with the lowest index has the highest priority and P0 has higher priority than P1.	0x0

Table 199: P1_DATA_REG (0x50003020)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P1_DATA	Set P1 output register when written; Returns the value of P1 port when read	0x0

Table 200: P1_SET_DATA_REG (0x50003022)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P1_SET	Writing a 1 to P1[y] sets P1[y] to 1. Writing 0 is discarded; Reading returns 0	0x0

Table 201: P1_RESET_DATA_REG (0x50003024)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0x0
7:0	r/w	P1_RESET	Writing a 1 to P1[y] sets P1[y] to 0. Writing 0 is discarded; Reading returns 0	0x0

Table 202: P10_MODE_REG (0x50003026)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 203: P11_MODE_REG (0x50003028)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 204: P12_MODE_REG (0x5000302A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 205: P13_MODE_REG (0x5000302C)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x2
7:5	-	-	Reserved	0x0

Table 205: P13_MODE_REG (0x5000302C)

Bit	Mode	Symbol	Description	Reset
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 206: P14_MODE_REG (0x5000302E)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 207: P15_MODE_REG (0x50003030)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care P14_MODE_REG and P15_MODE_REG reset value is 1 (i.e. pulled up)	0x1
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 208: P2_DATA_REG (0x50003040)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	P2_DATA	Set P2 output register when written; Returns the value of P2 port when read	0x0

Table 209: P2_SET_DATA_REG (0x50003042)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	P2_SET	Writing a 1 to P2[y] sets P2[y] to 1. Writing 0 is discarded; Reading returns 0	0x0

Table 210: P2_RESET_DATA_REG (0x50003044)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	P2_RESET	Writing a 1 to P2[y] sets P2[y] to 0. Writing 0 is discarded; Reading returns 0	0x0

Table 211: P20_MODE_REG (0x50003046)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 212: P21_MODE_REG (0x50003048)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 213: P22_MODE_REG (0x5000304A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 214: P23_MODE_REG (0x5000304C)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 215: P24_MODE_REG (0x5000304E)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0

Table 215: P24_MODE_REG (0x5000304E)

Bit	Mode	Symbol	Description	Reset
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 216: P25_MODE_REG (0x50003050)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 217: P26_MODE_REG (0x50003052)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 218: P27_MODE_REG (0x50003054)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 219: P28_MODE_REG (0x50003056)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0

Table 219: P28_MODE_REG (0x50003056)

Bit	Mode	Symbol	Description	Reset
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 220: P29_MODE_REG (0x50003058)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In analog mode, these bits are don't care	0x2
7:5	-	-	Reserved	0x0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0x0

Table 221: P01_PADPWR_CTRL_REG (0x50003070)

Bit	Mode	Symbol	Description	Reset
15:14	-	-	Reserved	0x0
13:8	r/w	P1_OUT_CTRL	1 = P1_x port output is powered by the 1 V rail 0 = P1_x port output is powered by the 3 V rail bit 8 controls the power of P1[0], bit 13 controls the power of P1[5] **(Note 3)**	0x0
7:0	r/w	P0_OUT_CTRL	1 = P0_x port output is powered by the 1 V rail 0 = P0_x port output is powered by the 3 V rail bit 0 controls the power of P0[0], bit 7 controls the power of P0[7] **(Note 4)**	0x0

Note 3: In Buck mode the output must be powered by the 3 V rail. In Boost mode the outputs can be powered by the 1 V rail or by the 3 V rail. In Boost mode the 3 V rail can only supply a limited current, e.g. for switching a high-impedance input of an external device. See table 'Digital input/output characteristics'.

Note 4: In Buck mode the output must be powered by the 3 V rail. In Boost mode the outputs can be powered by the 1 V rail or by the 3 V rail. In Boost mode the 3 V rail can only supply a limited current, e.g. for switching a high-impedance input of an external device. See table 'Digital input/output characteristics'.

Table 222: P2_PADPWR_CTRL_REG (0x50003072)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0x0
9:0	r/w	P2_OUT_CTRL	1 = P2_x port output is powered by the 1 V rail 0 = P2_x port output is powered by the 3 V rail bit 0 controls the power of P2[0], bit 9 controls the power of P2[9], **(Note 5)**	0x0

Note 5: In Buck mode the output must be powered by the 3 V rail. In Boost mode the outputs can be powered by the 1 V rail or by the 3 V rail. In Boost mode the 3 V rail can only supply a limited current, e.g. for switching a high-impedance input of an external device. See table 'Digital

input/output characteristics'.

Table 223: P3_PADPWR_CTRL_REG (0x50003074)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0
7:0	r/w	P3_OUT_CTRL	1 = P3_x port output is powered by the 1 V rail 0 = P3_x port output is powered by the 3 V rail bit 0 controls the power of P3[0], bit 7 controls the power of P3[7], **(Note 6)**	0

Note 6: In Buck mode the output must be powered by the 3 V rail. In Boost mode the outputs can be powered by the 1 V rail or by the 3 V rail. In Boost mode the 3 V rail can only supply a limited current, e.g. for switching a high-impedance input of an external device. See table 'Digital input/output characteristics'.

Table 224: P3_DATA_REG (0x50003080)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0
7:0	r/w	P3_DATA	Set P3 output register when written; Returns the value of P3 port when read	0

Table 225: P3_SET_DATA_REG (0x50003082)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0
7:0	r0/w	P3_SET	Writing a 1 to P3[y] sets P3[y] to 1. Writing 0 is discarded; Reading returns 0	0

Table 226: P3_RESET_DATA_REG (0x50003084)

Bit	Mode	Symbol	Description	Reset
15:8	-	-	Reserved	0
7:0	r0/w	P3_RESET	Writing a 1 to P0[y] sets P0[y] to 0. Writing 0 is discarded; Reading returns 0	0

Table 227: P30_MODE_REG (0x50003086)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 228: P31_MODE_REG (0x50003088)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0

Table 228: P31_MODE_REG (0x50003088)

Bit	Mode	Symbol	Description	Reset
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 229: P32_MODE_REG (0x5000308A)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 230: P33_MODE_REG (0x5000308C)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 231: P34_MODE_REG (0x5000308E)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 232: P35_MODE_REG (0x50003090)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0

Table 232: P35_MODE_REG (0x50003090)

Bit	Mode	Symbol	Description	Reset
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 233: P36_MODE_REG (0x50003092)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 234: P37_MODE_REG (0x50003094)

Bit	Mode	Symbol	Description	Reset
15:10	-	-	Reserved	0
9:8	r/w	PUPD	00 = Input, no resistors selected 01 = Input, pull-up selected 10 = Input, Pull-down selected 11 = Output, no resistors selected In ADC mode, these bits are don't care	2
7:5	-	-	Reserved	0
4:0	r/w	PID	See P0x_MODE_REG[PID]	0

Table 235: WATCHDOG_REG (0x50003100)

Bit	Mode	Symbol	Description	Reset
15:9	r0/w	WDOG_WEN	0000.000 = Write enable for Watchdog timer else Write disable. This filter prevents unintentional preset-ting the watchdog with a SW run-away.	0x0
8	r/w	WDOG_VAL_NEG	0 = Watchdog timer value is positive. 1 = Watchdog timer value is negative.	0x0

