BRM1553D
Hardware-Software Interface Document (HSID)

Mil-Std-1553 Bus Controller, Remote Terminal and Bus Monitor

SW Driver Compatible with DDC® ENHANCED MINIACE®

USER'S MANUAL FOR:

BRM1553D IP Cores, OCTAVA™ Family, Minuet™ Family, Orchestra™ Boards

Version 8 –With optional wiring fault detection and location

Oct 2017 – Diagrams bug fixes

Jan 2018 – Base address selection for stacks

* DDC® and MINI-ACE® are registered trademarks of Data Device Corporation, Bohemia, NY, USA.
There is not any affiliation between Data Device Corporation and Sital Technology, Ltd.

Copyright (C) 2018 by Sital Technology Ltd.

All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Sital Technology Ltd.

TABLE OF CONTENTS

1 INTRODUCTION 8
1.1 About this manual 8
1.2 About BRM1553D – Mil-Std-1553 Core 9
1.3 SmartWiring 10
1.4 Built-In testing 11
1.5 Terms used in this document 12

Part I: Remote Terminal and Monitor Mode - BRM1553D-RTMT 13

2 Remote Terminal and Monitor 14
2.1 Basic Functionality 14
2.1.1 RT and its Setup 14
2.1.2 Message Monitor and its Setup 14
2.1.3 Non MIL-STD-1553B settings 16
2.2 Block Diagram 17
2.2.1 Decoder/Encoder 17
2.2.2 The RT State Machine 17
2.2.3 The MT State Machine 17
2.3 software Interface 18
2.3.1 General Description 18
2.3.2 Registers Operation 18
2.3.2.1 Interrupt Mask Register #1 Address 0x0 20
2.3.2.2 Configuration Register #1 Address 0x1 21
2.3.2.3 Configuration Register #2 Address 0x2 22
2.3.2.4 Start Reset Register Writing to Address 0x3 23
2.3.2.5 Stack Pointer Register Reading from Address 0x3 23
2.3.2.6 RT Sub-address control word Address 0x4 23
2.3.2.7 Time Tag Register reading from Address 0x5 24
2.3.2.8 Interrupt Status Register #1 address 0x6 25
2.3.2.9 Configuration Register #3 address 0x7 26
2.3.2.10 Configuration Register #4 address 0x8 27
2.3.2.11 Configuration Register #5 address 0x9 28
2.3.2.12 RT/Monitor Data Stack pointer address 0xA 29
2.3.2.13 RT Last command register address 0xD 29
2.3.2.14 RT Status word address 0xE 29
2.3.2.15 RT BIT word address 0xF 30
2.3.2.16 BC Encoder test register 0x11 30
2.3.2.17 Passive TDR Level 0x12 30
2.3.2.18 Constant Value 0x13 31
2.3.2.19 latched time tag register 0x15 31
2.3.2.20 Configuration Register #6 address 0x18 32
2.3.2.21 Configuration Register #7 address 0x19 33
2.3.2.22 Version of Core Register address 0x1A 33
2.3.2.23 Self Test Status register 0x1C 33
2.3.2.24 Interrupt mask register #2 Address 0x1D 34
2.3.2.25 Interrupt status register #2 Address 0x1E 34
2.3.3 Memory Data Structure and Modes of Operation 35
2.3.3.1 Memory Mapping 35
2.3.3.2 RT Stack 36
2.3.3.3 Message Monitor Stack 36
2.3.3.4 Stack contents 38
2.3.3.5 Mode Code Data 39
2.3.3.6 Mode Commands Interrupt Enable Lookup table 40
2.3.3.7 Sub-address Lookup Table 41

2.3.3.8 Sub-address Control Word 41
2.3.4 Memory management modes: 43
2.3.5 Illegalization look up table 44
2.3.6 Message Monitor look up table 46
2.3.7 Single Transmit Message Mode 47
2.3.8 Single Receive Message Mode 48
2.3.9 Receive Double Buffering Message Mode 49
2.3.10 Circular Buffer Message Mode 50
2.3.11 Global Circular Buffer Message Mode 51

Part II: Legacy Bus Controller Mode - BRM1553D-BC 52

3 Bus Controller Introduction 53
3.1 Overview 53
3.1.1 Non MIL-STD-1553B settings 53
3.2 Block Diagram 54
3.2.1 The State Machines 54
3.3 Bus Controller Operation Modes 56
3.3.1 Base Operation 56
3.3.2 Multi Frame Setup 56
3.3.3 Auto-repeat mode 56
3.3.4 Multi-frame & Auto-repeat 57
3.3.5 Retry 57
3.3.6 Stop on error 58
3.3.7 Stop on status bits 58

