1. Introduction

The WL102B is a highly integrated low power CMOS Wireless Data Controller, designed to dramatically reduce the cost of radio data applications. It forms the DE6038 chip set together with the WL600C receiver chip and WL800 transmitter chip to offer a complete solution for a frequency-hopping spread spectrum radio in the 2.4 to 2.5 GHz ISM band. It is also interfaces with the GPS DE6003 radio transceiver.

Its flexible design means that it can also be used in a wide range of other applications using a range of microprocessors, protocols and additional memory options, as well as radios at other frequencies. It is designed for low power and minimum cost systems, it is particularly suited for hand-held applications.

1.1 Features

- Complete Low Power CMOS Single-Chip Radio Transceiver Controller
- Embedded 8051 Processor for Control of Media Access Control (MAC) Protocol
- On-chip Dual Port 6.4kbyte Network Data Buffer
- On-chip 4kbyte System RAM
- Provision for External Processor, 64kbyte Data Buffer and System RAM
- Capability for In-system Re-programming of External Flash Code Memory
- 8-bit Host Processor Interface to Buffer, with Support for PCMCIA Interface
- Radio Transceiver Interface Provided by Generic Communications Control Block (CCB)
- Up to 1 Mbit/s with 2-level, or 2 Mbit/s with 4-level GFSK Operation
- Block Power-down Facility to Reduce Power
- Power Efficient Dedicated DMA Transfer Mode between Buffer RAM and CCB
- Available in 100 and 144 pin Thin Quad Flatpack Packages

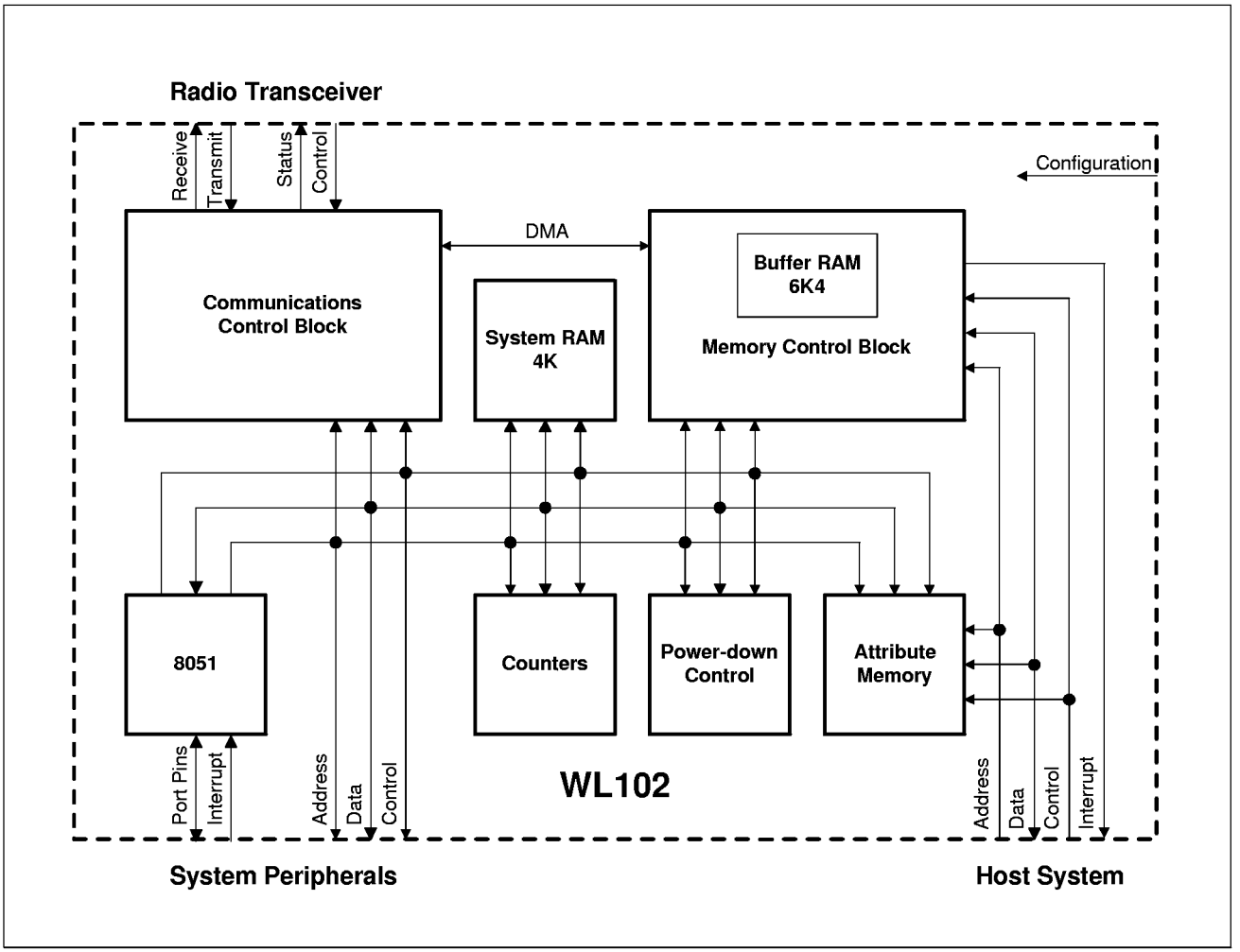

Figure 1. WL102 Block Diagram

1.2 WL102 System Overview

The WL102 is a highly integrated digital wireless transceiver controller suitable for controlling multiple channel radios with data rates up to 1Mbit/s 2-level data or 2Mbits/s 4-level data. It has been designed to interface easily with the WL600 and WL800 radio transceiver chips.

The use of low power synchronous on-chip RAM and selective power down of system blocks makes the WL102 ideally suited to power sensitive applications. Also, the WL102 can be operated from 3V or 5V supply rails with level translation between the Host and MAC system built in to the Host interface. A minimum configuration of the WL102 with a 64Kbtye external ROM for code storage makes it ideal for small footprint applications, and the option of 100 pin or 144 pin TQFP 1mm packages make it suitable for PC-Card applications.

It is envisaged that the WL102 hardware will allow the development of a wireless network to the Media Access layer of the IEEE model for networking, with the addition of an LLC layer providing the equivalent of the Data Link layer of the ISO network model. In such a system the Host interface buffer RAM would provide the boundary between the LLC layer running on the Host and the MAC layer running on the internal WL102 8051 processor, or an optional external MAC system processor.

Figure 1 shows the internal architecture of the WL102.

The WL102 provides an interface between a Host system and the radio transceiver, controlled by the MAC system processor.

The host interface, such as to a PC or PDA, can be either a PC-Card interface or an 8 bit microprocessor bus. The Memory Control Block (MCB) provides a buffer and interrupt signalling mechanisms between the Host and Mac system processors. The Attribute Memory provides PC-Card Card Information Structure (CIS) and Configuration Option Register (COR).

The Communications Control Block (CCB) interfaces with either the DE6038 (WL600/WL800) or the DE6003 radio transceivers. The CCB performs transmit and receive operations directly to the MCB buffer RAM, using a dedicated DMA bus, so minimising the load on the MAC system processor.

The MAC system processor is then only required to provide the timing and control functions of Media Access Control. A minimum controller system configuration would be the WL102 plus an external code memory device.

Figure 2 shows the control and network packet data flows through the WL102. The network data packet transfer between the Radio Transceiver and the Host system is directly between the CCB and the MCB Host Buffer RAM. The system processor provides the management control; it has direct access to the network management packets.

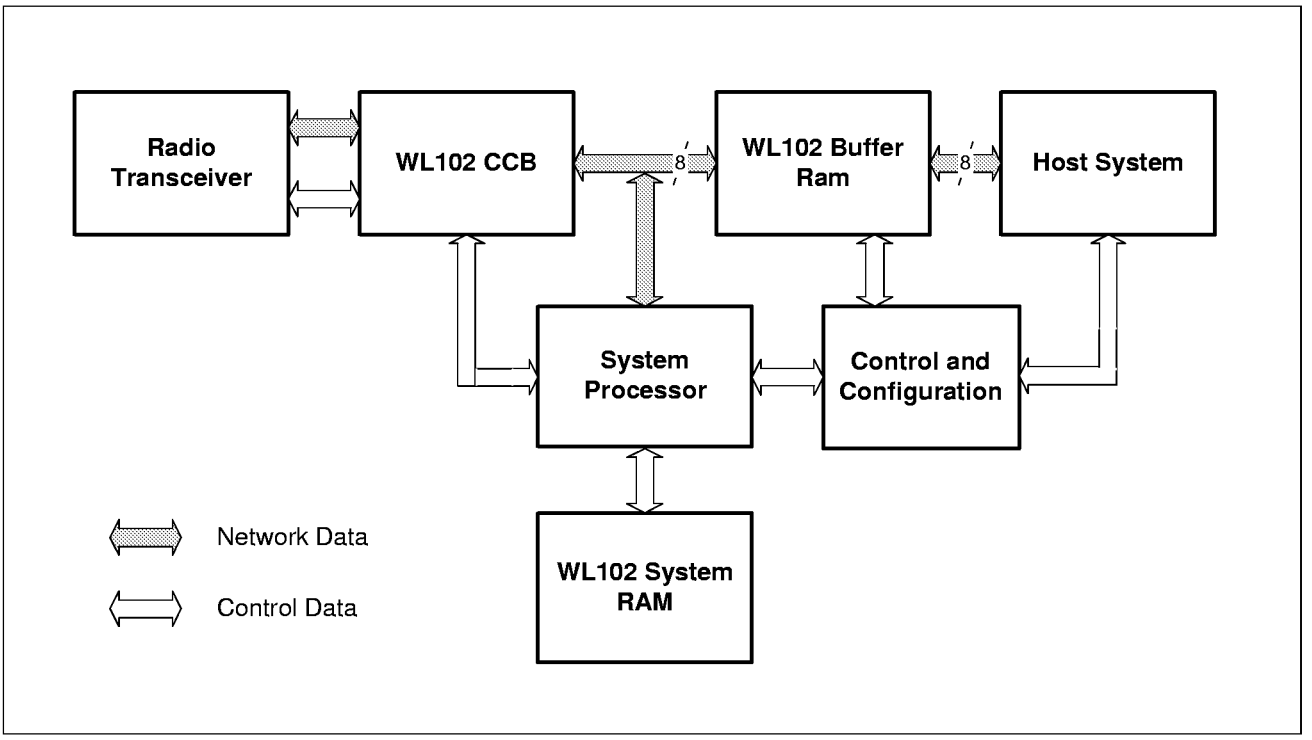

Figure 2. Data Flow in a WL102 Based System

1.3 WL102 Hardware Blocks

1.3.1 System Processor

The WL102 contains an embedded processor; identical in operation to an industry-standard 8051microcontroller, with the exception that the internal data memory is increased from 128 to 256 bytes. System busses and strobes allow the 8051 to access the full 64 kbytes of external code memory.

Internal connections exist to the processor's two interrupt inputs, and to the two timer inputs. Two I/O port pins are internally connected to blocks within the WL102, whilst four of the remaining I/O port pins are brought out to external pins, including the serial receive and transmit port pins.

Internal address decoding provides access to the other blocks of the device, with 36 kbytes of the processor's 64 kbytes address range being un-assigned and available for the addition of external peripheral devices.

The 8051 demultiplexed address and data bus are available externally to allow for the addition of extra memory or other peripheral devices.

The 8051 is clocked at the internal system clock frequency of 10MHz.

If additional processing power is required for a particular application, the WL102 may also be used with an external processor instead of the internal 8051 as shown in Figure 3. This external processor also accesses the WL102 blocks via the internal address decode. Interrupt and Counter outputs are brought to external pins, and two control input signals must also be provided to the WL102. The internal 8051 is disabled to save power.

1.3.2 Data RAM

The WL102 contains 4 kbytes of static RAM which the processor may use for variable storage etc. If additional storage space is required, an external static RAM may also be used, and mapped to an unused area of the WL102 memory map.

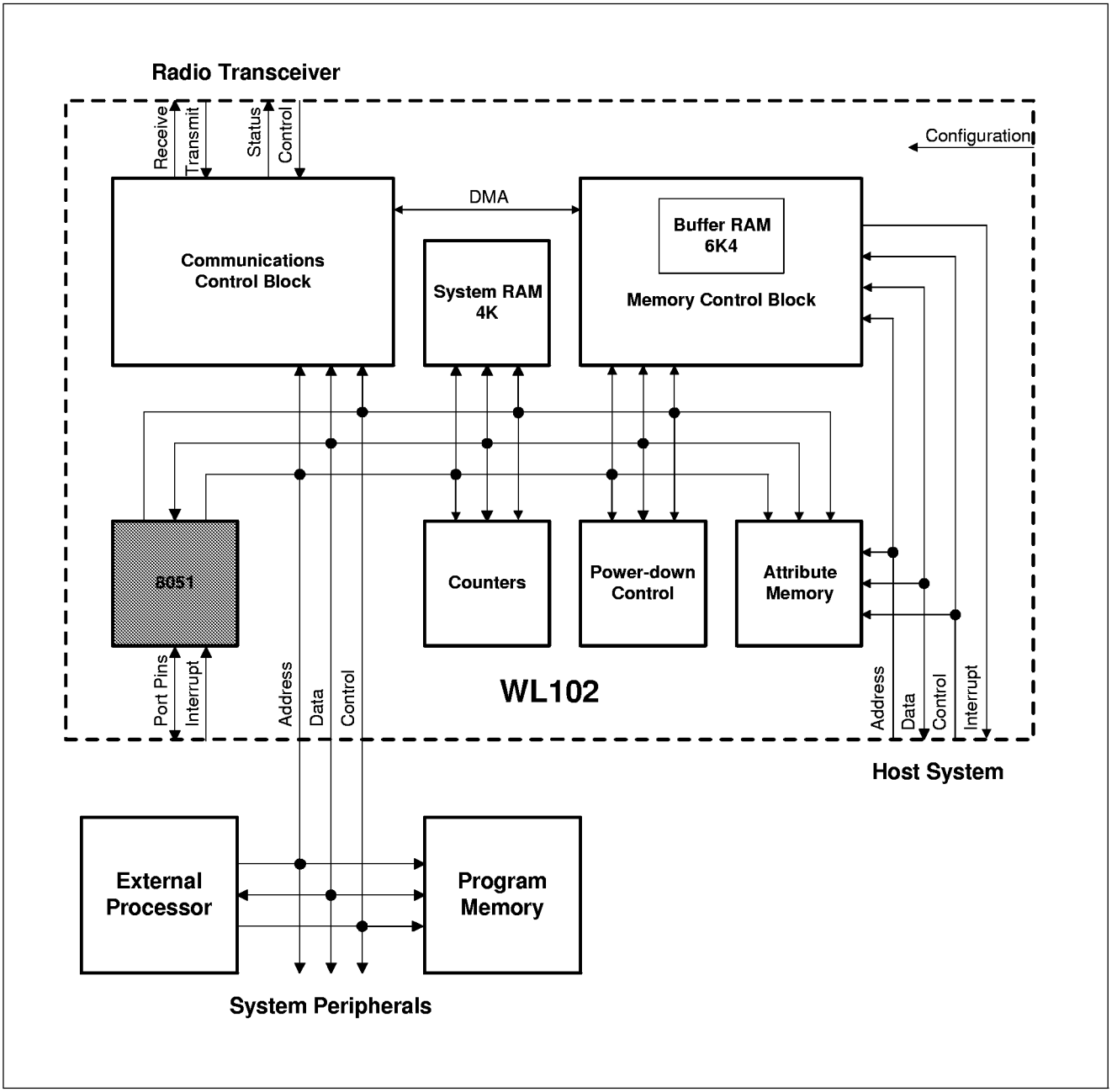

Figure 3. WL102 Using External Processor

1.3.3 Memory Control Block

The Memory Control Block (MCB) allows access to the dual port buffer used to communicate between the Host and WL102 System processes. The control logic of the buffer RAM allows the Host System to asynchronously read or write data at the same time as the WL102 MAC system. To help arbitrate the access to this buffer space, a hardware semaphore system is also included.

The WL102 contains 6784 bytes of low power buffer RAM on chip. The 144-pin package option also allows for an external (single port) SRAM to be used to increase the buffer RAM space, as shown inFigure 4, up to 64KBytes. The MCB provides access to the Buffer RAM, which is accessible by both the Host and the MAC systems simultaneously, and also to a control register which allows the Host to reset or interrupt the MAC processor and performs some other control functions on the WL102.

Packets of data are typically buffered in the buffer RAM before being transmitted or transferred from the MAC system to the host.

The MCB has been designed to be flexible enough to allow its use from small microprocessor systems to PC-Card slots. In a minimum configuration it uses only 4 address locations and standard microprocessor-type read or write cycles.

The 8-bit interface is microprocessor and PCMCIA-compatible. The MCB may be used by any microprocessor-type interface which can supply two bits of address, a chip enable signal, read/write strobes and 8 bits of data, subject to timing requirements of the WL102. The interface also provides an interrupt signalling mechanism between the Host and MAC system as well as some other control functions, such as a hardware semaphore and reset circuitry.

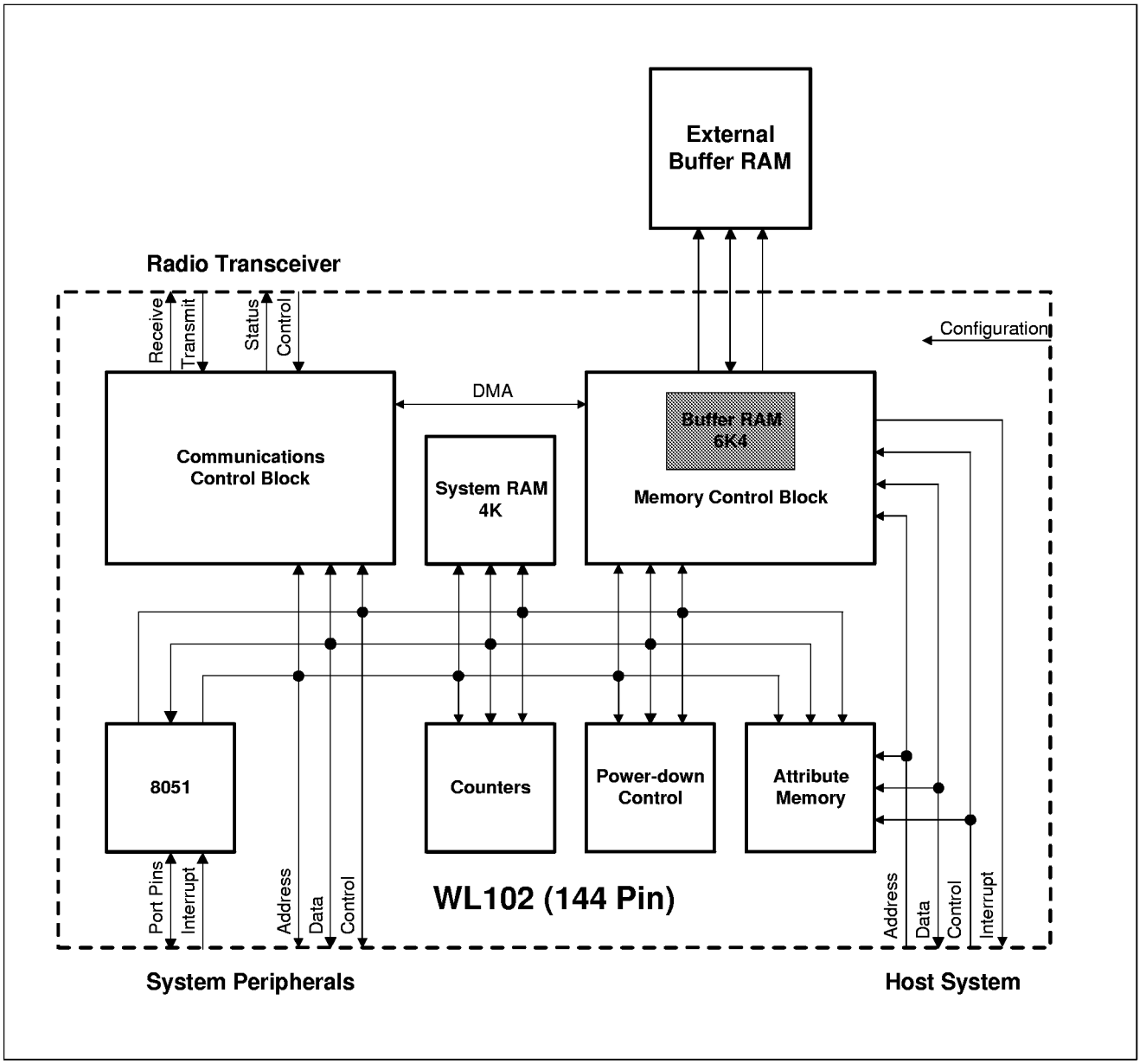

Figure 4. WL102 Using External Buffer RAM

1.3.4 Attribute Memory

The attribute memory is PCMCIA-compatible providing a dedicated 255 bytes RAM for Card Information Structure(CIS) and Configuration Option Register(COR) used for the configuration and control of a PC-Card.

1.3.5 Communications Control Block

This block of the WL102 performs many of the functions required for transmission and reception of data packets, and interfaces directly to the radio transceiver. For power sensitive applications the configuration registers can have their clocks switched off once the initial configuration is complete and for protocols which allow the transceiver to sleep for periods of time, the clocks to the entire block can be disabled via the Power Control register.

The CCB handles all the control signals for the radio and can be configured for the timings required by the transceiver being used. The CCB can directly access data stored in the buffer RAM, via a DMA bus separate to the processor bus. Once configured, the CCB will fully handle a transmission or reception, including CRC checking, address matching, optional data scrambling and Bias suppression encoding, and transfer of the data to/from the buffer RAM. For shorter network management packets the MAC system processor can directly read/write up to 64 bytes of data from/to the CCB FIFO.

The CCB generates maskable interrupts to the MAC system processor at defined points in the receive/transmit process to allow the processor to perform any additional processing required to successfully complete the receive or transmit sequence.

The high level of hardware integration dramatically reduces the load on the WL102 system processor and provides a number of configuration options to suit different wireless protocols.

The features of the CCB include:

Transmission and reception of 2-level and 4-level GFSK bit streams at 625Kbps, 1Mbps and 2Mbps
Configurable Preamble/Frame word generation and recognition
Checksum generation and validation (CRC-16 and CRC-32)
Optional data coding schemes: bit-stuffing, scrambling and bias suppression encoding (as per draft IEEE 802.11)
Dedicated data path for DMA transfer to and from buffer RAM
Address matching on received data packets
Analysis of received signal for performing clear channel assessment, including 16-bit countdown timer
8 (maskable) interrupt sources to optimise operation of the system software
Automatic synthesiser channel loading for Rx/Tx when using WL600/WL800.

1.3.6 Counters

The two 16-bit counters are provided which are configured by setting a reload value. When enabled, the counters decrement continuously at 10MHz, generating a terminal count signal to the processor and automatically reloading the reload value when the count reaches zero.

The terminal count outputs are internally connected to the 8051 Timer inputs when using the internal 8051, and to external pins when using an external processor.

1.3.7 Power Control

A power-down function is provided under the control of the WL102 system. Power-down enable signals are provided to each major block; the clocks of the individual blocks being disabled when the appropriate power-down bit is set. In addition, the internal processor's PCON register will disable the 8051 core in the usual manner.

1.3.8 System Supply Voltages

The WL102 has two separate digital supply rails, allowing for use in mixed 3V/5V systems. The host interface has its own independent supply, whilst a second supply powers the core of the WL102 and its interface to the memory and radio transceiver subsystems. Both power supplies may be operated at either 3 or 5 volts (nominal); with the restriction that the host interface supply voltage must be greater than or equal to the voltage powering the WL102 core and radio transceiver.

1.4 Manufacturing Enhancements

1.4.1 Radio Test Mode

The WL102 radio test mode is a mode of operation which allows an attached radio transceiver to be tested more easily in a complete assembled unit. In this mode the usual functions of the WL102 are suspended and the control and status signals of the CCB radio interface are connected to the pins of the Host interface. This allows the operation of the radio to be directly controlled by a test system connected to the Host interface, with the WL102 providing the status signals back from the radio interface. The comparators on the receive data stream are still used and so the test system is provided with the digital version of this radio output. In addition the operation of the CCB preamble and frame word recognition can be tested as well as the clock recovery and recovered data stream.

1.4.2 MCB Execute Mode

This mode of operation is used to allow programming of the external code memory if a re-programmable device is used, via special software run on the Host. Using this mode, the WL102's code memory may be completely reprogrammed regardless of its current state i.e. no 'boot-block' code is required in the WL102 system. This allows complete radio modules to be manufactured without the need for pre-programming the code memory. At final test the MAC software image can be downloaded or test software can be downloaded and used prior to the final software.

This facility also allows the completed radio module to be upgraded in the field or for diagnostic software to be downloaded as necessary.

2. Pin Descriptions

2.1 Host Interface Pins

All logic high and low levels correspond to HVdd and Gnd respectively.

Pin	Type	Description
HA7 - 0 Host Address Bus	Input (Pull down)	HA0 through HA7 are the host address bus input lines.
HD7 - 0 Host Data Bus	Input /Output (Pull down)	HD0 through HD7 are the bi-directional host data bus.
HREGB PCMCIA Attribute Memory / I/O Select	Input (Pull Up)	HREGB is an active-low input which selects attribute memory and I/O space. The WL102 does not respond to common memory access (when HREGB is high.)
HRDB I/O Read Strobe	Input (Pull Up)	The HRDB input is made active to read data from the chip's I/O space. The HREGB and HCEB inputs must also be active for I/O transfer to take place.
HWRB I/O Write Strobe	Input (Pull Up)	The HWRB input is made active to write data to the chip's I/O space. The HREGB and HCEB inputs must also be active for the I/O transfer to take place.
HOEB PCMCIA Attribute Memory Read Strobe	Input (Pull Up)	HOEB is an active-low output enable input. HOEB is used to gate attribute memory read data from the WL102 onto the HD bus.
HWEB PCMCIA Attribute Memory Write Strobe	Input (Pull Up)	HWEB is an active-low write enable input. HWEB is used to strobe attribute memory write data into the WL102 from the HD bus.
HCEB Chip Enable	Input (Pull Up)	The HCEB line is an active-low chip enable.
HRESET System Reset	Input (Pull Up)	HRESET is an active-high input that resets the entire WL102.
HRDY (HIREQ) PCMCIA Ready / Interrupt Request	Output	HRDY (PCMCIA Memory Mode) is an active-high output. On reset the WL102 sets this output low until the system microprocessor sets it high after configuring the CIS memory. HIREQ (PCMCIA I/O Mode) is an active-low output. HIREQ is used to indicate to the host that software service is required. The control of this pin is described in Section 6

Pin	Type	Description
HWAITB Extend Bus Cycle	Output	The HWAITB output is asserted by the chip to delay completion of the access currently in progress.
HVdd Host Interface Supply		HVdd is the Host Interface power supply pin. The voltage on this pin must be greater than or equal to that on SVdd. If a 5 volt supply is used, the LOWVDD bit of the host MCB control register should be set to '0', or the configuration Option register bit should be set to '1'

2.2 System Interface Pins

All logic high and low levels correspond to SVdd and Gnd respectively, unless otherwise stated.

Pin	Type	Description
SA15 - 0 *Internal Processor Mode* System Memory Address Bus	Output	SA15 through SA0 are the demultiplexed system address lines.
External Processor Mode System Address Bus	Input	SA15 through SA0 are the system address lines.
SD7 - 0 *Internal/ External Processor Mode* System Memory Data Bus	Input/ Output (Pull Down)	SD7 through SD0 are the bi-directional system data lines
SRDB *Internal Processor Mode* System Data Memory Read Strobe	Open Drain Output (Pull Up)	SRDB is an active-low data memory read strobe.
External Processor Mode System Read Strobe	Input	SRDB is an active-low data output enable input.
SWRB *Internal Processor Mode* System Data Memory Write Strobe	Output (Pull Up)	SWRB is an active-low data memory write strobe.
External Processor Mode System Write Strobe	Input	SWRB is an active-low write strobe.