7:0	r/w	WDOG_VAL	Write: Watchdog timer reload value. Note that all bits 15-9 must be 0 to reload this register.
			Read: Actual Watchdog timer value. Decremented by 1 every 10.24 msec. Bit 8 indicates a negative counter value. 2, 1, 0, $1FF_{16}$, $1FE_{16}$ etc. An NMI or WDOG (SYS) reset is generated under the following conditions:
			If WATCHDOG_CTRL_REG[NMI_RST] = 0 then
			If WDOG_VAL = 0 -> NMI (Non Maskable Interrupt)
			if WDOG_VAL = $1F0_{16}$ -> WDOG reset -> reload FF_{16}
			If WATCHDOG_CTRL_REG[NMI_RST] = 1 then
			if WDOG_VAL <= 0 -> WDOG reset -> reload FF_{16}
				0xFF

Table 236: WATCHDOG_CTRL_REG (0x50003102)

Bit	Mode	Symbol	Description	Reset
15:2	-	-	Reserved	0x0
1	-	-	Reserved	0x0
0	r/w	NMI_RST	0 = Watchdog timer generates NMI at value 0, and WDOG (SYS) reset at <=-16. Timer can be frozen /resumed using SET_FREEZE_REG[FRZ_WDOG]/ RESET_FREEZE_REG[FRZ_WDOG]. 1 = Watchdog timer generates a WDOG (SYS) reset at value 0 and can not be frozen by Software. Note that this bit can only be set to 1 by SW and only be reset with a WDOG (SYS) reset or SW reset. The watchdog is always frozen when the Cortex-M0 is halted in DEBUG State.	0x0

Table 237: CHIP_ID1_REG (0x50003200)

Bit	Mode	Symbol	Description	Reset
7:0	r	CHIP_ID1	First character of device type "580" in ASCII.	0x35

Table 238: CHIP_ID2_REG (0x50003201)

Bit	Mode	Symbol	Description	Reset
7:0	r	CHIP_ID2	Second character of device type "580" in ASCII.	0x38

Table 239: CHIP_ID3_REG (0x50003202)

Bit	Mode	Symbol	Description	Reset
7:0	r	CHIP_ID3	Third character of device type "580" in ASCII.	0x30

Table 240: CHIP_SWC_REG (0x50003203)

Bit	Mode	Symbol	Description	Reset
7:4	-	-	Reserved	0x0
3:0	r	CHIP_SWC	**S**oft**Wa**re **C**ompatibility code. Integer (default = 0) which is incremented if a silicon change has impact on the CPU Firmware. Can be used by software developers to write silicon revision dependent code.	0x0

Table 241: CHIP_REVISION_REG (0x50003204)

Bit	Mode	Symbol	Description	Reset
7:0	r	REVISION_ID	Chip version, corresponds with type number in ASCII. 0x41 = 'A', 0x42 = 'B'	0x41

Table 242: SET_FREEZE_REG (0x50003300)

Bit	Mode	Symbol	Description	Reset
15:4	-	-	Reserved	0x0

Table 242: SET_FREEZE_REG (0x50003300)

Bit	Mode	Symbol	Description	Reset
3	r/w	FRZ_WDOG	If '1', the watchdog timer is frozen, '0' is discarded. WATCHDOG_CTRL_REG[NMI_RST] must be '0' to allow the freeze function.	0x0
2	r/w	FRZ_BLETIM	If '1', the BLE master clock is frozen, '0' is discarded.	0x0
1	r/w	FRZ_SWTIM	If '1', the SW Timer (TIMER0) is frozen, '0' is discarded.	0x0
0	r/w	FRZ_WKUPTIM	If '1', the Wake Up Timer is frozen, '0' is discarded.	0x0

Table 243: RESET_FREEZE_REG (0x50003302)

Bit	Mode	Symbol	Description	Reset
15:4	-	-	Reserved	0x0
3	r/w	FRZ_WDOG	If '1', the watchdog timer continues, '0' is discarded.	0x0
2	r/w	FRZ_BLETIM	If '1', the the BLE master clock continues, '0' is discarded.	0x0
1	r/w	FRZ_SWTIM	If '1', the SW Timer (TIMER0) continues, '0' is discarded.	0x0
0	r/w	FRZ_WKUPTIM	If '1', the Wake Up Timer continues, '0' is discarded.	0x0

Table 244: DEBUG_REG (0x50003304)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	DEBUGS_FREEZE_EN	Default '1', freezing of the on-chip timers is enabled when the Cortex-M0 is halted in DEBUG State. If '0', freezing of the on-chip timers is depending on FREEZE_REG when the Cortex-M0 is halted in DEBUG State except the watchdog timer. The watchdog timer is always frozen when the Cortex-M0 is halted in DEBUG State.	0x1

Table 245: GP_STATUS_REG (0x50003306)

Bit	Mode	Symbol	Description	Reset
15:1	-	-	Reserved	0x0
0	r/w	CAL_PHASE	If '1', it designates that the chip is in Calibration Phase i.e. the OTP has been initially programmed but no Calibration has occured.	0x0

Table 246: GP_CONTROL_REG (0x50003308)

Bit	Mode	Symbol	Description	Reset
15:6	-	-	Reserved	0

Table 246: GP_CONTROL_REG (0x50003308)

Bit	Mode	Symbol	Description	Reset
5:1	r/w	EM_MAP	Select the mapping of the Exchange memory pages. 0: EM size 0 kB, SysRAM size 42 kB 1: EM size 2 kB, SysRAM size 48 kB 2: EM size 3 kB, SysRAM size 47 kB 3: EM size 4 kB, SysRAM size 46 kB 4: EM size 5 kB, SysRAM size 45 kB 5: EM size 6 kB, SysRAM size 44 kB 6: EM size 7 kB, SysRAM size 43 kB 7: EM size 8 kB, SysRAM size 42 kB 8: Reserved 9: EM size 4 kB, SysRAM size 40 kB 10: EM size 5 kB, SysRAM size 40 kB 11: EM size 6 kB, SysRAM size 40 kB 12: EM size 7 kB, SysRAM size 40 kB 13: EM size 8 kB, SysRAM size 40 kB 14: EM size 9 kB, SysRAM size 40 kB 15: EM size 10 kB, SysRAM size 40 kB 16: Reserved 17: EM size 6 kB, SysRAM size 38 kB 18: EM size 7 kB, SysRAM size 38 kB 19: EM size 8 kB, SysRAM size 38 kB 20: EM size 9 kB, SysRAM size 38 kB 21: EM size 10 kB, SysRAM size 38 kB 22: EM size 11 kB, SysRAM size 38 kB 23: EM size 12 kB, SysRAM size 38 kB other: Reserved.	0x1
0	r/w	BLE_WAKEUP_REQ	If '1', the BLE wakes up. Must be kept high at least for 1 low power clock period. If the BLE is in deep sleep state, then by setting this bit it will cause the wakeup LP IRQ to be asserted with a delay of 3 to 4 low power cycles.	0x0

Table 247: TIMER0_CTRL_REG (0x50003400)

Bit	Mode	Symbol	Description	Reset
15:4	-	-	Reserved	0x0
3	r/w	PWM_MODE	0 = PWM signals are '1' during high time. 1 = PWM signals send out the (fast) clock divided by 2 during high time. So it will be in the range of 1 to 8 MHz.	0x0
2	r/w	TIM0_CLK_DIV	1 = Timer0 uses selected clock frequency as is. 0 = Timer0 uses selected clock frequency divided by 10. Note that this applies only to the ON-counter.	0x0
1	r/w	TIM0_CLK_SEL	1 = Timer0 uses 16, 8, 4 or 2 MHz (fast) clock frequency. 0 = Timer0 uses 32 kHz (slow) clock frequency.	0x0
0	r/w	TIM0_CTRL	0 = Timer0 is off and in reset state. 1 = Timer0 is running.	0x0