4 BC Programming and Setup 59
4.1 Registers 59
4.2 Registers Map 59
4.3 Register's Short Form Look Up Table 61
4.3.1 Interrupt enable register #1 Address 0x0 62
4.3.2 Configuration Register #1 Address 0x1 63
4.3.3 Configuration Register #2 Address 0x2 64
4.3.4 Start Reset Register Writing to Address 0x3 65
4.3.5 Stack Pointer Register Reading from Address 0x3 65
4.3.6 Current BC control word. Address 0x4 65
4.3.7 Time Tag Register – Address 0x5 66
4.3.8 Interrupt Status Register #1 address 0x6 67
4.3.9 Configuration Register #3 address 0x7 68
4.3.10 Configuration Register #4 address 0x8 69
4.3.11 Configuration Register #5 address 0x9 70
4.3.12 BC Frame Timing Remaining address 0xB 70
4.3.13 BC Message Timing Remaining address 0xC 70
4.3.14 BC frame time register address 0xD 70
4.3.15 BC Encoder test register 0x11 71
4.3.16 Passive TDR Level 0x12 71
4.3.17 Constant Value 0x13 71
4.3.18 latched time tag register 0x15 71
4.3.19 Configuration Register #6 address 0x18 71
4.3.20 Configuration Register #7 address 0x19 72
4.3.21 Version of Core Register address 0x1A 72
4.3.22 BC Condition code reading from register address 0x1B 72
4.3.23 Self Test Status register 0x1C 73
4.3.24 interrupt mask register #2 0x1d 73
4.3.25 interrupt status register #2 0x1E 74
4.4 Memory 75
4.4.1 Memory Mapping 75
4.4.2 Stack contents 76
4.4.3 The Message Block 79

4.4.4 The BC Control word 80

Part III: Enhanced Bus Controller Mode - (eBC) *81*

5 Enhanced Bus Controller 82
5.1 Introduction 82
5.1.1 Non MIL-STD-1553B settings 82
5.2 Bus Controller operation modes 83
5.2.1 Basic operation 83
5.3 Programming and Setup 83
5.3.1 Registers 83
5.3.1.1 Interrupt enable register #1 Address 0x0 84
5.3.1.2 Configuration Register #1 Address 0x1 85
5.3.1.3 Configuration Register #2 Address 0x2 86
5.3.1.4 Start Reset Register Writing to Address 0x3 86
5.3.1.5 Stack Pointer Register Reading from Address 0x3 87
5.3.1.6 Last\current BC message control word. Address 0x4 87
5.3.1.7 Time Tag Register – Address 0x5 87
5.3.1.8 Interrupt Status Register #1 address 0x6 88
5.3.1.9 Configuration Register #3 address 0x7 89
5.3.1.10 Configuration Register #4 address 0x8 89
5.3.1.11 Configuration Register #5 address 0x9 90
5.3.1.12 BC Frame Timing remaining address 0xB 90
5.3.1.13 BC Message Timing Remaining address 0xC 90
5.3.1.14 eBC initial instruction register address 0xD 91
5.3.1.15 Test register address 0x11 91
5.3.1.16 Passive TDR Level 0x12 91
5.3.1.17 Constant Value 0x13 91
5.3.1.18 Latched time tag register 0x15 91
5.3.1.19 Configuration Register #6 address 0x18 91
5.3.1.20 Configuration Register #7 address 0x19 92
5.3.1.21 Back End Version of Core Register address 0x1A 93
5.3.1.22 BC Condition code reading from register address 0x1B 93
5.3.1.23 BC GP flags control by writing to register address 0x1B 94
5.3.1.24 Self Test Status register 0x1C 94
5.3.1.25 Interrupt mask register #2 Address 0x1D 95
5.3.1.26 Interrupt status register #2 Address 0x1E 95
5.3.1.27 BC General Purpose Queue pointer register Address 0x1F 96
5.4 Memory 97
5.4.1 Message Sequence Control 97
5.4.2 Variables 98
5.4.3 Instructions 99
5.4.4 Message Block 103
5.4.5 General Purpose Queue 105
5.4.6 Minor-Major message gap control 106