Pin	Type	Description
CCEB *Internal / External Processor Mode* Program Memory Enable	Output	CCEB is an active-low program memory chip enable.
SPSENB *Internal Processor Mode* System Program Memory Read Strobe	Open drain Output (Pull Up)	SPSENB is an active-low program memory read strobe.
External Processor Mode System Program Memory Read Strobe	Input (Pull Up)	SPSENB is an active-low program memory read strobe. This input is used to generate CCEB, and is also used to enable the MCB Execute mode operation.
SWAKEUP / EPTCNT2 *Internal Processor Mode* Wakeup Interrupt	Input	SWAKEUP is internally inverted and ORed with the CCB interrupt to drive the INT0 input of the internal 8051.
External Processor Mode Counter2 Terminal Count	Output	EPTCNT2 is the active-high terminal count output of Counter2.
P1[3] / EPCINTR *Internal Processor Mode* 8051 Port 1.3	Open Drain I/O	P1[3] is the Port1 bit 3 open drain I/O of the internal 8051.
External Processor Mode CCB Interrupt	Open Drain Output	EPCINTR is the active-low CCB interrupt output.
P1[5] / EPTCNT1 *Internal Processor Mode* 8051 Port 1.5	Open Drain I/O	P1[5] is the Port1 bit 5 open drain I/O of the internal 8051.
External Processor Mode Counter1 Terminal Count	Open Drain Output	EPTCNT1 is the active-high terminal count output of Counter1.
P3[1] / EPMINTR *Internal Processor Mode* 8051 Port 3.1	Open Drain I/O (Pull Up)	P3[1] is the Port3 bit 1 open drain I/O of the internal 8051.
External Processor Mode MCB Interrupt	Open Drain Output (Pull Up)	EPMINTR is the active-low MCB interrupt output

Pin	Type	Description
RSSISTRB *Internal / External Processor Mode* External ADC Strobe for RSSI	Output	RSSISTRB is an active-low read strobe generated by a read from the RSSI Strobe memory area.
SYSCLK_X *External Processor Mode* 10MHz Reference Clock	Output	SYSCLK_X is the 10MHz system clock output. This clock is generated from the SYSCLK_D (10MHz) or SYSCLK (20MHz) input. In External processor mode SWRB, SRDB, SPSENB must be synchronised to this clock (see timing diagrams.)
Internal Processor Mode		Disabled
P3[0] EPSYSINC *Internal Processor Mode* 8051 Port 3.0	Input	P3[0] is the Port 3 bit 0 input of the internal 8051
External Processor Mode MCB System Increment	Input	EPSYSINC is an active-high input which enables auto-increment for the system MCB address counter. In addition it controls the HRDY output as detailed in Section 6
EPRSTB *External Processor Mode Only* Processor Reset	Output	EPRSTB is an active-low reset output, generated from the internal reset logic. Following a host reset it is guaranteed to be 64 clocks long. This pin is also controlled by the Host MCB control register bit 0, allowing the host to reset the system microprocessor independently of the rest of WL102 chip.
EPRDYB *External Processor Mode Only* Control of HRDY(HIREQ)	Input	EPRDYB is used by an external processor to control the HRDY (HIREQ) output. During external processor reset its function is disabled, after which EPRDYB controls the HIREQ signal as detailed in Section 6.
SYSCLK 20MHz System Clock	Analogue Input	SYSCLK is the 20MHz single-ended system clock input. If this input is used SYSCLK_D must be tied to Gnd.
SYSCLK_GND 20MHz System Clock Ground	Analogue Input	SYSCLK_GND is the 20MHz system clock ground
SYSCLK_D 10MHz System Clock	Input	SYSCLK_D is the 10MHz digital system clock input. If this input is used SYSCLK pin must be tied to Gnd and SYSCLK_GND must be tied to SVdd.
EBA7 - 0 External Buffer RAM Address Bus	Output (Pull down)	EBA0 through EBA7 are the external buffer RAM address outputs. These pins are only available on the 144-pin package

Pin	Type	Description
EBD7 - 0 External Buffer RAM Data Bus	Input /Output (Pull Down)	EBD0 through EBD7 are the bi-directional external buffer RAM data bus. These pins are only available on the 144-pin package.
EBRDB External Buffer RAM Read Strobe	Output	EBRDB is an active low read strobe for the external buffer RAM. This pin is only available on the 144 pin package.
EBWRB External Buffer RAM Write Strobe	Output	EBWRB is an active low write strobe for the external buffer RAM. This pin is only available on the 144 pin package.
EBCEB External Buffer RAM Chip Enable	Output	EBCEB is an active low chip enable signal for the external buffer RAM. This pin is only available on the 144 pin package.
ENIBRAMB Enable Internal Buffer RAM	Input (Pull Down)	ENIBRAMB is an active-low input which enables the internal buffer RAM. This pin must be connected to SVdd if an external buffer RAM is used.
ENI8051B Enable Internal 8051	Input	ENI8051B is an active-low input which enables the internal 8051 processor. This pin must be connected to the Host Interface supply HVdd or Gnd.
THREEV System 3 Volt Supply Mode	Input	The THREEV input must be set according to the SVdd voltage; '1' = 3 volt operation, '0' = 5 volt operation.
TEST1-0 Enable Test Mode	Input	Test0 and Test1 are used to enable the WL102 test modes, and must be set to '0' for normal operation. These pins must be connected to the Host Interface supply HVdd or Gnd.
SVdd System Interface Supply		SVdd is the System power supply pin. The THREEV pin must be connected according to the voltage on this pin. When using the WL102 with the DE6003 radio, SVdd (and HVdd) must be 5 volts.
Gnd Ground		

2.3 Radio Transceiver Interface Pins

All logic high and low levels correspond to SVdd and Gnd respectively.

Pin	Function	Description
RXD_L & RXD_LB *4-Level Modulation Mode* LSB of Receive Data *2-Level Modulation Mode*	Analogue Input	The Least Significant Bit of the decoded symbol from the demodulated signal. These pins are un-used and must be tied to opposite

Pin	Function	Description
		known states.
RXD_H & RXd_HB *4-Level Modulation Mode* MSB of Receive Data	Analogue Input	The Most Significant Bit of the decoded symbol from the demodulated signal.
2-Level Modulation Mode Serial Receive Data	Analogue Input	Receive Data from the demodulated signal. Note: If this differential input is used the RXD_D input must be tied to SVdd.
RXD_D *2-Level Mode Only* Serial Receive Data	Input	Receive Data input. Note: If this input is used the RXD_L and RXD_H must be tied to SVdd and RXD_HB and RXD_LB must be tied to Gnd
TXD_L *4-Level Modulation Mode Only* LSB of Transmit Data	Output	The Least Significant Bit of the Transmit data.
2-Level Modulation Mode. Not Used.		
TXD_H / TXD *4-Level Modulation Mode* MSB of Transmit Data	Output	The Most Significant Bit of the Transmit data.
2-Level Modulation Mode 4-Level Modulation Mode Only		Transmit Data output. This pin connects to the WL800 TXD input via a 20KOhm resistor
DE6003 Mode Serial Transmit Data	Output	Transmit Data output.
TX/RXB / RX/TXB *DE6038 Mode* Transmit / Receive Control	Output	TX/RXB is used to control the Transmit or Receive state of the radio. ('1' = Transmit, '0' = Receive).
DE6003 Mode Receive / Transmit Control	Output	RX/TXB is used to control the Transmit or Receive state of the radio. ('1' = Receive, '0' = Transmit).
PA_ON PA On / Off Control	Output	Power Amplifier control. ('1' = On, '0' = Off)
LCK_DETB Lock Detect	Input	LCK_DETB is an active-low input which indicates that the radio's on-board synthesiser is in lock.
STDBYB Radio Power Down Control	Output	STDBYB is an active-low output which puts the DE6003/DE6038 into standby mode.

Pin	Function	Description
MOD2LEVB *DE6038 Mode* Modulation Mode Control *DE6003 Mode* Un-used, remains at 2-Level Modulation.	Output	MOD2LEVB switches the transceiver between 2 Level and 4-Level modulation modes.('0' = 2-Level, 1' = 4-Level).
CS_CLK / ANT_SEL *DE6038 Mode* Channel Select Clock *DE6003 Mode* Antenna Select	Output Output	Clocks channel select data into synthesiser A & M registers. Selects between Antenna1 & Antenna2. ('0' = ANT1, '1' = ANT2)
CCAB Clear Channel Assessment	Input	This input signal can control the operation of the CCA timer and is reflected in the CCB status register (21h).
CS_DATA / PWR_LOB *DE6038 Mode* Channel Select Serial Data *DE6003 Mode* High/Low Transmit Power Select	Output Output	Channel select data for the WL800 synthesiser registers. PWR_LOB selects between 10mW and 100mW output power ('0' = 10mW, '1' = 100mW).
CS_LOADB / LOADB *DE6038 Mode* Channel Select Latch Enable *DE6003 Mode* Channel Load Strobe	Output Output	CS_LOADB is held low to enable channel load. Data is latched on rising edge. LOADB latches parallel channel data into the synthesiser.
CSD6 - 0 *DE6038 Mode* Un-used. *DE6003 Mode* Channel Select Data	Output	CSD0 through CSD6 are the parallel channel select data. These pins are only available on the 144-pin package, and are powered from the HVdd power supply.
AVdd System Analogue Supply		AVdd is the System analogue power supply pin. The voltage on this pin must be equal to that on SVdd.
AGnd System Analogue Ground		AGnd is the System Analogue Ground.

2.4 Device Pin Out

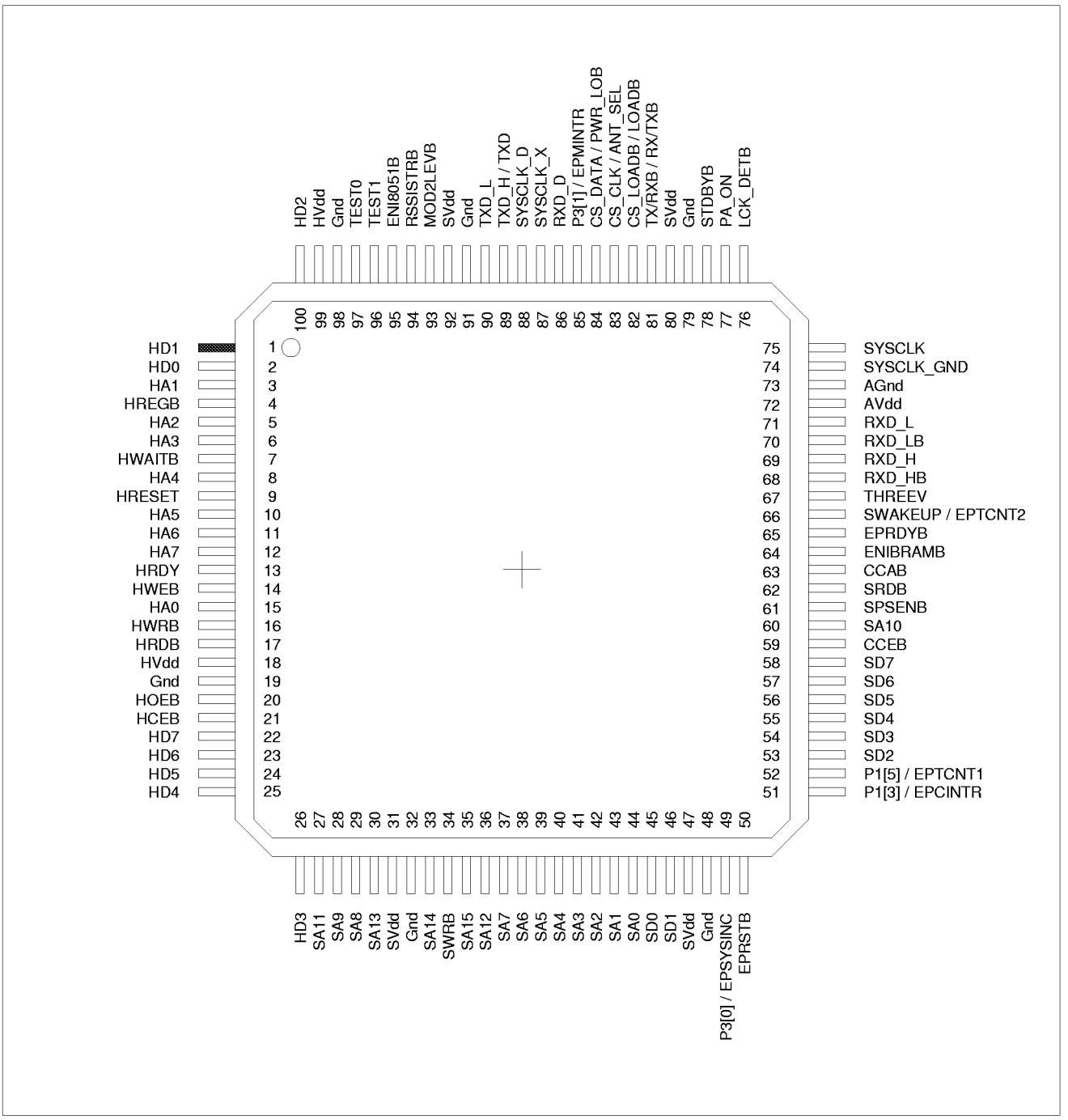

Figure 5. WL102 100 Pin Package Pin Out

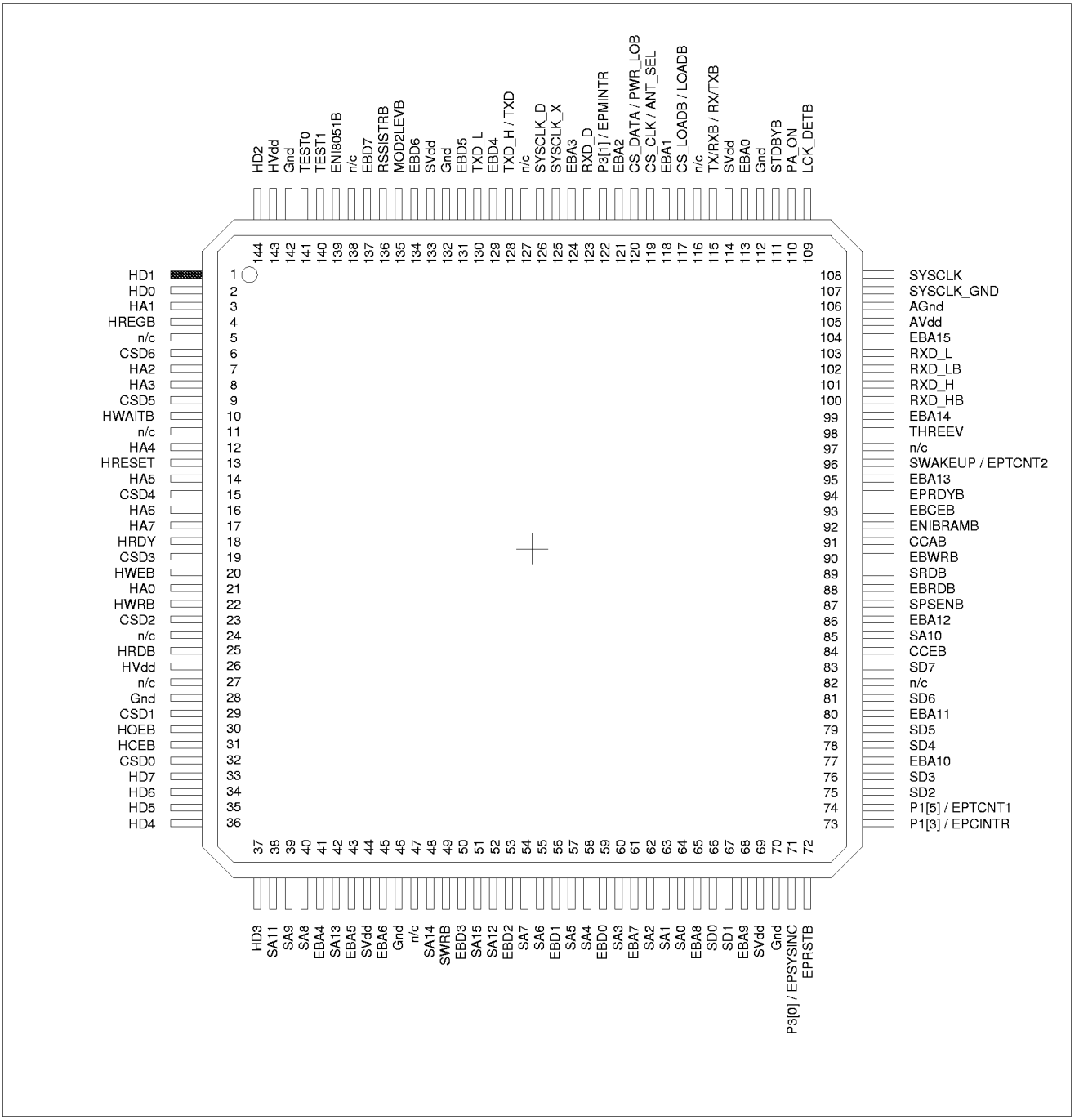

Figure 6. WL102 144 Pin Package Pin Out

3. Functional Block Description

3.1 Internal 8051 Processor

The WL102 contains an embedded 8051 microcontroller core, operating at a 10MHz clock frequency, and containing 256 bytes of internal RAM. Connections to the 8051 which are internal to the WL102 are:

ALE	Latches the high address byte.
Port1.4	Output only. Drives the RDYB/HIREQ logic.
Port1.2	Output only. Drives the MCB address increment control.
INT0 / Port3.2	Input only. Driven by logical OR of the CCB interrupt output and the SWAKEUP pin (inverted).
INT1 / Port3.3	Input only. Driven by the MCB interrupt output.
T0 / Port3.4	Input only. Driven by the Counter1 terminal count output.
T1 / Port3.5	Input only. Driven by the Counter2 terminal count output.

Available external connections to the 8051 (when in internal processor mode only) are:

Data bus and demultiplexed Address bus

Memory strobes

Port1.3

Port1.5

Port 3.1 / TxD

Port 3.0 / RxD (Input only)

Internal decode logic latches the processor address and provides enable signals to the blocks defined by the WL102 memory map.

3.1.1 System Memory Map in Normal Mode

All program memory accesses (SPSENB active) select the code memory enable (CCEB), in the range 0000 to FFFF. Data memory accesses (SRDB or SWRB active) are decoded as follows:

Address (hex)	Decoded block
0000 - 7FFF	Not assigned
8000 - 8FFF	MCB
9000 - 9FFF	Attribute Memory
A000 - BFFF	Internal Data RAM
C000 - CFFF	Power Control + Counters
D000 - DFFF	RSSISTRB output (on SRDB only)
E000 - EFFF	CCB
F000 - FFFF	Not assigned

Table 1. System Memory Map

3.2 System Memory Map in MCB Execute Mode

All program memory accesses (SPSENB active) select the MCB data register, whilst all data memory accesses (SRDB or SWRB active) select the code memory enable (CCEB).

3.3 Data RAM

The WL102 contains 4K bytes of static RAM, mapped into the data memory space of the processor (accessed by SRDB or SWRB cycles) which the processor may use for variable storage etc. If additional storage space is required, an external static RAM may also be used, and mapped to an unused area of the WL102 memory map (see 3.1.1).

3.4 Buffer RAM

The WL102 contains 6784 bytes of RAM, which is accessed via the Memory Control Block, providing a dual-port buffer area, accessible by the system processor and via the host interface. Alternatively, with the 144-pin package only, an external RAM may be used (up to 64K bytes). In this case, the internal buffer RAM must be disabled by connecting the ENIBRAMB pin to SVdd.

3.5 Power Control

A power-down function is provided under the control of the system processor. Power-down enable signals are provided to each major block. Clocks of individual blocks are disabled when the appropriate power-down bit is cleared. The reset condition is all powered up (ones).

3.6 Counters

The two 16 bit counters can be loaded by writing to address locations specified below. Both counters decrement until a terminal count of zero is detected. The terminal count then reloads the latest count start value and generates a 3.2µs active high pulse on the terminal count output. This output is connected to the 8051 Timer input (internal processor) or to the EPTCNT1/EPTCNT2 output pin (external processor). Each counter is stopped by resetting the "Enable counter" bit. The counters load the start value at each enable.

Registers are decoded as follows:

Address		Function
00	Bit 0	CCB power-down
	Bit 1	Not used
	Bit 2	Counter 1 power-down
	Bit 3	Counter 2 power-down
01	Bit 0	Enable Counter1
02		Counter 1 start value count high byte
03		Counter 1 start value count low byte
09		Enable Counter2
0A		Counter 2 start value count high byte
0B		Counter 2 start value count low byte

Table 2. Power Down / Counter Registers

4. Memory Control Block

4.1 I/O Address Space

The MCB I/O space allows access to the dual port buffer RAM used to communicate between the Host and System processes. Access to the buffer RAM is fully dual port, with the Host side being able to asynchronously read or write data at the same time as the system. This potentially leads to design issues when deciding on how to arbitrate between the Host and System having access to the buffer RAM space. Therefore a hardware semaphore system is also included is this block, although the semaphore does not electrically prevent or allow access to the buffer RAM. The WL102 has 6784 bytes of RAM on chip, or an external single port SRAM can be added for increased buffer RAM space, up to 64KBytes. Using external buffer RAM disables the internal RAM block.

The following address decode is used for external and internal access to the buffer RAM for each interface.

Host Access	System Access	Description
00	N/A	Control register
01	01	Address pointer (high byte)
02	02	Address pointer (low byte)
03	03	Data register
04	04	Semaphore

Table 3. Memory Control Block Register Addresses

In addition the system side of the MCB has the following signals:

INT1/EPMINTR	This is the output of Host control register bit 3
8051RESET/EPRSTB	This is the output of Host control register bit 0
Port1.2/EPSYSINC	This input controls auto increment of the system MCB address

4.1.1 Host Control Register

The Host control register is used to control the type of access to the buffer RAM, and also to control various miscellaneous aspects of the WL102 operation. The register contains the following bits:

Bit	Name	Reset Condition
0	SYS_RESET	1
1	HST_INC	0
2	MCB_EXECUTE	0
3	SYS_INT	1
4	CLEAR	0
5	IRQ_CLR	0
6	LOWVDD	1
7	SEMA4	1

Table 4. Host Control Register

4.1.1.1 SYS_RESET

This bit controls the Reset line of the internal 8051 block, when in internal 8051 mode, once the Power-on reset has completed. Therefore a '0' in this bit will hold the internal processor in the reset state. Care should be taken that sufficient clocks are allowed to correctly reset the internal 8051 (24 minimum, 2.4uS @ 10MHz). In external processor mode this bit will control the state of the EPRSTB signal directly.

4.1.1.2 HST_INC

This bit controls the operation of the Host MCB address counter. If this bit is high then the MCB address will automatically increment after each read/write cycle. If this bit is low then the address will not increment (although the data will be re-read/re-written).

4.1.1.3 MCB_EXECUTE

This bit is used to switch the address map of the WL102 so that instruction read cycles from the system processor will fetch data from the MCB buffer RAM. A high in this bit will switch to MCB Execute mode. In this mode any SPSENB strobe will cause a read of the MCB buffer RAM. The system address counter is set into auto-increment mode (regardless of the state of the internal 8051 Port1 bit6 or the EPSYSINC pin) and cannot be loaded by the system processor. In order to allow a Flash ROM to be written in this mode all system write cycles cause the CCEB ROM enable signal to become active.

In summary, in MCB Execute mode:

The system address counter is forced into increment mode.
The system counter cannot be loaded.
All SPSENB cycles cause MCB buffer RAM reads.
All SWRB cycles cause the CCEB signal to become active.

For a full description of the MCB Execute mode see Section 8.

4.1.1.4 SYS_INT

This bit is used to drive the internal processors INT1 pin, which can be configured in system software to be edge or level sensitive. In external processor mode this bit directly controls the EPMINTR pin.

4.1.1.5 IRQ_CLR

This bit controls the generation of interrupts from the WL102 to the Host. See Section 6 for a full description.

4.1.1.6 LOWVDD

This bit controls the slew rate of the Host interface outputs in order to minimise power and noise. A '0' is the slow 5V option, a logic '1' is the fast 3V option. On power up the WL102 defaults to the fast speed for the Host interface pins. See Section 11 for a full description.

4.1.1.7 SEMA4

This bit allows access to the semaphore logic, in addition to address 04 of the MCB block. The semaphore logic is a dual port lock which guarantees that only one port can have a successful lock at any time. In order to queue a lock request a '0' is written to this bit. Its value is then read back with a '0' indicating a successful lock and a '1' indicating a pending lock. If a '1' is returned the bit can be repeatedly polled until a lock is granted, or the queued lock request can be cancelled by writing a '1' to the bit. To release a granted lock a '1' is written to this bit.

4.2 Operation

The System and Host Address pointers are used to set the address of the buffer RAM access. Both pointers are read/write which allows for easier interrupt handling, since the address pointers can be read and restored around any interrupt handling code.

The Host and System Data registers are used to read and write data to or from the buffer RAM itself. If auto-increment is enabled, any read or write cycle to these registers causes an increment for the relevant address pointer.

For a write cycle the data written to the data register is used to program the location pointed to by the address pointer.

For a read cycle the data read is always for the previous address. This is a one cycle latency for the read cycle allowing a fast read time which forces the next byte to be pre-loaded into the read register between read accesses. This means that in order to read an address (or addresses) the following must be performed:

Switch on/off auto-increment for the port	Optional
Load the address pointers with the address required.	
Perform a dummy read cycle on the data register.	The value read will be for the address held before the address pointers were loaded - so discard.
Perform a read cycle.	The data read will be for the address loaded into the address pointers.
Perform a read cycle.	If increment is on the data read will be from the address loaded + 1. If increment was off then the address will be re-read. If the other port has modified the data at this address then this will be reflected in the data read.

In this way the MCB buffer RAM can be polled at a particular location or a packet of data can be read or written.

It is envisaged that buffer management can be performed by allocating a portion of the buffer RAM to be lockable, by the semaphore, in software. This will allow the memory in the buffer to be allocated dynamically or statically as the particular system configuration requires.

5. Communications Control Block

5.1 Introduction

The CCB interfaces directly with the radio transceiver. It has a number of configuration options which allow it to use a variety of transmission/reception protocols. It has principally been designed to be compatible with both the DE6003 radio and the DE6038 radio.

The CCB is configured and controlled by either the internal 8051 micro-controller or an external processor (in the remainder of this section the word 'processor' will apply to either controller). After the processor switches the CCB into receive/transmit, the CCB directly transfers received/transmitted data to/from the MCB. The functions performed by the CCB are described below, together with some recommendations on how to configure the CCB.

5.2 Generic Packet Format

The CCB has principally been designed to work with data packets in the format illustrated in Figure 7.

The 'preamble' consists of a Sync. Word, repeated several times, followed by a Frame Word. The 'header' must contain the data length field, which can be between 8 and 16 bits long. The HCS (Header Checksum) and FCS (Frame Checksum) fields are CRC generated codes which the CCB can add in order to detect errors in the received data.

Whenever possible, the sub-blocks have been designed to work in any configuration. However, some blocks expect certain information in order to work properly. In particular, the 4-level blocks have only been designed to work with an IEEE 802.11 type of packet (see Section 5.11 for further details on restrictions for 4-level transmission/reception).

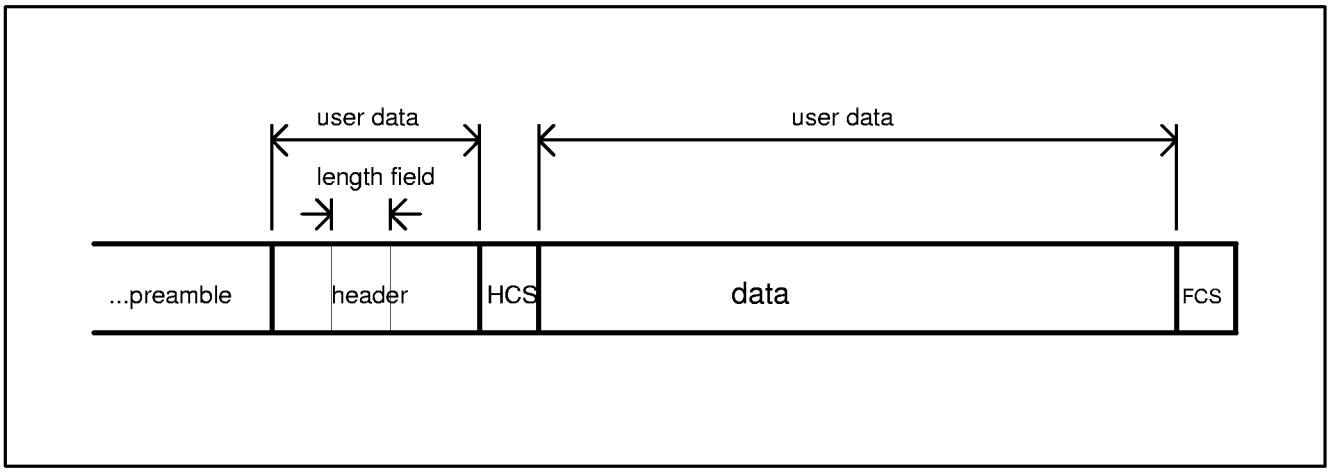

Figure 7. Generic Packet Format

5.3 CCB Sub-blocks

Figure 8 illustrates the sub-blocks of the CCB.

