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

Smart Technology. Delivered.™

Datasheet

BT900-SA-0x, BT900-SC-0x

Intelligent BTv4.0 Dual-Mode Module

Version 1.12

Revision History

Version	Date	Notes	Approver
1.0		Initial Version	Jonathan Kaye
1.1	13 Feb 2015	Added system clock and tick count period table.	Jonathan Kaye
1.2	24 Feb 2015	Edits to clarify OTA app download works over VSP (command mode)	Jonathan Kaye
1.3	01 July 2015	Updated SPP range to reflect *Up to 600 kpbs*	Ben Whitten
1.4	21 July 2015	Added Tape and Reel information	Maggie Teng
1.5	15 Oct 2015	Updated SIG Qualification section	Jonathan Kaye
1.6	19 Nov 2015	Updated Reel photos with correct labels	Maggie Teng
1.7	01 July 2015	Converted from HIG to Datasheet; changed to new template Minor updates throughout. Added section on BLE vSP.	Raj Khatri
1.8	30 Aug 2016	Updated Declarations of Conformity	Sue White
1.9	01 Feb 2017	Fixed error to Pin Definition table; removed *Do not connect* from the Comment column of Pin 39	Raj Khatri
1.10	03 May 2017	Updated the Declaration of Conformity with new RED standards	Jonathan Kaye
1.11	04 May 2017	Fixed typo in DoC	Sue White
1.12	31 Oct 2017	Fixed error in FCC/IC regulatory section	Jonathan Kaye

CONTENTS

1 Overview and Key Features

BT900 Series modules from Laird make it easy to add Classic BT and Bluetooth Low Energy (BLE) functionality to small, portable, power-conscious devices, including those powered by batteries. The fully approved, programmable modules feature Laird's innovative, event-driven *smart*BASIC programming language, which significantly reduces OEM development risk and speeds time to market.

Based on the Cambridge Silicon Radio (CSR) 8811 silicon and a low power Cortex M3 microcontroller, the BT900 modules provide exceptionally low power consumption with outstanding wireless range, all within a compact footprint of 19 mm x 12.5 mm. The modules incorporate all the hardware and firmware required to support development of Dual Mode applications, including:

- Complete radio hardware
- UART, I2C, SPI, ADC, and GPIO interfaces
- Embedded BTv4.0 software stack
  - Classic BT profile - SPP
  - GATT Client and Peripheral modes

What makes the modules truly innovative is *smart*BASIC, an event-driven programming language that enables standalone operation of the module. Laird has extended the implementation of *smart*BASIC from the popular BL6xx series of single mode BLE modules into the BT900 series. This allows developers the flexibility of utilising the Core and BLE specific *smart*BASIC functions from the BL6xx series to create fully interchangeable BLE applications between these product ranges.

Without the need for any external processor, a simple *smart*BASIC application encapsulates the complete end-to-end process of reading, writing, and processing of sensor data and then using Classic Bluetooth or BLE to transfer it to / from any Bluetooth device. Ultimately *smart*BASIC accelerates initial development, creation of prototypes, and mass production by providing you with your own Bluetooth expert within the module.

In addition to carrying FCC modular, IC, CE and MIC approvals, BT900 modules are fully qualified as a Bluetooth product, enabling designers to integrate the modules in devices without the need for further Bluetooth testing. A low-cost developer's kit including simple software tools simplifies module integration and guarantees the fastest route to market.

Features and Benefits

- Bluetooth v4.0 - Dual Mode (Classic Bluetooth and BLE)
- External or Internal Antennas
- smartBASIC programming language
- Full Bluetooth EPL
- Compact Footprint
- Programmable TX power 8 dBm to -20 dBm
- RX sensitivity: -90 dBm
- Ultra low power consumption
- TX: 85 mA peak (at +8dBm)
- Standby Doze: 2.8 mA (see Power Consumption Note 2)
- Deep Sleep: 2.7 uA
- UART, GPIO, ADC, PWM, FREQ output, TIMERS, I2C, and SPI interfaces
- Fast Time to Market
- FCC, CE, IC, and Japan certified; other certs on request
- No external components required

Application Areas

- Medical devices
- Wellness devices
- Automotive Diagnostic Equipment
- Bar Code Scanners
- Industrial Cable Replacement
- Home automation

2 SPECIFICATIONS

2.1. Specification Summary

Table 1: Specifications

Categories	Feature	Implementation
Wireless Specification	Bluetooth®	V4.0 – Dual-Mode
	Frequency	2.402 - 2.480 GHz
	Transmit Power	+ 8 dBm (maximum) Configurable down to -20 dBm
	Receive Sensitivity	-90 dBm (typical)
	Link Budget	98 dB
	Raw Data Rates (Air)	3 Mbps (Classic BT – BR/EDR)
Host Interface and Peripherals	UART Interface	TX, RX, CTS, RTS DTR, DSR, DCD, RI can be implemented in *smart*BASIC- using General Purpose I/O Default 115200, N, ,8, 1 From 1,200 to 921600 RX buffer size (1024 bytes)
	GPIO	18 (maximum – configurable) lines. O/P drive strength (4 mA) Pull-up resistor (33 KOhms) control (via *smart*BASIC) Read pin-level
	I2C Interface	1 (configurable from GPIO total). Up to 400 kbps
	SPI	1 (configurable from GPIO total). Up to 4 Mbps
	ADC Interface	2 channels (configured from GPIO total). Up to 12-bit resolution Conversion time 2.0uS (at VCC 2.7V to 3.6V) Reference voltage AVCC (external, same as VCC) pre-scaling to match BL600 ADC
	PWM or FREQ output	Output a PWM or FREQ on up to 3 GPIO output pins. PWM output duty cycle: 0%-100% PWM output frequency: 500 kHz FREQ output frequency: 0 MHz to 4 MHz (50% duty cycle)
	Wi-Fi-BT coexistence	3 dedicated pins
Profiles	Classic Bluetooth	SPP (Serial Port Profile) – Up to 600 kbps
	Bluetooth Low Energy	GATT Client & Peripheral – Any Custom Services
Maximum Connections	Classic Bluetooth Bluetooth Low Energy	7 clients 5 clients
Programmability	*smart*BASIC	On-board programming language similar to BASIC
	*smart*BASIC application	Via UART or Over the Air

Categories	Feature	Implementation
Control Protocols		Any that can be implemented using *smart*BASIC vSP – Virtual Serial Port for BLE – select Command Mode or Bridge Mode.
FW upgrade	*smart*BASIC runtime engine FW upgrade	Via UART
Coexistence	802.11 (Wi-Fi)	3 wire CSR schemes supported (Unity-3 for classic BT, Unity-3e for BLE)
Operating Modes	Self-contained Run Mode	Selected by nAutoRUN pin status: LOW (0V). Then runs $autorun$ (*smart*BASIC application) if it exists.
	Interactive Development Mode	HIGH (VCC). Then runs via at+run (and "file name" of *smart*BASIC application script).
Supply Voltage	Supply	1.8V – 3.6V (Note 6) 1.8V operation not supported in current FW. 3.3V operation only (2.8V-3.6V).
Power Consumption	Current	Max Peak Current (TX Power @ +8 dBm TX): 85 mA
		Standby Doze (waitevent) – 2.8mA (at 4MHz clock) (Note 5)
		Deep Sleep – 2.7 uA (external signal wakeup) See Note 5
	User Configurable Clocking	User configurable clocking (40MHz, 20MHz, 4MHz), so user can reduce current consumption further.
Physical	Dimensions	19 mm x 12.5 mm x 2.5 mm; Pad Pitch 0.8 mm
Environmental	Operating	-40°C to +85°C
	Storage	-40°C to +85°C
Miscellaneous	Lead Free	Lead-free and RoHS compliant
	Warranty	One Year
Development Tools	Development Kit	Development board and free software tools
Software Tools	Utilities	Windows, Android and iOS applications UART Firmware Upgrade
Approvals	Bluetooth®	Complete Declaration ID
	FCC / IC / CE / MIC	All BT900 Series

Module Specification Notes:

Note 1 DSR, DTR, RI, and DCD can be implemented in the *smart*BASIC application.