Table 248: TIMER0_ON_REG (0x50003402)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	TIM0_ON	Timer0 On reload value: If read the actual counter value ON_CNTer is returned	0x0

Table 249: TIMER0_RELOAD_M_REG (0x50003404)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	TIM0_M	Timer0 'high' reload valueIf read the actual counter value T0_CNTer is returned	0x0

Table 250: TIMER0_RELOAD_N_REG (0x50003406)

Bit	Mode	Symbol	Description	Reset
15:0	r0/w	TIM0_N	Timer0 'low' reload value: If read the actual counter value T0_CNTer is returned	0x0

Table 251: PWM2_DUTY_CYCLE (0x50003408)

Bit	Mode	Symbol	Description	Reset
13:0	r/w	DUTY_CYCLE	duty cycle for PWM	0x0

Table 252: PWM3_DUTY_CYCLE (0x5000340A)

Bit	Mode	Symbol	Description	Reset
13:0	r/w	DUTY_CYCLE	duty cycle for PWM	0x0

Table 253: PWM4_DUTY_CYCLE (0x5000340C)

Bit	Mode	Symbol	Description	Reset
13:0	r/w	DUTY_CYCLE	duty cycle for PWM	0x0

Table 254: TRIPLE_PWM_FREQUENCY (0x5000340E)

Bit	Mode	Symbol	Description	Reset
13:0	r/w	FREQ	Freq for PWM 2 3 4	0x0

Table 255: TRIPLE_PWM_CTRL_REG (0x50003410)

Bit	Mode	Symbol	Description	Reset
2	r/w	HW_PAUSE_EN	'1' = HW can pause PWM 2,3,4	0x1
1	r/w	SW_PAUSE_EN	'1' = PWM 2 3 4 is paused	0x0
0	r/w	TRIPLE_PWM_ENABLE	'1' = PWM 2 3 4 is enabled	0x0

6 Specifications

All MIN/MAX specification limits are guaranteed by design, production testing and/or statistical characterization. Typical values are based on characterization results at default measurement conditions and are informative only.

Default measurement conditions (unless otherwise specified): V_{BAT}(VBAT3V) = 3.0 V (buck mode), V_{BAT}(VBAT1V) = 1.2 V (boost mode), T_A = 25 °C. All radio measurements are performed with standard RF measurement equipment providing a source/load impedance of 50 Ω.

The specifications in the following tables are valid for the reference circuits shown in Figure 10 (Boost mode) and Figure 11 (Buck mode).

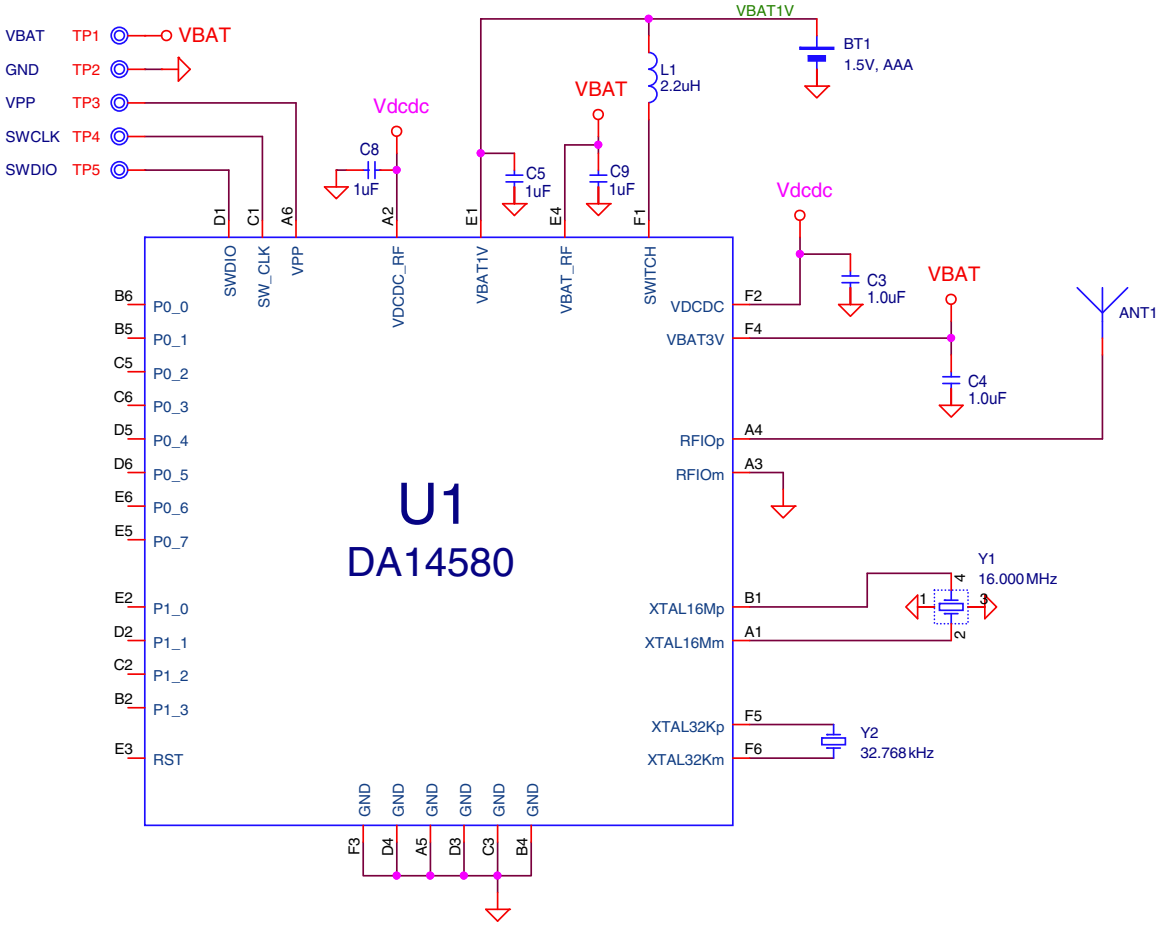


Figure 10: Alkaline Battery Cell Powered System Diagram (Boost Mode)