Part IV: Other Modes. *108*

6 External Loop-back 109

Part V: Extended Bit Rate - EBR1553 Option *112*

7 EBR Option 113
7.1 Introduction 113
7.2 BC Mode and its Hub 113
7.3 RT Mode 114
7.4 MT Mode 114
7.5 RT / MON Mode 114
7.6 External Loopback 114

Appendix A: Changes tracking 115

TABLE OF FIGURES

Figure 1: BRM1553D-RTMT Block Diagram 17
Figure 2: Stuck Filling Up....39
Figure 3: Message Monitor lookup table....46
Figure 4: Single Message mode transmit command....47
Figure 5: Single message mode transmit command....48
Figure 6: Receive Double Buffering Message Mode....49
Figure 7: Circular Buffer Message Mode 50
Figure 8: BRM1553D-BC Block Diagram....54
Figure 9: 3rd Word Gap Time....78
Figure 10: Execute BC sequencing 97
Figure 11: General purpose Queue....106
Figure 12: 4th word example time-to-next-message....107
Figure 13: Resistors Network for Loop-Back....109

1 INTRODUCTION

1.1 ABOUT THIS MANUAL

This document is the user's manual for BRM1553D, core for Sital's Mil-Std-1553 products. It covers the software interface – modes of operation, memory structure and registers operation - for Bus Controller (BC), Enhanced Bus Controller (eBC), Remote Terminal (RT) and Monitor (MT) parts of the core.

This manual is applicable for various Sital products, including Sital's IP Cores for FPGA, OCTAVA™ and Minuet™ components and Sital's board products.

This manual does not cover the hardware interface for these products.

The hardware interface of each product (IP cores, components, boards and other products from Sital) is covered under each product's Hardware Interface Manual.

In order to reduce the complexity of this manual, each part is covered in a different section. Some software registers and bits are used in a different manner between the various modes. It is assumed that if a user requires Bus Controller mode, for example, then it is not relevant to cover the functionality of a Remote Terminal mode in the same chapter, although it may be using the same set of registers and memory.

Therefore, the RT and MT Modes are covered in Part I, Legacy Bus Controller mode is covered on Part II, and Enhanced BC Mode is covered in Part III.

In some configurations of the core, depending on your product, the BC, RT and MT are packed together as a single IP core (BRM1553D-BCRTMT) or product. All reference made for BRM1553D-RTMT is equal to BRM1553D-BCRTMT, when in RT & MT Mode, and all reference made for BRM1553D-BC is equal to BRM1553D-BCRTMT, when in BC Mode.

Note:
Please note that it is assumed that the user of this manual is knowledgeable about the DDC® Mini-ACE® and Enhanced Mini-ACE® components, and their software interface. It is also assumed that the user of this manual has knowledge of Mil-Std-1553 protocol.

1.2 ABOUT BRM1553D – MIL-STD-1553 CORE

The BRM1553D core is used in various Sital products, including Sital's IP Cores for FPGA, OCTAVA™ and Minuet™ components and Sital's board products. This core provides a simple to use link for the MIL-STD-1553B Notice 2 bus interface.

The core supports all modes of Bus Controller (BC), Enhanced Bus Controller (eBC), Remote Terminal (RT), Message Monitor and Word Monitor (MT).

The BRM1553D core is designed to have the same Memory and Registers mapping and contents as DDC's Enhanced mini-ACE has. This compatibility allows the usage of software drivers developed for DDC's device to work with the core without any SW changes. Some of Sital's components, boards and other products are also pin-to-pin compatible to respective DDC® components and boards, making this a true replacement.

Note:
BRM1553D does not implement all of DDC's mini-ACE features, but rather the features which are used for the application. In most designs only a very small set of features are used from the DDC's interface. These features are implemented in the core. This document describes the parts that are implemented. If there is a feature that is required by your software, but is not supported by the core, then please contact Sital Technology.

The BRM1553D as a stand-alone core was validated to meet the MIL-STD-1553B Notice 2 Remote Terminal Validation test plan, thus reliving the user from mastering the standard.

The BRM1553D IP was developed with state machine approach and does not include any micro code, thus all configuration register values are constantly available inside the hardware. This allows the configuration setup procedure to be performed in any order. Obviously, the go command when relevant should be the last command.

The BRM1553D IP was developed in vendor independent VHDL supporting all FPGA vendors and families of FPGA in a wide and flexible form. All Validation testing test benches in VHDL were also developed to support the safe addition of features and capabilities over time.