Note 2 With I2C interface selected, pull-up resistors on I2C SDA and I2C SCL ***must*** be connected externally as per I2C standard.

Note 3 SPI interface (master) consists of SPI MOSI, SPI MISO, and SPI CLK. SPI CS is created by using any spare SIO pin within the *smart*BASIC application script allowing multi-dropping.

Module Specification Notes:

Note 4	The BT900 module comes loaded with *smart*BASIC runtime engine firmware but does not come loaded with any *smart*BASIC application script (as that is dependent on customer-end application or use). Laird provides many sample *smart*BASIC application scripts covering the services listed. Additional BLE services are being added every quarter.
Note 5	Deep sleep consumes 2.7uA of power when the BT900 internal radio chip 32.768kHz is used. The *smart*BASIC runtime engine firmware has SIO (DIO default function) input pins that are PULL-UP enabled by default. You may disable the internal PULL_UP through a *smart*BASIC application script.
Note 6	1.8V operation not supported in current *smart*BASIC runtime engine FW. 3.3V operation only (2.8V-3.6V).

3 Hardware Specifications

3.1. Block Diagram and Pin-out

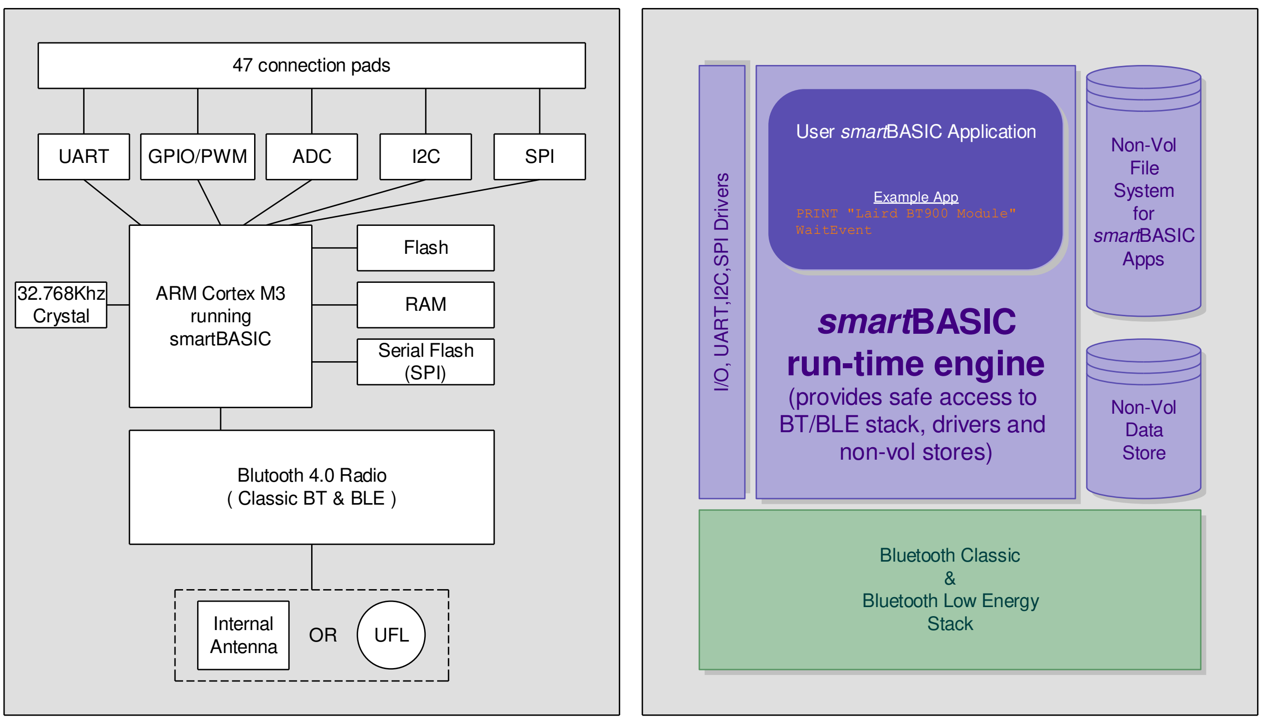


Figure 1: Functional HW and SW block Diagram for BT900 series Dual-Mode BT/ BLE smartBASIC module

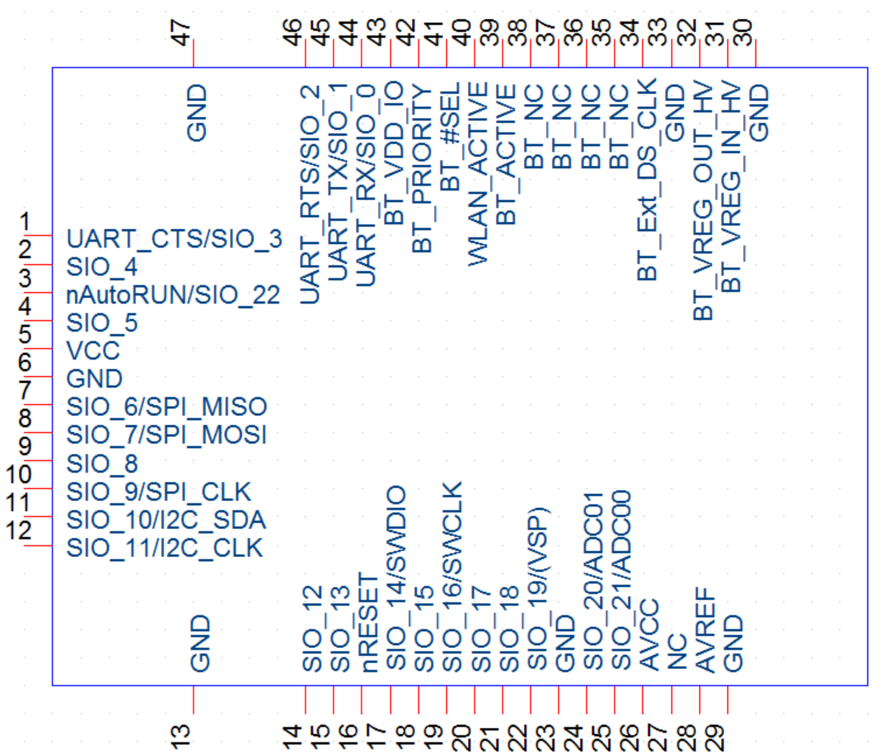


Figure 2: BT900-Sx module pin-out (top view)