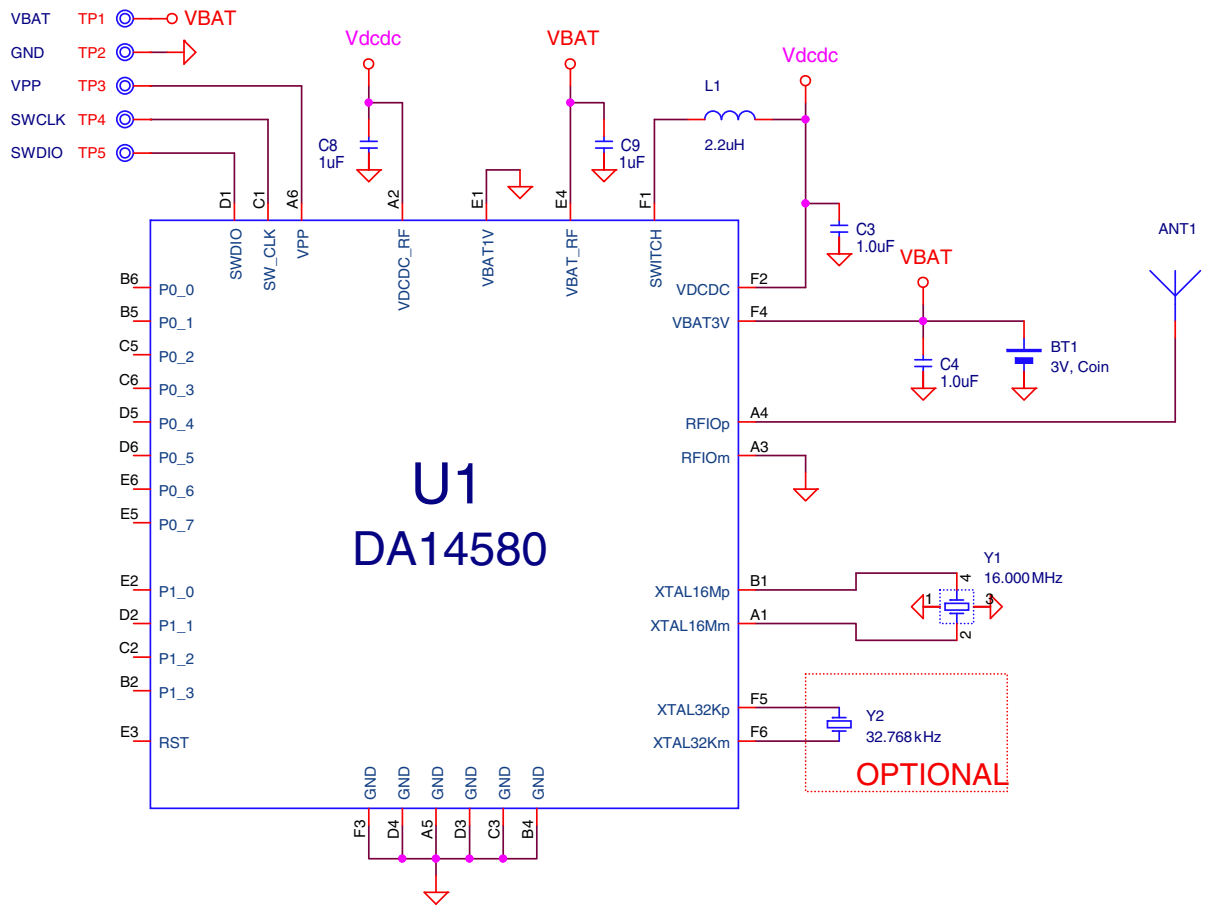


Figure 11: Lithium Coin Cell Powered System Diagram (Buck Mode)

Table 256: Absolute Maximum Ratings

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{PIN(LIM)}$(default)	limiting voltage on a pin	Voltage between pin and GND **(Note 7)**	-0.1		min{3.6, VBAT_RF+0.2}	V
T_{STG}	storage temperature		-50		150	°C
$t_{R(SUP)}$	supply rise time	Power supply rise time			100	ms
$V_{BAT(LIM)}$(VBAT1V)	limiting battery supply voltage	Supply voltage on VBAT1V in a boost converter application (VBAT3V is second output of boost-converter in this case) **(Note 7)**	-0.1		3.6	V
$V_{BAT(LIM)}$(VBAT3V)	limiting battery supply voltage	Supply voltage on VBAT3V in a buck-converter application, pin VBAT1V is connected to ground **(Note 7)**	-0.1		3.6	V
$V_{PIN(LIM)}$(1V2)	limiting voltage on a pin	XTAL32Km, XTAL16Mp, XTAL16Mm **(Note 7)**	-0.2		min(1.2,VBAT_RF+0.2)	V
$V_{PIN(LIM)}$(XTAL32Kp)	limiting voltage on a pin	XTAL32Kp	-0.2		min(1.5,VBAT_RF+0.2)	V
$V_{PIN(LIM)}$VDCDC_RF	limiting voltage on the VDCDC_RF pin	Supply voltage on VDCDC_RF **(Note 7)**	-0.2		min(3.3,VBAT_RF+0.2)	V
$V_{ESD(HBM)}$(WLCSP34)	electrostatic discharge voltage (Human Body Model)				2000	V
$V_{ESD(HBM)}$(QFN40)	electrostatic discharge voltage (Human Body Model)				4000	V
$V_{ESD(HBM)}$(QFN48)	electrostatic discharge voltage (Human Body Model)				4000	V
$V_{ESD(MM)}$(WLCSP34)	electrostatic discharge voltage (Machine Model)				200	V
$V_{ESD(MM)}$(QFN40)	electrostatic discharge voltage (Machine Model)				200	V
$V_{ESD(MM)}$(QFN48)	electrostatic discharge voltage (Machine Model)				200	V
$V_{ESD(CDM)}$(WLCSP34)	electrostatic discharge voltage (Charged Device Model)				500	V
$V_{ESD(CDM)}$(QFN40)	electrostatic discharge voltage (Charged Device Model)				1000	V
$V_{ESD(CDM)}$(QFN48)	electrostatic discharge voltage (Charged Device Model)				1000	V

Note 7: The device should not be exposed for prolonged periods of time to voltages between the Recommended Operating Conditions and the Absolute Maximum Ratings range.

Table 257: Recommended Operating Conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_{PP}	programming voltage	Supply voltage on pin VPP during OTP programming; $T_J \leq 50$ °C	6.6	6.7	6.8	V
V_{BAT}(VBAT1V)	battery supply voltage	Supply voltage on VBAT1V in a boost converter application (VBAT3V is second output of boost-converter in this case)	0.9		3.3	V
V_{BAT}(VBAT3V)	battery supply voltage	Supply voltage on VBAT3V and VBAT_RF in a buck-converter application, pin VBAT1V is connected to ground	2.35 (Note 8)		3.3	V
V_{PIN}(default)	voltage on a pin	Voltage between pin and GND	0		min(3.3,VBAT_RF+0.2)	V
V_{PIN}(1V2)	voltage on a pin	XTAL32Km, XTAL16Mp, XTAL16Mm	0		1.2	V
V_{PIN}(VDCDC_RF)	voltage on the VDCDC_RF pin	Supply voltage on VDCDC_RF	0		3.3	V
T_A	ambient temperature		-40		85	°C

Note 8: Cold boot should not be performed if voltage is less than 2.5 V because of possible corruption during OTP data mirroring. Trim values programmed in the OTP as well as the application image, should be copied into RAM while VBAT3V >= 2.5 V.