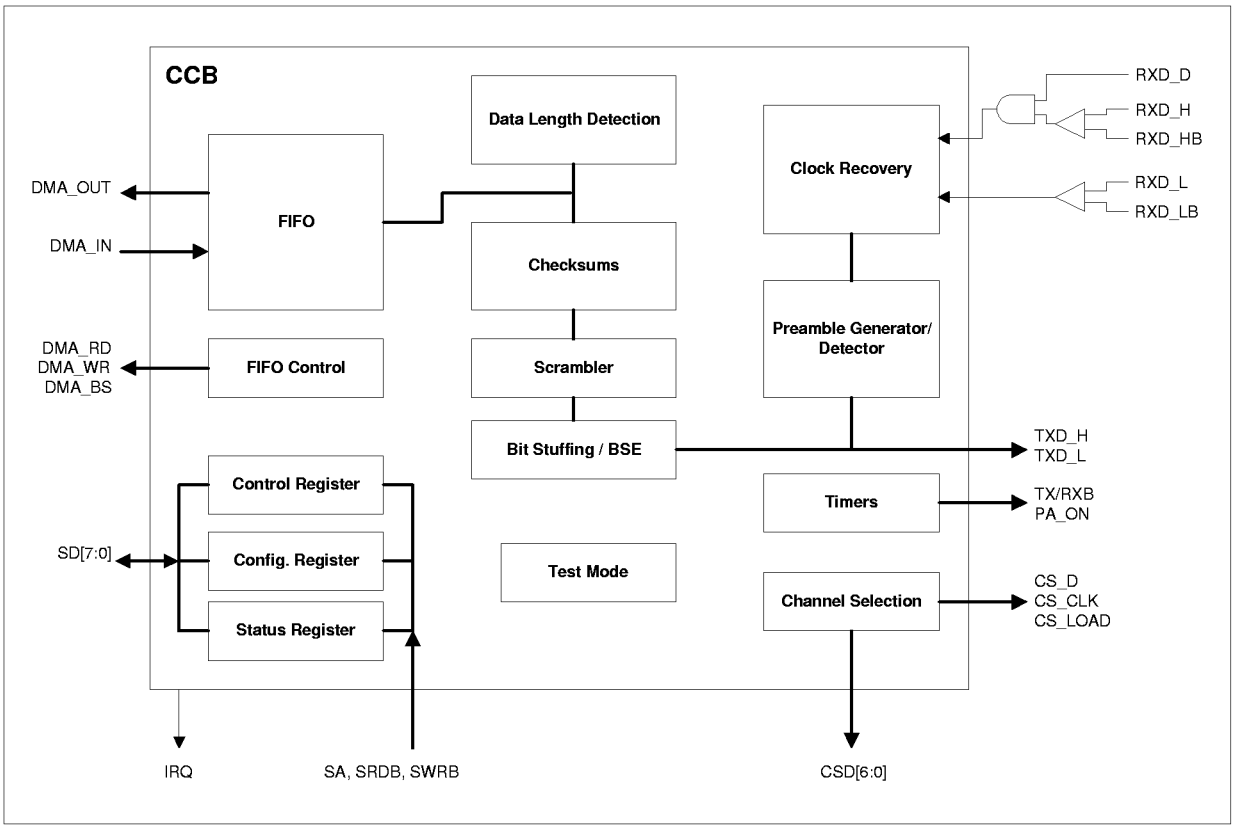

Figure 8. CCB Sub-blocks

5.3.1 CCB control

The operations of the CCB are controlled by either the internal 8051 micro-controller, or by an external processor. The processor can read from or write to the Control Registers, which control the operation of the CCB (discussed in Section 5.4). The processor can also read and write to the Configuration Registers, which are discussed in Section 5.5. The processor can read the Status Registers (discussed in Section 5.6) which provide feedback on the current state of the CCB.

5.3.2 FIFO

The CCB includes a 64-byte FIFO, to provide a buffer in the transfer of data between the radio and the Memory Control Block. Reading and writing to the FIFO from the radio side is independent of DMA transfers to the MCB.

In transmit mode, if the FIFO is not full, a byte will be read from the MCB on every other 'bit_clk' cycle (a 'bit_clk' cycle will either be 1µS or 1.6µS depending on the CCB configuration (Section 5.5.18.1). When the FIFO is full, the CCB will wait until the radio side removes a byte, and then the CCB will 'top-up' the FIFO from the MCB side. This will continue until the number of bytes read from the MCB equals the number of bytes expected by the 'Data Length' detection block (Section 5.3.3).

In receive mode, data received by the radio will be stored in the FIFO until the processor gives permission to transfer the data to the MCB. Once the processor gives permission, the CCB will wait for a pre-programmed period. It will then write to the MCB on every other 'bit_clk' cycle until the FIFO is empty. The CCB will then transfer data to the MCB as soon as it is available from the radio side, until all of the received packet has been transferred to the MCB.

In FIFO access mode (Section 5.5.18.4), the processor can write to Addr. 05h, which writes bytes into the FIFO. The pointers in the CCB will be updated in the same way as they would be for a DMA read. When the CCB starts a transmission, as soon as a byte is available in the FIFO, the CCB pre-loads four bytes into the BSE registers. With the usual DMA delay mechanism this would wait until the DMA starts. However, if the processor writes bytes directly into the FIFO, at the start of a transmission the CCB will immediately try to pre-load the BSE registers. Therefore, either the DMA delay should be very small, or preferably, the processor should write a minimum of four bytes to the FIFO prior to starting the transmission (otherwise a FIFO read error will be generated).

Also, in FIFO access mode, Addr. 40h to 7Fh are mapped to the FIFO read address. When the processor reads the FIFO this way it does not affect the CCB pointers (i.e. the data is left in the FIFO, waiting to be transferred to the MCB).

Addr. 28h contains the FIFO in-pointer, and Addr. 29h contains the FIFO out-pointer. The in-pointer indicates the address of the last byte written to the FIFO (either from the radio side in receive mode, from the MCB during transmit, or from ADDR. 05h in FIFO access mode). The out-pointer indicates the address of the next byte to be read from the FIFO (either by the radio side in transmit mode, or by the MCB after an 'Upload Grant' has been given).

The FIFO access mode should not be used to read the FIFO during either transmit, or after an 'Upload Grant' has been given in receive. The FIFO access mode should not be used to write to the FIFO while either the Commence Transmit or Commence Receive bits are low.

5.3.3 Data Length Detection

All packets must contain a 'Data Length' field in the Header. The Data Length field consists of a pair of bytes which specify how many bytes are in the frame. The location of the 'Data Length' field is specified in Addr. 1Bh (Section5.5.14). The second byte of the 'Data Length' field is 'ANDed' with the contents of Addr. 1Ah (Section5.5.13). If the Legacy bit (Section 5.5.18.2) is set, the 'Data Length' field applies to the frame length excluding the length of the Frame Checksum. When the Legacy mode bit is clear, the 'Data Length' applies to the frame length including a 32-bit Frame Checksum. (N.B. Four is added to the data length excluding the Frame Checksum, irrespective of the type of Frame Checksum being used (Addr. 1Eh, Bit 0). Therefore, when the Legacy mode bit is clear, a 32-bit Frame Checksum should always be used). In Legacy mode (bit set), if any of the three MSB of the second byte of the 'Data Length' field are set, the 'Data Length' field is cleared to zero. If the 'Data Length' field is zero the transmission/reception state machine will terminate after the Header Checksum.

5.3.4 CRC Checksum

The CCB actually contains three CRC Checksum generators, which are used to detect errors in the received data. All three Checksum generators are initially loaded with the 'Net Key', which is the seed of the CRC. One CRC Checksum generator is used to generate/check the Checksum of the header. It can be configured to be either 16 or 32 bits long. The second Checksum generator is used to generate/check the Checksum of the frame. It can also be configured to be either 16 or 32 bits long. The advantage of having the Frame Checksum (FCS) generator independent of the Header Checksum (HCS) generator is that the FCS can start before the HCS finishes (when the FCS starts in the header, it is disabled while the HCS is being received/transmitted). The third CRC Checksum is used in 4-level GFSK mode. The 4-level Checksum starts when the CCB switches to 4-level mode (discussed in Section 5.11), and the Checksum can only be 32 bits long.

All three Checksum's implement the same CRC code, which is the same as the code used in the IEEE 802.11 standard. The 16-bit CRC uses the generator polynomial:

$$x^{16}+x^{12}+x^5+1$$

The checksum transmitted can either be the Checksum remainder, in which case the remainder after reception should be zero, or the ones complement of the Checksum remainder, in which case the remainder after reception should be:

$$x^{12}+x^{11}+x^{10}+x^8+x^3+x^2+x^1+1$$

The 32-bit CRC uses the generator polynomial:

$$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1.$$

The checksum transmitted can either be the Checksum remainder, in which case the remainder after reception should be zero, or the ones complement of the Checksum remainder, in which case the remainder after reception should be:

$$x^{31}+x^{30}+x^{26}+x^{25}+x^{24}+x1^8+x^{15}+x^{14}+x^{12}+x^{11}+x^{10}+x^8+x^6+x^5+x^4+x^3+x+1$$

However, in the CCB, CRC's are implemented in reverse order (i.e. L.S.B. is actually M.S.B.). Therefore, the 'Net Key' and the CRC remainder registers are in reverse order to those specified in the IEEE 802.11 standard. The CRC remainder register should contain F0B8h for a ones complement 16-bit CRC, and DEBB20E3h for a ones complement 32-bit CRC.

5.3.5 Scrambler

The scrambler implements the following generator polynomial, which generates a 127-bit repeating pattern:

$$x^7+x^4+1$$

The generator is initialised to all ones at the start of reception/transmission, and serially scrambles all bits except the BSE stuff symbol. If BSE is used, scrambling will start after the first stuff bit (i.e. after 17 bits if 16/17 BSE is used, or 33 bits if 32/33 BSE is used). If BSE is not used, scrambling starts with the first bit after the Sync. Frame Delimiter.

5.3.6 Bit Stuffing/BSE

Two mutually exclusive encoding options are included to make the data more suitable for radio transmission. The Bit stuffing option can be configured to either add an extra bit after 8 or 16 bits. The stuff bit added will be of the inverse polarity to the last bit transmitted. (N.B. When data scrambling is used with bit stuffing, the scrambler generator is incremented during the stuff bit, but the newly generated scramble bit is ignored so that the bit output is the inverse of the last bit output). The main advantage of bit stuffing is that it ensures there will be at least one edge in the data every 9 or 17 bits, which helps with clock recovery (and there is a requirement for the DE6003 radio that the data polarity should change within 27us).

Bias Suppression Encoding (BSE) is implemented as described in the IEEE 802.11 (Draft 5) standard. It minimises the cumulative bias of a packet (i.e. ratio of '1's to '0's) by deciding whether inverting a block of data will increase or decrease the bias. The block size can be configured to be either 16 or 32 symbols (plus a stuff symbol to indicate if the block has been inverted).

5.4 Control Registers

The CCB has five control registers (from Addr. 00h to Addr. 04h), which are used to control the operation of the CCB after the initial configuration period. The registers control whether the CCB is trying to transmit or receive, which channel is to be used, the address matching options, and which conditions cause an interrupt. In addition, there is the FIFO Write register (which can only be used in FIFO access mode), and the CCA Timer register.

The registers can be both read and written to, and all the control registers will be set to a specified default condition by a hardware reset. The control registers are not affected by a software reset (A software reset is explained in Section 5.4.1.7).

5.4.1 Control Register 00h

5.4.1.1 Configuration Enabled bit (Addr. 00h, Bit 0)

When this bit is set, the configuration registers discussed in Section 5.5 can be loaded. When the bit is cleared, the clock to the configuration registers is disabled. The registers retain the value they were last loaded with, but cannot be loaded with a new value. Disabling the configuration registers saves power.

The Configuration Enabled bit is set by a hardware reset.

5.4.1.2 Channel Hop bit (Addr. 00h, Bit 1)

The WL102 has two different channel selection methods, one for the DE6003 radio, and the other for the DE6038 radio (DE6003/DE6038 configuration discussed in Section 5.5.18.1). In either case, the Channel Hop bit will be set by a hardware reset.

5.4.1.2.1 Channel selection in DE6038 mode

The DE6038 radio uses the WL800 Frequency Synthesiser which is programmed serially, using a three wire bus to load an 8-bit number. The value loaded for receive mode should be 43MHz less than the value loaded for transmit mode (The values loaded are discussed further in Section 5.4.2.1). The Channel Hop bit (CH) is used to enable the automatic channel load sequence. When CH is low, the WL800 will be loaded with a new channel every time either CT (Commence Transmission) or CR (Commence Receive) are taken low. If CH is high, a new channel will not be loaded.

The CH bit can be cleared either at the same time as, or before CT or CR is cleared to enable the channel load sequence.

5.4.1.2.2 Channel selection in DE6003 mode

The DE6003 uses a 7-bit bus to specify the channel, plus a strobe signal to load the channel. The falling edge of the Channel Hop bit triggers the state machine, which causes the channel strobe signal to pulse low for 1µS, which loads the DE6003 channel. The CH bit then needs to be set again before the next channel load sequence can be triggered. The channel loaded is discussed in Section 5.4.2.2.

5.4.1.3 Commence Receive (Addr. 00h, Bit 2)

The Commence Receive bit (active low) is the bit which tells the CCB to try to receive a packet. The Commence Receive bit being low causes an internal bit called Commence Receive (CR) to go low. CR being low enables the RXD buffers and the clock recovery block. However, CR can be over-ridden by the Commence Transmit bit and subsequent transmit sequence. The state of the internal CR bit can be read from Addr. 21, bit 6 (see Section 5.6.2.7).

The Commence Receive bit is set by a hardware reset.

5.4.1.4 Commence Transmit (Addr. 00h, Bit 3)

The Commence Transmit bit (active low) is the bit which tells the CCB to transmit a packet. As with the Commence Receive bit, there is an internal bit called Commence Transmit (CT). If CR was low, CT will wait three 'bit_clk' cycles (3µs if CCB configured for 1 or 2Mbit/s, 4.8µs if CCB configured for 625kbit/s) to allow CR to terminate cleanly before CT goes low. The CT bit will always over-ride the CR bit. Clearing the Commence Transmit bit also acts as the DMA Grant trigger. After the delay period programmed into Addr. 3Eh (Section 5.5.21), the CCB will assume it has access to the MCB and begin DMA transfers.

The Commence Transmit bit is automatically set high when the Tx. DMA Transfer Complete interrupt is generated. The Commence Transmit bit should not be set low again until the radio finishes transmitting, indicated by Tx. DMA Transfer Complete status bit being cleared (Addr. 20h, bit 4) and the RXTX bit being set high (Addr. 21h, bit 4).

The Commence Transmit bit is set by a hardware reset.

5.4.1.5 CCB Software Reset (Addr. 00h, Bit 4)

The CCB Software Reset bit (active low) resets all the internal registers and state machines in the CCB (e.g. clock recovery, BSE, CRC, FIFO, Receive/Transmit state machines). However, it does not effect the Control or Configuration registers (i.e. the registers described in Sections 5.4 and 5.5). A reset is caused by clearing the bit. The reset bit will automatically be set again on the clock cycle after the access.

The CCB Software Reset bit is set by a hardware reset. (The hardware reset is applied to all the registers reset by the software reset as well as the Control and Configuration registers).

5.4.1.6 Interrupt Clear (Addr. 00h, Bit 5)

The Interrupt Clear bit will clear an interrupt due to CCA Timer Equal Zero, a Frame Upload Complete, a Frame Upload Request, a Header Upload Request, or a Tx. DMA Transfer Complete. It will not clear an interrupt due to a FIFO Read Error or a FIFO Write Error. An interrupt due to the PLL lock detect signal will be cleared, but if the PLL lock detect signal is still in the fail state (i.e. set) the interrupt will immediately re-occur. Interrupt masks are discussed in Section 5.4.5, and interrupt identification is discussed in Section 5.6.1.

To clear an interrupt the Interrupt Clear bit needs to be set. Two 'bit_clk' cycles later, or after writing to Addr. 00h is complete (which ever is longest), the Interrupt Clear bit will automatically be cleared. The Interrupt Clear bit is cleared by a hardware reset.

5.4.1.7 Standby (Addr. 00h, Bit 6)

The Standby bit controls an output pin to the radio. It does not affect any of the functions of the CCB. Powering-down the CCB will not affect the state of the Standby pin to the radio. The Standby bit is cleared by a hardware reset, which is sleep mode for both the DE6003 and the DE6038 radios.

5.4.1.8 Upload Grant (Addr. 00h, Bit 7)

The Upload Grant bit should be set in response to an Upload Request. In receive mode, the CCB will generate a Header Upload Request when it finds valid preamble plus the frame word. At this point it will store data in it's 64 byte FIFO until it has been given an Upload Grant. The Upload Grant signal gives the CCB permission to transfer the header data into the MCB (after the delay period specified in Addr. 3Eh).

After the header data has been transferred to the MCB, the CCB generates a Frame Upload Request. The Upload Grant bit should then be set again to permit the transfer of the frame to the MCB. In either case of upload request, if an Upload Grant isn't given and the FIFO becomes full, a FIFO Write Error will occur.

The Upload Grant bit is automatically cleared after the DMA state machine has started. The Upload Grant bit is also cleared by a hardware reset.

5.4.2 Channel Select register "SDT" (Addr. 01h)

The Channel Select Register "SDT" holds the code for the frequency which will be loaded into the radio. In DE6003 mode, the "SDT" register is used to store the channel for both transmit and receive mode. In DE6038 mode, the "SDT" register is used to store the transmit channel only. Section 5.12 contains a table showing how the code relates to the final frequency. A hardware reset will load the "SDT" register with 6Bh.

5.4.2.1 Channel selection in DE6038 mode

The DE6038 radio uses a frequency synthesiser which is serially programmed by the WL102. The frequency synthesiser can synthesise 144 frequencies in 1MHz steps. The 8-bit number serially loaded into the synthesiser gives the division ratios required for the two counters in the synthesiser's programmable divider system to generate the required channel. The division ratio required to transmit on a specific channel is different from the divide ratio required to receive on the same channel. Table 22 shows the values which should be loaded into the WL800 frequency synthesiser.

The "SDT" register holds the value which will be loaded in transmit mode. On the falling edge of CT (Commence Transmit discussed in Section 5.4.1.4), if CH (Channel Hop bit discussed in Section 5.4.1.2) is low, the channel load state machine is started. The "SDT" register will be clocked out of the "CS_D" pin at the system clock rate (i.e. one bit every 100ns), MSB (Most Significant Bit) first. The "CS_CLK" pin will be the inverted system clock, active during the channel load sequence, with the rising edge approximately 50nS after "CS_D" changes. The "CS_LOAD" pin will go low at the start of the channel load sequence, and then high again at the end of the channel load sequence.

5.4.2.2 Channel selection in DE6003 mode

In DE6003 mode the frequency synthesiser is programmed by a 7-bit bus, with a strobe signal. The same value is used for both the transmit and receive channel. Therefore, "SDT" is used both in transmit mode and receive mode. Section 5.12 shows how the value loaded relates to a channel. The falling edge of CH (discussed in Section 5.4.1.2.2) starts the channel load state machine. The seven LSBs are output on the CSD pins. "CS_LOAD" will pulse low for 1µS, which loads CSD into the DE6003. After the channel load state machine finishes, the CH bit needs to be set manually before another channel load sequence is triggered.

5.4.3 Channel Select Register "SDR" (Addr. 02h)

The Channel Select register "SDR" holds the code for the frequency which will be loaded at the start of receive in DE6038 mode. Section 5.12 contains a table showing how the code relates to the final frequency. The channel load sequence is as described in Section 5.4.2.1, except it is triggered by the falling edge of CR (discussed in Section 5.4.1.3) instead of CT. A hardware reset will load the "SDR" register with 30h.

5.4.3.1 Antenna Select (Addr. 02h, Bit 0)

In DE6003 mode, Bit 0 of the SDR register is used as the Antenna Select bit. In DE6003 mode, the "ANT_SEL" pin is driven by the Antenna Select bit. A low selects "ANT1" and a high selects "ANT2". The Antenna Select bit is cleared by a hardware reset.

5.4.3.2 RF Power Level (Addr. 02h, Bit 1)

In DE6003 mode, Bit 1 of the SDR register is used as the RF Power Level bit. In DE6003 mode, the "PWR_LOB" pin is driven by the RF Power Level bit. When the bit is set, the DE6003 radio is in high power mode. The RF Power Level bit is cleared by a hardware reset.

5.4.4 Control Register 03h

5.4.4.1 Receive Broadcast (Addr. 03h, Bit 0)

The Receive Broadcast bit indicates whether the Address Hit bit should be set if a broadcast address has been identified. Using the Address Location field (Section 5.5.14) to identify the first byte of the address, if all the bits of all six bytes of the address are set, and the Receive Broadcast bit is set, the Address Hit bit will be set. The Receive Broadcast bit is cleared by hardware reset.

5.4.4.2 Receive Unique Address (Addr. 03h, Bit 1)

The Receive Unique Address bit indicates whether the Address Hit bit should be set if the received address is identical to the expected address. Using the Address Location field (Section 5.5.14) to identify the first byte of the address, if the six bytes of the received address are identical to the address specified in the Unique Address field (Section 5.5.20), the Address Hit bit will be set. The Receive Unique Address bit is cleared by hardware reset.

5.4.4.3 Receive Unicast (hash lookup/all) (Addr. 03h, Bit 2)

The Receive Unicast bit is one of the Address matching options. An address is considered to be a unicast address if the LSB of the first byte of the address field is cleared (i.e. an address can be either a multicast address or a unicast address). The first byte of the address field is identified by the Address Location field (discussed in Section5.5.14). If the Receive All bit is set (Addr. 03h, Bit 4), the Address Hit bit will be set if the multicast/unicast bit is clear. If the Receive All bit is clear, the unicast hash lookup option will be used. In the hash-lookup option, the six bytes of the address will be passed through a hash algorithm (described in Section 5.8.7) to generate a six-bit pointer. The pointer identifies a location in the hash lookup table (Section 5.5.19). If the bit pointed to in the hash lookup table is set and the multicast/unicast bit was clear, the Address Hit bit will be set. The Receive Unicast bit is set by a hardware reset.

5.4.4.4 Receive Multicast (hash lookup/all) (Addr. 03h, Bit 3)

The Receive Multicast bit is one of the Address matching options. An address is considered to be a multicast address if the LSB of the first byte of the address field is set. The first byte of the address field is identified by the Address Location field (discussed in Section 5.5.14.1). If the Receive All bit is set (Addr. 03h, Bit 4), the Address Hit bit will be set if the multicast bit is set. If the Receive All bit is clear, the multicast hash lookup option will be used. In the hash-lookup option, the six bytes of the address will be passed through a hash algorithm (described in Section5.8.7) to generate a six-bit pointer. The pointer identifies a location in the hash lookup table (Section5.5.19). If the bit pointed to in the hash lookup table is set and the multicast bit was set, the Address Hit bit will be set. The Receive Multicast bit is set by a hardware reset.

5.4.4.5 Receive All (Addr. 03h, Bit 4)

When the Receive All bit is set, the multicast and unicast address matching options described above will rely solely on the multicast/unicast bit. If the Receive All bit is cleared, the multicast and unicast address matching options described above will use the hash lookup option. If both the Receive Multicast and Receive Unicast bits are set and the Receive All bit is clear, the hash look-up table can be used irrespective of the state of the multicast bit. If both the Receive Multicast and Receive Unicast bits are clear, the Receive All bit will have no effect. (The receive all option is effectively the same as setting all the bits of the hash look-up table.) The Receive All bit is set by a hardware reset.

5.4.4.6 Define Header Upload Length (Addr. 03h, Bit 5)

The Define Header Upload Length bit can be used to specify how many bytes should be transferred to the MCB as the header. When the bit is set, Addr. 0Ch (Section 5.5.2) indicates how many bytes should be transferred to the MCB when the Header Upload Grant has been given. If the Define Header Upload Length bit is not set, the bytes received up to the Checksum insertion point (Addr. 1Dh) will be transferred to the MCB. The Define Header Upload Length bit is set by the hardware reset.

5.4.4.7 CCA Timer Run (/not Stop) (Addr. 03h, Bit 6)

When the CCA Timer Run is set, the 16-bit CCA Timer is enabled (Section 5.4.7). If both the CCA Timer Run bit and the CCA Timer Always bit are set, the CCA Timer will be decremented every 'bit_clk' cycle until it reaches zero. If the CCA Timer Run bit is set, but the CCA Timer Always bit is clear, the CCA Timer will only be decremented if both CCA (Section 5.6.2.1) and Sync (Section 5.6.2.4) are clear (i.e. received RF power small and Sync pattern not detected). When CCA Timer Run is clear, the CCA Timer will be stopped irrespective of the state of CCA Timer Always.

CCA Timer Run is cleared by a hardware reset.

5.4.4.8 CCA Timer Always (/not CCA test) (Addr. 03h, Bit 7)

When the CCA Timer Always bit is set, the CCA Timer will run when CCA Timer Run is set. In this mode the CCA Timer can be used to generate an interrupt after a specified delay. When CCA Timer Always bit is clear, the CCA Timer can be used to detect if the channel is clear to transmit for a specified period. This is achieved by only decrementing the counter if CCA Timer Run is set, CCA is clear (indicating low received power), and Sync is clear (indicating the receiver has not detected valid Sync. Words).

CCA Timer Always is cleared by a hardware reset.

5.4.5 Interrupt Mask Registers (Addr. 04h)

Address 04h can be used to disable each individual cause of an interrupt.

5.4.5.1 CCA Timer Equal Zero interrupt mask (Addr. 04h, Bit 0)

An interrupt is generated when the CCA Timer is decremented from one to zero. If the mask bit is set, the interrupt will not be generated. A hardware reset clears the CCA Timer Equal Zero interrupt mask.

5.4.5.2 Upload Header Request interrupt mask (Addr. 04h, Bit 1)

An interrupt is generated when the CCB is in receive mode and it detects the correct preamble pattern (discussed in Section 5.9). If the mask bit is set, the interrupt will not be generated. In this configuration, the status bits in Addr. 20h need to be read frequently enough to ensure the processor has time to respond to the upload request before the FIFO overflows. A hardware reset clears the Upload Header Request interrupt mask.

5.4.5.3 Upload Frame Request interrupt mask (Addr. 04h, Bit 2)

An interrupt is generated after the CCB has transferred the received header to the MCB. The interrupt indicates both that the DMA transfer to the MCB is complete and that the frame is ready to be transferred to the MCB. If the mask bit is

set, the interrupt will not be generated. If the interrupt is masked, the processor has to read Addr. 20h frequently enough to ensure the FIFO doesn't overflow. A hardware reset clears the Upload Frame Request interrupt mask.

5.4.5.4 Frame Upload Complete interrupt mask (Addr. 04h, Bit 3)

An interrupt is generated after the CCB has transferred the received frame to the MCB. The interrupt indicates both that the DMA transfer to the MCB is complete and that the CCB has finished the receive cycle. If the mask bit is set, the interrupt will not be generated. A hardware reset clears the Frame Upload Complete interrupt mask.

5.4.5.5 Tx. DMA Transfer Complete interrupt mask (Addr. 04h, Bit 4)

In transmit mode, an interrupt is generated to indicate when the CCB has read the last byte from the MCB, and the DMA transfer is complete. However, the CCB will still be transmitting data which is stored in it's internal FIFO. If the mask bit is set, the interrupt will not be generated. A hardware reset clears the Tx. DMA Transfer Complete interrupt mask.

5.4.5.6 FIFO Read Error interrupt mask (Addr. 04h, Bit 5)

The FIFO will generate a read error interrupt if it is read when it is empty. The most likely cause of a FIFO read error is at the start of transmit. If the Rx to Tx switch time (Addr. 0Dh) plus the preamble time is shorter than the DMA delay time (Addr. 3Eh), the CCB can attempt to read the FIFO to start transmitting data before the DMA transfer has started. When the FIFO Read Error interrupt occurs, any DMA transfer will be aborted. However, if the mask bit is set, the interrupt will not be generated, and the DMA transfer will not be aborted. A hardware reset clears the FIFO Read Error interrupt mask.

5.4.5.7 FIFO Write Error interrupt mask (Addr. 04h, Bit 6)

The FIFO will generate a write error interrupt if it is written to when it is full. The most likely cause of a FIFO write error is during receive. If the Upload Grant is given late, the received data can fill the FIFO before the DMA transfer starts. A FIFO Write Error interrupt will cause any DMA transfer to be aborted. However, if the mask bit is set, the interrupt will not be generated, and the DMA transfer will not be aborted. A hardware reset clears the FIFO Write Error interrupt mask.

5.4.5.8 Synth. Unlock interrupt mask (Addr. 04h, Bit 7)

Both the DE6003 and the DE6038 radios generate a signal which goes high if the Phase Locked Loops on the radio become unlocked. This is an error condition and could mean that the radio is using an illegal frequency. However, the signal may also go high during the channel transition phase. An interrupt will be generated when the signal goes high, unless the mask bit is set. A hardware reset sets the Synth. Unlock interrupt mask.

5.4.6 FIFO Write (Addr. 05h)

The FIFO Write register will only work if the FIFO access mode bit (Section 5.5.18.4) is set. The processor can write to the register in the same way it writes to any other register. However, there must be at least three 'bit_clk' cycles between consecutive writes to the register. Reading the register will return a temporary variable which will be the last value written into either Addr. 05h, Addr. 06h, or Addr. 07h; it does not read from the FIFO.

Writing to Addr. 05h updates both the FIFO in-pointer and the DMA byte count. This means the FIFO Write register can be used by the processor to load a packet header into the FIFO. Then, when Commence Transmit is cleared, the data transferred from the MCB will be appended to the data already written into the FIFO. For packets of 64 bytes or less, the whole packet can be loaded by the processor, and the DMA state machine will not be started. The processor must not load more bytes than the DMA transfer would load (i.e. load more bytes than indicated by the 'Data Length' field).

The FIFO Write register is cleared to 0 by a hardware reset.

5.4.7 CCA Timer Register (Addr. 06h and 07h)

Addr. 06h (for 8 MSB) and Addr. 07h (for 8 LSB) are used to load and read the 16-bit CCA Timer. The CCA Timer is enabled by a combination of two bits. The CCA Timer Run bit (Section 5.4.4.6) switches between 'run if enabled' and 'stop'. The CCA Timer Always bit (Section 5.4.4.7) switches between always enabled and only enabled if channel is clear for transmission. When it is enabled, the CCA timer counts down to zero, in 'bit_clk' cycles (Section 5.5.18.3). As it is decremented from one to zero it can generate an interrupt (Section 5.4.5.1). The counter stops at zero.

The CCA Timer is loaded via a temporary register to re-synchronise the loaded value with the CCA Timer clock. Addr 05h, Addr. 06h and Addr. 07h share a common temporary register. After writing to either Addr. 06h or Addr. 07h, the processor should wait at least 1.5 'bit_clk' cycles before attempting another write to either Addr. 05h, Addr. 06h, or Addr. 07h.

The CCA Timer is loaded with zero by a hardware reset.