3.2. Pin Definitions

Table 2: Pin definitions

Pin #	Pin Name	Default Funct.	Alternate Funct.	Default Direction	Supply Domain	Internal Pull-up or Pull-down State	Notes	Comment
1	UART_CTS	UART	SIO_3 or WKUP4 or Ext Interrupt	IN	VCC	Pull-up	1, 2, 6, 7, 12	
2	SIO_4	DIO		IN	VCC	Pull-up	2	Laird Devkit: UART_DTR via CON12
3	nAutoRUN		SIO_22 or Ext Interrupt	IN	VCC	Pull-up	In ONLY 1, 2, 12	Laird Devkit: UART_DSR via CON12
4	SIO_5	DIO	Ext Interrupt	IN	VCC	Pull Up	1, 2, 12	Laird Devkit: UART_DCD via CON12
5	VCC			IN	1.75V-3.6V	See Table 4	16	
6	GND	-	-	-	-	-	-	-

Pin #	Pin Name	Default Funct.	Alternate Funct.	Default Direction	Supply Domain	Internal Pull-up or Pull-down State	Notes	Comment
7	SIO_6	DIO	SPI MISO	IN	VCC	Pull Up	1, 2, 6, 9	SPIOPEN() in *smart*BASIC selects SPI function, MOSI and CLK are outputs when in SPI master mode. See Note 9
8	SIO_7	DIO	SPI MOSI	IN	VCC	Pull Up	1, 2, 6, 9	
9	SIO_8	DIO	Ext Interrupt	IN	VCC	Pull Up	1, 2, 12	Laird Devkit: UART_RI via CON12 or SPI_CS via CON16
10	SIO_9	DIO	SPI CLK	IN	VCC	Pull Up	1, 2, 6, 9	
11	SIO_10	DIO	I2C SDA	IN	VCC	Pull Up	1, 2, 6,	I2COPEN() in *smart*BASIC selects I2C function
12	SIO_11	DIO	I2C SCL	IN	VCC	Pull Up	1, 2, 6,	
13	GND	-	-	-	-	-	-	-
14	SIO_12	DIO	FREQ or PWM	IN	VCC	Pull Up	1, 2, 13	Laird Devkit: Buzzer output via CON15
15	SIO_13	DIO	FREQ or PWM	IN	VCC	Pull Up	1, 2, 13	Laird Devkit: Button1 input
16	nRESET			IN	VCC	Pull Up	8	System Reset (Active low)
17	SIO_14	DIO	*****	IN	VCC	N/A	2, 14	
18	SIO_15	DIO		IN	VCC	Pull Up	2	
19	SIO_16	DIO	*****	IN	VCC	N/A	2, 14	
20	SIO_17	DIO	FREQ or PWM	IN	VCC	Pull Up	1, 2, 13	Laird Devkit: LED1 via CON14
21	SIO_18	DIO		IN	VCC	Pull Up	2	Laird Devkit: LED2 via CON14
22	SIO_19	DIO	VSP	IN	VCC	Pull Up	1, 2, 10	Pull to GND externally (at power-up) to enter VSP Command mode (enable OTA functionality)
23	GND	-	-	-	-	-	-	-

Pin #	Pin Name	Default Funct.	Alternate Funct.	Default Direction	Supply Domain	Internal Pull-up or Pull-down State	Notes	Comment
24	SIO_20	DIO	AIN (ADC01) or WKUP1 or Ext Interrupt	IN	VCC	Pull Up	1, 2, 3, 4, 12	Laird Devkit: Button 2 input; Trim Pot via CON14
25	SIO_21	DIO	AIN (ADC00)	IN	VCC	Pull Up	1, 2, 3, 4	Laird Devkit: Temp Sensor input via CON14
26	AVCC			IN	1.7V-3.6V	See Table 4	16	
27	NC	NC						Reserved for future use. Do NOT connect.
28	AVREF			IN		See Table 4	16	
29	GND	-	-	-	-	-	-	-
30	GND	-	-	-	-	-	-	-
31	BT_VREG_IN_HV			IN only	3.3V	See Table 4	16	
32	BT_VREG_OUT_HV	DIO		IN only	1.8V	See Table 4	16	
33	GND	DIO	-	-	-	-	-	-
34	BT_Ext_DS_CLK	DIO		IN	BT_VDD_IO	Weak Pull-down		Do not connect
35	BT_NC	DIO		OUT	BT_VDD_IO	Weak Pull-down		Do not connect
36	BT_NC	DIO		OUT	BT_VDD_IO	Weak Pull-down		Do not connect
37	BT_NC			OUT	BT_VDD_IO	Weak Pull-down		Do not connect
38	BT_NC	NC		IN	BT_VDD_IO	Weak Pull-down		Do not connect
39	BT_ACTIVE	DIO		OUT	BT_VDD_IO	Weak Pull-down	17	
40	WLAN_ACTIVE	DIO		INs	BT_VDD_IO	Weak Pull-down	17	Also called WLAN_DENY
41	BT_#SEL	DIO		IN	BT_VDD_IO	Weak Pull-down	11	Must add 100K to GND externally
42	BT_PRIORITY	DIO		OUT	BT_VDD_IO	Weak Pull-down	17	Also called BT_STATUS

Pin #	Pin Name	Default Funct.	Alternate Funct.	Default Direction	Supply Domain	Internal Pull-up or Pull-down State	Notes	Comment
43	BT_VDD_IO			IN only	3.3V or 1.8V	See Table 4	16	
44	UART_RX	DIO	SIO_0 or WKUP2	IN	VCC	Pull-up	1, 2, 6, 7, 12, 15	UARTCLOSE() selects DIO functionality and UARTOPEN() selects UART comms behaviour
45	UART_TX	DIO	SIO_1	OUT	VCC	Set high in FW	1, 2, 6, 7, 15	
46	UART_RTS	DIO	SIO_2	OUT	VCC	Set low in FW	1, 2, 6, 7, 15	
47	GND	-	-	-	-	-	-	-

Module Pin Notes:

Note 1 Alternate function is selectable in the *smart*BASIC application.

Note 2 DIO – Digital Input or Output. I/O voltage level tracks VCC

Note 3 AIN – Analog Input.

Note 4 DIO or AIN functionality is selected using the GpioSetFunc() function in *smart*BASIC.

Note 5 AIN configuration selected using GpioSetFunc() function.

Note 6 I2C, UART, SPI controlled by xxxOPEN() functions in *smart*BASIC

Note 7 SIO_0 to SIO_3 are DIO by default when $autorun$ app runs on power up.

Note 8 Pull the nRESET pin low for minimum 500 nS in order for the BT900 to reset.

The BT900 module start-up time is ~1.6 seconds. Start-up time is the time taken from power-up to being able to run a *smart*BASIC command. Out of this, 1.6 seconds, ~1.3 seconds is for radio initialisation. 1.6 seconds is also the time when coming out of reset through AT command (ATZ) or AT command for factory default (at&f*).