Table 258: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
I_{BAT}(DP_SLP)_BOOST_1kB	battery supply current	Boost configuration in deep-sleep with 1 kB retention RAM active, running from RC32K oscillator at lowest frequency		0.48		µA
I_{BAT}(DP_SLP)_BOOST_2kB	battery supply current	Boost configuration in deep-sleep with 2 kB retention RAM active, running from XTAL32K oscillator		0.55		µA
I_{BAT}(DP_SLP)_BOOST_8kB	battery supply current	Typical boost-application in deep-sleep with 8 kB retention RAM active, running from XTAL32K oscillator		0.7	2	µA
I_{BAT}(EXT_SLP)_BOOST_43KB	battery supply current	Typical boost-application in extended-sleep mode with 42 kB (SysRAM) and 1 kB (RetRAM) retained		1.37		µA

Table 258: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
I_{BAT}(EXT_SLP)_BOOST_50kB	battery supply current	Typical boost-application in extended-sleep mode with 42 kB (SysRAM) and 8 kB (RetRAM) retained		1.5	3	µA
I_{BAT}(DP_SLP)_BUCK_1kB	battery supply current	Buck configuration deep-sleep with 1 kB retention RAM active, running from RC32K oscillator at lowest frequency		0.4		µA
I_{BAT}(DP_SLP)_BUCK_2kB	battery supply current	Buck configuration in deep-sleep with 2 kB retention RAM active, running from XTAL32K oscillator		0.45		µA
I_{BAT}(DP_SLP)_BUCK_8kB	battery supply current	Typical buck-application in deep-sleep with 8 kB retention RAM active, running from XTAL32K oscillator		0.6	2	µA
I_{BAT}(EXT_SLP)_BUCK_43KB	battery supply current	Typical buck-application in extended-sleep mode with 42 kB (SysRAM) and 1 kB (RetRAM) retained		1.2		µA
I_{BAT}(EXT_SLP)_BUCK_50kB	battery supply current	Typical buck-application in extended-sleep mode with 42 kB (SysRAM) and 8 kB (RetRAM) retained		1.4	3	µA
I_{BAT}(ACT_RX)_BOOST	battery supply current	Typical application with boost converter and receiver active		13.4	16	mA
I_{BAT}(ACT_TX)_BOOST	battery supply current	Typical application with boost converter and transmitter active		12.4	15	mA
I_{BAT}(ACT_RX)_BUCK	battery supply current	Typical application with buck converter and receiver active		5.1	6	mA
I_{BAT}(ACT_TX)_BUCK	battery supply current	Typical application with buck converter and transmitter active		4.8	6	mA

Table 259: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
t_{STA}(BOOST)	startup time	Boost-mode; time from deep-sleep to software start. Typical application, running from retention RAM on 16 MHz RC oscillator		1.2 **(Note 9)**		ms

Table 259: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
t_{STA}(BUCK)	startup time	Buck-mode; time from deep-sleep to software start. Typical application, running from retention RAM on 16 MHz RC oscillator		1 **(Note 9)**		ms

Note 9: Worst-case value under Normal Operating Conditions.

Table 260: 16 MHz Crystal Oscillator: Recommended Operating Conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
f_{XTAL}(16M)	crystal oscillator frequency			16		MHz
ESR(16M)	equivalent series resistance				100	Ω
C_L(16M)	load capacitance	No external capacitors are required	10		12	pF
C_0(16M)	shunt capacitance				5	pF
Δf_{XTAL}(16M)	crystal frequency tolerance	After optional trimming; including aging and temperature drift **(Note 10)**	-20		20	ppm
Δf_{XTAL}(16M)UNT	crystal frequency tolerance	Untrimmed; including aging and temperature drift **(Note 11)**	-40		40	ppm
$P_{DRV(MAX)}$(16M)	maximum drive power	**(Note 12)**	100			µW
$V_{CLK(EXT)}$(16M)	external clock voltage	Only in case of an external reference clock on XTAL16Mp (XTAL16Mm floating or connected to mid-level 0.6 V)	1	1.2		V
φ_N(EXTERNAL)16M	phase noise	f_C = 50 kHz in case of an external reference clock			-130	dBc/Hz

Note 10: Using the internal varicaps a wide range of crystals can be trimmed to the required tolerance.
Note 11: Maximum allowed frequency tolerance for compensation by the internal varicap trimming mechanism.
Note 12: Select a crystal which can handle a drive-level equal or more than this specification

Table 261: 16 MHz Crystal Oscillator: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$t_{STA(XTAL)}$(16M)	crystal oscillator startup time		0.5	2	3	ms

Table 262: 32 kHz Crystal Oscillator: Recommended Operating Conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{CLK(EXT)}$(32K)	external clock voltage	peak-peak voltage of external clock at XTAL32Kp, pin XTAL32Km floating. note: XTAL32Kp is internally AC coupled	0.1	0.2	1.5	V
f_{XTAL}(32k)	crystal oscillator frequency	frequency range for an external clock (for a crystal, use either 32.000 kHz or 32.768 kHz)	10	32.768	100	kHz
ESR(32k)	equivalent series resistance				100	kΩ
C_L(32k)	load capacitance	no external capacitors are required for a 6 pF or 7 pF crystal	6	7	9	pF
C_0(32k)	shunt capacitance			1	2	pF
Δf_{XTAL}(32k)	crystal frequency tolerance (including aging)	Timing accuracy is dominated by crystal accuracy. A much smaller value is preferred	-250		250	ppm
$P_{DRV(MAX)}$(32k)	maximum drive power	**(Note 13)**	0.1			µW

Note 13: Select a crystal that can handle a drive-level of at least this specification.

Table 263: 32 kHz Crystal Oscillator: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$t_{STA(XTAL)}$(32k)	crystal oscillator startup time	Typical application, time until 1000 clocks are detected. **(Note 14)**		0.4		s

Note 14: This parameter is very much dependent on crystal parameters

Table 264: DC-DC Converter: Recommended Operating Conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
L	effective inductance		1.5	2.2	3	µH
C_{OUT}(VDCDC)	effective load capacitance	VDCDC and VDDCRF combined **(Note 15)**	0.5	1	10	µF
C_{OUT}(VBAT3V)	effective load capacitance	VBATRF and VBAT3V combined are the second output of the boost-converter **(Note 15)**	0.5	1	10	µF

Note 15: A low value will result in lowest power consumption, keep this value at 1 uF or 2 uF.

Table 265: DC-DC Converter: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_O(BUCK)	output voltage	default settings		1.41		V

Table 265: DC-DC Converter: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_O(BOOST)	output voltage	default settings, VDCDC		1.41		V
η_{CONV_MAX}(BUCK)	maximum conversion efficiency			86		%
η_{CONV_MAX}(BOOST)	maximum conversion efficiency			80		%
$\Delta V_O / \Delta V_I$(BUCK)	line regulation	2.35 V ≤ VBAT3V ≤ 3.3 V	-2	0.7	2	%/V
$\Delta V_O / \Delta V_I$(BOOST)	line regulation	0.9 V ≤ VBAT1V ≤ 1.2 V **(Note 16)**	-2	1	4	%/V
$\Delta V_O / \Delta I_L$(BUCK)	load regulation	VBAT3V = 2.5 V	-0.2	-0.02	0.2	%/mA
$\Delta V_O / \Delta I_L$(BOOST)	load regulation	VBAT1V = 1.2 V	-0.2	-0.07	0.2	%/mA
V_{RPL}(BUCK)	ripple voltage	buck mode; RMS ripple voltage		5		mV
V_{RPL}(BOOST)	ripple voltage	VBAT1V ≤ 1.2 V, boost mode; RMS ripple voltage **(Note 16)**		8		mV

Note 16: When VBAT1V > VDCDC_nominal, VDCDC will follow VBAT1V.