5.5 Configuration Registers

The CCB has a number of configuration registers (from Addr. 08h to Addr. 1Fh and from Addr. 30h to 3Eh), which are used to configure the CCB to enable it to implement a wide variety of transmission/reception protocols. The configuration registers are intended to be written to during an initial start-up phase, and then have their clocks disabled by clearing the Configuration Enabled bit (discussed in Section 5.4.1.1). This saves power, whilst still allowing the contents to be read. The configuration registers should not be changed during a transmit or receive cycle.

The registers can be both read and written to if the Configuration Enabled bit (Addr. 00h, Bit 0) is set. If the Configuration Enabled bit is low, the configuration registers can only be read. All the configuration registers will be set to a specified default condition by a hardware reset. The configuration registers are not affected by a software reset (A software reset is explained in Section 5.4.1.5).

5.5.1 Net Key (Addr. 08h to 0Bh)

The Net Key register is a 32-bit register which is used to seed both the Header CRC and the Frame CRC. The LSB of Addr. 08h is the LSB of the Net Key, and the MSB of Addr. 0Bh is the MSB of the Net Key. At the start of a transmit or receive cycle, the CRC registers are loaded with the Net Key (The CRC registers are discussed further in Section 5.3.4). The 16 bit CRC use the 16 LSBs of the Net Key (i.e. Addr. 08h and Addr. 09h).

A hardware reset sets all the bits of the Net Key (which is the seed used by the IEEE 802.11 protocol).

5.5.2 Header Upload Length

5.5.2.1 Header Upload Field (Addr. 0Ch, Bit 7)

The CCB starts by counting the number of bytes before the Header Checksum should be inserted during transmit or checked during receive (Section 5.5.16). It stops the counter just before the start of the Header Checksum, and then resets the counter to start counting the number of bytes after the Header Checksum (i.e. in what the CCB considers to be the frame). If the Header Checksum is not used, the counter applies to the frame.

If the Header Upload Field bit is clear, the Header Upload Position applies to the header counter. The specified position should be less than the Header Checksum insertion position. If the Header Upload Field bit is set, the Header Upload Position applies to the frame counter. All the bytes in the Header are uploaded to the MCB, plus the specified number of bytes in the frame.

A hardware reset sets the Header Upload Field bit.

5.5.2.2 Header Upload Position (Addr. 0Ch, Bits 0-6)

The Header Upload Position relates directly to the number of bytes from the specified field which should be transferred to the MCB (as indicated in Table 5).

A hardware reset sets the Header Upload Position to 10h.

H.U. Field	H.U. Position	No. bytes transferred to MCB
0	0	Should not be used
0	1	1 (first byte of header)
0	2-127	2 - 127 bytes (but number of bytes must be less than or equal to Header Checksum insertion point (Addr. 1Dh))
1	0	Should not be used
1	1	No. bytes in header (specified in Addr. 1Dh) + 1
1	2-127	No. bytes in header + 2 to 127 bytes

Table 5. Header Upload Length

5.5.3 Rx. to Tx. Switch Time (Addr. 0Dh)

The Rx. to Tx. Switch Time register is an eight bit register which specifies the delay (in multiples of 2 'bit_clk' cycles) between the RXTX pin switching to transmit mode, and the PA signal turning on. A 'bit_clk' cycle will either be 1µS or 1.6µS depending on the data rate configuration (Section 5.5.18.3). In addition to the specified delay, there is an additional 1 'bit_clk' cycle delay due to an internal transition state delay. A hardware reset sets the Rx. to Tx. Switch Time register to 6Eh (which results in a total delay of 221 'bit_clk' cycles).

5.5.4 Ramp-up Time (Addr. 0Eh, Bits 0-3)

The Ramp-up Time register is a four bit register which specifies the delay (in 'bit_clk' cycles) between the PA signal being set and the start of the preamble. In addition to the specified delay, there is an additional 4 'bit_clk' cycles delay. A hardware reset sets the Ramp-up Time register to 6h (which results in a total delay of 10 'bit_clk' cycles).

5.5.5 Ramp-down Time (Addr. 0Fh)

The Ramp-down Time register is an eight bit register which specifies the delay (in 'bit_clk' cycles) between the PA signal being cleared (at the end of Tx.) and the RXTX pin switching to the receive state. In addition to the specified delay, there is an additional 1 'bit_clk' cycle delay due to an internal transition state delay. A hardware reset sets the Ramp-down Time register to 09h (which results in a total delay of 10 'bit_clk' cycles).

5.5.6 Sync. Word (Addr. 10h)

The Sync. Word is the first part of the preamble pattern. A specified number of bits of the Sync. Word (up to 8) are repeatedly output (a specified number of times) to improve the ability to recover the data (preamble is discussed further in Section 9). The Sync. Word is output LSB first, up to the bit specified in Section 5.5.7.1. A hardware reset sets the Sync. Word to 02h.

5.5.7 No. of Bits of Sync and Frame Word (Addr. 11h)

5.5.7.1 No. of Bits of Sync Word (Addr. 11h, Bits 0-2)

The No. of Bits of Sync. Word specifies how many of the bits of Sync. Word are used (from LSB up) (sync words are discussed further in Section 5.9). A hardware reset sets the Number of Bits of Sync Word used to 1, which corresponds to two bits, as shown in Table 6.

No. Bits of S.W. (B2=MSB, B0=LSB)	No. bits used	Illustration of 'U'sed bits of Sync. Word (MSB - LSB)
0	---	Should not be used
1	2	xxxx xxUU
2	3	xxxx xUUU
3	4	xxxx UUUU
4	5	xxxU UUUU
5	6	xxUU UUUU
6	7	xUUU UUUU
7	8	UUUU UUUU

Table 6. Effect of No. of Bits of Sync. Word register

5.5.7.2 No. of Bits of Frame Word (Addr. 11h, Bits 3-7)

The No. of Bits of Frame Word specifies how many of the bits of the Frame Word are used (from LSB up) (frame words are discussed further in Section 5.9). A hardware reset sets the Number of Bits of Frame Word used to 0Fh, which corresponds to sixteen bits, as shown in Table 7.

No. bits of S.W. (B7=MSB, B3=LSB)	No. bits used	Illustration of 'U'sed bits of Frame Word (MSB - LSB)
0	---	Should not be used
1	2	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxUU
2	3	xxxx xxxx xxxx xxxx xxxx xxxx xxxx xUUU
...	...	...
0Fh	16	xxxx xxxx xxxx xxxx UUUU UUUU UUUU UUUU
...	...	...
1Eh	31	xUUU UUUU UUUU UUUU UUUU UUUU UUUU UUUU
1Fh	32	UUUU UUUU UUUU UUUU UUUU UUUU UUUU UUUU

Table 7. Effect of No. of Bits of Frame Word register

5.5.8 Frame Word (Addr. 12h to 15h)

The Frame Word is the last part of the preamble pattern. A specified number of bits of the Frame Word (up to 32) are used to identify the end of the preamble pattern, and hence the start of the data (preamble is discussed further in Section 5.9). The Frame Word is output LSB first, up to the bit specified in Section 5.5.7.2 (i.e. LSB of Addr. 12h is first bit, up to MSB of Addr. 15h if all 32 bits are used). A hardware reset sets the Frame Word to 0000BD30h (which is the Frame Word used by the IEEE 802.11 protocol).

5.5.9 No. of Transmitted Sync. Words (Addr. 16h)

The Number of Transmitted Sync. Words register specifies how many times the Sync Word is transmitted (from 0 to 255). It is discussed further in Section 5.9. A hardware reset sets the Number of Transmitted Sync. Words to 28h (which outputs 40 Sync. Words).

5.5.10 Received Sync. Words (Addr. 17h)

5.5.10.1 No. Sync. Words Required for 'Sync' (Addr. 17h, Bits 0-2)

The Number of Sync. Words Required for 'Sync' specifies the number of Sync. Words the receiver must detect before it sets the 'Sync' bit. When the 'Sync' bit is set, the clock recovery circuit switches to 'narrow loop' mode, and the CCB can start searching for the Frame Word. A hardware reset sets the No. of Sync. Words Required for 'Sync' register to 2, which corresponds to three Sync. Words.

No. of S.W. Required (B2=MSB, B0=LSB)	Actual no. S.W. req'd
0	1
1	2
...	...
6	7
7	8

Table 8. Effect of No. of Sync. Words Required for 'Sync' register

5.5.10.2 Bit Errors Allowed (Addr. 17h, Bit 3)

If this bit is set, single bit errors are allowed in any two consecutive Sync. Words. If more than a single bit error is detected, the 'Sync' bit will be cleared. If the Bit Errors allowed bit is cleared, Sync will be lost on the first error. A hardware reset clears the Bit Errors Allowed bit.

5.5.11 Valid Data Tolerances (Addr. 18h)

5.5.11.1 Jitter Tolerance (Addr. 18h, Bits 0-1)

The Jitter Tolerance specifies how much jitter (the difference between actual edge in data stream and expected edge in data stream) is required before edges are considered to be noise rather than valid early/late edges.

A hardware reset sets the Jitter Tolerance register to 1.

Jitter Tolerance (MSB = Bit 1)	Tolerance for 625kbit/s D.R.	Tolerance for 1Mbit/s D.R.
0	12.5%	20%
1	25%	40%
2	37.5%	Not used
3	Not used	Not used

Table 9. Jitter Tolerance (as percentage of bit period)

5.5.11.2 Noise Threshold (Addr. 18h, Bits 2-3)

The Noise Threshold specifies how many occurrences of noise will cause the 'Noise' bit of status register Addr. 21h to be set. A hardware reset sets the Noise Threshold to 3.

Noise Threshold (MSB = Bit 3)	No. of occurrences of noise
0	1
1	4
2	8
3	16

Table 10. Noise Threshold

5.5.11.3 Data Threshold (Addr. 18h, Bits 4-5)

The Data Threshold specifies how many occurrences of valid data (i.e. edges within the jitter tolerance) are required before the 'Noise' and 'Long' bits are cleared again. A hardware reset sets the Data Threshold to 1.

Data Threshold (MSB = Bit 5)	No. of occurrences of valid data
0	1
1	4
2	8
3	16

Table 11. Data Threshold

5.5.12 Packet Length Start Byte Position (Addr. 19h)

This register specifies the position in the packet header where the two byte "Data Length" field is located. The value loaded indicates how many bytes into the packet the first byte of the "Data Length" field is located. A hardware reset sets the value to 1, which means the first byte of the header contains the eight LSBs of the data length, and the second byte contains the eight MSBs (the data length register is discussed further in Section 5.3.3).

5.5.13 Data Length Mask (Addr. 1Ah)

The Data Length Mask is 'ANDed' with the eight MSBs of the Data Length field. This enables any of the MSBs of the Data Length field which aren't required to be used for other purposes. A hardware reset loads the Data Length Mask with 0Fh, which gives a 12 bit data length field (as used by IEEE 802.11) and the 4 MSBs can therefore be used to hold other information.

5.5.14 Address Location (Addr. 1Bh)

The Address Location is used in receive mode, to indicate the first byte at which Address Matching should start (the Address Matching options are described in Sections 5.4.4.1 to 5.4.4.4). The Address Location is defined in terms of field (header/frame) and position.

5.5.14.1 Address Location Field (Addr. 1Bh, Bit 7)

The Address Location Field indicates whether the Address is in the header (bit cleared) or the frame (bit set). However, in the case of the Address starting at the first byte of the frame, the field should be specified as header (because the position specified is actually the byte before the first byte of the Address). A hardware reset sets the Address Location Field bit.

5.5.14.2 Address Location Position (Addr. 1Bh, Bits 0-6)

The Address Location Position specified should be the number of byte count before the first byte of the Address, as shown in Table 12. A hardware reset sets the position register to 4, which makes the default Address Location start at the 5th byte of the frame.

A.L. Field	A.L. Position	Actual location of 1st byte of address
0	0	First byte of header
0	1 - 127	2 - 128th byte (but number of bytes must be six less than the Header Checksum insertion point (Addr. 1Dh))
0	Value in Addr. 1Dh	First byte of frame
1	0	Should not be used
1	1 - 127	2nd - 128th byte of frame

Table 12. Address Location

5.5.15 Frame Checksum Start Position (Addr. 1Ch)

The Frame Checksum Start Position is the location that the Frame Checksum is started. The Frame Checksum is independent from the Header Checksum. It can be started anywhere in the header, or up to the 127th byte of the frame. If the Frame Checksum is started in the header field, it will be disabled during the header field. The Frame Checksum position starts the byte after the specified position, identical to the way the Address Location is defined (Section 5.5.14). However, if all the bits of Addr. 1Ch are set, the Frame Checksum will not be calculated.

A hardware reset clears the Frame Checksum Start Field (Bit 7), and sets the Frame Checksum Start Position to 2 (which is the same as the default Header Checksum insertion point). Therefore, the default condition is that the Frame Checksum starts calculating from the first byte of the frame.

FCSS Field	FCSS Position	Actual location at which CRC starts
0	0	First byte of header
0	1 - 127	2 - 128th byte (but number of bytes must be less than the Header Checksum insertion point (Addr. 1Dh))
0	Value in Addr. 1Dh	First byte of frame
1	0	Should not be used
1	1 - 126	2nd - 127th byte of frame
1	127	No CRC calculation

Table 13. Frame Checksum start position

5.5.16 Header Checksum Insertion Position (Addr. 1Dh)

The Header Checksum Insertion Position is an eight bit register which specifies the byte before the Header Checksum is to be inserted (Tx.) or Checked (Rx.). The “Data Length” field must be before the Header Checksum, and so the lowest value which should be loaded is shown in Table below. A hardware reset loads a value of 2.

Addr. 1D	Location of 1st byte of Header Checksum
0-1	Invalid
2-254(2h - FEh)	3rd to 255th byte
FFh	No Header Checksum

Table 14. Header Checksum Insertion Position

5.5.17 Checksum and BSE/Bit Stuffing Configuration (Addr. 1Eh)

5.5.17.1 16/32 bit Frame Checksum (Addr. 1Eh, Bit 0)

When this bit is cleared, a 32-bit Frame Checksum is used. When the bit is set, a 16-bit Frame Checksum is used. A hardware reset clears the bit (i.e. default is 32-bit Frame Checksum).

5.5.17.2 16/32 bit Header Checksum (Addr. 1Eh, Bit 1)

When this bit is cleared, a 32-bit Header Checksum is used. When the bit is set, a 16-bit Header Checksum is used. A hardware reset sets the bit (i.e. default is 16-bit Header Checksum).

5.5.17.3 Ones Complement of Checksum (Addr. 1Eh, Bit 2)

When this bit is set, both the Header Checksum and the Frame Checksum are inverted (as specified in IEEE 802.11 protocol). The main advantage of using the Complemented Checksum is that the remainder is non-zero, and so gives a more positive indication of a pass (The remainder from a Checksum which isn't complemented is zero). A hardware reset sets the Ones Complement of Checksum bit.

5.5.17.4 Scrambler enabled (Addr. 1Eh, Bit 3)

When this bit is set, a 127-length scrambler circuit is enabled. The scrambler circuit is as defined for the IEEE 802.11 protocol. The purpose of the scrambler is to make the transmitted data as 'white' as possible (i.e. no. of '1's and no. '0's is balanced), so as to avoid adverse effects on the radio. The scrambler circuit can be used with both bit stuffing and Bias Suppression Encoding (discussed in Section5.5.17.5). A hardware reset enables the scrambler.

5.5.17.5 Bit Stuffing/Bias Suppression Encoding (Addr. 1Eh, Bits 4-6)

There are two mutually exclusive options for adding bits to the transmitted data stream to reduce any adverse effects the data stream may have on the radio, and to ensure the clock recovery circuit can keep track of the ideal sampling period. A hardware reset loads 3, which is BSE 32/33 as used for the IEEE 802.11 protocol.

Bit Stuffing/BSE (MSB = Bit 6)	Option selected
0	No BSE or Bit stuffing
1	BSE 16/17
2	Invalid
3	BSE 32/33
4	Bit stuff after 8
5	Invalid
6	Bit stuff after 16
7	Invalid

Table 15. Bit stuffing/BSE options

5.5.17.6 Bit Stuffing

The bit stuffing option inserts an extra bit after every 8 or 16 bits, of the opposite polarity to the previous bit. Therefore, bit stuffing ensures the maximum number of consecutive bits of the same state is either 9 or 17. However, bit stuffing cannot control the cumulative bias.

5.5.18 CCB Modes and Data Polarity (Addr. 1Fh)

5.5.18.1 DE6003/DE6038 Mode (Addr. 1Fh, Bit 0)

The DE6003 (0) / DE6038 (1) Mode register principally effects the output pins to the radio, and in particular the channel load sequence.

In DE6038 mode, RXTX is high in transmit mode, but in DE6003 mode it is low in transmit. In DE6038 mode, the channel load sequence uses three pins to serially load a channel, which needs to be changed each time the radio switches between transmit and receive. In DE6003 mode, 8pins are required to load the channel in parallel. Also, in DE6003 mode, two of the pins used for the serial channel load in DE6038 mode are used as ANTSEL and PWRLO.

A hardware reset sets the bit (i.e. DE6038 mode).

5.5.18.2 Legacy Mode (Addr. 1Fh, Bit 1)

The CCB has largely been designed to support sub-blocks of both the IEEE 802.11 protocol and an existing GPS protocol. When the Legacy Mode bit is set, the "Data Length" field applies to the frame length excluding the length of the Frame Checksum. When the Legacy Mode bit is clear, the "Data Length" applies to the frame length, excluding the length of the Frame Checksum, plus four (which assumes a 32-bit Frame Checksum is being used). Also in Legacy Mode (bit set), if any of the three MSBs of the second byte of the "Data Length" field are set, only a header is received, with the rest of the length field being ignored.

In Legacy Mode, the 4-level GFSK blocks are disabled. The 2Mbit/s clock is disabled, and the 4-level GFSK mode detection is disabled.

A hardware reset clears the Legacy Mode bit.

5.5.18.3 Data Rate (Addr. 1Fh, Bit 2)

The Data Rate bit switches between 1Mbit/s (data rate bit clear) and 625kbit/s (data rate bit set). The Data Rate bit configures an internal clock called 'bit_clk' which switches between 1µs cycle times and 1.6µs cycle times. The 4-level blocks assume the data rate bit is set to 1Mbit/s mode. A hardware reset clears the Data Rate bit to 1Mbit/s mode.

5.5.18.4 FIFO Access Mode (Addr. 1Fh, Bit 3)

When the FIFO Access Mode bit is set, the processor can write bytes to the FIFO and update the pointers in the same way the MCB does (Section 5.3.2). Also, in FIFO Access Mode, the processor can 'snoop' the contents of the FIFO (read a byte without updating the pointers). When the FIFO Access Mode bit is clear, the FIFO Access Mode functions are disabled. A hardware reset clears the FIFO Access Mode bit.

5.5.18.5 Txd Inverted (Addr. 1Fh, Bit 4)

When the Txd Inverted bit is set, the TXD_H output pin is inverted. The TXD_H pin is the pin used for two level transmissions (including preamble), and is the MSB of 4-level transmissions. The Txd Inverted bit does not effect TXD_L. A hardware reset clears the Txd Inverted bit (i.e. data not inverted).

5.5.18.6 Rxd Inverted (Addr. 1Fh, Bit 5)

When the Rxd Inverted bit is set, the RXD_H input pin is inverted. The RXD_H pin is the pin to receive two level transmissions (including preamble), and is the MSB when receiving 4-level transmissions. The Rxd Inverted bit does not effect RXD_L. A hardware reset clears the Rxd Inverted bit (i.e. data not inverted).

5.5.18.7 TXD_L 'XORed' With TXD_H (Addr. 1Fh, Bit 6)

The TXD_L exclusive OR option is enabled when this bit is set. In 4-level transmissions, if this bit is set, TXD_L will be inverted if the internal TXD_H is set. This option uses TXD_H before it has been effected by Txd Inverted (Section 5.5.18.5). The XOR option will result in the Gray code numbering system used for BSE being translated to a Binary code (see Table 17). A hardware reset enables the TXD_L exclusive OR option.

5.5.18.8 RXD_L 'XORed' With RXD_H (Addr. 1Fh, Bit 7)

The RXD_L exclusive OR option is enabled when this bit is set. When receiving 4-level transmissions, if this bit is set, RXD_L will be inverted if the internal RXD_H is set. This option uses RXD_H after the Rxd Inverted option has been applied (Section 5.5.18.6). The XOR option will result in a Binary coding system being translated to the Gray code numbering system used for BSE. A hardware reset disables the RXD_L exclusive OR option.

Table 16 shows the relationship between the TXD output and the carrier deviation generated by the radio in order to correctly minimise the bias.

Data bit 0,2,4,6 (becomes TXD_H)	Data bit 1,3,5,7 (becomes TXD_L)	Carrier deviation expected by BSE
0	0	-3/2 * h4 * Fclk
0	1	-1/2 * h4 * Fclk
1	1	1/2 * h4 * Fclk
1	0	3/2 * h4 * Fclk

Table 16. TXD output to carrier deviation expected for BSE

Table 17 shows how the BSE coding system can be translated into different sets of codes on the TXD pins. However, the result of applying the new codes to the radio should still result in the same carrier deviations. For example, if the radio uses a binary sequence (instead of a Gray code sequence) to determine carrier deviation, TXD_L XOR should be set, and Txd invert should be clear.

Txd invert (Addr. 1Fh, Bit 4)	TXD_L XOR (Addr. 1Fh, Bit 6)	Car. dev. -3/2*h4*Fclk (TXD H,L)	Car. dev. -1/2*h4*Fclk (TXD H,L)	Car. dev. 1/2*h4*Fclk (TXD H,L)	Car. dev. 3/2*h4*Fclk (TXD H,L)
0	0	0,0	0,1	1,1	1,0
0	1	0,0	0,1	1,0	1,1
1	0	1,0	1,1	0,1	0,0
1	1	1,0	1,1	0,0	0,1

Table 17. TXD output manipulation for alternative carrier deviation encoding

Table 18 shows how the CCB should be configured to translate the code produced for a given received deviation back into the BSE coding system.

Car. dev. -3/2*h4*Fclk (RXD H,L)	Car. dev. -1/2*h4*Fclk (RXD H,L)	Car. dev. 1/2*h4*Fclk (RXD H,L)	Car. dev. 3/2*h4*Fclk (RXD H,L)	Rxd invert (Addr. 1Fh, Bit 5)	RXD_L XOR (Addr. 1Fh, Bit 7)
0,0	0,1	1,1	1,0	0	0
0,0	0,1	1,0	1,1	0	1
1,0	1,1	0,1	0,0	1	0
1,0	1,1	0,0	0,1	1	1

Table 18. Received carrier deviation to BSE coding system

5.5.19 Hash Lookup Table (Addr. 30h to 37h)

The Hash Lookup Table is used as part of the Address Matching options (discussed in Section 5.4.4 and Section 5.8). The lookup table consists of 64 bits which are pointed to by a 6-bit pointer generated from a hash algorithm (the hash algorithm is explained in Section 5.8.8). A pointer value of 0 points to the LSB of Addr.30h, a pointer value of 7 points to the MSB of Addr. 30h, a pointer value of 56 points to the LSB of Addr. 37h, and a pointer value of 63 points to the MSB of Addr.37h. A hardware reset clears all the bits in the Hash Lookup Table.

5.5.20 Unique Address (Addr. 38h to 3Dh)

The Unique Address is used by one of the Address Matching options (it is discussed in Section 5.4.4.2). The Unique Address is used to indicate if the address of the received packet is an exact match with the Unique Address (it would typically be set to the system's own address). The address consists of six bytes, where the first byte received is compared with the contents of Addr. 38h, up to the last byte of the address being compared to Addr. 3Dh. A hardware reset clears all the bits of the Unique Address.

5.5.21 DMA Delay (Addr. 3Eh)

The DMA Delay is the delay between the processor giving the CCB permission to start a DMA transfer (either by setting Upload grant (Section 5.4.1.8) in response to an Upload Request, or by clearing the Commence Transmit bit (Section 5.4.1.4)), and the actual DMA transfer starting. The delay is specified in 'bit_clk' cycles, as discussed in Section 5.5.18.3. The actual delay will be slightly longer than specified, due to internal delays.

A hardware reset loads 28h into the DMA Delay register, which corresponds to a 40 'bit_clk' cycle delay.

5.6 Status Registers

The status registers provide feedback on the state of the CCB, and are read-only. The registers indicate the cause of an interrupt, give an indication of radio state (and whether a channel is likely to be clear to transmit), and the result of the Checksums. The status registers also indicate the state of the FIFO pointers.

5.6.1 Interrupt States (Addr. 20h)

5.6.1.1 CCA Timer Equal Zero (Addr. 20h, Bit 0)

The CCA Timer Equal Zero bit will be set when the CCA Timer is equal to zero. If the CCA Timer Equal Zero interrupt mask bit is cleared (Section 5.4.5.1), an interrupt will be generated when the CCA timer is decremented from one to zero.

The CCA Timer Equal Zero bit is generated by inverting the result of 'ORing' all of the bits of the CCA Timer. Therefore, to clear the CCA Timer Equal zero bit, a non-zero value should be loaded into the CCA Timer. The CCA Timer is loaded with zero by a hardware reset, and so the CCA Timer Equal Zero bit will be set by a hardware reset.

5.6.1.2 Header Upload Request (Addr. 20h, Bit 1)

The Header Upload Request bit is set as soon as the Frame Word has been found, to indicate to the processor that a header has been received and is ready to be transferred to the MCB. If the Upload Header Request interrupt mask bit is cleared (Section 5.4.5.2), an interrupt will be generated at the same time the Header Upload Request bit is set.

It is cleared by a software or hardware reset, at the time the frame Upload Grant is given, or as a result of setting the Commence Receive bit (i.e. at the end of the receive cycle).

5.6.1.3 Upload Request (Addr. 20h, Bit 2)

The Upload Request bit is set at the same time as the Header Upload Request bit, and also as soon as the Header upload completes, to indicate that the Frame is ready to be uploaded. Interrupts due to either a Header Upload Request or Frame Upload Request can be masked by the Upload Header Request interrupt mask bit (Section 5.4.5.2) and the Upload Frame Request interrupt mask bit (Section 5.4.5.3).

The Upload Request bit is cleared by a software or hardware reset, after an Upload Grant sequence has started, or as a result of setting the Commence Receive bit (i.e. at the end of the receive cycle).

5.6.1.4 Frame Upload Complete (Addr. 20h, Bit 3)

The Frame Upload Complete bit is set after the received packet has been transferred to the MCB, to indicate to the processor that the CCB has finished using the Memory Control Block (MCB). An interrupt will be generated unless the Frame Upload Complete interrupt mask (Section 5.4.4.4) is set. N.B. In the event of both the Frame Upload Complete bit and the Upload Request bit being set, the Upload Request bit should be ignored.

The Frame Upload Complete bit is cleared by a software or hardware reset, or as a result of setting the Commence Receive bit (i.e. at the end of the receive cycle).

5.6.1.5 Tx. DMA Transfer Complete (Addr. 20h, Bit 4)

The Tx. DMA Transfer Complete bit is set in transmit mode, when the last byte of data has been transferred from the MCB to the CCB. It indicates to the processor that the CCB has finished using the MCB. However, the CCB may still be in the process of transmitting the last of the data. An interrupt will be generated at the same time as the Tx. DMA Transfer Complete bit is set unless the Tx. DMA Transfer Complete interrupt mask bit is set.

The Tx. DMA Transfer Complete bit is cleared by a software or hardware reset, or by the CCB finishing the Tx. cycle (indicated by RXTX bit of Addr. 21h).

5.6.1.6 FIFO Read Error (Addr. 20h, Bit 5)

The FIFO Read Error bit will be set if the FIFO is read when it is empty. An interrupt will be generated at the same as the read error is generated unless the FIFO Read Error interrupt mask bit is set. If a FIFO Read Error Interrupt is generated during a DMA transfer, the DMA transfer will be aborted.

A FIFO read error can only be cleared by a software or hardware reset.

5.6.1.7 FIFO Write Error (Addr. 20h, Bit 6)

The FIFO Write Error bit will be set if the FIFO is written to when it is full. An interrupt will be generated at the same as the write error is generated unless the FIFO Write Error interrupt mask bit is set. If a FIFO Write Error interrupt is generated during a DMA transfer, the DMA transfer will be aborted.

A FIFO Write Error can only be cleared by a software or hardware reset.

5.6.1.8 Radio PLL Unlocked Latch (Addr. 20h, Bit 7)

If a phase-locked loop (PLL) on the radio becomes unlocked, the radio will set the LCK_DETB input to the CCB high. The Radio PLL Unlocked Latch bit will stay high as soon as it detects the LCK_DETB signal is high. It will also generate an interrupt if the Synth. Unlock interrupt mask bit is low. Bit 5 of Addr. 21h indicates the un-latched state of LCK_DETB.

The Radio PLL Unlocked Latch bit is cleared by a software or hardware reset, or the interrupt clear bit (Section 5.4.1.6).

5.6.2 Radio State (Addr. 21h)

5.6.2.1 Clear Channel Assessment Indicator (Addr. 21h, Bit 0)

The DE6038 radio produces a signal called CCA, which gives an indication of the RF power received at the antenna port. When Commence Receive is low, the CCA Indicator will be the same as the CCA input pin. When Commence Receive is high, the CCA Indicator will be low.

5.6.2.2 'Long' Indicator (Addr. 21h, Bit 1)

The 'Long' Indicator will be set if the number of consecutive '1's or '0's received exceeds the limit for the Bit stuffing or BSE option (Section 5.5.17.5). If no bit stuffing or BSE option is used, 'Long' will never be set. If BSE 16/17 is used, the maximum period with no edges in the data is 42 bit periods. If BSE 32/33 is used, the maximum period with no edges in the data is 83 bit periods. If Bit stuffing after 8 is used, the maximum period with no edges in the data is 9.5 bit periods. If Bit stuffing after 16 is used, the maximum period with no edges in the data is 17.5 bit periods.