For robustness against external interference, you must fit an external pull-up resistor (10K) on nRESET (pin 16) to VCC for BT900 to be out of reset. By default, the module is out of reset (internal weak-pull-up, 33k) when power is applied to the VCC pin

Note 9 SPI CS is created by the customer using any spare SIO pin within their *smart*BASIC application script allowing multi-dropping.

Note 10 It is possible to download smart BASIC applications Over the Air (OTA) to the BT900. To enable this feature, SIO_19 must be pulled low to GND externally (on power up). Refer to the firmware release documentation for details.

Note 11 You must connect 100 K pull-down resistor on BT_#SEL externally to GND.

Note 12 UART_CTS (pin 1), UART_RX (pin 44) and SIO_20/ADC01 (pin 24) are WKUP (wake-up) pins that allow the BT900 module to be woken up from Deep Sleep by the host. *smart*BASIC function will be added in the future to allow you to select which WKUP pin (or all) from which to wake up.

Module Pin Notes:

Note 13	PWM output signal is an alternative function on SIO_12, SIO_13 and SIO_17. FREQ output signal is an alternative function on SIO pins SIO_12, SIO_13, SIO_17. Up to three SIO pins are allowed to output FREQ signal or PWM signal. Refer to *smart*BASIC User Guide for details.
Note 14	It is mandatory that you specifically set script SIO_14 and SIO_16 as either input or output in your *smart*BASIC application to make SIO_14 and SIO_16 as GPIO's.
Note 15	*smart*BASIC runtime engine firmware has DIO (default function) input pins that are PULL-UP enabled by default. You can disable internal PULL_UP through your *smart*BASIC application script All the SIO pins (with a default function of DIO) are mostly inputs (unless stated otherwise in Table 2) – with no internal pull-up. SIO_1 and SIO_2 are outputs: ▪ SIO_1 (alternative function UART_TX) is an output, set high (in FW) ▪ SIO_2 (alternative function UART_RTS) is an output, set low (in FW) ▪ SIO_0 (alternative function UART_RX) is an input, set with internal ▪ SIO_3 (alternative function UART_CTS) is an input, set with internal pull-up ▪ SIO_19 is an input, needs an external pull-down. It is used for download *smart*BASIC applications over-the-air. See the latest FW release documentation for details.
Note 16	1.8V operation not supported in current *smart*BASIC runtime engine FW hence Customer must operate BT900 from nominal 3.3V supply (2.8V-3.6V, refer to Table4, note4). To operate BT900 from 3.3V connect the external 3.3V supply to pin 31 (BT_VREG_IN_HV), pin 5 (VCC), and pin 43 (BT_VDD_IO). Customer **MUST** leave pin 32 (BT_VREG_OUT_HV) unconnected.
Note 17	Dedicated BT900 BT-WiFi coexistence pins for CSR scheme Unity3 (used for classic BT) and Unity3e (used for BLE). Refer to *smart*BASIC user manual for details on how to enable coexistence.

The BT900 module is delivered with the integrated *smart* BASIC runtime engine FW loaded (but no onboard *smart*BASIC application script). Because of this, it starts up in AT command mode by default.

At reset, all SIO lines are configured as the defaults shown above.

SIO lines can be configured through the *smart* BASIC application script to be either inputs (with pull-ups or none) or outputs. When an alternative SIO function is selected (such as I2C or SPI), the firmware does not allow the setup of internal pull-up. Therefore, when I2C interface is selected, pull-up resistors on I2C SDA and I2C SCL **MUST** be connected externally as per I2C standard.

UART_RX, UART_TX, UART_CTS are 3.3 V level logic (if VCC is 3.3 V, i.e. SIO pin I/O levels track VCC). For example, when RX and TX are idle, they sit at 3.3 V (if VCC is 3.3 V). Conversely, handshaking pins CTS and RTS at 0 V are treated as assertions.

Pin 3 (nAutoRUN) is an input, with active low logic. In the development kit (DVK-BT900-sx) it is connected so that the state is driven by the host’s DTR output line. The nAutoRUN pin must be externally held high or low to select between the following two BT900 operating modes:

- Self-contained Run mode (nAutoRUN pin held at 0 V).
- Interactive / development mode (nAutoRUN pin held at VCC).

*smart*BASIC runtime engine firmware checks for the status of nAutoRUN during power-up or reset. If it is low and if there is a *smart*BASIC application script named **$autorun$**, then the *smart*BASIC runtime engine FW executes the application script automatically; hence the name S*elf-contained Run Mode*.

3.3. Electrical Specifications

3.3.1. Absolute Maximum Ratings

Absolute maximum ratings for supply voltage and voltages on digital and analogue pins of the module are listed below. Exceeding these values causes permanent damage.

The average SIO pin output current is defined as the average current value flowing through any one of the corresponding pins for a 100mS period. The total average SIO pin output current is defined as the average current value flowing through all of the corresponding pins for a 100mS period. The maximum output current is defined as the value of the peak current flowing through any one of the corresponding pins.

Table 3: Maximum Current Ratings

Parameter	Min	Max	Unit
Voltage at VCC pin	-0.3	+3.6	V
AVCC	VSS-0.5	VSS+4.6	V
AVREF	VSS-0.5	VSS+4.6	V
BT_VREG_IN_HV	2.3	4.8	V
BT_VREG_OUT_HV	1.7	2.0	V
BT_VDD_IO	-0.4	3.6	V
Voltage at GND pin		0	V
Voltage at SIO pin	-0.3	VCC+0.3	V
SIO “L” level average output current		4	mA
SIO “H” level average output current		-4	mA
SIO “L” level maximum output current		10	mA
SIO “H” level maximum output current		-10	mA
SIO “L” level total average output current		50	mA
SIO “H” level total average output current		-50	mA
SIO “L” level total maximum output current		100	mA
SIO “H” level total maximum output current		-100	mA
Storage temperature	-40	+85	ºC

3.3.2. Recommended Operating Parameters

Table 4: Power Supply Operating Parameters

Parameter	Min	Typ	Max	Unit
VCC (Note 1, Note4)	1.75	3.3	3.6	V
AVCC (AVCC=VCC) (Note 1)	1.75	3.3	3.6	V
AVREF[1] (when AVCC>=2.7V AVREF[1] (when AVCC<2.7V)	2.7V AVCC		AVCC AVCC	V
VCC Maximum ripple or noise (Note 2)			<10%of VCC	%
VCC rise time (0 to 1.8V) (Note 2)			0.1	mS
VCC shut down time (1.8V to 0V) (Note 2)			1	mS

Parameter	Min	Typ	Max	Unit
BT_VREG_IN_HV (Note 4)	2.3		3.6	V
BT_VREG_OUT_HV (Note 4)	1.75		1.95	V
BT_VDD_IO (Note 4)	1.2		3.6	V
Operating Temperature Range	-40	-	+85	ºC

Recommended Operating Parameters Notes:

Note 1 Notes on power on. Turn on/off in the following order or at same time.
Turning on: VCC > AVCC > AVRH. Turning off: AVRH > AVCC > VCC.
If not using the ADC convertor, connect AVCC and AVREF=VCC.
1.8V operation is not supported in current *smart*BASIC runtime engine FW, see Note 4.