Table 266: Digital Input/Output: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
V_{IH}	HIGH level input voltage		0.84			V
V_{IL}	LOW level input voltage				0.36	V
V_{IH}(RST)	HIGH level input voltage	RST pin	0.84			V
V_{IL}(RST)	LOW level input voltage	RST pin			0.36	V
V_{OH}(VBAT1V)	HIGH level output voltage	Iout = -250 µA, VBAT3V = 2.35 V, VBAT1V = 0.9 V	0.72			V
V_{OH}(VBAT3V)	HIGH level output voltage	Iout = -4.8 mA, VBAT3V = 2.35 V, VBAT1V = 0 V **(Note 17)**	1.88			V
V_{OL}(VBAT1V)	LOW level output voltage	Iout = 250 µA, VBAT3V = 2.35 V, VBAT1V = 0.9 V			0.18	V
V_{OL}(VBAT3V)	LOW level output voltage	Iout = 4.8 mA, VBAT3V = 2.35 V, VBAT1V = 0 V **(Note 18)**			0.47	V
I_{IH}	HIGH level input current	Vin = VBAT3V = 2.5 V	-1		1	µA
I_{IL}	LOW level input current	Vin = VSS = 0 V	-1		1	µA
I_{IH}(PD)	HIGH level input current	Vin = VBAT3V = 2.5 V	50		150	µA
I_{IL}(PU)	LOW level input current	Vin = VSS = 0 V	-150		-50	µA
I_{IH}(RST)	HIGH level input current	RST pin, V(RST) = 1.2 V	25		75	µA

Note 17: In Boost mode the output source current is limited to Iout = -250 uA.
Note 18: In Boost mode the output sink current is limited to Iout = 250 uA.

Table 267: General Purpose ADC: Recommended Operating Conditions

Parameter	Description	Conditions	Min	Typ	Max	Unit
N_{BIT}(ADC)	number of bits (resolution)			10		bit

Table 268: General Purpose ADC: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$V_{I(ZS)}$	zero-scale input voltage	single-ended, calibrated at zero input	-2.5	0	2.5	mV
$V_{I(FS)}$	full-scale input voltage	single-ended, calibrated at zero input	1150	1180	1250	mV
$V_{I(FSN)}$	negative full-scale input voltage	differential, calibrated at zero input		-1180		mV
$V_{I(FSP)}$	positive full-scale input voltage	differential, calibrated at zero input		1180		mV
INL	integral non-linearity		-2		2	LSB
DNL	differential non-linearity		-2		2	LSB

Table 269: General Purpose ADC: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
t_{CONV}(ADC)	conversion time	Excluding initial settling time of the LDO and the 3x-attenuation (if used): LDO settling time is 20 µs (max), 3x-attenuation settling time = 1 µs (max) Using internal ADC-clock (~200 MHz)		0.25	0.4	µs

Table 270: Radio: DC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
I_{BAT}(RF)RX	battery supply current	receive mode; radio receiver and synthesizer active; DC-DC converter assumed ideal; T_A = 25 °C **(Note 19)**		3.7	4.3	mA
I_{BAT}(RF)TX	battery supply current	transmit mode; radio transmitter and synthesizer active; DC-DC converter assumed ideal; T_A = 25 °C **(Note 19)**		3.4	4	mA

Note 19: The DC-DC-converter efficiency is assumed to be 100 % to enable benchmarking of the radio currents at battery supply domain (VBAT3V = 3 V).

Table 271: Radio: AC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
P_{SENS}(CLEAN)	sensitivity level	DC-DC converter enabled; Dirty Transmitter disabled; T_A = 25 °C; PER = 30.8 % **(Note 20)**		-93		dBm
P_{SENS}	sensitivity level	Normal Operating Conditions; DC-DC converter enabled; T_A = 25 °C; PER = 30.8 % **(Note 20)**		-92.5		dBm
P_I(max)	input power level	DC-DC converter disabled; T_A = 25 °C; PER ≤ 30.8 % **(Note 20)**	10			dBm
$P_{INT(IMD)}$	intermodulation distortion interferer power level	worst case interferer level @ f_1, f_2 with $2f_1$-f_2 = f_0, $\|f_1-f_2\|$ = n MHz and n = 3,4,5; P_{WANTED} = -64 dBm @ f_0; PER = 30.8 %; T_A = 25 °C **(Note 22)**	-35	-31		dBm
CIR(0)	carrier to interferer ratio	n = 0; interferer @ f_1 = f_0 + n*1 MHz; T_A = 25 °C **(Note 23)**		7	21	dB
CIR(1)	carrier to interferer ratio	n = ±1; interferer @ f_1 = f_0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-3	15	dB
CIR(P2)	carrier to interferer ratio	n = +2 (image frequency); interferer @ f1 = f0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-20	-9	dB
CIR(M2)	carrier to interferer ratio	n = -2; interferer @ f1 = f0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-30	-17	dB
CIR(P3)	carrier to interferer ratio	n = +3 (image frequency + 1 MHz); interferer @ f1 = f0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-30	-15	dB
CIR(M3)	carrier to interferer ratio	n = -3; interferer @ f1 = f0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-35	-27	dB
CIR(4)	carrier to interferer ratio	|n| >= 4 (any other BLE channel); interferer @ f_1 = f_0 + n*1 MHz; T_A = 25 °C **(Note 23)**		-37	-27	dB

Table 271: Radio: AC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
$P_{BL}(I)$	blocker power level	30 MHz ≤ f_{BL} ≤ 2000 MHz; P_{WANTED} = -67 dBm; T_A = 25 °C **(Note 24)**	-5			dBm
$P_{BL}(II)$	blocker power level	2003 MHz ≤ f_{BL} ≤ 2399 MHz; P_{WANTED} = -67 dBm; T_A = 25 °C **(Note 24)**	-15			dBm
$P_{BL}(III)$	blocker power level	2484 MHz ≤ f_{BL} ≤ 2997 MHz; P_{WANTED} = -67 dBm; T_A = 25 °C **(Note 24)**	-15			dBm
$P_{BL}(IV)$	blocker power level	3000 MHz ≤ f_{BL} ≤ 12.75 GHz; P_{WANTED} = -67 dBm; T_A = 25 °C **(Note 24)**	-5			dBm
$P_{RSSI}(min)$	RSSI power level	absolute power level for RXRSSI[7:0] = 0; T_A = 25 °C **(Note 25)**	-115	-112	-109	dBm
$P_{RSSI}(max)$	RSSI power level	upper limit of monotonous range; T_A = 25 °C	-26	-19		dBm
$L_{ACC}(RSSI)BOOST$	level accuracy	tolerance of 5 % to 95 % confidence interval of P_{RF}: when RXRSSI[7:0] = *X*, 50 < X < 175; burst mode 1500 packets; T_A = 25 °C; DC-DC converter in BOOST mode		0	3	dB
$L_{ACC}(RSSI)BUCK$	level accuracy	tolerance of 5 % to 95 % confidence interval of P_{RF}: when RXRSSI[7:0] = *X*, 50 < X < 175; burst mode 1500 packets; T_A = 25 °C; DC-DC converter in BUCK mode		0	2	dB
$L_{RES}(RSSI)$	level resolution	gradient of monotonous range for RXRSSI[7:0] = X, 50 < X < 175; burst mode 1500 packets; T_A = 25 °C	0.46	0.474	0.485	dB/ LSB
ACP(2M)	adjacent channel power level	f_{OFFSET} = 2 MHz; T_A = 25 °C **(Note 26)**		-53	-50	dBm
ACP(2M)(EOC)	adjacent channel power level	f_{OFFSET} = 2 MHz; -40 °C ≤ T_A ≤ +85 °C **(Note 26)**		-53	-47	dBm
ACP(3M)	adjacent channel power level	f_{OFFSET} ≥ 3 MHz; T_A = 25 °C **(Note 26)**		-57	-55	dBm