The 'Long' Indicator is cleared by a software or hardware reset, if Commence Receive is high, or if the number of occurrences of consecutive valid edges meets the data threshold (Section 5.5.11.3).

5.6.2.3 'Noise' Indicator (Addr. 21h, Bit 2)

The 'Noise' Indicator will be set if the number of occurrences of invalid data (discussed in Section 5.5.11.1) exceeds the noise threshold (Section 5.5.11.2).

The 'Noise' Indicator is cleared by a software or hardware reset, if Commence Receive is high, when the 'Long' bit is set, or if the number of occurrences of consecutive valid edges meets the data threshold (Section 5.5.11.3).

5.6.2.4 'Sync' Indicator (Addr. 21h, Bit 3)

The 'Sync' Indicator will be set in receive mode, when the number of consecutive 'good' Sync. Words (Section 5.5.6) exceeds the number specified in the No. Sync. Words Required for 'Sync' register (Section 5.5.10.1). If the Bit Errors Allowed bit is set, every other Sync. Word received can contain a single bit error, and still be considered good (i.e. a single bit error will be allowed in a Sync. Word, but if more than one bit error occurs, or if two consecutive Sync. Words contain single bit errors, the Sync. Word will be considered to be wrong). Once the 'Sync' bit is set, the clock recovery circuit switches into it's narrow loop mode (i.e. it re-calibrates ideal sampling time slowly), and it can start to try to identify the Frame Word (Section 5.5.8). During reception of the Frame Word, the Sync bit will remain set unless their is an error in the received Frame Word. If the Frame Word is identified correctly, the 'Sync' indicator will remain set until the end of the receive cycle.

The 'Sync' Indicator is cleared by a software or hardware reset, if bit errors are not allowed and the received data is not equal to either the Sync. Word or the Frame Word, if bit errors are allowed and their is more than one error in two consecutive Sync. Words (no errors allowed in Frame Word), or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.2.5 RXTX Indicator (Addr. 21h, Bit 4)

The RXTX Indicator indicates the state of the radio. The RXTX Indicator will be low in transmit mode and high in receive/inactive mode. (The signal which is output to the radio will be the inverted version of this in DE6038 mode)

5.6.2.6 LCK_DET Indicator (Addr. 21h, Bit 5)

The LCK_DET Indicator shows the state of the input pin from the radio. A high indicates an error condition for both the DE6003 radio and the DE6038 radio.

5.6.2.7 Internal Commence Receive Indicator (Addr. 21h, Bit 6)

The Internal Commence Receive bit is active low. It is the signal which triggers the receive cycle. The Internal Commence Receive bit will only go low if the Commence Transmit Register is high (Section 5.4.1.4), the Commence Receive Register is low (Section 5.4.1.3), and the RXTX bit indicates the radio is not in transmit mode.

5.6.2.8 DMA Transfer (Addr. 21h, Bit 7)

The DMA Transfer bit is active low. In transmit mode, when Commence Transmit is cleared, the DMA Transfer bit will go low after a period specified by the DMA delay (Section 5.5.21). The DMA Transfer bit will then stay low until all the bytes have been transferred from the MCB and the Tx. DMA Transfer Complete bit is set (Section 5.6.1.5). In receive mode, when the Upload Grant is given, the DMA Transfer bit will go low after a period specified by the DMA delay (Section

5.5.21). The DMA Transfer bit will then stay low until the transfer to the MCB completes, and either the Frame Upload Complete or Upload Request bits are set (Section 5.6.1.3, 5.6.1.4).

The processor should not attempt a FIFO access or an MCB transfer while the DMA Transfer bit is low.

5.6.3 CRC State (Addr. 22h)

5.6.3.1 Frame Checksum Error (Addr. 22h, Bit 0)

The Frame Checksum Error bit will be set if an error is detected in the received Frame Checksum. If a non-complemented Checksum is used 5.5.17.3), the expected result for both a 16-bit Checksum and a 32-bit Checksum is zero. If a complemented Checksum is used, the expected remainder for a 16-bit Checksum is F0B8h. The expected remainder for a complemented 32-bit Checksum is DEBB20E3h.

The Frame Checksum Error bit will be cleared by a software or hardware reset, or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.3.2 Header Checksum Error (Addr. 22h, Bit 1)

The Header Checksum Error bit will be set if an error is detected in the received Header Checksum. The expected remainder for the Header Checksum is identical to the expected remainder for a Frame Checksum (Section 5.6.3.1).

The Header Checksum Error bit will be cleared by a software or hardware reset, or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.3.3 Frame Checksum Valid (Addr. 22h, Bit 2)

The Frame Checksum Valid bit will be set when the received Frame Checksum has been tested. When the Frame Checksum Valid bit is set, both the Header and the Frame Checksum Error indicators will be valid, and the Checksum register (Section 5.6.4) will contain the remainder from the Frame Checksum.

The Frame Checksum Valid bit will be cleared by a software or hardware reset, or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.3.4 Header Checksum Valid (Addr. 22h, Bit 3)

The Header Checksum Valid bit will be set when the received Header Checksum has been tested. When the Header Checksum Valid bit is set, the Header Checksum Error indicator will be valid, and the Checksum register (Section 5.6.4) will contain the remainder from the Header Checksum.

The Header Checksum Valid bit will be cleared by a software or hardware reset, when the Frame Checksum Valid bit is set, or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.3.5 Address Hit Indicator (Addr. 22h, Bit 4)

The Address Hit Indicator will be set in receive mode if the selected Address Matching Scheme results in a match. The Address Hit bit will be cleared by a software or hardware reset, or when the Commence Receive bit is set high (i.e. to end receive cycle).

5.6.4 Checksum Register (Addr. 24h to 27h)

The Checksum register will be loaded with the remainder from the Head Checksum at the same time as the Header Checksum Valid indicator is set (Section 5.6.3.4). The Checksum register will be overwritten with the remainder from the Frame Checksum at the same time as the Frame Checksum Valid Indicator is set (Section 5.6.3.3). Addr. 24h will contain the eight LSBs of the Checksum, Addr. 25h will contain the eight MSBs of a 16-bit Checksum, and Addr. 27h will contain the eight MSBs of a 32-bit Checksum.

A software or hardware reset will set all the bits of the Checksum register.

5.6.5 FIFO In-Pointer (Addr. 28h)

The FIFO In-Pointer is a 6-bit number (Bits 5-0) which indicates the location at which the next byte written to the FIFO will be stored (i.e. used by radio side during receive and MCB side during transmit). Addr. 28h only contains a copy of the In-Pointer. The processor cannot write to the FIFO In-pointer.

5.6.6 FIFO Out-Pointer (Addr. 29h)

The FIFO Out-Pointer is a 6-bit number (Bits 5-0) which indicates the location of the next byte to be read from the FIFO (i.e. used by radio side during transmit and MCB side during receive). Addr. 29h only contains a copy of the Out-Pointer. The processor cannot write to the FIFO Out-Pointer.

5.7 CCB Address Map

Address (in Hex.)	Bit	Type	Name (common abbreviation)	Reset Value	Section reference
00	0	Cont.	Configuration Enabled	1	5.4.1.1
	1	Cont.	Channel Hop (CH)	1	5.4.1.2
	2	Cont.	Commence Receive (CR)	1	5.4.1.3
	3	Cont.	Commence Transmit (CT)	1	5.4.1.4
	4	Cont.	CCB Software Reset	1	5.4.1.5
	5	Cont.	Interrupt Clear	0	5.4.1.6
	6	Cont.	Standby (radio)	0	5.4.1.7
	7	Cont.	Upload Grant	0	5.4.1.8
01	0-7	Cont.	Channel Select	6Bh	5.4.2
02	0-7	Cont.	Channel Select (DE6038 Rx. only)	30h	5.4.3
	0	Cont.	Antenna Select (DE6003 only)	0	5.4.3.1
	1	Cont.	RF Power Level (DE6003 only)	0	5.4.3.2
03	0	Cont.	Receive Broadcast	0	5.4.4.1
	1	Cont.	Receive Unique Address	0	5.4.4.2
	2	Cont.	Receive Unicast (hash lookup/all)	1	5.4.4.3
	3	Cont.	Receive Multicast(hash lookup/all)	1	5.4.4.4
	4	Cont.	Receive All	1	5.4.4.5
	5	Cont.	Define Header Upload Length	1	5.4.4.6
	6	Cont.	CCA Timer Run	0	5.4.4.7
	7	Cont.	CCA Timer Always(1) / CCA(0)	0	5.4.4.8
04	0	Cont.	CCA Timer Interrupt Mask	0	5.4.5.1
	1	Cont.	Upload Header Request Interrupt Mask	0	5.4.5.2
	2	Cont.	Upload Frame Request Interrupt Mask	0	5.4.5.3
	3	Cont.	Frame Upload Complete Interrupt Mask	0	5.4.5.4
	4	Cont.	Tx. DMA Transfer Complete Intr. Mask	0	5.4.5.5
	5	Cont.	FIFO Read Error Interrupt Mask	0	5.4.5.6
	6	Cont.	FIFO Write Error Interrupt Mask	0	5.4.5.7
	7	Cont.	Synth Unlocked Interrupt Mask	1	5.4.5.8
05	0-7	FIFO	FIFO Write register	(00h)	5.4.6
06	8-15	Cont.	8 MSB of CCA Timer Register	00h	5.4.7
07	0-7	Cont.	8 LSB of CCA Timer Register	00h	5.4.7
08-0B	0-31	Config.	Net Key (Checksum seed)	FFFF-FFFFh	5.5.1
0C	0-6	Config.	Header Upload Position	10h	5.5.2.2
0C	7	Config.	Header Upload Field	1	5.5.2.1
0D	0-7	Config.	Rx. to Tx. Switch Time	6Eh	5.5.3
0E	0-3	Config.	Ramp-up Time	06h	5.5.4
	4-7	---	Not used	(0)	
0F	0-7	Config.	Ramp-down Time	09h	5.5.5
10	0-7	Config.	Sync. Word (SW)	02h	5.5.6
11	0-2	Config.	No. of Bits of Sync Word (BSW)	01h	5.5.7.1
	3-7	Config.	No. of Bits of Frame Word (BFW)	0Fh	5.5.7.2
12-15	0-31	Config.	Frame Word (FW)	0000-BD30h	5.5.8
16	0-7	Config.	No. of Transmitted Sync. Words	28h	5.5.9
17	0-2	Config.	No. Sync. Words Required for 'Sync'	2	5.5.10.1
	3	Config.	Bit Errors Allowed (BE)	0	5.5.10.2
	4-7	---	Not used	(0)	
18	0-1	Config.	Jitter Tolerance (JT)	1	5.5.11.1
	2-3	Config.	Noise Threshold (NT)	3	5.5.11.2
	4-5	Config.	Data Threshold (DT)	1	5.5.11.3
	6-7	---	Not used	(0)	
19	0-7	Config.	Packet Length Start Byte Position	01h	5.5.12

Address (in Hex.)	Bit	Type	Name (common abbreviation)	Reset Value	Section reference
1A	0-7	Config.	Data Length Mask	0Fh	5.5.13
1B	0-6	Config.	Address Location Position	04h	5.5.14.2
	7	Config.	Address Location Field	1	5.5.14.1
1C	0-6	Config.	Frame Checksum Start Position	2	5.5.15
	7	Config.	Frame Checksum Start Field	0	5.5.15
1D	0-7	Config.	Header Checksum Insertion Position	2	5.5.16
1E	0	Config.	16/32 bit Frame Checksum	0	5.5.17.1
	1	Config.	16/32 bit Header Checksum	1	5.5.17.2
	2	Config.	Ones Complement of Checksum	1	5.5.17.3
	3	Config.	Scrambler Enabled	0	5.5.17.4
	4-6	Config.	Bit stuffing / BSE	3	5.5.17.5
	7	---	Not used	(0)	
1F	0	Config.	DE6003/DE6038 Mode	1	5.5.18.1
	1	Config.	Legacy Mode	0	5.5.18.2
	2	Config.	Data Rate	0	5.5.18.3
	3	Config.	FIFO Access Mode	0	5.5.18.4
	4	Config.	Txd Inverted	0	5.5.18.5
	5	Config.	Rxd Inverted	0	5.5.18.6
	6	Config.	TXD_L 'XORed' with TXD_H	1	5.5.18.7
	7	Config.	RXD_L 'XORed' with RXD_H	0	5.5.18.8
20	0	Status	CCA Timer Equal Zero	(1)	5.6.1.1
	1	Status	Header Upload Request	(0)	5.6.1.2
	2	Status	Upload Request	(0)	5.6.1.3
	3	Status	Frame Upload Complete	(0)	5.6.1.4
	4	Status	Tx. DMA Transfer Complete	(0)	5.6.1.5
	5	Status	FIFO Read Error	(0)	5.6.1.6
	6	Status	FIFO Write Error	(0)	5.6.1.7
	7	Status	Radio PLL unlocked latch	(0)	5.6.1.8
21	0	Status	Clear Channel Assessment (CCA)	(i/p)	5.6.2.1
	1	Status	'Long' Indicator	(0)	5.6.2.2
	2	Status	'Noise' Indicator	(0)	5.6.2.3
	3	Status	'Sync' Indicator	(0)	5.6.2.4
	4	Status	RXTX Indicator	(1)	5.6.2.5
	5	Status	LCK_DET Indicator	(i/p)	5.6.2.6
	6	Status	Internal Commence Receive Indicator	(1)	5.6.2.7
	7	Status	DMA Transfer	(1)	5.6.2.8
22	0	Status	Frame Checksum Error	(0)	5.6.3.1
	1	Status	Header Checksum Error	(0)	5.6.3.2
	2	Status	Frame Checksum Valid	(0)	5.6.3.3
	3	Status	Header Checksum Valid	(0)	5.6.3.4
	4	Status	Address Hit Indicator	(0)	5.6.3.5
	5-7	---	Not used	(0)	
23	0-7	---	Not used	(0)	
24-27	0-31	Status	Checksum Register	(FFFF-FFFFh)	5.6.4
28	0-5	Status	FIFO In-Pointer	(0)	5.6.5
	6-7	---	Not used	(0)	
29	0-5	Status	FIFO Out -Pointer	(0)	5.6.6
	6-7	---	Not used	(0)	
2A-2F		---	Not used	(0)	
30-37	0-63	Config.	Hash Lookup Table	0000h	5.5.19
38-3D	0-47	Config.	Unique Address	0000h	5.5.20
3E	0-7	Config.	DMA Delay	28h	5.5.21
3F		---	Not used		
40-7F		FIFO	FIFO read		5.3.2

Table 19. CCB Address Map

5.8 Address Matching schemes

During the reception of a packet, the CCB provides the facility to give an indication to the processor if it is likely to want the packet based on the address contained in the packet.

The CCB provides the following address matching schemes:

- Broadcast Address
- Unique Address
- Unicast Address hash lookup
- Any Unicast Address
- Multicast Address hash lookup
- Any Multicast Address
- All Address hash lookup

A combination of address matching schemes can be used by enabling more than one of the address matching options.

5.8.1 Broadcast Address match

A Broadcast Address is defined as all the bits of all 48 bits of the address being set. If the received address (see Section 5.5.14 for Address Location) is a broadcast address and if the Receive Broadcast bit is set (see Section 5.4.4.1), the Address Hit indicator will be set (Section 5.6.3.5).

5.8.1.1 Unique Address match

Addr. 38h to 3Dh can be loaded with a unique six byte address (usually the unit's own address). If the received address (see Section 5.5.14 for Address Location) is an exact match to the previously loaded unique address and the Receive Unique Address bit is set (see Section 5.4.4.2), the Address Hit indicator will be set (Section 5.6.3.5).

5.8.2 Unicast Address hash lookup match

The Least Significant Bit of the first byte of the address is used to distinguish between a multicast address (if the bit is set) and a unicast address (if the bit is clear). If Receive Unicast is set (Section 5.4.4.3) and Receive All is clear (Section 5.4.4.5), the Address Hit indicator will be set if the received address is a unicast address which generates a hash hit (Section 5.8.8).

5.8.3 Any Unicast Address match

If Receive Unicast (Section 5.4.4.3) and Receive All (Section 5.4.4.5) are set, the Address Hit indicator will be set if the received address is a unicast address.

5.8.4 Multicast Address hash lookup match

If the multicast bit of an address is set (LSB of first byte), Receive Multicast is set (Section 5.4.4.4) and Receive All is clear (Section 5.4.4.5), the Address Hit indicator will be set if the received address is a multicast address which generates a hash hit (Section 5.8.8).

5.8.5 Any Multicast Address match

If Receive Multicast (Section5.4.4.4) and Receive All (Section 5.4.4.5) are set, the Address Hit indicator will be set if the received address is a multicast address.

5.8.6 All Address hash lookup match

To generate a hash lookup hit irrespective of the state of the multicast/unicast bit, both the Receive Unicast bit and the Receive Multicast bit should be set, and the receive all bit should be clear.

N.B. If Receive Unicast, Receive Multicast and Receive All are set, all addresses will generate an Address Hit.

5.8.7 Hash lookup options

The hash lookup options involve passing the six bytes of the address through an algorithm. The algorithm will generate a six bit number which is used to point to a bit in the hash lookup table (Section 5.5.19). If the bit pointed to in the hash lookup table is set, a hash hit is generated. The six bits of the hash pointer are generated as follows:

HashPoint[5] = AB1[7] ^ AB1[1] ^ AB2[3] ^ AB3[5] ^ AB4[7] ^ AB4[1] ^ AB5[3] ^ AB6[5]
HashPoint[4] = AB1[6] ^ AB1[0] ^ AB2[2] ^ AB3[4] ^ AB4[6] ^ AB4[0] ^ AB5[2] ^ AB6[4]
HashPoint[3] = AB1[5] ^ AB2[7] ^ AB2[1] ^ AB3[3] ^ AB4[5] ^ AB5[7] ^ AB5[1] ^ AB6[3]
HashPoint[2] = AB1[4] ^ AB2[6] ^ AB2[0] ^ AB3[2] ^ AB4[4] ^ AB5[6] ^ AB5[0] ^ AB6[2]
HashPoint[1] = AB1[3] ^ AB2[5] ^ AB3[7] ^ AB3[1] ^ AB4[3] ^ AB5[5] ^ AB6[7] ^ AB6[1]
HashPoint[0] = AB1[2] ^ AB2[4] ^ AB3[6] ^ AB3[0] ^ AB4[2] ^ AB5[4] ^ AB6[6] ^ AB6[0]

Where:- AB1[0] = LSB of first Byte of Address
AB6[7] = MSB of sixth Byte of Address
^ = Exclusive-OR

This is generated by:

HashPoint initialised to 0;
Then, for each byte of address:

New HashPoint[0] = Old HashPoint[4] XOR AB[0] XOR AB[6]
New HashPoint[1] = Old HashPoint[5] XOR AB[1] XOR AB[7]
New HashPoint[2] = Old HashPoint[0] XOR AB[2]
New HashPoint[3] = Old HashPoint[1] XOR AB[3]
New HashPoint[4] = Old HashPoint[2] XOR AB[4]
New HashPoint[5] = Old HashPoint[3] XOR AB[5]

5.9 Clock Recovery and Data Synchronisation

The Receiver must generate a clock for the Receiver circuitry at a clock rate determined by the transmitted data rate. Although the data rate is known by the Receiver, the Receiver must synchronise its clock with the incoming data stream so that both are in phase with each other. Only then can the Receiver reliably sample the incoming data stream at sampling times midway between bit transitions.

After synchronisation has been achieved, the clock recovery circuit switches into a narrow loop mode. In the narrow loop mode the Receiver's clock is only very slowly adjusted in response to a number of occurrences of early or late edges in the received data. This is to compensate for any differences between the receiver's reference clock and the Transmitter's reference clock, due to crystal tolerances.

Clock Recovery is hindered by:

1. Long runs of ones and zeros in the data stream
2. Multipath and bit period distortion
3. Channel noise

Due to these factors, tolerances must normally be allowed so that clock recovery may be achieved. However relaxing the tolerances increases the likelihood of false clock recovery.

5.9.1 Long runs of ones and zeros in the data stream

The Clock Recovery circuit is 'triggered' by edges in the received data stream. Therefore, if there are long runs of ones or zeros, there will not be enough edges for the clock recovery circuit to synchronise to. The clock recovery circuit would ideally see a change in data state every bit period in order to synchronise to the bit clock quickly.

N.B. The DE6003 specification requires that the maximum number of contiguous '1's or '0's should be limited to 16.

5.9.2 Multipath and bit period distortion

The RF signal that travels from the Transmitter to the Receiver suffers multipath distortion. The effect of this is that the demodulated data stream fed into the WL102 will suffer jitter. Jitter is a measure of the variation in the time of arrival of actual edges in the received data stream, compared to the expected time of arrival of edges (if all bit periods were identical). The actual edges in the received data may occur slightly sooner or slightly later than expected. To accommodate

this jitter, a jitter tolerance option can be used which allows a valid transition to occur anywhere within a temporal band around the expected transition time.

The Jitter Tolerance is set by JT (Addr. 18, bits 0 and 1), and can be setto +/-12.5%, +/-25% or +/-37.5% for 625kbit/s data rate, or +/-20% or +/-40% for 1Mbit/s (percentage of a symbol period for the current data rate). If clock recovery is difficult to achieve, and if the Receiver-Transmitter are being used in an environment where there are many reflective surfaces which promote multipath distortion, then a large jitter tolerance may be necessary. However, the jitter tolerance should be kept as small as possible to minimise the likelihood of false clock recovery.

5.9.3 Channel noise

Channel noise may be seen as glitches, or noise spikes, in the incoming data stream. These glitches are a product of the Receiver circuitry and the nature of RF transmission and cause problems for clock recovery.

Prior to synchronisation and the 'Sync' bit (Addr. 21, bit 3) being set high, the WL102 clock and data recovery circuit is in an open loop mode, coarsely re-adjusting the recovered clock on every bit. Occasionally, due to jitter, the clock may be off. Once synchronisation is achieved, the WL102 switches to a narrow loop mode and adjusts the clock at a slower rate based on the trend of the previous bits.

5.9.4 Synchronisation

Once clock recovery has been achieved and a clock is being generated, it is necessary to perform data synchronisation. The purpose of data synchronisation (from now on simply called synchronisation) is to allow the Receiver to know when the preamble ends and when the user data starts. Synchronisation is achieved by locking onto a repetitive stream of sync words, followed by detecting a frame delimiter bit sequence. The end of the frame delimiter bit sequence signifies the end of the preamble and the beginning of the user data (Section 5.2 describes generic packet format).

5.9.4.1 Locking onto the sync words

The sync word bit pattern is entered into SW (Addr. 10, bits 7 to 0). The number of bits of the sync word used is then entered into BSW (Addr. 13, bits 2 to 0). If the number of bits in the sync word is less than eight, then the sync word used is formed from the LSBs of SW. For example, if BSW is set to 0,1,1 then the sync word is taken as SW3 to SW0 inclusive.

BSW (BSW2,BSW1,BSW0)	Number of bits in a sync word
0,0,0	Not used
0,0,1	2
0,1,0	3
0,1,1	4
1,0,0	5
1,0,1	6
1,1,0	7
1,1,1	8

Table 20. The number of bits in a sync word for different BSW values

It should be noted that the sync word bit pattern should be chosen such that it includes sufficient transitions in the data to enable the clock recovery circuit to work efficiently. Also, the bit pattern for an individual sync word should not contain repetitive elements, so that the state machine receiving the data cannot mistake the middle of a sync word for the start of a sync word (see Section 5.9.5.1 for further details).

The total number of sync words to be transmitted in the synchronisation sequence is specified in Addr. 16h. The number of sync words that the Receiver must recover before it decides that synchronisation has been achieved (and sets the Sync. status bit high) must be specified in Addr. 17, bits 0 to 2. Once synchronisation has been achieved and the Sync bit (see Addr. 21, bit 3) set high, it remains high until the end of the data reception, or until synchronisation is lost.

The greater the number of sync words in the preamble (as specified in Addr. 16h), the more time there will be for the Receiver to lock onto the sync words and hence obtain synchronisation. It may be difficult to achieve synchronisation in noisy environments if the number of sync words used is too few. An increased number of sync words in the preamble will also allow the user to demand a greater number of sync words to be recovered before synchronisation is achieved. This strict demand on synchronisation will reduce the probability of false synchronisation. However, the user must remember that as the number of sync words is increased, so the redundancy overhead in the transmitted data is also increased, so reducing the information rate for a given data rate.

5.9.4.2 Setting the Bit Errors Allowed register (BE)

A further accommodation may be made for noisy environments. In a noisy environment, noise may corrupt the transmitted data packet. If this occurs, then it should be detected by the Checksums or alternatively a user-defined error control code. If the preamble is corrupted, then this does not affect the user data but will cause loss of synchronisation. To accommodate this, the BE configuration bit (Addr 17, bit 3) can be set high to allow single bit errors. This means that after synchronisation has been achieved, a single bit error in a sync word will not put the receiver out of synchronisation (if the corrupted sync word is followed by a correct sync word). If BE is not set high, then once a single bit error in a sync word is detected, then synchronisation will be lost and the Sync flag will be set low, independent of the following sync word. However, if more than one bit error occurs within two consecutive sync words, then Sync is set low independent of the setting of the BE configuration bit.

Therefore, if the BE configuration bit is set high and if synchronisation has been achieved, then a worst case sequence of S,s,S,s,S,s,S,s...(where 'S' is a correct sync word, and 's' is a sync word with one bit corrupted) will not throw the Receiver out of synchronisation. Further clarification should be possible by inspecting the state transition diagram in Figure 10. However, it should be noted that, if possible, BE should be set low to minimise the probability of false synchronisation.

5.9.4.3 Detecting the Frame Delimiter word

Once synchronisation has been achieved ('Sync' set high) the Receiver then starts looking for the next sync word or the frame delimiter word. When it detects the frame delimiter word, it sets the Header Upload Request status bit (Addr. 20, bit 1) high, and may generate an interrupt.

The bit sequence for the frame delimiter word must be entered into FW (Addr. 12h, 13h, 14h and 15h). The number of bits in the frame delimiter word must be entered into BFW (Addr. 11h, bits 3 to 7). The maximum size of the frame delimiter word is 32 bits. If BFW specifies a number of bits less than 32, then the LSBs of FW are used as the frame delimiter.

To increase performance, it is important to have a distinctive and unique pattern of bits for the frame delimiter word. The frame delimeter word is detected by a state machine. Therefore, the frame delimeter word should not start with a bit sequence identical to the Sync Word, because the state machine testing for the frame delimiter word may start in the wrong place i.e. mistake the last Sync. Word for the start of a frame delimiter, then when the test for the frame delimiter fails, it will reset the frame delimeter state machine at which point it may have already missed the start of the real frame delimeter.

5.9.5 Choosing the sync word and the frame delimiter word

To minimise false sync and frame detection, it is preferable that the sync and frame words be unique. Uniqueness means that the bit code to be detected is not part of the user-data field nor a data field from a 'foreign' network. The synchronisation and frame delimiter detection functions must meet requirements for minimum false detects. This can be accomplished by one of two options:

1. The first option involves using a pseudo-unique word of many bits as the entire preamble; this word must allow clock recovery, synchronisation and frame delimiter detection. If this single word is repeated elsewhere in the data packet, most probably in the user data field, then there will be a risk of false data recovery. Therefore to minimise this risk, the word chosen must contain many bits to decrease the possibility of the 'natural' occurrence of this bit sequence.
2. The alternative option for the preamble is to use a much shorter bit sequence which is a unique flag. The uniqueness is ensured by bit stuffing. If the short preamble is repeated 'naturally' as a bit sequence elsewhere in the data packet, then a redundant bit is inserted in a controlled manner into that bit sequence to change it. The net result is that we guarantee a particular bit sequence to be unique to the preamble. However this technique is often undesirable for wireless communications since the length of the packet is now dependent on the number of stuffed bits introduced and hence the packet length becomes variable and unpredictable. For this reason the former option is recommended.

Longer frame delimiters have a lower probability of causing a false frame detect and so are more reliable. However this requirement conflicts directly with the need for minimum detect time. Furthermore, clock recovery should be achieved before synchronisation and frame delimiter detection is attempted, otherwise erroneous bit boundaries will probably cause some false detection.

A sync bit pattern should ideally have a noticeably peaked auto-correlation function. Furthermore the bit pattern should ideally have a uniform, or flat, cross-correlation with the other parts of the data packet, including the user data field. If this is the case, then there is a minimum probability that the bits in the data stream will be corrupted into sync words, therefore the probability of false synchronisation is minimised. It should be noted that the synchronising circuit performs its synchronisation by, in effect, shifting the incoming data stream past the known sync word bit pattern. When there is an exact match between the two, the WL102 locks onto the data stream and will attempt to recover a programmed number of sync words before it sets the Sync bit high and acknowledges that synchronisation has been achieved.

As with the sync word, the frame delimiter word should display the property of a highly peaked auto-correlation function and a flat cross-correlation with all other data transmitted. Simply, the sync and frame delimiter words chosen must be distinctive and as unlikely as possible to occur 'naturally' in the transmitted data, be it friendly or foreign.

Clock recovery from sync words requires many bit transitions in the data stream. A dot clock, a long repeating 1010 symmetrical pattern, is often used to recover the clock from serial bit streams over various media. However in terms of auto-correlation, the auto-correlation function equals zero on odd shifts and one on even shifts; the peaking is not discernible and does little to identify the source of the transmission (i.e. friendly or competing network). Therefore some asymmetry in the bit pattern used may be desirable.