The VHDL IP has a few generic parameters that are set by Sital Technology during synthesis and are based on the mode of operations and various characteristics of the desired configuration. The customer must select the following parameters:

1. If to include the message monitor.
2. If to include the bus controllers.
3. The clock frequency of the IP.
4. If to support MIL-STD-1760, or not.
5. The level of Sital Smart Wiring, if to only detect a wiring fault, or also to locate it.

The IP is synthesized and delivered based on these parameters in VHDL gate level format. This unique delivery mode is also VHDL vendor independent gate level description of the IP implementing the above parameters. This makes the IP future prof and allows future retargeting of the netlist to newer FPGA families.

1.3 SMARTWIRING

Sital technology introduced a capability to the BRM1553D IP that detects and locates wiring failures of the 1553 bus connected with this IP.

The capability provided is a fully passive sensing circuit that constantly tracks the dynamic impedance along the wires to which this IP connects. This capability is fully implemented inside the IP, and does not require any external electronic circuits.

At a base level, if enabled, SmartWiring indicates to the Host CPU a single bit in one of the status reports, that there was a wiring fault, by means of setting a single bit on. The bit is set for the block status words of the BC, MT and RT.

At a higher level, if enabled, SmartWiring detection is provided to the BIT word in RT, supporting system level detection of wiring issue and report.

Yet at a higher level, if enabled, SmartWiring data base of 128 words is provided to the Host, allowing additional analysis and pin pointing the location of the wiring fault in the system.

SmartWiring is enabled during IP generation in a number of levels. First level is to report to the local Host through interrupt status register 1 bit 14, second level would set bit 12 in BIT register, thus supporting system level report in RT mode, and a third level would allow location analysis by accessing registers from address 128 to 255.

SmartWiring_Clock is an input to the IP, and should be a multiple of the main clock in the range of 100Mhz to 200Mhz.

Note:
The Smart Wiring data base is mapped to the registers address space. It is mapped from address 32 (0x20) to address 127 (0x7F).
This is specifically required in case more than 1 IP is mapped in the address space, such as with a 4 channel 1553 IP.

1.4 BUILT-IN TESTING

The mini-ACE devices developed by DDC were assembled from a few ASIC chips in a hybrid package. In order to reduce risk of hybrid wiring disconnections, DDC included a few built-in tests to verify that these connections are intact.

These tests cover all hybrid traces, but obviously not all internal ASIC chips traces.

Some tests have been executed automatically during power-up, such as protocol test, and a pass fail report was issued within 2 ms of power-up.

Some additional tests, which are not built in, that cover board level traces have been executed by the EMA API and the host CPU.

The Sital Technology BRM1553D IP is holey implemented in one FPGA ASIC fabric and thus do not require execution of such built-in tests. Thus the IP is not muted for 2 ms at power-up to perform these built-in tests. However, some user programs rely on the successful pass of the test, and ping for its success as part of the device power-up sequence.

For these backward compatibility reasons, the BRM1553D IP emulates the tests result report and indicates successful pass in register 0x1C as if the tests were executed.

It is highly recommended to perform power-up testing to the IP from the host side. These tests should cover the PCB traces between the IP and the host CPU. These may include the address-data bus lines, the interrupt line if used, the RT-Address lines if testable, and transceiver-transformer connections if testable, and any other relevant PCB traces.

External Loopback test

The BRM1553D IP supports the loopback testing, either internally or externally between bus A and B. Please refer to chapter 6 for details.

1.5 TERMS USED IN THIS DOCUMENT

- **Remote Terminal (RT)** – The part of the FPGA that manages the 1553 communications and implemented by the core.
- **Bus Controller (BC)** – the system that handles Mil-Std-1553 timing and manages the transmission of RT's on the bus.
- **eBC** – Enhanced Bus controller.
- **SubSystem** – The whole box that connects to the MIL-STD-1553 bus that contains the FPGA – part of it is the Remote Terminal.
- **Host** - the CPU running the SubSystem and managing the IP core interface.
- **FPGA** – Programmable device that contains the BRM1553D - RT & MT MODE core and user logic and is part of the Subsystem.
- **User Logic** – Logic Circuit that resides in the FPGA that is not the BRM1553D - RT & MT MODE core and connects to it.
- **Message** – the group of command data and status words that compose a 1553 message.
- **IP** – Intellectual Property
- **Core** – Supplied logic circuit that interfaces to MIL-STD-1553 bus.
- **ICD** - Interface Control Document.
- **TA** – Terminal Address of the command / status words. Bits 11 to 15.
- **SA** – Sub Address of the command word. Bits 5 to 9.
- **WC** – Word count field of command. Bits 0 to 4.
- **Muxbus** – Time multiplexed bus known as the MIL-STD-1553B Notice 2 bus.
- **SW** – software.
- **BCST** – Broadcast command.
- **TX** – Transmit.
- **RX** – Receive.
- **SACW** – The Sub-address control word.
- **LUT** – Look Up Table.
- When a reference is made to a particular register and a specific bit, the notation is R01B12 for Register address 0x0001 and bit 12.
- When a reference is made to a particular memory location, the notation is M0100 for memory address 0x0100.
- **SmartWiring** – the capability to sense wiring problems of bus connected to this IP.