Note 2 The maximum VCC ripple or noise (at any frequency) should not exceed 10% of VCC. Ensure transient fluctuation rate does not exceed 0.1V/uS.

Note 3 nRESET input time is minimum 500nS. Customer must fit an external pull-up resistor (10K) on nRESET (pin 16) to VCC for BT900 to be out of reset. BT900 module start-up time is ~1.6 seconds; start-up time is the time taken from power-up to being able to run a smart BASIC command. Most of this is for radio initialisation. 1.6 seconds is also the time when coming out of reset through AT command (atz) or AT command for factory default (at&f*).

Note 4 The Bluetooth chip in the BT900 has two internal regulators, a high voltage (input pin BT_VREG_IN_HV) and low voltage (input pin BT_VREG_OUT_HV) regulator. ONLY ONE regulator MUST be used to power the radio chip.

- **Method 1:** If the BT900 is required to operate from 3.3V, connect the external 3.3V supply (2.8V-3.6V) to pin 31 (BT_VREG_IN_HV), pin 5 (VCC), and pin 43 (BT_VDD_IO). Customer **MUST** leave pin 32 (BT_VREG_OUT_HV) unconnected.
- **Method 2:** If the BT900 is required to operate from 1.8V, connect the external 1.8V supply (1.75V-1.95V) to pin 32 (BT_VREG_OUT_HV), pin 5 (VCC) and pin 43 (BT_VDD_IO). Customer **MUST** leave pin 31 (BT_VREG_IN_HV) unconnected.

Note that 1.8V operation is not supported in current *smart*BASIC runtime engine FW.

Table 5: Signal Levels for Interface, SIO

Parameter	Condition	Min	Typ	Max	Unit
VIH Input high voltage	VCC < 2.7V VCC ≥ 2.7V	0.7VxCC 0.8VxCC		VCC+0.3 VCC+0.3	V
VIL Input low voltage	VCC < 2.7V VCC ≥ 2.7V	VSS-0.3		0.3xVCC 0.2xVCC	V V
VOH Output high voltage (std. drive, 4mA) See Note 1	VCC < 2.7V VCC ≥ 2.7V	VCC-0.45 VCC-0.5		VCC VCC	V V
VOL Output low voltage (std. drive, 4mA)	VCC < 2.7V VCC ≥ 2.7V	VSS VSS		0.4 0.4	V V
Pull up resistance	VCC < 2.7V VCC ≥ 2.7V	- 21	- 33	134 66	kΩ kΩ
Input capacitance			5	15	pF

Signal Levels for Interface, SDIO Notes:

Note 1	mA is the total average SIO pin output current which is defined as the average current value flowing through all of the corresponding pins for a 100mS period.

Table 6: SIO pin alternative function AIN (ADC) specification

Parameter	Min	Typ	Max	Unit
AVCC (AVCC = VCC)	1.75	3.3	3.6	V
AVCC current draw (ADC 1 unit operation)		0.27	0.42	mA
AVCC current draw (ADC stop)		0.03	10	uA
AVREF (when AVCC ≥ 2.7V	2.7 V		AVCC	V
AVREF (when AVCC < 2.7V)	AVCC		AVCC	V
AVREF current draw (ADC 1 unit operation)		0.72	1.29	mA
AVREF current draw (ADC stop)		0.02	2.6	uA
ADC input pin (AIN) voltage maximum	VSS		AVREF	V
ADC input port (AIN) current draw			5	uA
Time required to convert single sample 12 bit mode	2		10	uS
ADC input resistor impedance (during operation) (Note 1)				
AVCC ≥ 2.7V			2.2	kOhm
1.8V ≥ AVCC < 2.7V			5.5-10.5	kOhm
ADC input capacitance impedance (during operation)[1]			9.4	pF

SIO Pin Alternative Function AIN (ADC) Specification Notes:

Note 1	ADC input impedance is estimated mean impedance of the ADC (AIN) pins. The ADC is highly sensitive to the impedance of the source. The ADC (AIN) input impedance is 2.2-10.5k. Normally, when not sampling, the ADC (AIN) impedance will have very high value and can be considered an open circuit. The moment ADC is sampling, ADC(AIN) impedance is 2.2-10.5k.

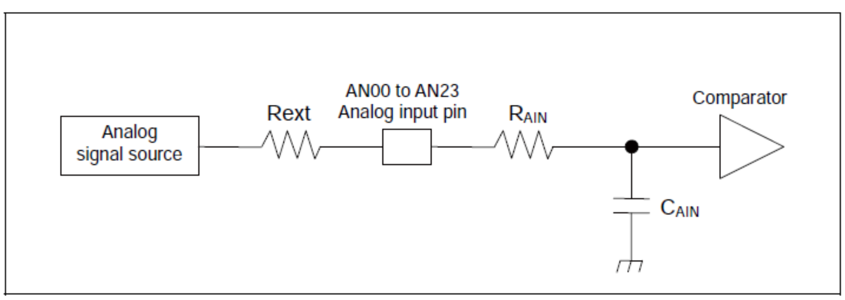


Figure 3: ADC Diagram

R_{ext}: Output impedance of external circuit (kOhms)

R_{ext}: Sampling time (nS)	
$T_s \geq (R_{AIN} + R_{ext}) \times C_{AIN} \times 9$	
R_{AIN}: Input resistor of ADC(kOhms)=2.2kOhms at 2.7V≤AVCC≤3.6V Input resistor of ADC(kOhms)=5.5kOhms at 1.8V≤AVCC≤2.7V	
C_{AIN}: Input capacity of ADC(pF)=9.4pF at .8V≤AVCC≤3.6V	

You ***must*** fit an external series resistor (R_{ext}) when using ADC pins, whose value is selected to get required Sample Time (T_s). 1K to 10K may be suitable.

Table 7: Digital I/O characteristics (ONLY those BT900 IO pins with names beginning with "BT_")

Normal Operation	Min	Typ	Max	Unit
Input Voltage				
VIL input logic level low	-0.4	-	0.4	V
VIH input logic level high	0.7 x BT_VDD_IO	-	BT_VDD_IO+ 0.4	V
Output Voltage				
VOL output logic level low, IOL = 4.0 mA	-	-	0.4	V
VOH output logic level high, IOL = 4.0 mA	0.75 x BT_VDD_IO	-	-	V
Input and Tristate Currents				
Strong pull-up	-150	-40	-10	µA
Strong pull-down	10	40	150	µA
Weak pull-up	-5	-1.0	-0.33	µA
Weak pull-down	0.33	1.0	5.0	µA
CI input capacitance	1.0	-	5.0	pF

This table applies to those BT900 pins ONLY with names beginning with ***BT_***:

- BT_Ext_DS_CLK (pin 34)
- BT_NC (pin 35)
- BT_NC (pin 36)
- BT_NC (pin 37)
- BT_NC (pin 38)
- BT_ACTIVE (pin 39)
- WLAN_ACTIVE (pin 40)
- BT_#SEL (pin 41)
- BT_PRIORITY (pin 42)

Note: BT900 IO pins with names beginning with *BT_* internal pull-up and pull-down resistors are not user-configurable via the *smart*BASIC application.

