Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

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

General Description 1

Features 1

System Diagram 1

Contents 2

1 Block Diagram 3

2 Pinout 4

3 Ordering Information 7

4 System Overview 8

4.1 ARM CORTEX–M0 CPU 8
4.2 BLUETOOTH LOW ENERGY 8
4.2.1 BLE Core 8
4.2.2 Radio Transceiver 9
4.2.3 SmartSnippets™ 9
4.3 MEMORIES 10
4.4 FUNCTIONAL MODES 10
4.5 POWER MODES 11
4.6 INTERFACES 11
4.6.1 UARTs 11
4.6.2 SPI+ 11
4.6.3 I2C Interface 11
4.6.4 General Purpose ADC 12
4.6.5 Quadrature Decoder 12
4.6.6 Keyboard Controller 12
4.6.7 Input/Output Ports 12
4.7 TIMERS 12
4.7.1 General Purpose Timers 12
4.7.2 Wake-Up Timer 13
4.7.3 Watchdog Timer 13
4.8 CLOCK/RESET 13
4.8.1 Clocks 13
4.8.2 Reset 13
4.9 POWER MANAGEMENT 14

5 Registers 16

6 Specifications 137

7 Package Information 149

7.1 MOISTURE SENSITIVITY LEVEL (MSL) . . . 149
7.2 WLCSP HANDLING 149
7.3 SOLDERING INFORMATION 149
7.4 PACKAGE OUTLINES 150