PART I:
REMOTE TERMINAL AND MONITOR MODE -
BRM1553D-RTMT

2 REMOTE TERMINAL AND MONITOR

2.1 BASIC FUNCTIONALITY

2.1.1 RT AND ITS SETUP

A Remote Terminal allows a subsystem to send and receive data to and from a Mil-Std-1553 bus. The communication is controlled by a Bus Controller (BC). A single BC controls the communications with up to 31 RTs, in accordance with its On Flight Program (OFP). An RT can never initiate any communication – receive or transmit data – without a direct command from a BC.

In some configurations it is demanded that BC Mode will be disabled by hardware in order to prevent accidental transmission of data to the bus.

The BRD1553D will power up in BC or IDLE mode. If 1760 mode is enabled (in some versions) the BRM1553D will power up in RT busy mode, as required by the Mil-Std-1760 standard. During the busy period, the RT will reply with the busy bit set in its status responses if it was assigned a valid RT Address. In this mode no data is sent and data received from the bus controller is dumped.

During the busy period, the software should setup all registers and memory locations that are needed for proper functionality of the core.

Once the setup of all registers and memory mappings is done, the busy bit can be reset (by writing to configuration register 1) and from that point on the core will start responding to 1553 bus commands.

The dual port shared memory array is reset on power-up. However the “Reset” signal does not reset the memories inside the core. If Reset is applied from the user logic, all registers in the core are reset.

The BRM1553D supports the Message Monitor mode, and this mode could be enabled concurrently to the RT running. Please refer to the next section for a description on the Message Monitor mode.

2.1.2 MESSAGE MONITOR AND ITS SETUP

The Message Monitor (MM) mode allows for recording selected messages from 1553 bus into the memory. The software defines a lookup table in memory where each bit enables the recording of a message with a specific Terminal Address, transmit or receive bit and a specific sub-address.

Message Monitor is useful in various ways. The simple usage is during flight when a Remote Terminal requires pieces of data from the ICD that is not sent directly to it, but rather to another Terminal. This mode reduces the amount of data on the bus, or allows adding features to a terminal without changing the ICD (changing the OFP software).

A second common usage is for recording purposes onto a recording media, and the ability to analyze the transactions on the Muxbus after flight.

The MM can operate as standalone, or concurrently with the RT.

The MM can be stopped and enabled during the operation of the device.

If the MM is enabled in the RT concurrent monitor mode then the MM will start monitoring after setting this mode to 'on'.

If the MM is operating as standalone, it will start recording transportation only after an MT start command.

In both modes, the MM can be stopped and started by the software.

When stopped or started, the 'on' and 'off' state will not hurt a currently running message.

The MM records all messages regardless of the RT operating or not. Messages meant for the concurrent RT are recorded if the appropriate 'enable' bits are set high (**Note**: This is different in the DDC® original MM).

Monitor setup involves programming the relevant configuration registers as well as setting up several memory addresses in the shared memory space.

Please refer to the register description and memory mapping locations to program the relevant memory locations.

The registers that need to be addressed are:

- Configuration register #3 to setup command and data stack sizes.
- Memory locations 0x0102 and 0x0103 for initial pointers.
- Selective monitor lists from Memory address 0x0280.
- Configuration register #1 to enable the Message monitor mode.

Optional:

- Reset register 0x03 bit 1 to starts the recording in standalone mode.
- MT Stop by writing 1 to bit 6 of Reset register 0x03.

Sital Unique mode: configuration register 6, bit 3 if set to '1' In Message Monitor only mode, the time tag is reset or updated based on selectively recorded mode messages. Only valid messages will load the time tag.

2.1.3 NON MIL-STD-1553B SETTINGS

The IP core is set to support different dialects than 1553B, by means of configuration.

These different settings affect both the RT and Message Monitor modes. The BC and eBC are also affected, please refer to the appropriate section.