Table 271: Radio: AC Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
ACP(3M)(EOC)	adjacent channel power level	$f_{OFFSET} \geq 3$ MHz; -40 °C ≤ T_A ≤ +85 °C **(Note 26)**		-57	-47	dBm
P_O	output power level	V_{DD} = 3 V; maximum gain; T_A = 25 °C	-2	-1	0	dBm
P_O(HD2)	output power level (second harmonic)	VDD = 3 V; maximum gain; T_A = 25 °C		-54	-40	dBm
P_O(HD3)	output power level (third harmonic)	VDD = 3 V; maximum gain; T_A = 25 °C		-56	-40	dBm
P_O(HD4)	output power level (fourth harmonic)	VDD = 3 V; maximum gain; T_A = 25 °C		-70	-40	dBm
P_O(HD5)	output power level (fifth harmonic)	VDD = 3 V; maximum gain; T_A = 25 °C		-70	-40	dBm
P_O(NFM)	output power level in 'Near Field Mode'	V_{DD} = 3 V; maximum gain; T_A = 25 °C **(Note 27)**	-25	-20	-15	dBm

Note 20: Measured according to Bluetooth® Low Energy Test Specification RF-PHY.TS/4.0.1, section 6.4.1.
Note 21: Measured according to Bluetooth® Low Energy Test Specification RF-PHY.TS/4.0.1, section 6.4.2.
Note 22: Measured according to Bluetooth® Core Technical Specification document, version 4.2, volume 6, section 4.4. Published value is for n = IXIT = 4 . IXIT = 5 gives the same results, IXIT = 3 gives results that are 5 dB lower.
Note 23: Measured according to Bluetooth® Core Technical Specification document, version 4.2, volume 6, section 4.2.
Note 24: Measured according to Bluetooth® Core Technical Specification document, version 4.2, volume 6, section 4.3. Due to limitations of the measurement equipment, levels of -5 dBm should be interpreted as > -5 dBm.
Note 25: PRF = PRSSI(min) + LRES(RSSI) x RXRSSI[7:0] ± LACC(RSSI). Thanks to constant gain biasing of RF part in the receiver, the RSSI can be used to estimate absolute power levels, rather than mere level changes. Even across the full temperature range the variation is limited.
Note 26: Measured according to Bluetooth® Low Energy Test Specification RF-PHY.TS/4.0.1, section 6.2.3.
Note 27: To activate the "Near Field Mode", program address 0x50002418 with the value 0x0030.

Table 272: Stable Low Frequency RCX Oscillator: Timing Characteristics

Parameter	Description	Conditions	Min	Typ	Max	Unit
f_{RC}(RCX)	RCX oscillator frequency	default setting, buck mode only	5	10	40	kHz
Δf_{RC}(RCX)	RCX oscillator frequency drift	buck mode only **(Note 28)**	-500		500	ppm
ΔT_A/ Δt(RCX)100ms	ambient temperature gradient	buck mode only; connection interval 100 ms			0.66	°C/s
$\Delta T_A/\Delta$t(RCX)4s	ambient temperature gradient	buck mode only; connection interval 4 s			0.33	°C/s

Note 28: Maximum recommended connection interval (including slave latency) for the RCX usage is 2 s.

7 Package Information

7.1 MOISTURE SENSITIVITY LEVEL (MSL)

The MSL is an indicator for the maximum allowable time period (floor life time) in which a moisture sensitive plastic device, once removed from the dry bag, can be exposed to an environment with a maximum temperature of 30 °C and a maximum relative humidity of 60 % RH. before the solder reflow process.

WLCSP packages are qualified for MSL 1.
QFN packages are qualified for MSL 3.

MSL Level	Floor Life Time
MSL 4	72 hours
MSL 3	168 hours
MSL 2A	4 weeks
MSL 2	1 year
MSL 1	Unlimited at 30 °C / 85 % RH

7.2 WLCSP HANDLING

Manual handling of WLCSP packages should be reduced to the absolute minimum. In cases where it is still necessary, a vacuum pick-up tool should be used. In extreme cases plastic tweezers could be used, but metal tweezers are not acceptable, since contact may easily damage the silicon chip.

Removal will cause damage to the solder balls and therefore a removed sample cannot be reused.

WLCSP is sensitive to visible and infrared light. Precautions should be taken to properly shield the chip in the final product.

7.3 SOLDERING INFORMATION

Refer to the JEDEC standard J-STD-020 for relevant soldering information.
This document can be downloaded from http://www.jedec.org