1 Block Diagram

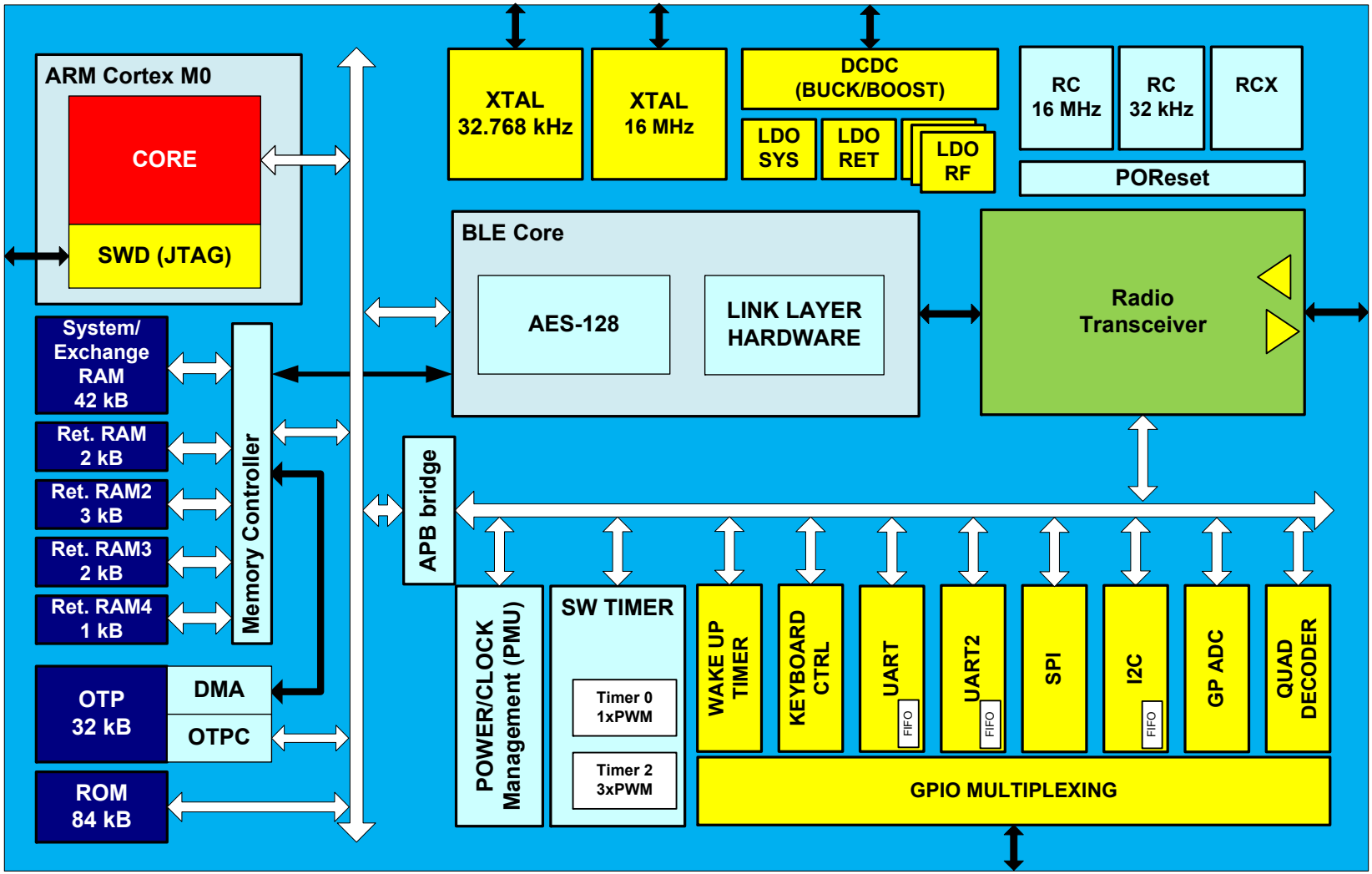


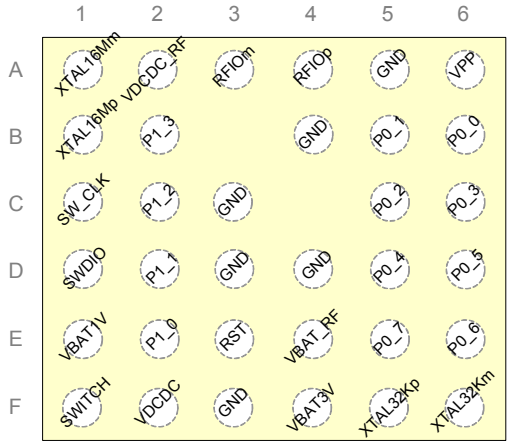
Figure 1: DA14581 Block Diagram

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

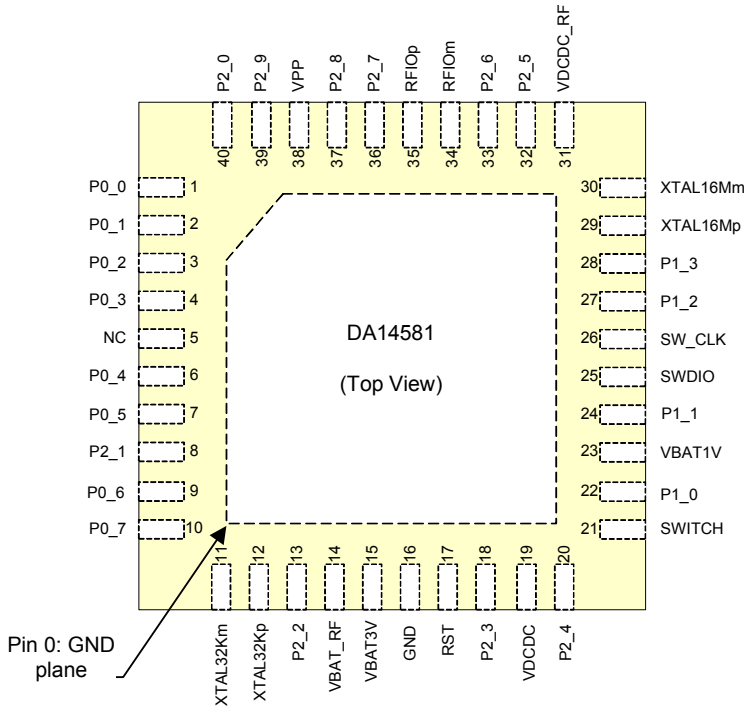


Figure 3: QFN40 Pin Assignment

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.