The configuration register 3 bit 1, when set high, sets SA 31 to be non-mode, but rather a regular SA. In addition, this setting will assume only mode commands without data, so the RT will not reply, nor accept data words for messages designated to it with SA=0.

When this bit is set, both transmit last command and transmit status mode codes will not retain the status bits as they do in 1553B, but rather treat as any other mode command and clear the status bits for the reply.

Using configuration register #3 bit 5 set high, it is possible to manage all 11 LSBs of the RT status reply, including the ME bit. In 1553A, the RT responds to a valid command with invalid data words with a status word and ME set. In 1553A only bits 10 and 0 are meaningful. The RT when set to 1553A mode, will follow this behavior and respond with ME in status word, but if the host wants to set ME as well, it is possible through this configuration and settings.

Using Configuration register 5 bit 7, it is possible to turn Broadcast address 31 to a standard RT as defined by 1553A.

The RT response time was shortened to be slightly less than 7 us (5 us dead time) in order to support both 1553A and 1553B.

2.2 BLOCK DIAGRAM

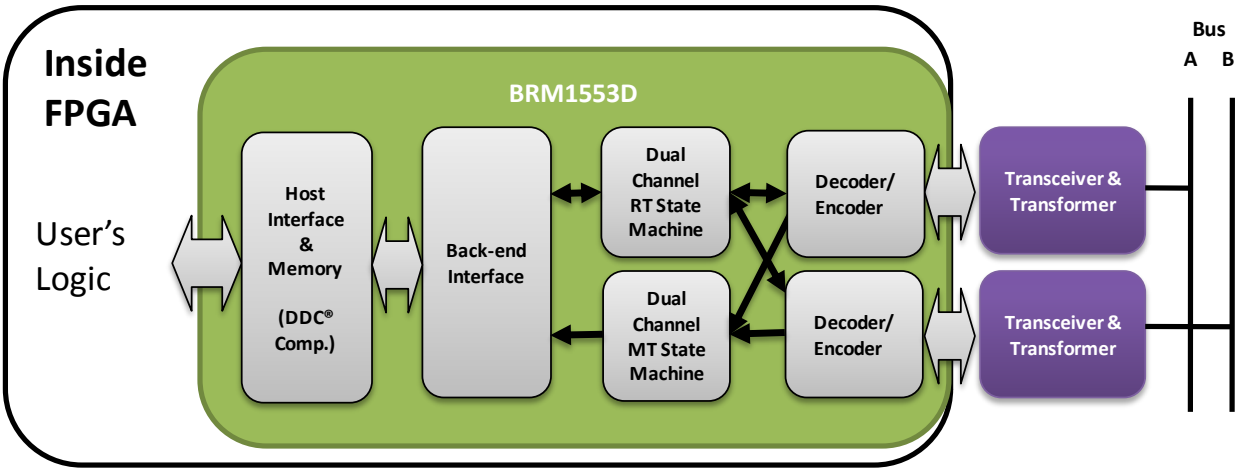

Figure 1: BRM1553D-RTMT Block Diagram

2.2.1 DECODER/ENCODER

The BRM1553D-RTMT core incorporates two decoders that translate the serial bus messages from the MIL-STD-1553 Manchester coding and format, into a 16 bit parallel data accompanied by status indications for each word. Two concurrently operated state machines, the RT state machine, and the MT state machines, analyze these words.

There are two identical blocks of Decoder/Encoder, in order to accommodate the dual-redundancy required by the Mil-Std-1553 standard.

2.2.2 THE RT STATE MACHINE

The RT state machine identifies a set of these bus words as a valid message, being 'transmit', 'receive', 'mode' or 'broadcast' message. When the RT State Machine decides it has to either store a word to the subsystem or fetch a word from the subsystem it interfaces through the backend to the internal memory array, and stores or reads that word.
When transmitting words, a word is fetched from the host interface by the back-end logic, and is loaded into the encoder by the Dual channel RT state machine. The encoder formats the words into serial Manchester 2 coding and emits them to the 1553 transceivers and onto bus A or B.

2.2.3 THE MT STATE MACHINE

The Dual channel Monitor state machine searches for valid commands. When a command is found, the state machine checks whether this command defines a message required for monitoring. In such case, the state machine manages the process of storing the words one by one in the interface memory into a pre-defined location.

The words received and transmitted are arranged in a dual port memory according to the memory mapping of the enhanced mini-ACE remote terminal. The BRM1553D-RTMT implements all memory modes of operations as will describe in the following sections.