5.9.5.1 Cautionary note and an example

The sync word uniqueness versus word length should not be overlooked. For instance take the extreme case of the two WL102 synchronisation configurations detailed in Table 21 and Figure 9. Both configurations will generate a preamble of 20 bytes of 0,1 bit sequence. Both require 2 bytes of 0,1s to achieve synchronisation. However, the first configuration will have difficulty maintaining synchronisation.

The sync word 0,1,0,1,0,1,0,1 is a non-distinctive pattern and it is very likely that the Receiver will slip the start and stop of a sync word byte by 2 to 6 bits. Then when the frame delimiter word arrives, the WL100 may confuse the first 2 to 6 bits of the frame delimiter word with the sync word. This confusion will result in a synchronisation failure.

As a solution to this, it is suggested that the sync word be set as a 2 bit word of 0,1 and the corresponding WL100 registers changed accordingly. In Table 21, SW is set to 0,0,0,0,0,0,0,1 and BSW set to 0,0,1. Alternatively, the bit sequence 0,1,0,1,0,1,0,1 could still be written into SW, because BSW would be set to 0,0,1 to specify only 2 bits were to be used in the sync word).

Configuration register	Address	Data (binary)		Comments
		Config. 1	Config. 2	
SW	12 (7:0)	01010101	00000001	Sync Word: 01010101, 01
BSW	13 (2:0)	111	001	# Of Bits In Sync Word : 8 Bits, 2 Bits
TSW	18 (7:0)	00010100	01010000	# Of Transmitted Sync Words: ~ 20 Bytes
NSW	19 (5:3)	001	111	# Of Sync Words For Sync: ~2 Bytes

Table 21. Sync word configurations

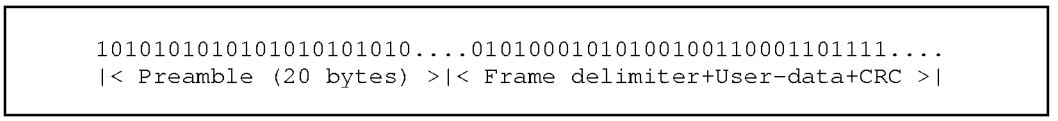

Figure 9. Example preamble and typical user data

The sync word bit pattern should not only have a low cross-correlation with the bit pattern of the frame delimiter word, but also have a low cross-correlation with the bit patterns of the user-data, and with the bit patterns produced by other wireless networks. It is particularly important that the cross-correlation between the frame delimiter word and the sync word for zero offset should be as low as possible (i.e. there should be minimum risk of mistaking the start of the frame delimiter word for a sync word, and conversely, a sync word for the start of the frame delimiter word). A low cross-correlation at zero-offset would minimise the risk of altogether failing to detect, or falsely detecting, the frame delimiter, and so would enable there to be a higher data throughput.

Furthermore, the auto-correlation function of the sync word should be highly peaked and should have only one peak. If the auto-correlation function had a number of peaks, then with slight corruption of the sync word, or with BE set high, the middle of the sync words may become mistaken for the start of the sync words. The Receiver may lock onto the middle of the sync words rather than the start of the sync words, and hence it would fail to pick up the start of the frame delimiter word and there would be failure in frame delimiter detection. Therefore frame delimiter detection failure is minimised if the sync word autocorrelation function is highly peaked, and with only one peak. This argument is the same as that which was applied in the above example, against using the non-distinctive sync word 0,1,0,1,0,1,0,1. In general sync words with repetitive elements should be avoided.

However it is not necessary for the frame delimiter word to have a singly and highly peaked auto-correlation function. This is because the Receiver should, by virtue of the sync words, know when to expect the start of the frame delimiter word. The Receiver should not be allowed to mistake the middle of the frame delimiter word for the start of the frame delimiter word.

The requirement for low cross-correlation between the sync word, or the frame delimiter word, and other data, is readily fulfilled by having words which contain many bits. However, in increasing the size of the preamble, various trade-offs come into consideration. These trade-offs are discussed below.

Probability and time calculations

s	=	the number of bits in a sync word
f	=	the number of bits in a frame delimiter word
n	=	the total number of bits in the data packet
N	=	the number of sync words to be recovered before setting SYNC high
T	=	the total number of sync words transmitted in the preamble

d = the data rate

For convenience, a 'detect sequence' is defined as 'N' sync words together with the frame delimiter word. This 'detect sequence' is the shortest bit sequence which can set FRM high. The number of bits in this 'detect sequence' is given by, p, where:

$p = sN + f$

The probability of false synchronisation, Pfs , is given by:

$P_{fs} = 1 - (1 - 1 / 2^{(sN)})^{(n-sN)}$

The probability of false frame detection, Pfd , is given by:

$P_{fd} = 1 - (1 - 1 / 2^{(sN+f)})^{(n-p)}$

The time taken to recover the clock is dependent on the sync word bit pattern chosen. Generally, the more transitions there are in the sync words, then the more quickly can the clock be recovered accurately. After the clock has been recovered, the time required to detect the frame delimiter and set FRM high, t_{frame}, is given by:

$t_{frame} = (((T - \text{missed sync words}) * s) + f) * d$

The probability calculations for P_{fs} and P_{fd} are based on the assumptions given below.

1. Each test for a sync word or a frame delimiter word is independent of the previous test for a sync word.

2. The probabilities for false detect are based on starting the test for sync dataat bit 1, and terminating the test at bit 'n'. It assumes all data received before the test is rejected (i.e. the system hasn't already detected sync words before bit 1) and sync/frame detect is aborted at bit 'n' irrespective of current state (i.e. even if some sync words have already been detected).

3. No errors are allowed in the synchronisation sequence (i.e. BE not used).

5.9.6 Summary of considerations and trade-offs

1. The longer the preamble is, the more time there is to both recover the clock and obtain synchronisation. However, a longer preamble means a greater redundancy overhead and so a lower information rate.
2. If the number of sync words that are required before SYNC is set high is increased, then the smaller is the probability that there will be false synchronisation and the resulting frame delimiter detection failure. However, if this number is increased, then the total length of the preamble must be increased. This increases the redundancy overhead and lowers the information rate.
3. As the length of the frame delimiter word is increased, so the probability of false frame delimiter detection decreases. However the redundancy overhead increases at the same time. However, if the length of the preamble is much greater than the length of the frame delimiter word, then the undesired increase in the redundancy overhead produced by adding extra bits to the frame delimiter, is more than outweighed by the desired decrease in the probability of false frame delimiter detection. In such cases where the preamble is much longer than the frame delimiter and the probability of frame delimiter detection failure must be minimised, then it is more worthwhile to increase the number of bits in the frame delimiter word rather than in the sync word. This is because the former will produce a smaller increase in the redundancy overhead than the latter, for the same decrease in detection failure.
4. It should be noted that BE set high, allowing single bit errors, is applicable to the sync words prior to Sync being set high, as well as to after Sync being set. If short sync words are used and many of them are required to set Sync, then there would be a higher probability of false synchronisation. However, if long sync words are used, with just a few required to set Sync, then BE will have a less significant effect on the probability of false synchronisation.

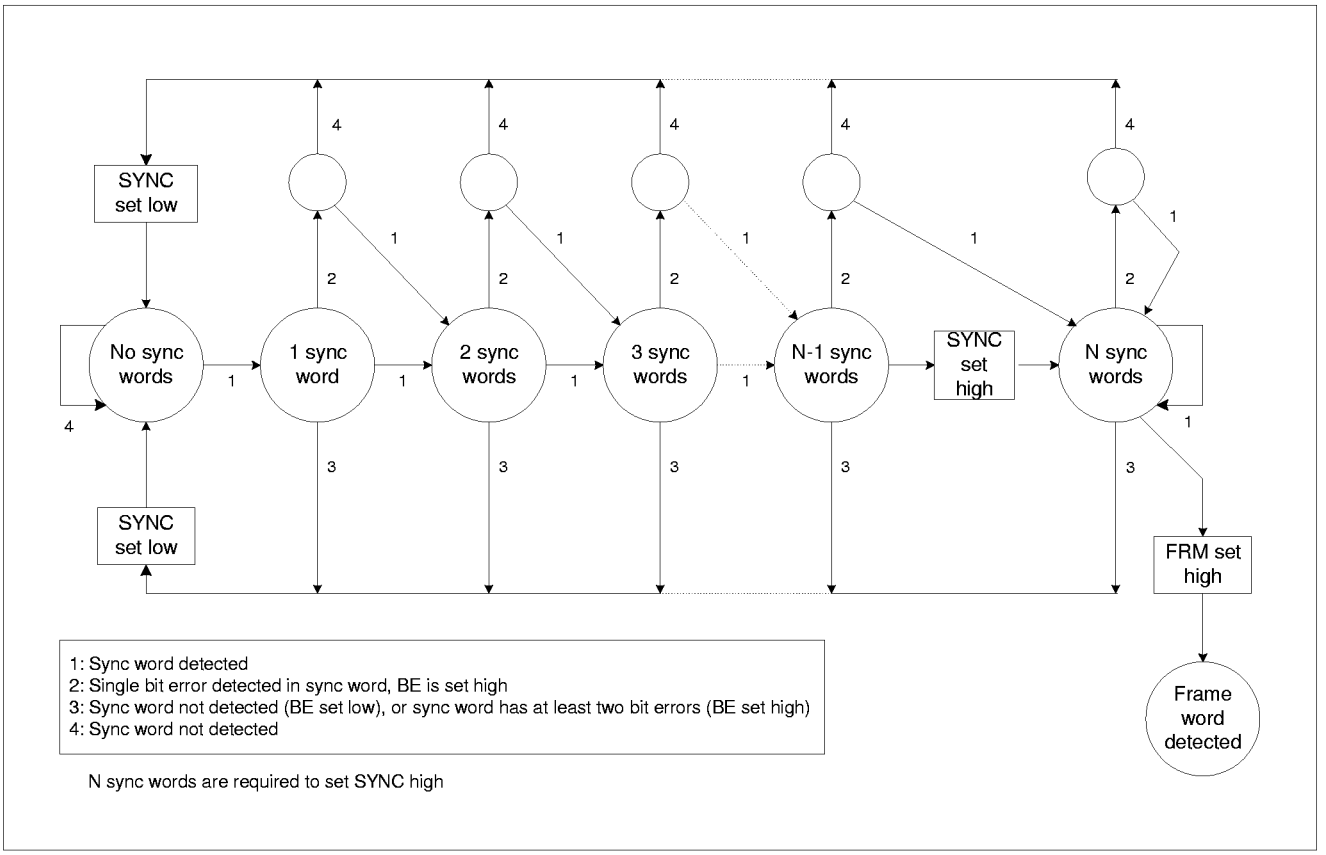

Figure 10. State Transition Diagram for Synchronisation and Frame Delimiter Detection

5.10 Clear Channel Assessment

The purpose of clear channel assessment is to detect if the current channel can be used to transmit. One way to decide if a channel is clear to transmit is to measure the RF power received at the antenna. If the received power is high, it is a good indicator that something else is using the channel, and any attempt to transmit is likely to be unsuccessful (i.e. the receiver is unlikely to be able to recover the data). However, even if a lower power signal is detected at the antenna, it may be a transmission which could nonetheless be recovered.

The CCB contains several features to give an indication of the type of signal being received on the antenna. The 'CCA' input pin of the WL102 is generated by a comparator in the WL600. A low indicates the RF power is below a pre-set limit (e.g. less than -60dBm). The comparison is done on a 'Received Signal Strength Indicator' (RSSI) signal. The WL102 could also latch the output of an external analogue-to-digital converter driven by the RSSI signal. The CCB generates two indicators, 'Long' and 'Noise', which indicate that the RXD input does not have edges which are consistent with valid data. The processor can use a combination of these signals to decide if the channel is clear to transmit.

5.10.1 CCA bit

The CCA bit, generated by the WL600, is the result of passing the RSSI signal through a comparator. If the received power is greater than a threshold set on the comparator, the CCA bit will be high. When CCA is high, the WL102 should not attempt to transmit, since the existing RF source is likely to make a transmission unrecoverable, and if the existing RF source is another wireless network system, it's transmission may also be corrupted.

5.10.2 'Long' indication (Addr. 21, bit 1)

An interfering signal may consistently be strong enough to pass through the radio demodulator and onto the Rx data line during transmission (e.g. a nearby microwave oven tends to cause a continuous stream of ones or zeros to appear to be generated over many bit times).

The WL102 will set the 'Long' indicator if no bit transitions are detected within a time period which is greater than the chosen encoding option should allow. For Bit Stuffing after 16 bits, 'Long' will be set if no edges are detected in the data within 17.5 bit periods. For Bit Stuffing after 8 bits, 'Long' will be set if no transitions are detected within 9.5 bits. For BSE 16/17, 'Long' will be set if no edges are detected in the data within 42 bit periods. For BSE 32/33, 'Long' will be set if no edges are detected in the data within 83 bit periods. If Bit Stuffing and BSE are disabled, 'Long' will not be used.

The 'Long' bit will be cleared if the valid data threshold is reached (Section 5.10.4).

5.10.3 'Noise' Indication

Maximising immunity to noise with respect to signal is important since noise will certainly be present in the recovered data and directly affects the usable range of a wireless network. Unlike a wired system where all nodes are considered friendly, while communicating over a wireless medium the Receiver will encounter several kinds of "noise" interferers.

In general, noise glitches (<125 nS) can occur on the digital input as a result of noise affecting the radio's data slicer circuit - i.e. digital glitches are likely to occur around the actual bit transitions. Synchronous noise may also occur as a result of digital switching within the controller.

No signal present on the channel (i.e. just background noise present) occurs during non-transmission and can exhibit itself as a continuous stream of random ones and zeros on the Rx data line. Noise determination of the channel is made by counting the number of invalid transitions over consecutive bit periods. Invalid transitions are edges on the Rx data line which are detected by the oversampling circuit at a time which is outside the specified Jitter Tolerance (Addr. 18h, bits 0 and 1) from an expected edge. The Noise Threshold setting NT (Addr. 18h, bits 2 and 3) can be configured so that it permits either 1, 4, 8 or 16 invalid transitions, accumulated while waiting for 'Sync', before it sets the 'Noise' flag (Addr. 21h, bit 2) to signify that noise is present on the channel. The count of the number of occurrences of invalid data will be reset if the valid data threshold is met (see Section 5.10.4).

5.10.4 'Valid Data' Indication

Valid data is considered to be when edges on Rx data line occur within the Jitter Tolerance of an expected edge. If the 'Noise' or 'Long' bits get set, they will be cleared again if the number of occurrences of valid data meet the Data Threshold (Addr. 18h, bits 4 and 5). The count of the number of occurrences of valid data will be reset by an occurrences of invalid data.

5.10.5 CCA timer

The CCA timer (Addr. 06h and 07h) can be configured to run only if both CCA and 'Sync' are low. While the radio is in receive mode, 'Sync' will be set if sync words are detected, or during the reception of a packet. Therefore, while in receive mode, if the CCA Timer Always bit (Addr. 03h, bit 7) is clear, the CCA timer will only decrement if the channel is clear to transmit (i.e. received power is below threshold and the preamble of a lower power reception is not detectable).

5.11 4-level GFSK support

The 4-level GFSK logic has only been designed to support certain packet formats.

The header must consist of two bytes, plus a 16-bit Header Checksum (Addr. 1Eh, Bit 1 set, Addr. 1Dh equal to 2). The two bytes should consist of the data length field (Addr. 19h equal to 1), plus the three MSBs of the second byte must be '010' (as specified in Draft 5 of 802.11). When the '010' pattern is detected, and the Legacy mode bit is clear (Addr. 1Fh, Bit 1), the CCB will switch to 4-level mode on the first bit of the first byte of the frame (i.e. the first bit after the Header Checksum).

The test for 4-level mode is done before the Data Length Mask (Addr. 1Ah) is applied. The three MSBs of the Data Length Mask should be cleared, so that the '010' pattern isn't interpreted as the three MSBs of the Data Length field.

The Frame Checksum start position (Addr. 1Ch) must be set to 02h, so that the checksum calculation starts with the first byte of the frame. The Frame Checksum must be 32-bit long (Addr. 1Eh, Bit 0 clear).

The Bit Stuffing option can not be used in 4-level mode (Addr. 1Eh, Bits 4-6).

5.12 Channel select codes

Table 22 shows how the codes loaded into SDT and SDR (Addr. 01h, 02h) relate to the frequencies used by the DE6003 and DE6038 radios:

Frequency used by radio (MHz)	Code loaded in SDT for DE6003 radio (both Tx. and Rx.) (MSB - LSB)	Code loaded in SDT for DE6038 radio in Tx. mode (MSB - LSB)	Code loaded in SDR for DE6038 radio in Rx. mode (MSB - LSB)
2400	X000 0000	0100 0000	0000 0101
2401	X000 0001	0100 0001	0000 0110
2402	X000 0010	0100 0010	0000 0111
:	:	:	:
2442	X010 1010	0110 1010	0010 1111
2443	X010 1011	0110 1011	0100 0000
2444	X010 1100	0110 1100	0100 0001
:	:	:	:
2447	X010 1111	0110 1111	0100 0100
2448	X011 0000	1000 0000	0100 0101
2449	X011 0001	1000 0001	0100 0110
:	:	:	:
2490	X101 1010	1010 1010	0110 1111
2491	X101 1011	1010 1011	1000 0000
2492	X101 1100	1010 1100	1000 0001
:	:	:	:
2495	X101 1111	1010 1111	1000 0100
2496	X110 0000	1100 0000	1000 0101
2497	X110 0001	1100 0001	1000 0110
2498	X110 0010	1100 0010	1000 0111
2499	X110 0011	1100 0011	1000 1000
2500	X110 0100	1100 0100	1000 1001

Table 22. Channel Select Codes

: = codes increment in binary count until next section.

The DE6038 channel is generated from the equation M*N + A

Where N = 48

M is dependent on the two MSBs of the code;- 00 = 49, 01 = 50, 10 = 51, 11 = 52

A is dependent on the 6 LSBs of the code, and should be in the range 0 to 47.

In receive mode, the code should generate a frequency 43MHz lower, due to the IF offset.

6. Host Interrupt Generation

The WL102 includes dedicated logic for the generation and acknowledgement of level-mode interrupts to the Host, and for the generation of 'Ready/Busy' indication to a PC-Card host.

The complete state diagram for the interrupt generation logic is shown inFigure 11. The state transitions are clocked by the 10MHz clock.

6.1 Operation Upon Reset

When the Host reset signal (HRESET) is released, the HIREQ output is set low and held in this state by an internal delay counter. After this delay (64 clock cycles), the System processor is given control of the interrupt logic. In order to set HIREQ to a high (inactive) state, the MCB auto-increment control must be set low once (Port1.2 for internal processor, or EPSYSINC for external processor), i.e. the first time this control is set low it will set the HIREQ pin.

6.2 Configuring Attribute Memory

When used in a PC-Card, this initial sequence allows the System processor time to write the Card Information Structure (CIS) into the Attribute Memory; HIREQ/HRDY low acts as the 'Busy' indication to the Host. After writing the CIS, the System clears the auto-increment control, indicating a 'Ready' condition.

6.3 Normal Operation

Once the internal reset delay has expired, and the System has cleared the auto-increment control once, two signals are used to control the HIREQ output: the (WL102) System interrupt control (Port1.4 for internal processor, or EPRDYB for external processor), and the IRQ_CLR bit in the Host's MCB Control Register (Bit 5).

In order to indicate an interrupt to the Host, the System clears the interrupt control. Provided IRQ_CLR is zero, this will cause the HIREQ output to be held low (active) until the Host sets IRQ_CLR. When the Host acknowledges the interrupt by setting IRQ_CLR, the HIREQ output returns to an inactive state, and is guaranteed to be inactive for at least 2 clock cycles.

6.4 Disabling Interrupts

The Host has the ability to disable interrupts from the WL102 by holding IRQ_CLR active (set to a '1'), thus forcing HIREQ high. When IRQ_CLR is then cleared, the state of HIREQ is once again determined by the System interrupt control.

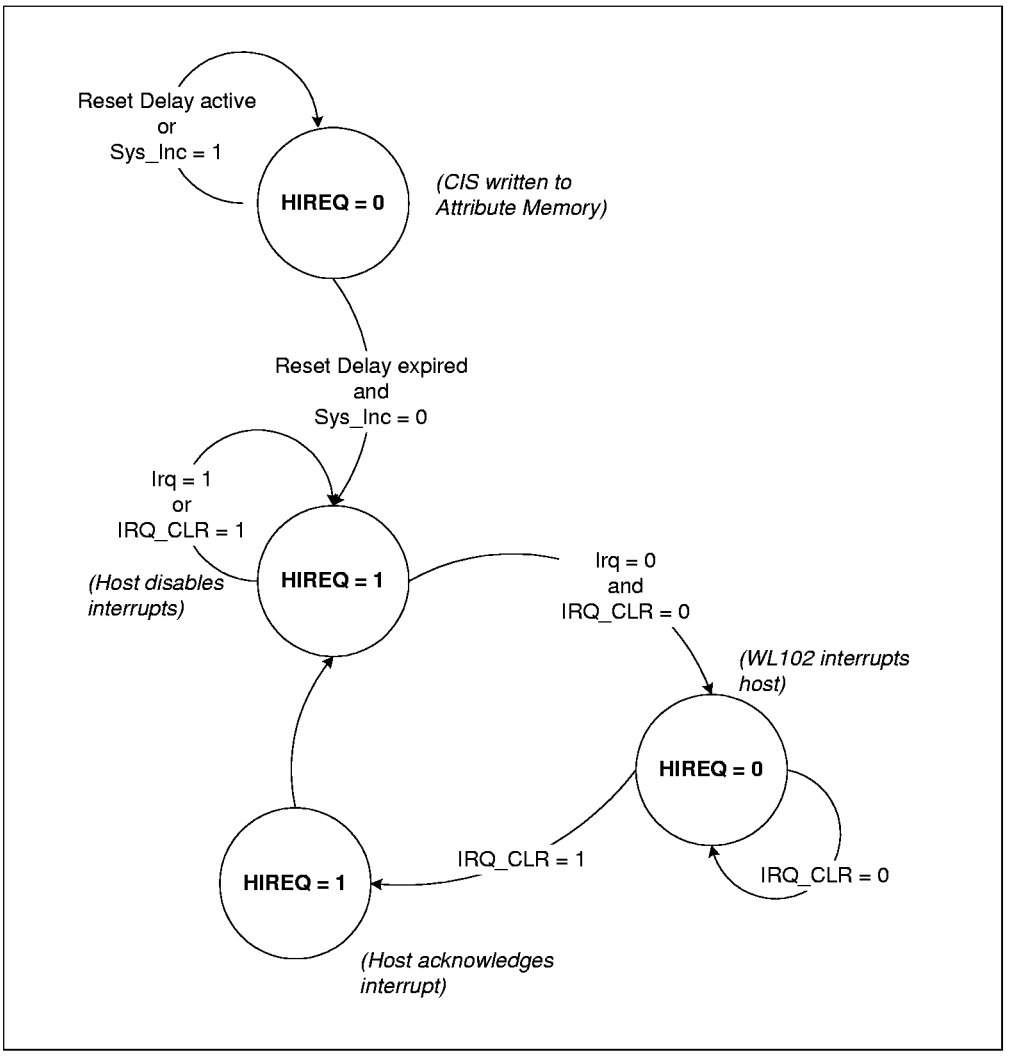

Figure 11. Host Interrupt State Diagram

7. Interfacing to Radio Transceivers

7.1 Clock Inputs

The WL102 must be clocked internally at 10MHz via one of two clock sources.

The SYSCLK/ SYCLK_GND inputs are for connection to a single-ended small signal 20MHz input, as provided by the WL800 synthesiser chip. When used with the WL800, these inputs must be capacitively coupled as shown inFigure 12. The minimum recommended capacitor value is 1nF.

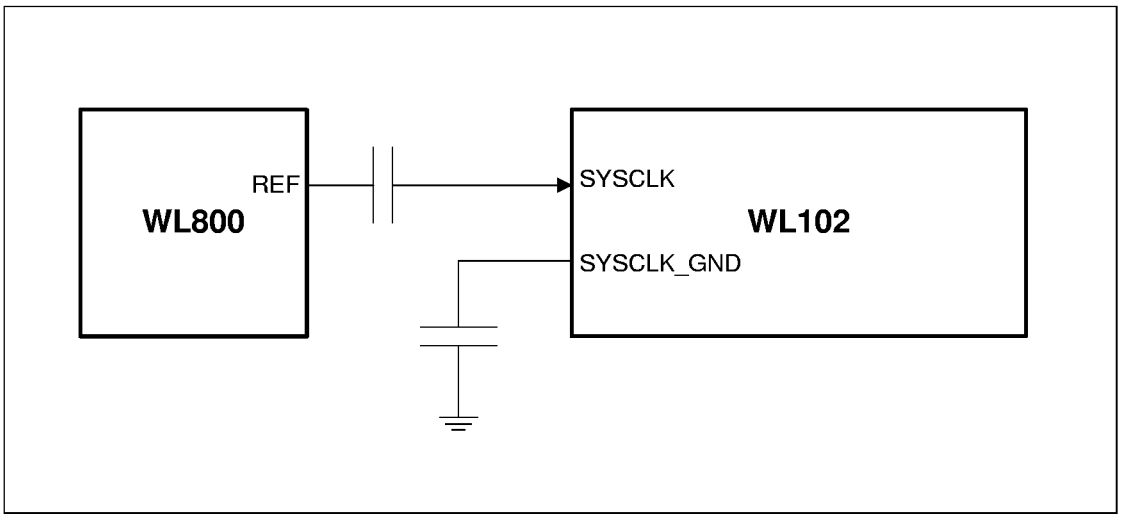

Figure 12. WL800 Reference Clock Interface

If not used, SYSCLK must be tied to AGnd, and SYSCLKGND must be tied to AVdd.

Alternatively SYSCLK_D, the 10MHz digital clock input, may be used. Note that in order to use the clock output of the DE6003 transceiver, the signal must be buffered using a circuit such as that shown in the DE6003 data sheet (DS-3506). If not used, this input must be connected to Gnd.

7.2 Received Data Inputs

The WL102 received data input may be provided from one of two sources.

The RXDH/RXDHB inputs are for connection to a common-mode small-signal input, such as that provided by a WL600/WL800 transceiver. If 4-level modulation is used, the most significant bit of the decoded symbol is connected to the RXDH/RXDHB inputs, and the least significant bit of the decoded symbol connected to the RXDL/RXDLB inputs. Any inputs not used must be connected either to AGnd (RXDHB and RXDLB) or to AVdd (RXDH and RXDL).

Alternatively, a digital received data stream may be applied to the RXD_D input. Note that in order to use the received data output of the DE6003 transceiver, the signal must be buffered using a circuit such as that shown in the DE6003 data sheet (DS-3506). If not used, this input must be connected to SVdd.

7.3 DE6003 Interfacing

When the CCB is configured for DE6003 mode, the following radio interface signals are used:

TXD
RX/TXB
PA_ON

ANT_SEL
PWR_LOB
LOADB
LCK_DETB
STDBYB
CSD[6:0]
RXD_D

Note that the CSD signals are only available with the 144-pin package. Hence this package option is required to use the WL102 with the DE6003.

7.4 WL600/WL800 Interfacing

When the CCB is configured for DE6038 mode, the following radio interface signals are used:

TXD_H
TX/RXB
PA_ON
CS_CLK
CS_DATA
CS_LOADB
CCAB
LCK_DETB
STDBYB
RXD_H
RXD_HB

7.5 Radio Test Mode

The WL102 features a special test mode which allows in-system testing of WL600/WL800 radio transceiver subsystem. When enabled (ENI8051B=0, Test[1:0]=11), the HA and HREGB host interface signals are connected internally through to the radio interface pins, as follows:

Outputs to Radio	**Driven by :**
TX/RXB	HA[0]
PA_ON	HA[1]
CS_LOADB	HA[2]
CS_DATA	HA[3]
CS_CLK	HA[4]
TXD_H	HA[5]
TXD_L	HA[6]
MOD2LEVB	HA[7]
STDBYB	HREGB

Inputs from Radio	**Accessed From :**
CCAB	HD[0]
CCB-RXD_L	HD[1]
CCB-RXD_H	HD[2]
LCK_DETB	HD[3]
CCB-BITCLK	HD[4]
CCB-SYNC	HD[5]
RXD_L	HD[6]
RXD_H	HD[7]

The signals RXD_L and RXD_H are the digital outputs from the two differential amplifiers on the RXD_L / RXD_LB and RXD_H / RXD_HB analogue input pairs. These two digital data streams are also fed into the clock and data recovery logic in the CCB, from which is generated: the recovered clock CCB-BITCLK, and the recovered (sampled) data CCB-RXD_L and CCB-RXD_H.

In addition, the CCB-SYNC output indicates the state of the CCB 'Sync' Indicator status bit.

8. MCB Execute Mode

This mode of operation is used to allow programming of the external code ROM if a Flash device is used. This mode is controlled by the Host system and ensures that the System processor can be recovered from any software state it may be in, providing the SPSENB signal is used to access the code ROM. The sequence of operations is:

1. Place the System processor in RESET by using the MCB control register bit 0.
2. Load the MCB buffer RAM with the code to be executed in order to program locations in the Flash ROM. The format of this code requires careful design since when the System processor executes it from the MCB buffer RAM it will only be able to fetch sequential bytes. This means that if the processor pipelines its instruction fetches, as the internal 8051 does, the code will have to contain all the re-fetches that the processor will perform. Also no conditional branches will be possible, and the first fetch from the MCB buffer RAM will be the dummy byte, due to the one cycle latency. In order for this to be predictable it is set to zero by the Clear bit.
3. Clear the read data holding register and System address counters by using the CLEAR bit in the Host control register. This ensures that the first byte fetched by the System processor will be 0 and that the subsequent fetches will start at address 0.
4. Place the System into MCB Execute mode by using the MCB_EXECUTE bit of the Host control register.
5. Release the RESET on the System processor.