3.3.3. nAutoRUN Pin and Operating Modes

Operating modes (refer to the *smart*BASIC manual for details):

- Self-contained mode
- Interactive / Development mode

Table 7: nAutoRUN pin

Signal Name	Pin No	I/O	Comments
nAutoRUN	3	I	Input with active low logic. Operating mode selected by nAutoRun pin status: If Low (0V), runs $autorun$ if it exists; If High (VCC), runs via at+run (and "file name" of application).

Pin 3 (nAutoRUN) is an input, with active low logic. In the development board (DVK-BT900-sx) it is connected so that the state is driven by the host's DTR output line. nAutoRUN pin needs to be externally held high or low to select between the two BT900 operating modes:

- Self-contained Run mode (nAutoRUN pin held at 0V).
- Interactive / Development mode (nAutoRUN pin held at VCC)

The *smart*BASIC runtime engine firmware checks for the status of nAutoRUN during power-up or reset. If it is low and if there is a *smart*BASIC application named \$autorun\$ then the *smart*BASIC runtime engine executes the application automatically; hence the name *self-contained run mode*.

3.3.4. OTA (Over the Air) *smart*BASIC application download

It is possible to download smart BASIC applications Over the Air (OTA) to the BT900. To enable this, SIO_19 must be pulled low to GND externally (on power up). OTA *smart*BASIC download is possible from a remote host when in vSP command mode only.

The OTA *smart*BASIC application download is useful because it allows the module to be soldered into an end product without pre-configuration; the application can then be downloaded over the air once the product has been pre-tested. It is the *smart*BASIC application that is downloaded over the air and NOT the firmware. Since this is primarily meant for production environments with multiple collocated programming stations, the transmit power is limited.

Table 8: VSP pin description

Signal Name	Pin No	I/O	Comments
SIO_19	22	I	Internal pull up (default). Enter VSP Command mode by externally pulling SIO_19 pin to GND at power-up. OTA functionality is enabled through VSP Command mode.

4 Power Consumption

The BT900 module has User configurable clocking (40 MHz, 20 MHz, 4 MHz), so user can reduce current consumption at expense of speed. The default is 40MHz. Please note that when using the 4MHz clock, the maximum supported board rate is 115200. This data was taken at VCC 3.3V and a temperature of 25°C.

4.1. Power Consumption across Clock Frequencies

Table 9: Power consumption at 40MHz, 20MHz, and 4 MHz

Parameter	At 40 MHz	At 20 MHz	At 4 MHz	Unit
	Typical	Typical	Typical	
Active Peak current (Note 1)				
TX only run peak current @TX power = +8 dBm	85	85	85	mA
TX only run peak current @TX power = +4 dBm	71	71	71	mA
TX only run peak current @TX power = 0 dBm	61	61	61	mA
TX only run peak current @TX power = -4 dBm	55	55	55	mA
TX only run peak current @TX power = -8 dBm	52	52	52	mA
TX only run peak current @TX power = -12 dBm	49	49	49	mA
TX only run peak current @TX power = -16 dBm	48	48	48	mA
TX only run peak current @TX power = -20 dBm	48	48	48	mA
RX only 'peak' current	TBD	TBD	TBD	
Low Power Mode 1				
Standby Doze (waitevent) (Note 2)	10.7	6.9	2.8	mA
Low Power Mode 2 (Note 3)				
Deep Sleep (Note 3**)**	2.7	2.7	2.7	uA
Classic BT Mode (Note 5)				
Inquiring Mode (AT+BTI)	23.9	19.5	6.4 (Note 6)	mA
Wait for Connection or Discoverable	33	30	25 (Note 6)	mA
BT900 Master Role (connection ACL) (Note 5)				
Connecting Mode (ATDxxx)	37.8	29.8	27 (Note 6)	mA
Connected Mode (No Data Transfer)	20.5	16.3	12.6 (Note 6)	mA
Connected Mode (Max Data Transfer)	31	19	12.9 (Note 6)	mA
BT900 Slave Role (connection ACL) (Note 5)				
Connecting Mode (ATDxxx)	42	38.5	32.6	mA
Connected Mode (No Data Transfer)	35.3	30.7	22.7 (Note 6)	mA

Parameter	At 40 MHz Typical	At 20 MHz Typical	At 4 MHz Typical	Unit
Connected Mode (Max Data Transfer)	30.4	22.6	11.2	mA
Inquiring (Note 5)				
Scan interval: 640 ms, Scan Window: 320 ms	18	18	Note 6	mA
Scan interval 1920 ms, Scan Window 960 ms	18	11		mA
BLE Mode				
Active Mode Average Current (Note 4)				
Advertising Average Current Draw				
Maximum with advertising interval (min) 20 ms	23.3	12.5	9.8	mA
Minimum with advertising interval (max) 10240 ms	10.6	6.7	2.5	mA
Connection Average Current Draw				
Maximum with connection interval (min) 8 ms	17.2	12.4	9.3	mA
with connection interval 68 ms	11.4	7.4	3.2	mA
Minimum with connection interval (max) 4000 ms	10.6	6.7	2.5	mA
Scanning (Note 5)				
Active Scan Interval = 80 ms Scan Window = 40 ms	40	34	28	mA

Power Consumption Notes:

Note 1 Peak current is the current seen only during the duration of radio activity burst where TX is on and transmit power in Table 9 is transmitted.

Note 2 Standby Doze is entered automatically (when a *waitevent* statement is encountered within a *smart*BASIC application script). In Standby Doze, all enabled peripherals remain on and may re-awaken the chip. The module wakes up from Standby Doze via an interrupt (such as a received character on the UART Rx line). The module wakes up every millisecond to service the interrupt. If the module receives a UART character from either the external UART or the radio, it wakes up.

Note 3 In Deep Sleep, everything is disabled and the only wake-up sources are reset and changed on pins on which sense is enabled. The current typical consumption is 2.7uA.

*smart*BASIC runtime engine firmware requires a hardware reset to come out of deep sleep. Firmware allows the module to transition from Deep Sleep to Standby Doze through GPIO signals through the reset vector. Enter Deep Sleep mode via a command in your *smart*BASIC application script.

Note 4 The BLE radio taken with a TX power of 8 dBm and all peripherals off (UART OFF after radio event), slave latency of 0 (in a connection).

Average current consumption depends on a number of factors including a TX power and VCC accuracy of 26 MHz and 32.768 kHz. With these factors fixed, the largest variable is the advertising or connection interval set. Factors include:

- **Advertising Interval range:**
  - 20 ms to 10240 ms in multiples of 0.625 ms for Advert type=ADV_IND and ADV_DIRECT_IND
  - 100 ms to 10240 ms in multiples of 0.625 ms for Advert type=ADV_SCAN_IND and ADV_NONCONN_IND

Power Consumption Notes:

- For advertising timeout, if the advert type is ADV_DIRECT_IND, the timeout is limited to 1.28 seconds (1280 ms).