2.3 SOFTWARE INTERFACE

2.3.1 GENERAL DESCRIPTION

The interface to the device is divided between several control registers and memory access. The registers are used to control the device and its operation, while the memory is used as the 1553 message interface and control.

The registers are mapped to address 0x0 to 0x20 and can be written or read (depending on their functionality). Memory can be in different size, depending on the device and ordered configuration. The IP core can be set between 2K, 4K, 8K, 16K, 32K or 64K – all by 16 bits, OCTAVA devices can be either 4K or 64K by 16 bits and Minuet devices can be 8K or 16K words. Sital boards vary between 8K to 64K Words RAM per channel. Please see the applicable hardware interface document for the specific memory configuration for your device.

Access to the registers or memory is done through the same address and data lines, however, in the IP core products and components, when accessing the registers, the MEM_nREG signal should be kept low, and when accessing the memory this signal should be high. At Sital's 1553 boards, the driver should set two different address blocks to distinguish between memory and registers.

There are several dependencies between the configuration of registers and the configuration and operation of the memory. The user must verify that all dependencies configured correctly with accordance to the required operation of the device.

2.3.2 REGISTERS OPERATION

The software interface of the core to the host processor consists of 20 internal operational registers for normal operation. These registers determine the device configuration, modes of operation, memory structure, interrupt control and status, etc.

The address mapping for the registers is detailed in the following table:

Note:
If a bit is defined as read only, then the core acts according to the specified level, and the read operation is available for software compatibility with DDC reasons. However, these bits may be written with values which will be ignored by the core.
If the bit is Write/Read, then the default value is as defined but could be changed to desired level for the required operation.

Address	Address Lines					Register Description (Read/Write)
	A4	A3	A2	A1	A0	
0x0	0	0	0	0	0	Interrupt Mask Register #1 (RD/WR)
0x1	0	0	0	0	1	Configuration Register #1 (RD/WR)
0x2	0	0	0	1	0	Configuration Register #2 (RD/WR)
0x3	0	0	0	1	1	Start/Reset Register (WR)
0x3	0	0	0	1	1	RT Command Stack Pointer (RD)
0x4	0	0	1	0	0	RT Subaddress Control Word Register (RD)
0x5	0	0	1	0	1	Time Tag Register (RD/WR)
0x6	0	0	1	1	0	Interrupt Status Register #1(RD)
0x7	0	0	1	1	1	Configuration Register #3 (RD/WR)
0x8	0	1	0	0	0	Configuration Register #4 (RD/WR)
0x9	0	1	0	0	1	Configuration Register #5 (RD/WR)
0xA	0	1	0	1	0	(RD) – all '0'
0xB	0	1	0	1	1	Not relevant for RT – see BC mode manual for details
0xC	0	1	1	0	0	Not relevant for RT – see BC mode manual for details
0xD	0	1	1	0	1	RT Last Command (RD)
0xE	0	1	1	1	0	RT Status Word Register (RD)
0xF	0	1	1	1	1	RT BIT Word Register (RD)
0x10	1	0	0	0	0	Not implemented
0x11	1	0	0	0	1	Test register. Supports external loopback test in Word Monitor mode. Requires BC
0x12	1	0	0	1	0	Bus A coupling to bus information.
0x13	1	0	0	1	1	Bus B coupling to bus information.
0x14	1	0	1	0	0	Not implemented
0x15	1	0	1	0	1	Latched Time Tag Register
0x16	1	0	1	1	0	Not implemented
0x17	1	0	1	1	1	Not implemented
0x18	1	1	0	0	0	Configuration Register #6 (RD/WR)
0x19	1	1	0	0	1	Configuration Register #7 (RD/WR) – bit 5 is set by bit 9
0x1A	1	1	0	1	0	version of Core (RD)
0x1B	1	1	0	1	1	Not relevant for RT – see BC mode manual for details
0x1C	1	1	1	0	0	BIT Test Status Register (RD)
0x1D	1	1	1	0	1	Interrupt Mask Register #2 (RD/WR)
0x1E	1	1	1	1	0	Interrupt Status Register #2 (RD)
0x1F	1	1	1	1	1	Not relevant for RT – see BC mode manual for details

Note:

Not all of the bits of the DDC® Mini-ACE® registers are implemented. Please see "1553 Core compatibility" document for details on all differences between the 1553 Core and DDC® Mini-ACE® and Enhanced Mini-ACE®.