A detailed description of how to use MCB Execute mode may be found in Applications Note AN-4833.

9. External System Options

9.1 Use with an External Microprocessor

The WL102 can be used with an external system processor if desired, by setting the ENI8051B pin high (connect to HVdd).

This causes the internal 8051 core to be disabled and all the other blocks of the WL102 to be addressable by the external processor via the system address and data bus. In addition, the following pin changes occur as a result of this:

SRDB	Read strobe input.
SWRB	Write strobe input.
SPSENB	Program memory read strobe input (only required for MCB_EXECUTE mode use).
SA[15:0]	Address bus input.
P3[0]/EPSYSINC	MCB auto-increment control input
EPRSTB	Output from the Host MCB control register bit 0 and the Power-on reset signal generated from the HRESET pin.
P1[3]/EPCINTR	CCB interrupt output.
P1[5]/EPTCNT1	Counter 1 terminal count output.
EPRDYB	Input control to the Host Interrupt generation logic. (see Section 6)
SWAKEUP/EPTCNT2	Counter 2 terminal count output.
P3[1]/EPMINTR	MCB Host Control register bit 3 output.
SYSCLK_X	10MHz system output clock (used for synchronisation).

The external processor can then be connected to the required signals so that it can function correctly with the WL102. The external processor can be run from the SYSCLK_X output of the WL102 and this will usually result in the SWRB,SRDB and SPSEN signals meeting the synchronisation requirements of the WL102 as specified in the AC characteristics. If the external processor is not run from the SYSCLK_X pin the synchronisation requirements must be guaranteed by external circuitry.

If the MCB_EXECUTE mode is to be used (see Section8) then the instruction fetches for the external processor must be performed through the WL102 by strobing the SPSENB input, which will access code ROM via CCEB in normal mode, and will access the MCB system data register in MCB_EXECUTE mode.

9.2 Adding External Peripherals

Since only part of the WL102 address map is decoded to internal functions, it is possible to add further peripheral devices to a system. External logic may be added to the system bus signals to decode the available address space as required (see Section 3.1.1).

Internal decode logic is provided for a particular area of the address map, intended for reading the received signal strength (RSSI) from an analogue-to-digital converter; a processor read from this address range causes a read pulse on the RSSISTRB output (see Figure 17 and Figure 21).

10. Using the WL102 in a PC-Card

The host interface of the WL102 has been designed primarily for use in a PC-Card, providing the necessary logic and signals. The WL102 provides 8-bit attribute memory and I/O space, and includes support for Rdy/Busy and Wait signals.

10.1 Pin Connections

The connections required for use of the WL102 in a PC-Card are:

PC-Card	WL102
A8 - A1	HA7 - HA0
D7 - D0	HD7 - HD0
REG	HREGB
CE1	HCEB
OE	HOEB
WE	HWEB
RDY/BSY	HRDY
WP/IOIS16	HVdd
INPACK	HCEB
CD1	Gnd
IORD	HRDB
IOWR	HWRB
RESET	HRESET
WAIT	HWAITB
CD2	Gnd
Gnd	Gnd
Vcc	HVdd

Note that the least significant bit of the HA bus (HA0) should be connected to the 2nd least significant bit of the Host A bus (A1), such that the card is accessed only at even addresses, which is required for 8-bit access to the attribute memory. As a consequence of this, the addresses of the Host MCB registers are also on even-byte boundaries only.

10.2 Attribute Memory

The attribute memory consists of 255 bytes of RAM for storing the PCMCIA Card Information Structure (CIS), and a PCMCIA Configuration Option Register (COR). The Attribute memory address decode is shown inTable 23. Because Host A0 must be connected to WL102 HA1, the Host accesses the COR at an address offset of 1FEh (2 x FFh).

Address (HA7-HA0)		Function	Reset State
00h - FEh		Start of CIS	Undefined
FFh		COR	
	Bit 7	Sreset	0
	Bit 6	Levelreq	1
	Bits [5:0]	Configuration Index	0

Table 23. Attribute Memory Address Map

10.2.1 Card Information Structure

On reset, the WL102 sets the HRDYB (HIREQ) pin high, indicating to the host that the card is not ready. The system processor is then responsible for loading the Card Information Structure (CIS) data into the attribute memory, before HRDYB may be set low. A host attribute memory read (CE1=0, HREG=0, HOE=0, HWE=1) can occur at any time after HRDY has been released by the card.

Once the host has configured the card as an I/O device (setting the Configuration Index in the COR register to a non-zero value), this signal may be used for its primary function i.e. host interrupt (HIREQ).

Simultaneous host and system access to the CIS RAM is not supported

Reference should be made to the PC-Card Standard to determine suitable contents for the Card Information Structure, however special attention is drawn to the following tuples:

Configuration Register Base Address (CISTPL_CONFIG:TPCC_RADR)

The base address specified for the configuration registers must take into account the connection of the address busses i.e. this value should be set to 1FEh.

Configuration Index (CISTPL_CFTABLE_ENTRY:TPCE_INDX)

When creating Configuration Table Entries, it is important to note the effect of the Index byte on the interface voltage mode, as described in Section 10.2.3.

10.2.2 Configuration Option Register

The Configuration Option Register (COR) is read by and written to by the PCMCIA host. The system also has read and write access to this register, although simultaneous access is not allowed. The fields within the COR are summarised below.

Sreset

This bit allows the WL102 to be reset. When set high, the HIREQ output is forced to a high (inactive) state. When Sreset is then set low, a normal reset sequence occurs in the WL102 (equivalent to releasing the HRESET input).

Levelreq

A host may set this bit to specify whether it requires level-mode or pulse-mode interrupts. Refer to the PC-Card Standard for further details.

Configuration Index

The least significant bit of the Configuration Index field may be used by the host to set the voltage mode of the host interface as described in Section 10.2.3.

10.2.3 Supply Voltage

The host interface pins can operate at either 5 volts or 3 volts, allowing for use in dual-voltage hosts. Control of the voltage mode is provided to reduce switching noise when a 5 volt interface is used; the reset condition being 3 volt mode.

The voltage mode of the host interface is selected by a combination of the LOWVDD bit of the MCB control register and the least significant bit of the COR Configuration Index field. This allows the host to set the voltage mode either via the Control Register or via selection of a particular PC-Card configuration, as shown in Table 24.

LOWVDD	COR[0]	Mode	
0	0	5 Volt	Non PC-Card. Host must write Control Register on start-up
0	1	5 Volt	-
1	0	3 Volt	3 Volt Card; Reset Condition
1	1	5 Volt	5 Volt PC-Card; Host selects 5V mode via Configuration Index

Table 24. Host Interface Voltage Control

11. System Supply Voltages

The WL102 has two separate digital supply rails, allowing for use in mixed 3V/5V systems. The HVdd rail is used for the host interface, whilst the SVdd rails is used for the core of the WL102 and its interface to the memory and radio transceiver subsystems. Valid combinations of these supply voltages are:

HVdd = 5volts; SVdd = 5volts
HVdd = 5volts; SVdd = 3volts
HVdd = 3volts; SVdd = 3volts

When using the DE6003 transceiver, both SVdd and HVdd must be 5volts. When using the WL600/WL800 transceiver, SVdd must be equal to the WL600/WL800 supply voltage (approx. 3volts) and HVdd may be 3 to 5.5V, but must not be less than SVdd.

The THREEV input must be connected according to the SVdd supply voltage; to SVdd for a 3 volt supply, and to Gnd for a 5 volt supply.

To reduce switching noise, the LOWVDD bit of the MCB Host control register should be set to a logic '0' when the HVdd supply voltage is 5volts (LOWVDD is set by default on reset). Alternatively, the voltage mode may be set via the Configuration Option Register. (See Section 0)

The WL102 also has an analogue supply, AVdd and AGnd, for powering the SYSCLK and RXD differential amplifiers. The AVdd/AGnd voltages should be equal to those on SVdd/Gnd.

12. DC Electrical Characteristics

12.1 Absolute Maximum Ratings

Symbol	Parameter	Minimum	Maximum	Units
VDD	Supply voltage (see note 1)	0.0	7.0	V
VIP	Voltage applied to input pin	-0.5	7.0	V
VOP	Voltage applied to output pin	-0.5	VDD+0.3	V
Osct	Output short circuit time (see note 2)		1	S
Ts	Storage temperature	-65	150	
Ta	Ambient operating temperature	-10	85	°C

Note 1: HVdd must be greater than or equal to SVdd
Note 2: Not more than one output should be shorted to VDD or VSS at any one time

12.2 DC Operating Ratings

Symbol	Parameter	Minimum	Typical	Maximum	Units	Conditions
IDDO1	Operating supply current (Note 1)				mA	SVdd=HVdd=5V
IDDO2	Operating supply current (Note 1)		10	20	mA	SVdd=3V HVdd=5V Internal processor mode All clocks enabled
IDDO3	Analogue Supply Current		1.3 0.5	3 1	mA mA	Receive mode. All other modes
VIL	Input Voltage Low			0.2 VDD		
VIH	Input Voltage High	0.7 VDD				
VOL	Output Voltage Low (Note 2)			0.4	V	IOL2 = 4mA IOL3 = 6mA IOL4 = 8mA
VOH	Output Voltage High (Notes 2 and 3)	0.85 VDD				IOH2 = -4mA IOL3 = -6mA IOL4 = -8mA
IIL	Input Leakage Current			1	µA	
IPD	Input Pull-down Current			135	µA	
IPU	Input Pull-up Current			-115	µA	
VICM1	Comparator Common Mode Input Voltage (Note 4)	1.1		2.8	V	SVdd = 3V
VICM2	Comparator Common Mode Input Voltage (Note 4)	1.1		4.75	V	SVdd = 5V
VIS	Comparator Input Voltage Swing (Note 4)	180			V pk-pk	

Note 1: All blocks enabled. Value given is the total for HVdd plus SVdd. Host Access cycle time 2uS.
Note 2: IOL3 applies to pins HIREQ, HWAITB, CSD and SYSCLKX. IOL4 applies to the HD bus. IOL2 applies to all other outputs.
Note 3: VOH does not apply to open drain outputs.
Note 4: Applies to comparator input pin pairs
RXD_H and RXD_HB
RXD_L and RXD_LB
SYSCLK and SYCLK_GND

13. AC Characteristics (Preliminary Data)

The maximum capacitive load for each outputs is 30pF with the following exceptions:

HD = 100pF
HWAITB= 50pF
HIREQ = 50pF

13.1 Host Interface

13.1.1 Host Memory Read

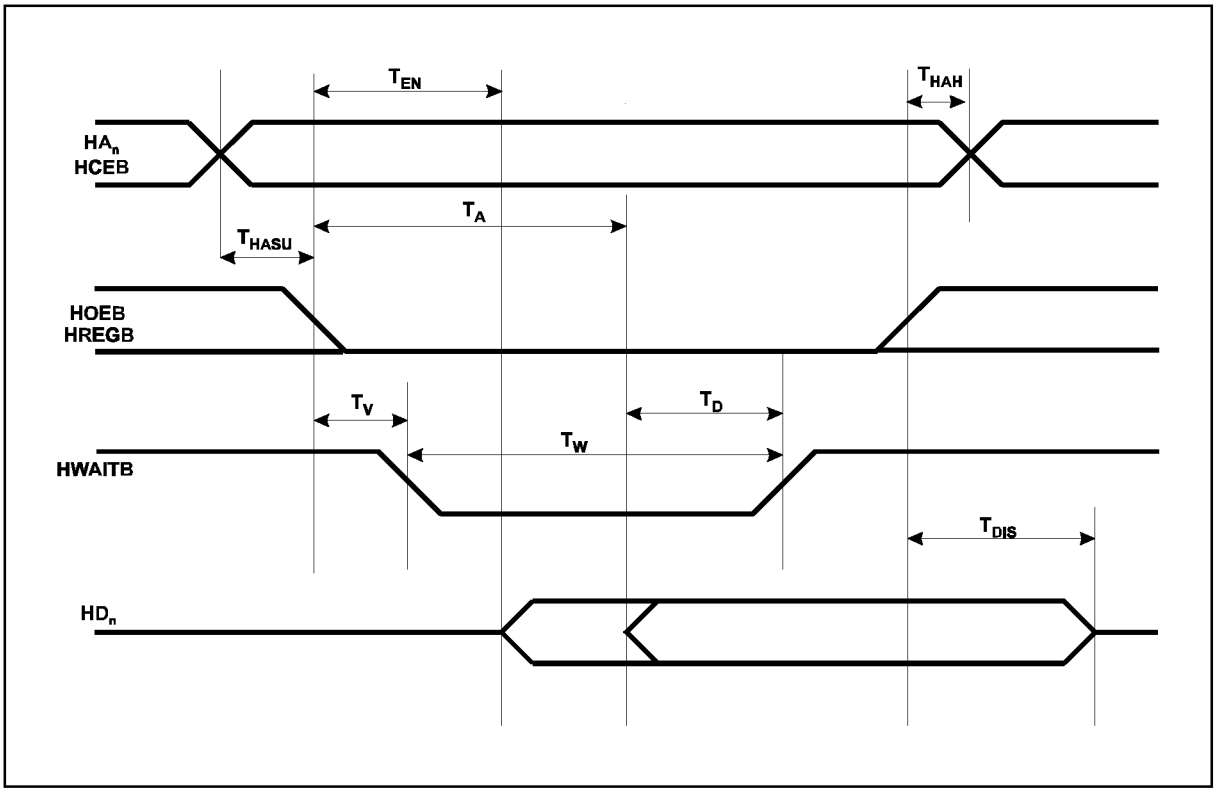

Figure 13. Host Memory Read Timings

Symbol	Description	Min	Max
TA	Access to data valid time (HWAITB not used)	-	700nS
TV	HWAITB valid from Access	-	30nS
THASU	Address setup to read access	0nS	-
THAH	Address hold after read access	20nS	-
TD	HD[ ] setup to HWAITB released	200nS	-
TW	HWAITB pulse width	-	550nS
TEN	HD[ ] enable from Access	5nS	-
TDIS	HD[ ] disable from Access	-	40nS

13.1.2 Host Memory Write

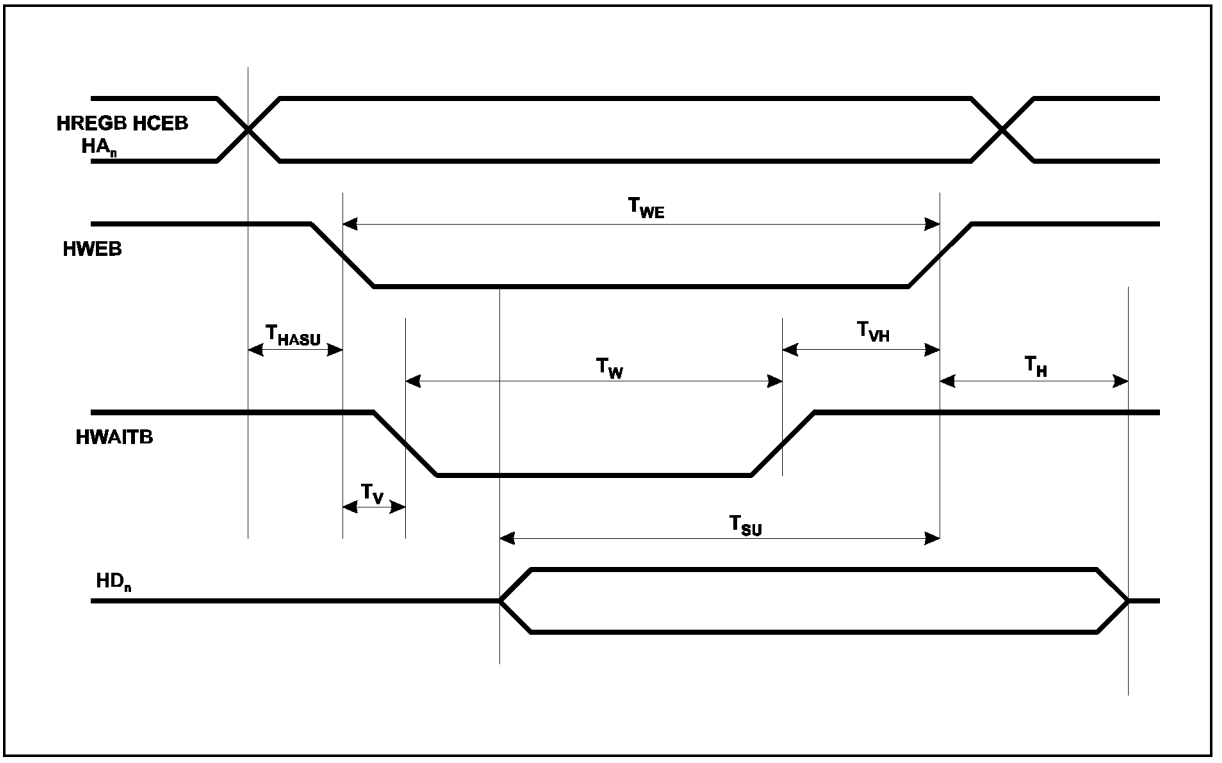

Figure 14. Host Memory Write Timings

Symbol	Description	Min	Max
TV	HWAITB valid from Access	-	30nS
THASU	Address setup to write access	0nS	-
THAH	Address hold after HWEB inactive	20nS	-
TSU	HD[ ] setup to Access end	20nS	-
TH	HD[ ] hold after Access end	20nS	-
TW	HWAITB pulse width	-	550nS
TWE	HWEB pulse width (HWAITB not used)	700nS	-
TVH	HWAITB released to HWEB high	0nS	-

13.1.3 Host I/O Read

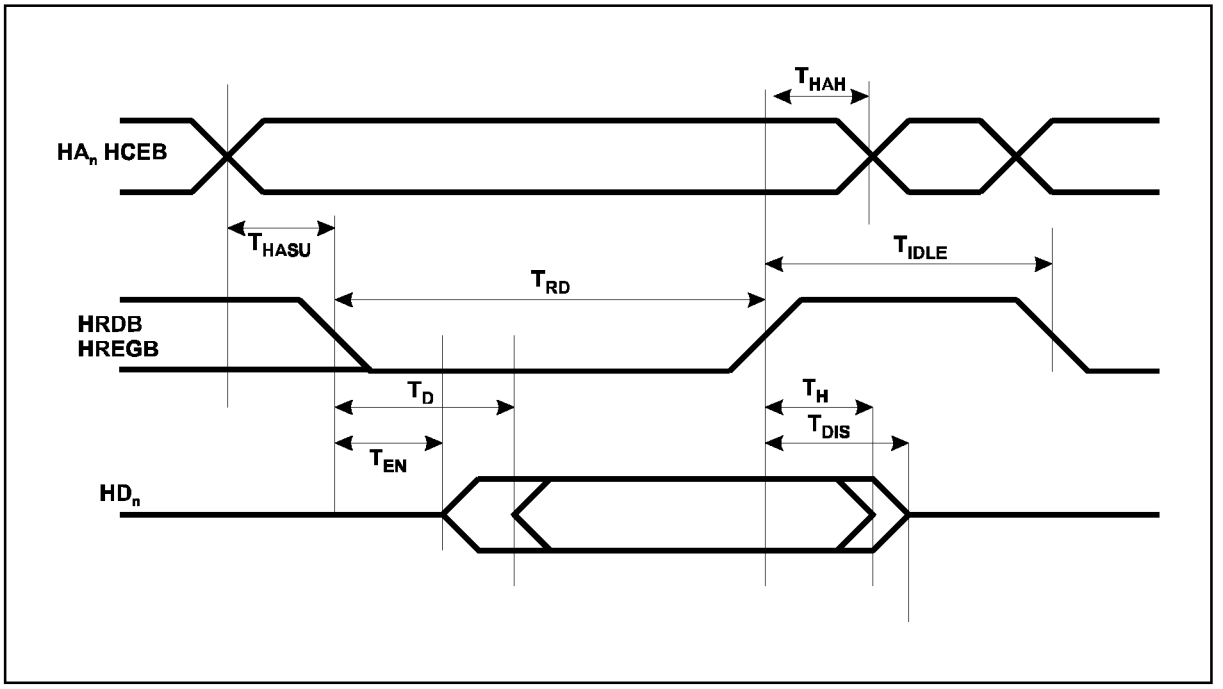

Figure 15. Host I/O Read Timings

Symbol	Description	Min	Max
TD	HD[ ] access to data valid	-	40nS
THASU	Address setup to read access	0nS	-
THAH	Address hold after read access	0nS	-
TEN	HD[ ] enable time	5nS	-
TDIS	HD[ ] disable time	-	40nS
TH	HD[ ] hold after Address	0nS	-
TRD	HRDB pulse width	150nS	-
TIDLE	I/O cycle idle time	500nS	-

13.1.4 Host I/O Write

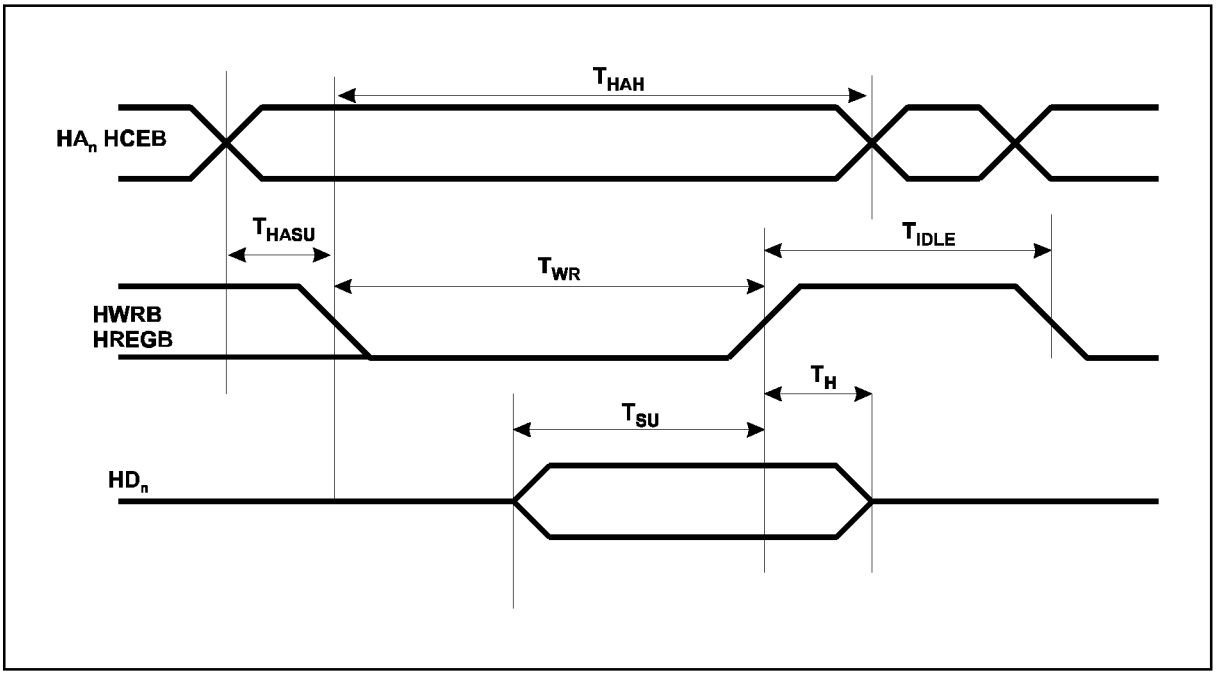

Figure 16. Host I/O Write Timings

Symbol	Description	Min	Max
TWR	HWRB pulse width	150nS	-
THASU	Address setup to write access	0nS	-
THAH	Address hold after start of write access	10nS	-
TSU	HD[ ] setup to HWRB high	20nS	-
TH	HD[ ] hold after HWRB	20nS	-
TIDLE	I/O cycle idle time	500nS	-

13.2 Internal Microprocessor

13.2.1 Internal Processor Data Memory Read

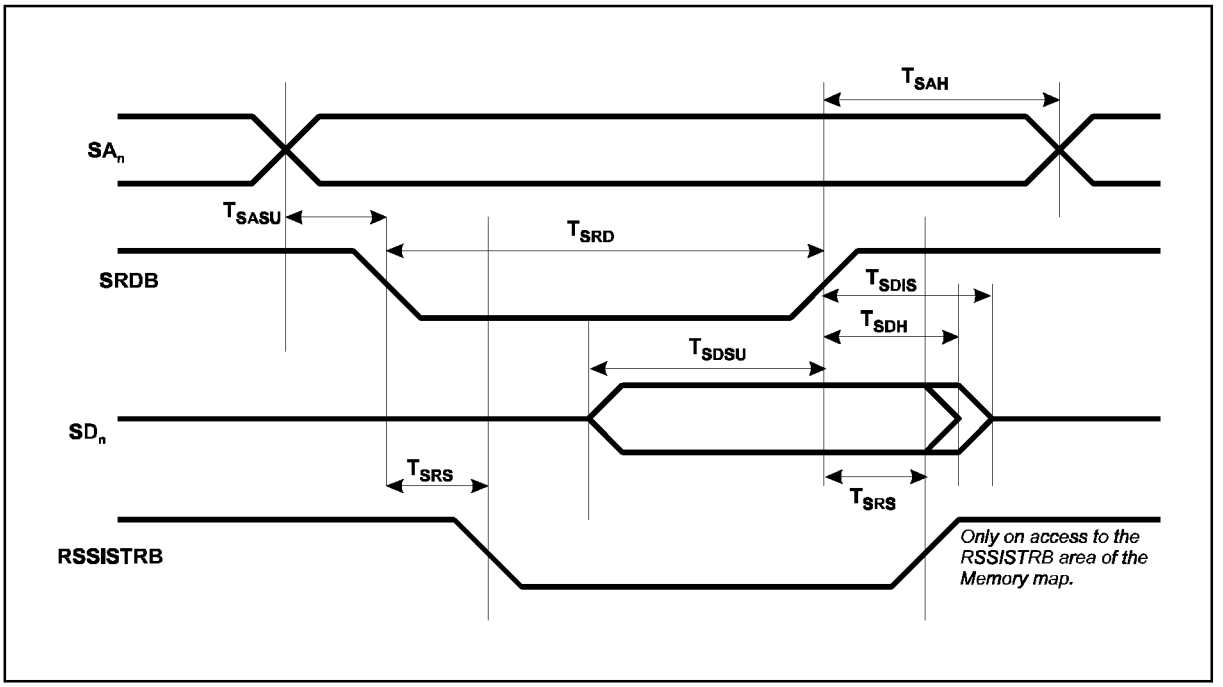

Figure 17. Internal Processor Data Memory Read

Symbol	Description	Min	Max
TSASU	Address to SRDB low	-	300nS
TSRD	SRDB pulse width	550nS	-
TSDSU	SD[ ] setup to SRDB high	150nS	-
TSDH	SD[ ] hold after SRDB high	0nS	-
TSAH	SA[ ] hold after SRDB high	250nS	-
TSDIS	SRDB high to SD[ ] disabled	-	50nS
TSRS	SRDB to RSSISTRB	-	10nS

13.2.2 Internal Processor Data Memory Write

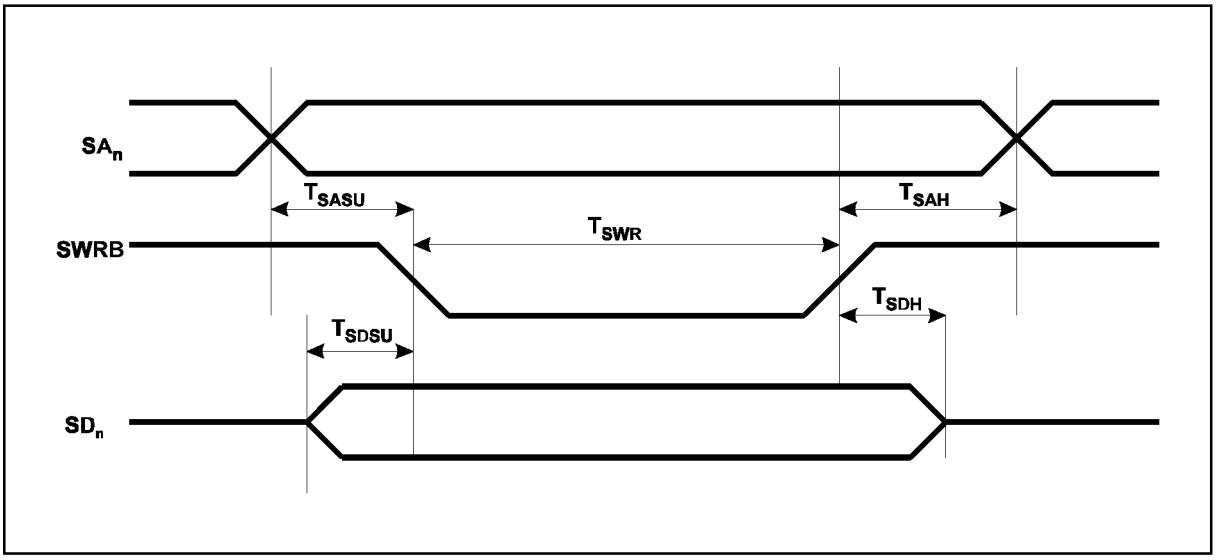

Figure 18. Internal Processor Data Memory Write

Symbol	Description	Min	Max
TSASU	SA[ ] to SWRB low	-	300nS
TSWR	SWRB pulse width	550nS	-
TSDSU	SD[ ] setup to SWRB	100nS	-
TSDH	SD[ ] hold after SWRB high	0nS	-
TSAH	SA[ ] hold after SWRB high	250nS	-
	Idle time for Access to MCB address 03 (see Section 4.1)	500nS	-
	Idle time for Access to CCB address 05,06 & 07 (see Sections 5.4.6 & 5.4.7)	3 Bit Clks	-