For an advertising event…

- The minimum average current consumption is when the advertising interval is large 10240 mS (this may cause long discover times for the advertising event by scanners.
- The maximum average current consumption is when the advertising interval is small (around 20 ms).
- Other factors that are also related to average current consumption include the advertising payload bytes in each advertising packet, as well as whether the BT900 is continuously advertising or periodically advertising.

- **Connection Interval range:**
  - 7.5 ms to 4000 ms in multiples of 1.25 ms.

For a connection event…

- The minimum average current consumption is when the connection interval is large (around 4000 ms)
- The maximum average current consumption is with the shortest connection interval of 7.5 ms; no slave latency.

Other factors related to average current consumption include whether transmitting 6 packets per connection interval and if each packet contains 20 bytes (which is the maximum for each packet). An inaccurate 32 kHz master clock accuracy would increase the average current consumption.

Note 5	Average current measurement using a current shunt IC (on DVK-BT900) and an oscilloscope.
Note 6	At 4 MHz clocking, slower throughput.

5 Functional Description

The BT900 dual mode (BT/BLE) module is a self-contained Bluetooth Low Energy product and requires only power and a user's *smart*BASIC application to implement full BLE functionality. The integrated, high performance antenna combined with the RF and base-band circuitry provides the Bluetooth Low Energy wireless link, and any of the SIO lines provide the OEM's chosen interface connection to the sensors. The user's *smart*BASIC application binds the sensors to the BLE wireless functionality.

The variety of hardware interfaces and the *smart*BASIC programming language allow the BT900 module to serve a wide range of wireless applications, while reducing overall time to market and the learning curve for developing dual-mode BT/ BLE products.

To provide the widest scope for integration, a variety of physical host interfaces/sensors are provided. The major BT900 series module functional blocks described below.

5.1. Power Management (includes brown-out and power-on-reset)

Power management features:

- System Standby Doze/Deep Sleep modes.
- Brownout Reset
- Open/Close peripherals (UART, SPI, I2C, SIO's and ADC) with a command in a smartBASIC application script
- Pin wake-up system from Deep sleep

Power supply features:

- Supervisor HW to manage power on reset, brownout (and power fail).
- 1.8V to 3.6V operating supply range. 1.8V operation is not supported in current *smart*BASIC runtime engine FW.

5.2. Clocks and Timers

5.2.1. Clocks

The integrated high accuracy (+/-20 ppm) 32.768 kHz crystal oscillator provides protocol timing and helps with radio power consumption in the system Standby Doze/Deep sleep modes by reducing the time that the RX window must be open. Standard accuracy clocks tend to have lower accuracy +/-250 ppm.

The integrated high accuracy 26 MHz (+/-10 ppm) crystal oscillator helps with Radio operation and also helps reduce power consumption in the Active modes.

5.2.2. Timers

In keeping with the event driven paradigm of *smart*BASIC, the timer subsystem enables the writing of *smart*BASIC which allows the generation of future events based on timeouts.

- Regular Timer – There are eight built-in timers (regular timer) derived from a single multifunction timer clock which are controlled solely by *smart*BASIC functions. The resolution of the regular timer is dependent on the selected system clock frequency can be obtained from Table 10.

Table 10: System Clock and Tick Count Period

System Clock (MHz)	Tick Count Period (uS)
40	6.4
20	12.8
4	64

- Tick Timer – This is a 31-bit free running counter that increments every one millisecond. The resolution of this counter is dependent on the selected system clock frequency and can be obtained from Table 10.

Refer to the *smart*BASIC user guide for more information.

5.3. Memory for *smart*BASIC Application Code and Data

Up to approximately 48 Kb of data memory is available for the *smart*BASIC application script and up to 4 Kb is available for data.

5.4. RF

- 2402–2480 MHz Bluetooth 4.0 Dual Mode (BT and BLE); 1 Mbps to 3 Mbps over the air data rate.
- TX output power of +8 dBm programmable (via *smart*BASIC command) to -20 dBm in steps of four dB.
- Receiver (with integrated channel filters) to achieve maximum sensitivity -90 dBm @ 1 Mbps BLE or Classic BT, 2 Mbps, 3 Mbps).
- RF conducted interface available in 2-ways:
  - BT900-SA: RF connected to on-board antenna on the BT900-SA
  - BT900-SC: RF connected to on-board uFL RF connector on the BT900-SC

- Antenna options:
  - Integrated monopole chip antenna on the BT900-SA
  - External dipole antenna connected with to uFL RF connector on the BT900-SC.

5.5. UART Interface

The Universal Asynchronous Receiver/Transmitter (UART) offers fast, full-duplex, asynchronous serial communication with built-in flow control support (UART_CTS, UART_RTS) in hardware up to 2 Mbps baud. No parity checking, 8 data bits, and 1 stop bit are supported.

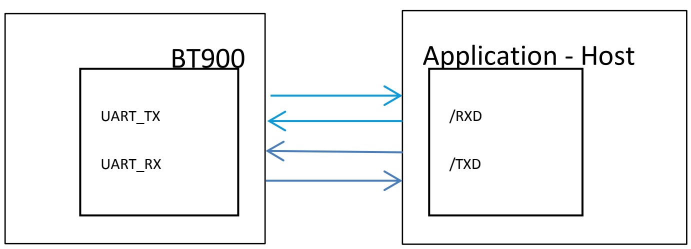
UART_TX, UART_RX, UART_RTS, and UART_CTS form a conventional asynchronous serial data port with handshaking. The interface is designed to operate correctly when connected to other UART devices such as the 16550A. The signalling levels are nominal 0 V and 3.3 V (tracks VCC) and are inverted with respect to the signalling on an RS232 cable.

Two-way hardware flow control is implemented by UART_RTS and UART_CTS. UART_RTS is an output and UART_CTS is an input. Both are active low.

These signals operate according to normal industry convention. UART_RX, UART_TX, UART_CTS, and UART_RTS are 3.3 V level logic (tracks VCC). For example, when RX and TX are idle they sit at 3.3 V. Conversely for handshaking pins CTS, RTS at 0 V is treated as an assertion.

The module communicates with the customer application using the following signals:

- Port/TXD of the application sends data to the module's UART_RX signal line
- Port/RXD of the application receives data from the module's UART_TX signal line



Note: The BT900 serial module output is at 3.3V CMOS logic levels (tracks VCC). Level conversion must be added to interface with an RS-232 level compliant interface.

Some serial implementations link CTS and RTS to remove the need for handshaking. We do not recommend linking CTS and RTS except for testing and prototyping. If these pins are linked and the host sends data when the BT900 deasserts its RTS signal, there is significant risk that internal receive buffers will overflow, which could lead to an internal processor crash. This drops the connection and may require a power cycle to reset the module. We recommend that you adhere to the correct CTS/RTS handshaking protocol for proper operation.

Table 11: UART Interface

Signal Name	Pin No	I/O	Comments
SIO_1 / UART_TX	45	O	SIO_1 (alternative function UART_TX) – Output, set high (in FW).
SIO_0 / UART_RX	44	I	SIO_0 (alternative function UART_RX) – Input, set with internal pull-up (in FW).
SIO_2 / UART_RTS	46	O	SIO_2 (alternative function UART_RTS) – Output, set low (in FW).
SIO_3 / UART_CTS	1	I	SIO_3 (alternative function UART_CTS) – Input, set with internal pull-up (in FW).