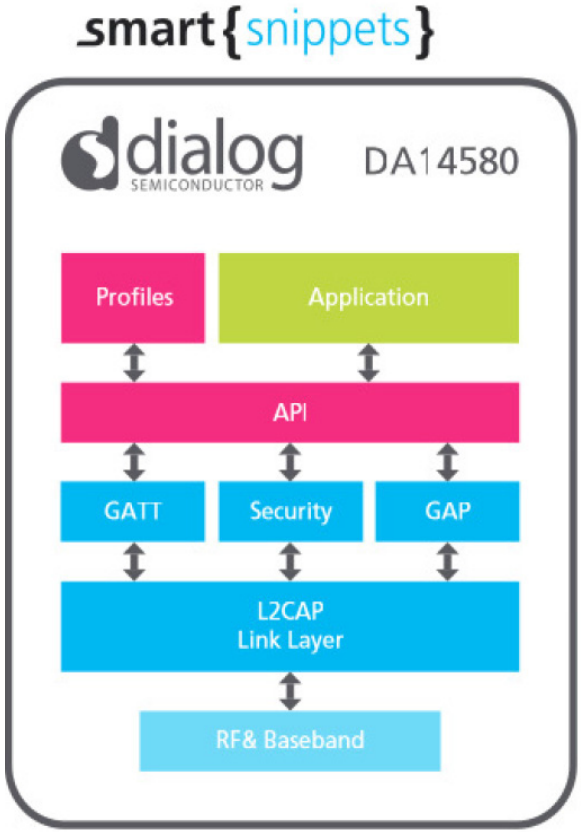


Figure 4: SmartSnippets Stack

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.

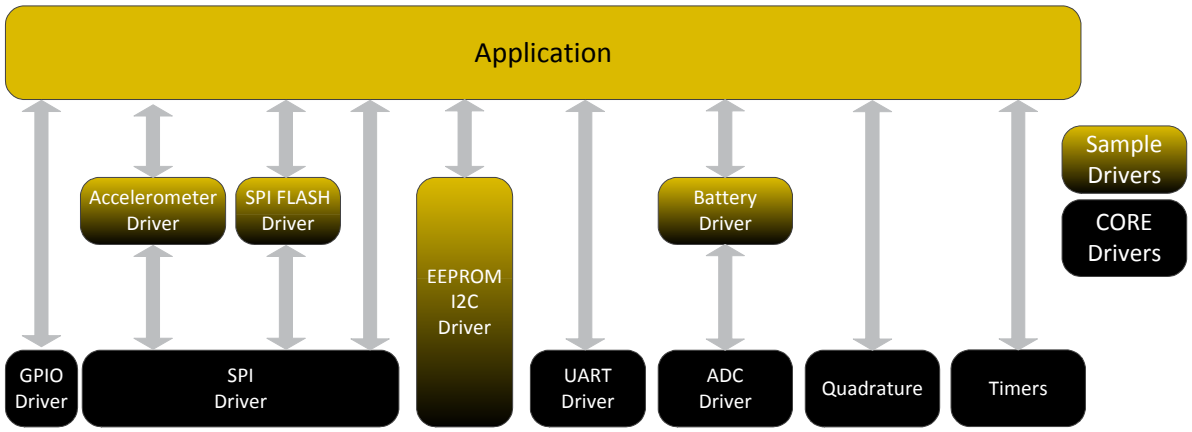


Figure 5: Hardware Abstraction Layer

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.

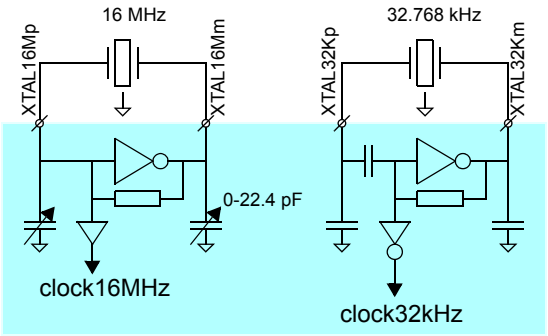


Figure 6: Crystal Oscillator Circuits

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:

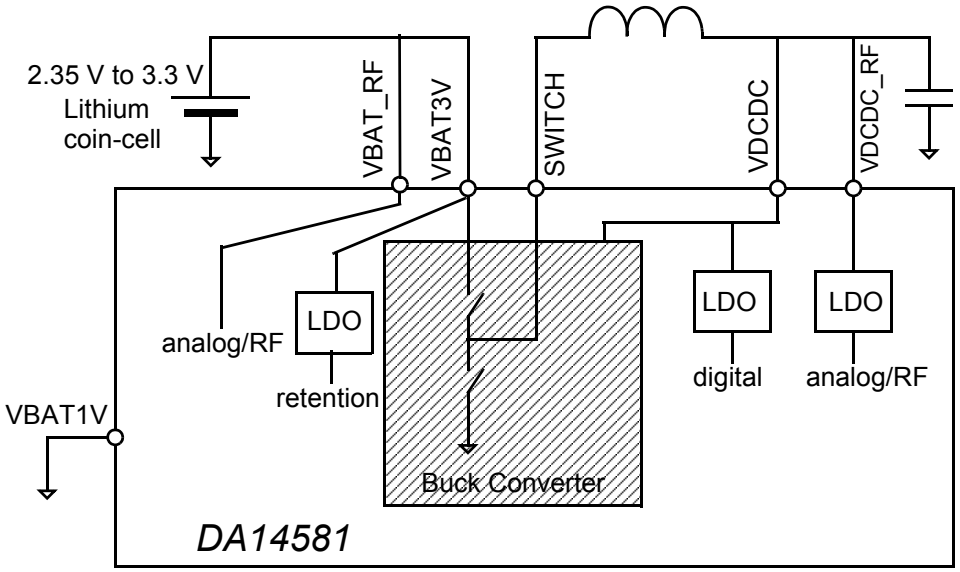


Figure 7: Supply Overview, Coin-Cell Application

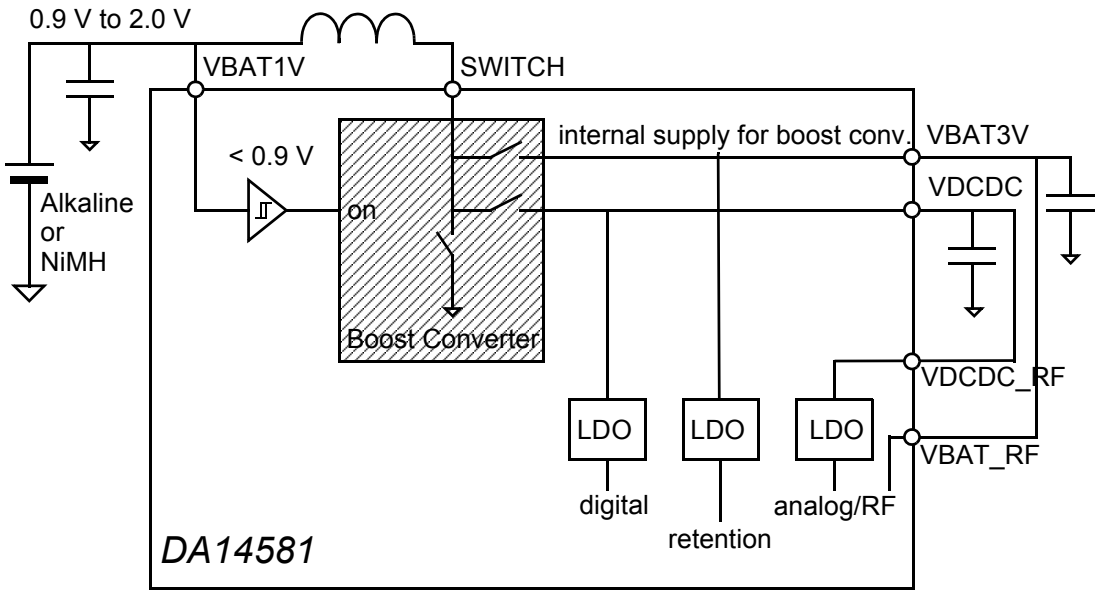


Figure 8: Supply Overview, Alkaline-Cell Application

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.

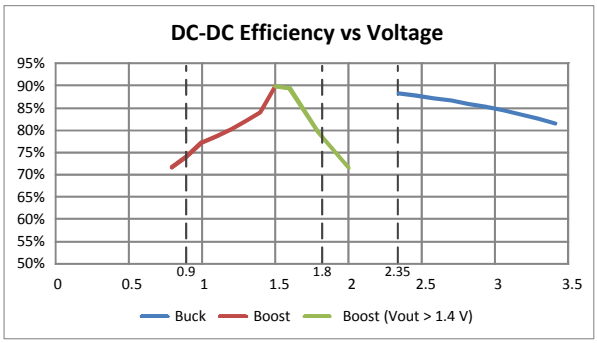


Figure 9: DC-DC Efficiency in Buck/Boost Mode at Various Voltage Levels

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