13.2.3 Internal Processor Program Memory Read

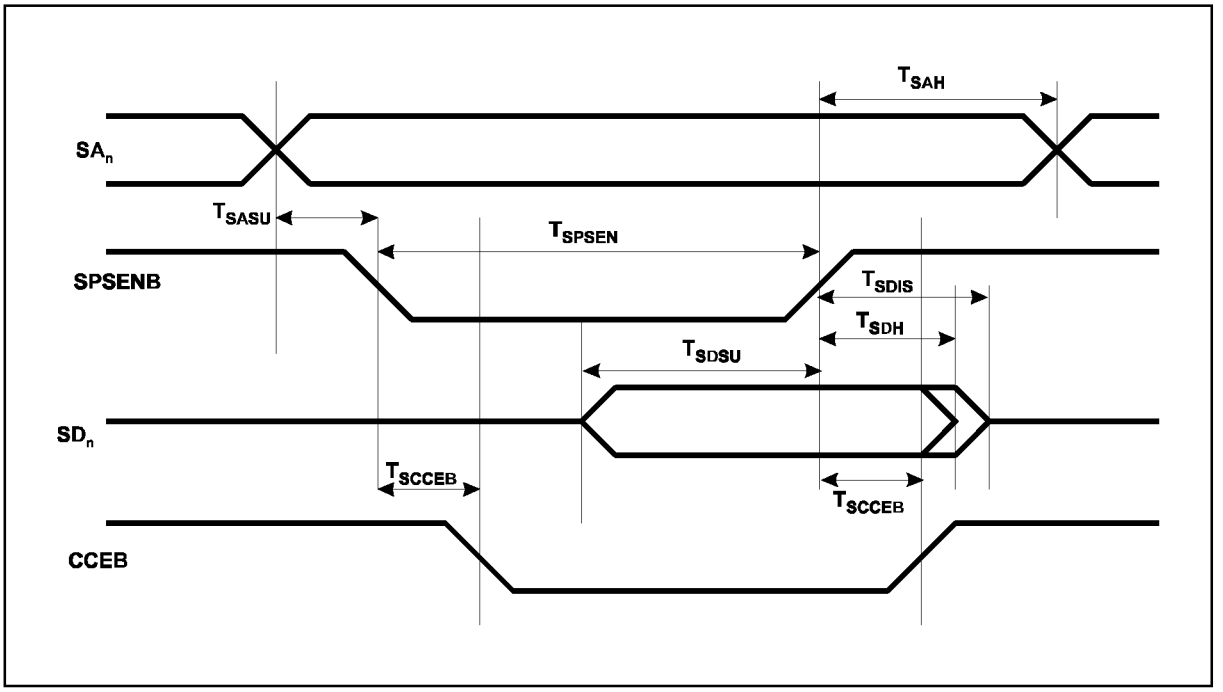

Figure 19. Internal Processor Program Memory Read

Symbol	Description	Min	Max
TSASU	SA[ ] to SPSENB low	-	100nS
TSCCEB	SPSENB to CCEB	-	20nS
TSPSEN	SPSENB pulse width	285nS	-
TSDSU	SD[ ] setup to SPSENB high	35nS	-
TSDH	SD[ ] hold after SPSENB high	0nS	-
TSAH	SA[ ] hold after SPSENB high	150nS	-
TSDIS	SPSENB high to SD[ ] disabled	-	10nS

13.2.4 Internal Processor Program Memory Write in MCB Execute mode

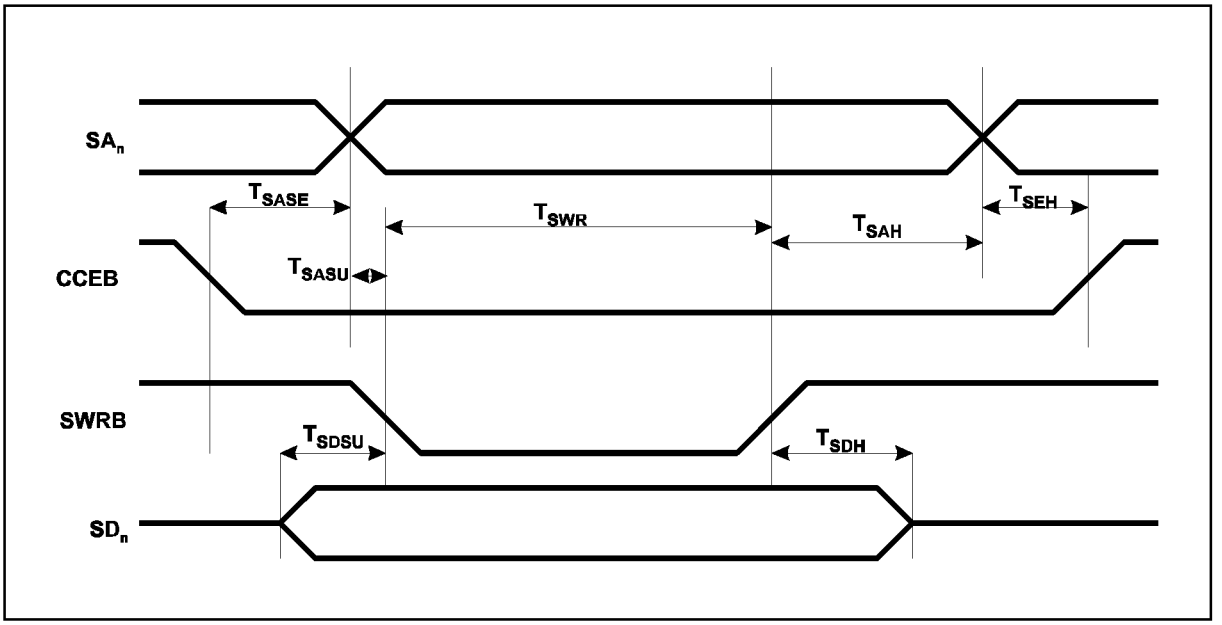

Figure 20. Internal Processor Program Memory Write in MCB Execute mode

Symbol	Description	Min	Max
TSASU	SA[ ] to SWRB low	-	350nS
TSASE	CCEB to SA[ ] valid	150nS	-
TSWR	SWRB pulse width	550nS	-
TSDSU	SD[ ] setup to SWRB low	100nS	-
TSDH	SD[ ] hold after SWRB high	0nS	-
TSAH	SA[ ] hold after SWRB high	250nS	-
TSEH	CCEB hold after SA[ ] invalid	50nS	-

13.3 External Microprocessor

Note: All input transitions on SWRB, SRDB and SPSENB must be synchronised to SYSCLK_X (see Figure 21).

13.3.1 External Processor Read

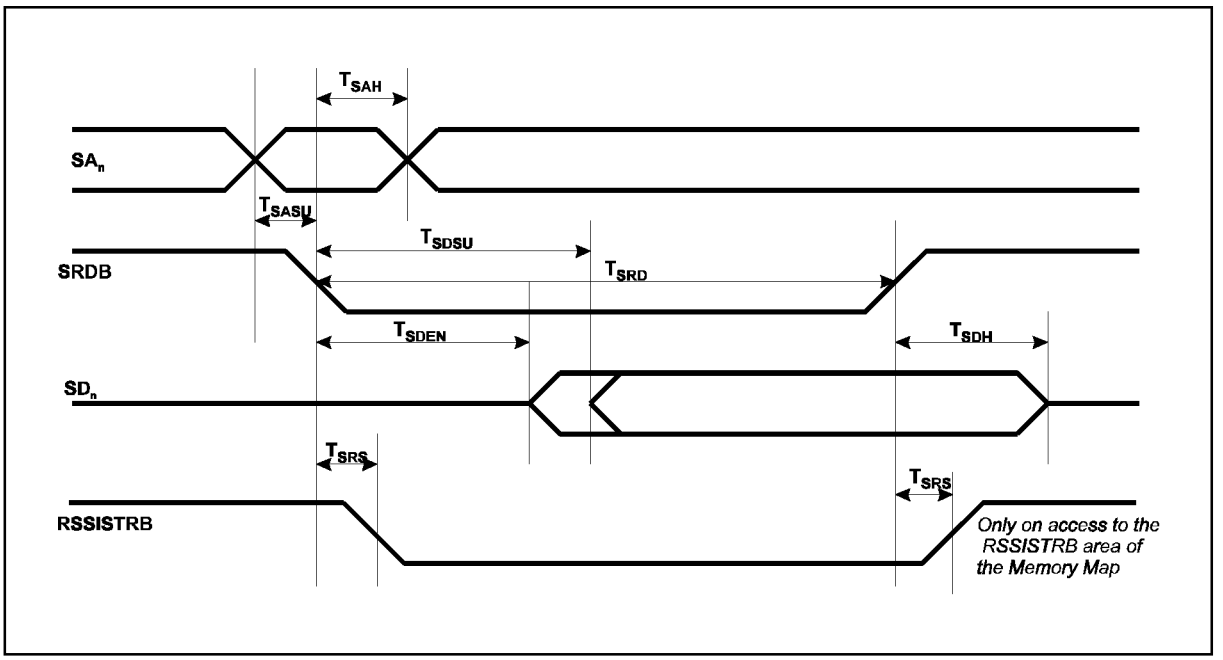

Figure 21. External Processor Read

Symbol	Description	Min	Max
TSASU	SA[ ] setup to SRDB low	10nS	-
TSDEN	SRDB low to SD[ ] enabled	0nS	-
TSDSU	SRDB low to SD[ ] valid	-	150nS
TSRD	SRDB pulse width	150nS	-
TSDH	SD[ ] valid after SRDB high	0nS	-
TSAH	SA[ ] hold after SRDB low	50nS	-
TSRS	SRDB to RSSISTRB	-	40nS

13.3.2 External Processor Write

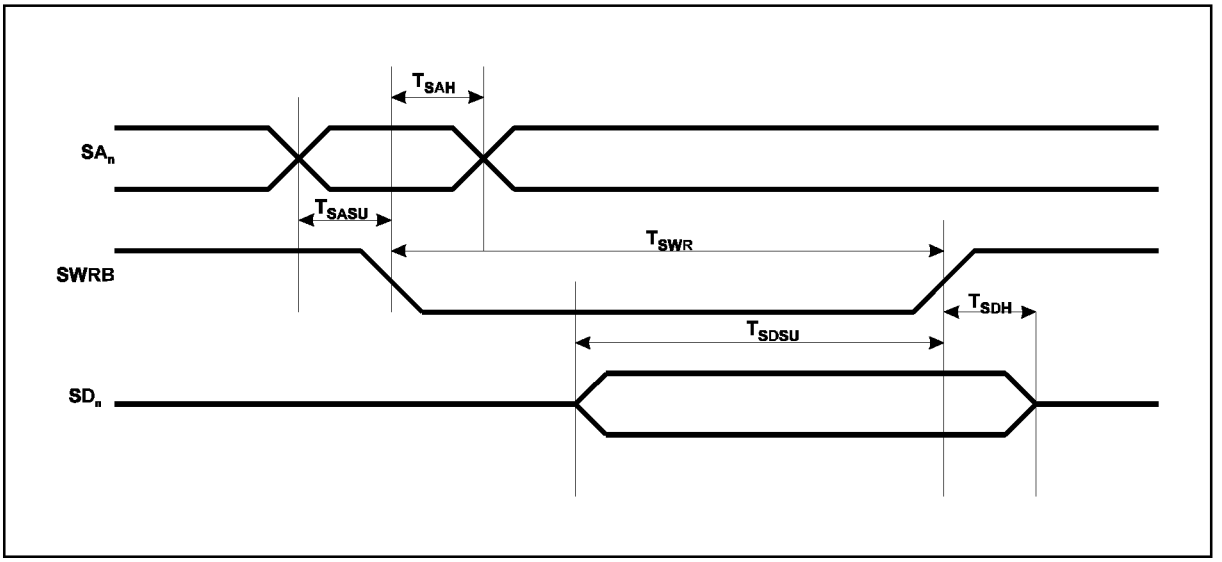

Figure 22. External Processor Write

Symbol	Description	Min	Max
TSASU	SA[ ] to SWRB active	10nS	-
TSWR	SWRB pulse width	150nS	-
TSAH	SA[ ] hold after SWRB low	50nS	-
TSDSU	SD[ ] setup to SWRB high	150nS	-
TSDH	SD[ ] hold after SWRB high	20nS	-
	Idle time for Access to MCB address 3	500nS	-
	Idle time for Access to CCB address 5,6 & 7	3 Bit Clks	-
	See Sections 5.4.5 & 5.4.6		

13.3.3 External Processor Instruction Read/Write

Figure 23. External Processor Instruction Read

Figure 24. External Processor Program Memory Write Timings in Execute Mode

Symbol	Description	Min	Max
TCCEB	SPSENB/ WRB to CCEB	-	30nS

13.3.4 External Processor Instruction Read in MCB Execute mode

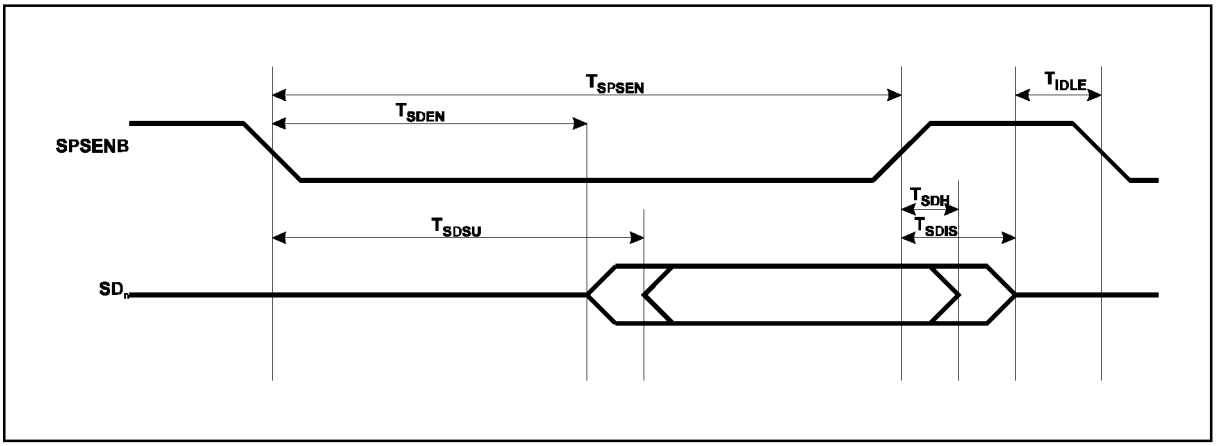

Figure 25. External Processor Instruction Read Timings in MCB Execute mode

Symbol	Description	Min	Max
TSDSU	SPSENB low to SD[ ] valid	-	50nS
TSDEN	SPSENB low to SD[ ] enabled	0nS	-
TSDH	SD[ ] hold after SPSENB high	0nS	-
TSPSEN	SPSENB width	150nS	-
TSDIS	SPSENB high to SD[ ] disabled	-	30nS
TIDLE	SPSENB high to next SPSENB low	500nS	-

13.4 External Buffer

13.4.1 External Buffer Read

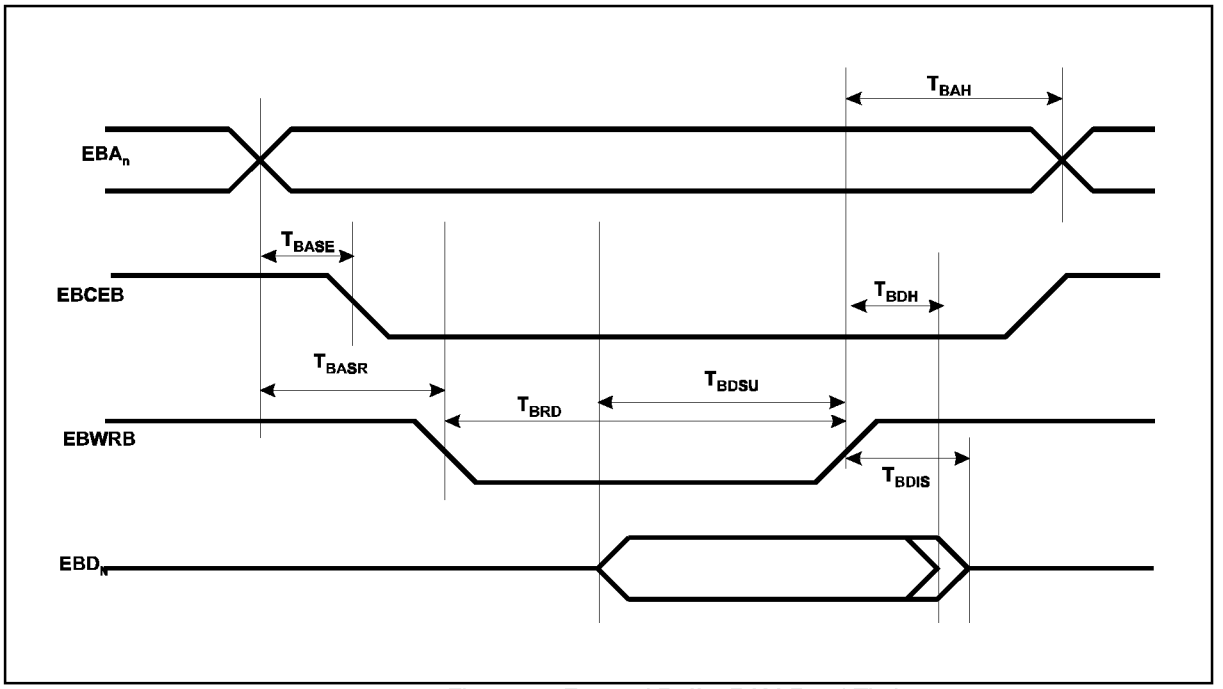

Figure 26. External Buffer RAM Read Timings

Symbol	Description	Min	Max
TBASE	EBA[ ] to EBCEB low	-	10nS
TBASR	EBA[ ] to EBRDB low	-	130nS
TBDSU	EBD[ ] setup to EBRDB high	10nS	150nS
TBDH	EBD[ ] hold after EBRDB high	0nS	-
TBAH	EBA[ ] hold after EBRDB high	80nS	-
TBRD	EBRDB pulse width	90nS	-
TBDIS	EBRDB high to EBD[ ] disabled	-	100nS

13.4.2 External Buffer Write

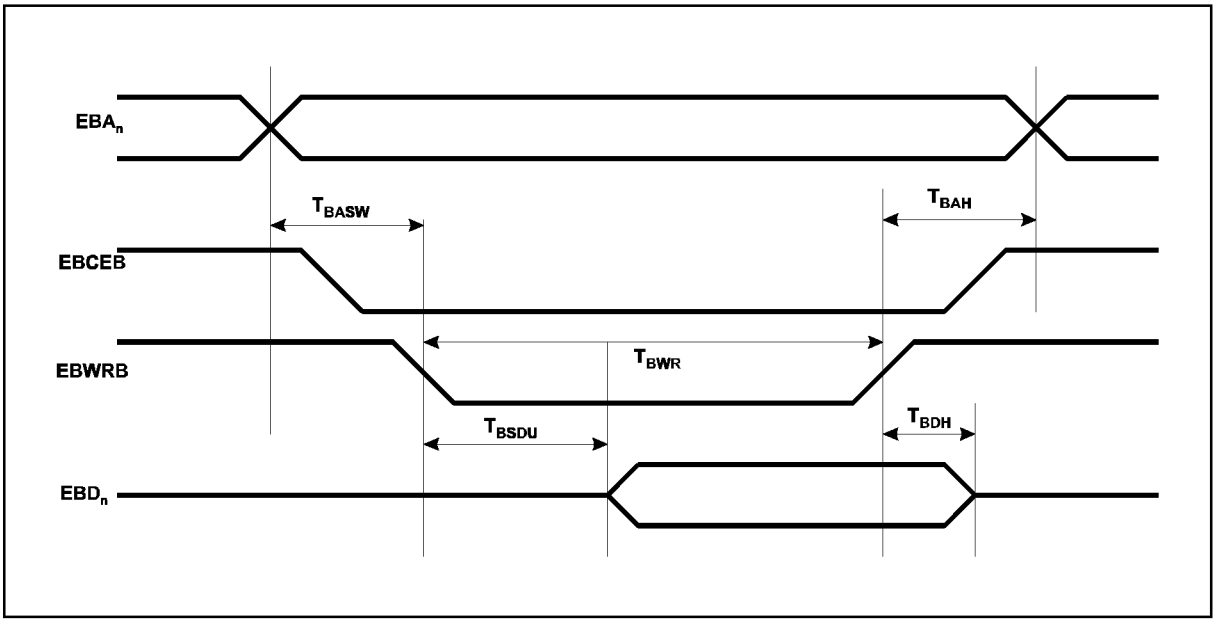

Figure 27. External Buffer RAM Write Timings

Symbol	Description	Min	Max
TBASW	EBA[ ] to EBWRB low	-	10nS
TBWR	EBWRB pulse width	90nS	-
TBDSU	EBWRBB low to EBD[ ] valid	-	10nS
TBDH	EBD[ ] hold after EBWRB high	0nS	-
TBAH	EBA[ ] hold after EBWRB high	70nS	-

13.5 Clock & Reset

13.5.1 Digital Clock Input (SYSCLK_D)

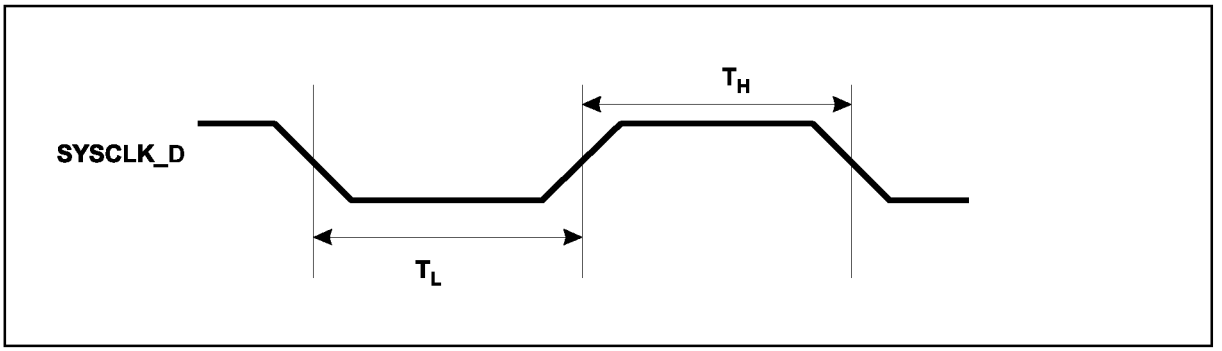

Figure 28. SYSCLK_D Timings

Symbol	Description	Min	Max
TH	High time (at Freq=10MHz	40nS	60nS
TL	Low time (at Freq=10MHz)	40nS	60nS
	Frequency tolerance	-	100ppm
	Operating Frequency	0	10MHz

13.5.2 Clock Output (SYSCLK_X), External µP mode only

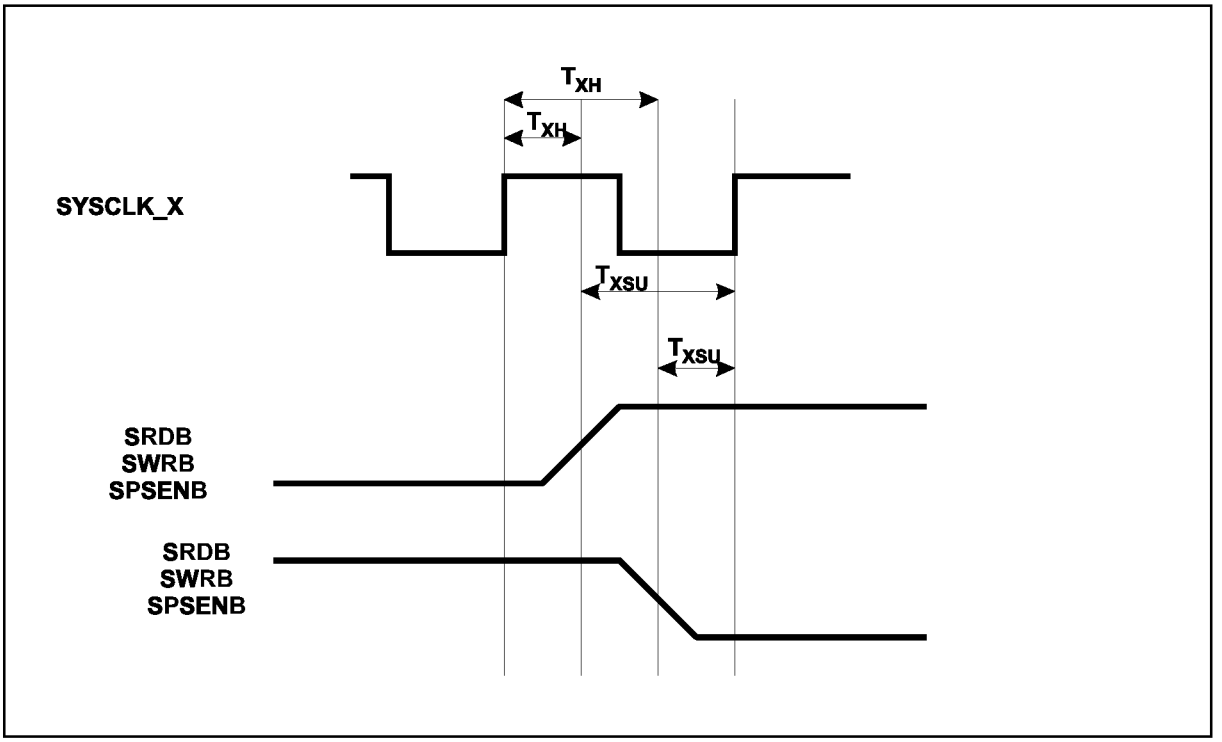

Figure 29. External Processor Synchronisation

Symbol	Description	Min	Max
TXSU	SRDB, SWRB, SPSENB setup to SYSCLK_X	50nS	-
TXH	SRDB, SWRB, SPSENDB hold after SYSCLK_X	0nS	-

13.5.3 Reset

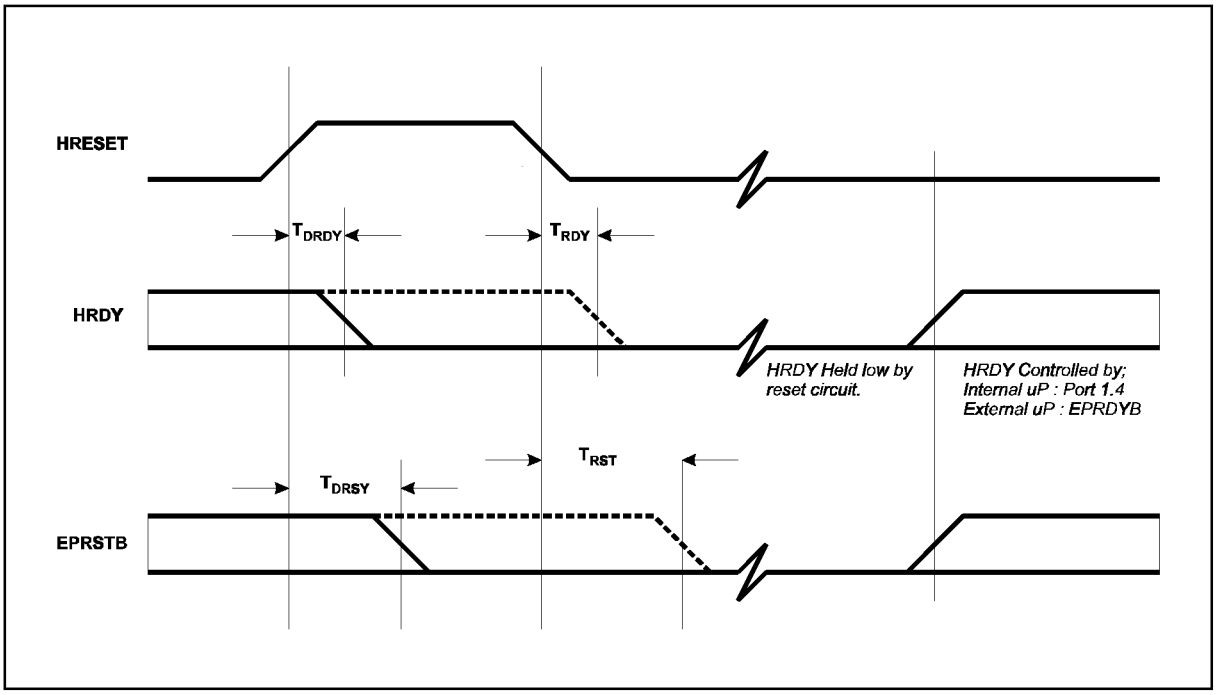

Figure 30. Reset Timings

Symbol	Description	Min	Max
TDRDY	HRESET high to HRDY low (using SYSCLK_D)	-	130nS
TRDY	HRESET low to HRDY low (using SYSCLK)	-	100nS
TDRST	HRESET high to EPRSTB low (using SYSCLK_D)	-	130nS
TRST	HRESET high to EPRSTB high (using SYSCLK)	-	100nS