The UART interface is also used to load customer developed *smart* BASIC application script.

UART has a deep buffer (UART_RX deep buffer) of 1024 bytes.

5.6. SPI Bus

The SPI interface is an alternate function on SIO pins, configurable by *smart* BASIC.

The module is a master device that uses terminals SPI_MOSI, SPI_MISO, and SPI_CLK. SPI_CS is implemented using any spare SIO digital output pins to allow for multi-dropping. On DVK-BT900 devboard, SIO_8 is used at the SPI_CS.

The SPI interface enables full duplex synchronous communication between devices. It supports a 3-wire (SPI_MOSI, SPI_MISO, SPI_SCK,) bi-directional bus with fast data transfers to and from multiple slaves. Individual chip select signals are necessary for each of the slave devices attached to a bus, but control of these is left to the application through use of SIO signals. I/O data is double buffered.

The SPI peripheral supports SPI mode 0, 1, 2, and 3.

Table 12: Peripheral supports

Signal Name	Pin No	I/O	Comments
SPI_MOSI	8	O	This interface is an alternate function configurable by *smart* BASIC. Default in the FW pin 8 and 10 are inputs. SPIOPEN() in smart BASIC selects SPI function and changes pin 8 and 10 to outputs (when in SPI master mode). SPI_CS is implemented using any spare SIO digital output pins to allow for multi-dropping. On DVK-BT900 devboard, SIO_8 (pin9) is used at the SPI_CS.
SPI_MISO	7	I	
SPI_CLK	10	O	

5.7. I2C Interface

The I2C interface is an alternate function on SIO pins, configurable by *smart* BASIC command.

The two-wire interface can interface a bi-directional wired-OR bus with two lines (SCL, SDA) and has master/slave topology. The interface is capable of clock stretching. Data rates of 100 kbps and 400 kbps are supported.

An I2C interface allows multiple masters and slaves to communicate over a shared wired-OR type bus consisting of two lines which normally sit at VCC. The BT900 module can only be configured as an I2C master and can be the **only** master on the bus. The SCL is the clock line which is always sourced by the master; the SDA is a bi-directional data line which can be driven by any device on the bus.

IMPORTANT: It is essential to remember that pull-up resistors on both SCL and SDA lines are not provided in the module and MUST be provided external to the module.

Table 13: I2C Interface

Signal Name	Pin #	I/O	Comments
I2C_SDA	11	I/O	This interface is an alternate function on each pin, configurable by *smart*BASIC. I2COPEN() in *smart*BASIC selects I2C function.
I2C_SCL	12	I/O	

5.8. General Purpose I/O, ADC, PWM/FREQ and Host-wakeup

5.8.1. GPIO

The 18 SIO pins are configurable by *smart*BASIC and can be accessed individually. Each has the following user configured features:

- Input/output direction (output drive strength – 4mA).
- For inputs, Internal pull up resistors (33K typical) or no pull-up.

5.8.2. ADC

The ADC is an alternate function on SIO pins and is configurable by *smart*BASIC.

The BT900 provides access to 2-channel 12-bit incremental ADC. This enables sampling multiple external signals through a front end MUX. The ADC has configurable input.

Note: Current *smart*BASIC runtime engine firmware provides access to 12-bit mode resolution.

Analog Interface (ADC)

Table 14: Analog interface

Signal Name	Pin #	I/O	Comments
AIN – Analog Input	24	I	This interface is an alternate function on each pin, configurable by *smart*BASIC. AIN configuration selected using GpioSetFunc() function. 12 bit resolution.
AIN – Analog Input	25	I	

PWM and FREQ signal output on up to two SIO pins

The PWM and FREQ output is an alternate function on SIO pins and is configurable by *smart*BASIC.

The ability to output a PWM (Pulse Width Modulated) signal or FREQ output signal on up to three GPIO (SIO) output pins available via *smart*BASIC runtime engine firmware and can be selected using the *smart*BASIC commandGpioSetFunc().

PWM output signal has a frequency and duty cycle property. PWM output is generated using 32-bit hardware timers. The timers are clocked by a 4 MHz clock source. Frequency is adjustable (up to 1 MHz) and the Duty cycle can be set over range from 0% to 100% (both configurable by *smart*BASIC command).

Note: The frequency driving the two SIO pins is the same but the duty cycle can be independently set for each pin.

FREQ output signal frequency can be set over a range of 0 Hz to 4 MHz (with 50% mark-space ratio).

5.9. nRESET pin

Table 15: nRESET pin

Signal Name	Pin No	I/O	Comments
nRESET	16	I	BT900 HW reset (active low). Pull the nRESET pin low for minimum 500 nS in order for the BT900 to reset. By default, the module is out of reset (internal weak-pull-up, 33k) when power is applied to the VCC pin

Note: For robustness against external interference, you MUST fit an external pull-up resistor (10K) on nRESET (pin 16) to VCC for the BT900 to be out of reset. nRESET needs to be held low (0V) for greater than 500 nS to reset the module.

5.10. nAutoRUN pin

Refer to section *nAutoRUN pin and Operating Modes* regarding operating modes and the nAutoRUN pin.

- Self-contained Run mode
- Interactive/Development mode

5.11. *smart*BASIC Runtime Engine Firmware Upgrade

The BT900 software consists of the following:

- BT900 *smart*BASIC runtime engine firware (loaded at production, may be upgraded by the customer).
- BT900 *smart*BASIC application script developed by customer (loaded through UART by the customer).

To allow customer the capability to upgrade the BT900 *smart*BASIC runtime engine FW to the latest version released from Laird, the current *smart*BASIC runtime engine firmware only allows this upgrade via the UART.

5.12. Wake-up BT900

5.12.1. Waking up BT900 from Host

Wake-up the BT900 from the host using wake-up pins (UART_CTS, UART_RX, SIO_20 (ADC01)). Refer to the *smart*BASIC user manual for details. You may configure the BT900's wakeup pins via *smart*BASIC to:

- Wake up when signal is low
- Wake up when signal is high
- Wake up when signal changes

BT900 also has pins that are external interrupts; refer to the *smart*BASIC user manual for details.

5.12.2. Wake up Host from BT900

This may be done by use of the BT900 SIO pin. Refer to the *smart*BASIC user manual for details.

5.13. Low Power Modes

The BT900 has three power modes: Run, Standby Doze and Deep Sleep. Further, the BT900 has user configurable clocking (40MHz, 20MHz, 4MHz) allowing power consumption trade-off in Run and Standby Doze modes.

The module is placed automatically in Standby Doze if there are no events pending (when *waitevent* statement is encountered within a customer's *smart*BASIC script). The module will wake up from Standby Doze via an interrupt e.g. received character on the UART Rx line. The module wakes up every millisecond to service the interrupt. If the module receives a UART character from either the external UART or the radio, that will cause it to wake up.

Deep sleep is the lowest power mode. Once awakened, the system will go through a system reset.