7.4 PACKAGE OUTLINES

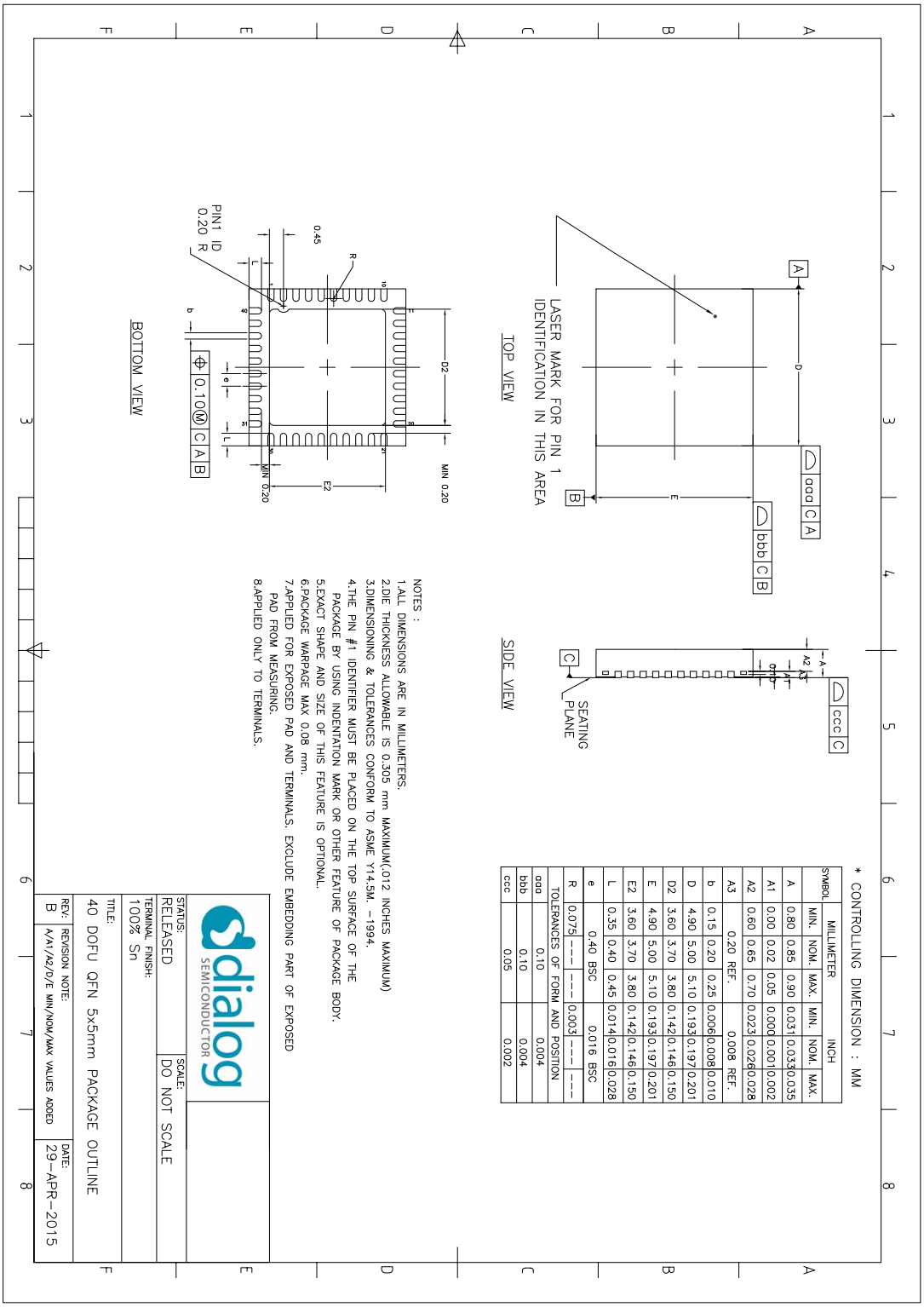
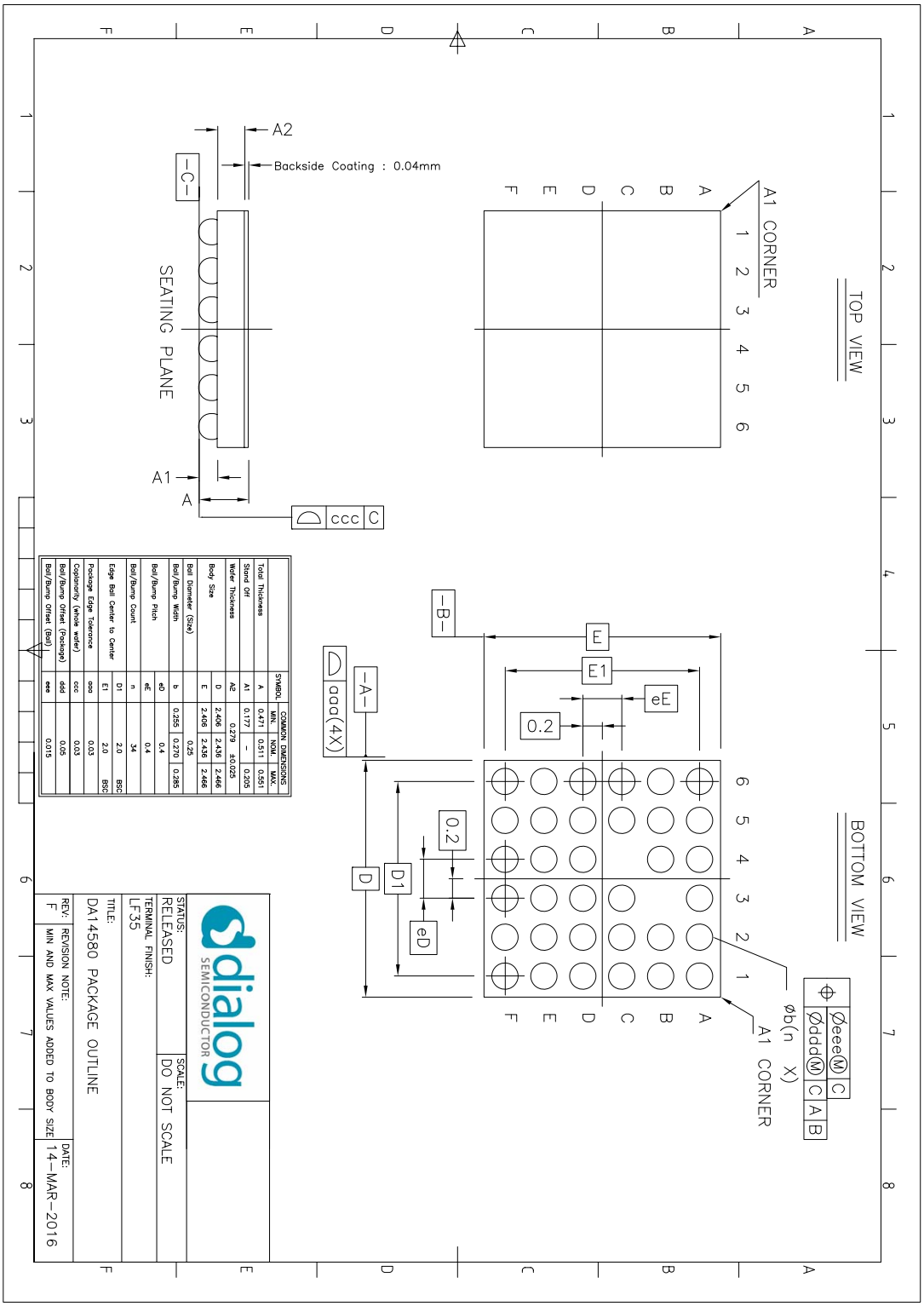


Figure 12: QFN40 Package Outline Drawing



	SYMBOL	COMMON DIMENSIONS MIN.	NOM.	MAX.
Total Thickness	A	0.471	0.511	0.551
Stand Off	A1	0.177	-	0.205
Wafer Thickness	A2		0.279	±0.025
Body Size	D	2.406	2.436	2.466
	E	2.406	2.436	2.466
Ball Diameter (Size)	b		0.25	
Ball/Bump Width		0.255	0.270	0.285
Ball/Bump Pitch	eD		0.4	
	eE		0.4	
Ball/Bump Count	n		34	
Edge Ball Center to Center	D1		2.0	BSC
	E1		2.0	BSC
Package Edge Tolerance	aaa		0.03	
Coplanarity (Whole wafer)	ccc		0.03	
Ball/Bump Offset (Package)	ddd		0.05	
Ball/Bump Offset (Ball)	eee		0.015	

Figure 13: WLCSP34 Package Outline Drawing

Status Definitions

Version	Datasheet Status	Product Status	Definition
1.<n>	Target	Development	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
2.<n>	Preliminary	Qualification	This datasheet contains the specifications and preliminary characterization data for products in pre-production. Specifications may be changed at any time without notice in order to improve the design.
3.<n>	Final	Production	This datasheet contains the final specifications for products in volume production. The specifications may be changed at any time in order to improve the design, manufacturing and supply. Major specification changes are communicated via Customer Product Notifications. Datasheet changes are communicated via www.dialog-semiconductor.com.
4.<n>	Obsolete	Archived	This datasheet contains the specifications for discontinued products. The information is provided for reference only.

Disclaimer

RoHS Compliance

Dialog Semiconductor's suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

Contacting Dialog Semiconductor

United Kingdom (Headquarters)
Dialog Semiconductor (UK) LTD
Phone: +44 1793 757700

Germany
Dialog Semiconductor GmbH
Phone: +49 7021 805-0

The Netherlands
Dialog Semiconductor B.V.
Phone: +31 73 640 8822

Email:
enquiry@diasemi.com

North America
Dialog Semiconductor Inc.
Phone: +1 408 845 8500

Japan
Dialog Semiconductor K. K.
Phone: +81 3 5425 4567

Taiwan
Dialog Semiconductor Taiwan
Phone: +886 281 786 222

Web site:
www.dialog-semiconductor.com

Singapore
Dialog Semiconductor Singapore
Phone: +65 64 8499 29

Hong Kong
Dialog Semiconductor Hong Kong
Phone: +852 3769 5200

Korea
Dialog Semiconductor Korea
Phone: +82 2 3469 8200

China (Shenzhen)
Dialog Semiconductor China
Phone: +86 755 2981 3669

China (Shanghai)
Dialog Semiconductor China
Phone: +86 21 5424 9058