2.3.2.1 INTERRUPT MASK REGISTER #1 ADDRESS 0X0

If the Host enables one of the bits below to '1' and the specified event occurs, an interrupt will be generated to the Host CPU. The status bits will be set regardless of the masking bit.

Bit #	Read/Write/Default	What event triggers the interrupt when enabled by '1'.
13	R/W/'0'	Transmitter timeout occurred
12	R/W/'0'	RT Stack Rollover occurred
11	R/W/'0'	Monitor command stack rollover occurred
10	R/W/'0'	Monitor data stack rollover occurred
9	R/W/'0'	Memory protection fault mask **
7	R/W/'0'	RT Address parity error found
6	R/W/'0'	Time tag counter rollover
4	R/W/'0'	End Of Message, if control word is enabled.
2	R/W/'0'	Transmitted words Loop back error, or Message Error in received words.
1	R/W/'0'	Mode command received.
0	R/W/'0'	EOM – End Of Message found

All other bits are read only and read value is zero.

Optional for the core: all bits can be written and read, but only the specified bits have effect.

Default Value: 0x0000 – all events masked.

** When Data pointer points to a non data block area (such as stack, LUT, 0x0100, etc…) for example – due to wrong memory setup by the host, the core detects this, and asserts bit 9 of the interrupt status register interrupt to '1'. If this bit is not masked than an interrupt will occur.

This bit (#9) is used in DDC Mini-ACE as memory handshake error. Since this cannot occur with the core, as the memory is internal to the FPGA, bit #9 is used for this more relevant issue.

Memory protection detection will also prevent the illegal write, so no memory overrun occurs.

2.3.2.2 CONFIGURATION REGISTER #1 ADDRESS 0X1

Configuration Register #1 starts in IDLE BC mode. In Mil-STD-1760 mode this register starts in RT busy mode.
For this document it is assumed that RT or MT functions are initiated.
For BC function details please refer to Part II: Bus Controller Mode.

RT without alternate status (configuration register #3 bit 5 = '0') is bits:

Bit #	Read/Write/Default	Description
15	R/W/'0'	'1' For Enhanced RT. A hardware input 'M1760' (if exists on the IP core) when set high before HW reset cycle is complete, powers up to RT + Busy for MIL-STD-1760 compatibility. '0' For legacy BC. In this case some of the other registers and bits have different meaning and usage.
14	R/W/'0'	'1' for Message Monitor **Only** function. If bit 15 is high (RT) this bit should be '0' and bit 12 enables MT.
13	R/W/'0'	'1' area B. Fixed memory locations => 0x0104, 0x0105 '0' area A. Fixed memory locations => 0x0100, 0x0101 Area Change will take place after message has ended. The read value is the actual area used.
12	R/W/'0'	'1' Message Monitor Enable. '0' Message Monitor Disabled.
11	Assumed 1	'1' – dynamic bus control bit always low.
10	R/W/'0'	Busy bit. If '0' is written, Busy bit and busy state of the RT is engaged. No words will be sent by the RT in response to commands only status word with busy bit set to '1'.
9	R/W/'0'	Service Request bit. If '0' is written, Service Request bit in status word is set to '1'.
8	R/W/'0'	Sub System Flag bit. If '0' is written, Sub System Flag bit in status word is set to '1'.
7	R/W/'0'	RT Flag bit. If '0' is written, RT Flag bit in status word is set to '1'.
2	Read/'0'	'1' – MT state machine has been started '0' – MT is idle. Before trigger on, or after trigger off, or just off.
0	Read/'0'	Message in Progress. Is set by the core to '1' on SOM and reset to '0' on EOM.

RT with alternate status (configuration register #3 bit 5 = '1') is bits:

Bit #	Read/Write/Default	Description
15	R/W/'0'	'1' For Enhanced RT.
14	R/W/'0'	'0' for Enhanced RT.
13	R*/W/'0'	'1' area B. Fixed memory locations => 0x0104, 0x0105 '0' area A. Fixed memory locations => 0x0100, 0x0101 Area Change will take place after message has ended. *The read value is the actual area used.
12	R/W/'0'	'1' Message Monitor Enable. '0' Message Monitor Disabled.
11	R/W/'0'	A '1' on this bit sets RT status bit 10 to '1'. Message Error bit.
10	R/W/'0'	A '1' on this bit sets RT status bit 9 to '1'. Instrumentation bit.
9	R/W/'0'	A '1' on this bit sets RT status bit 8 to '1'. Service Request bit.
4	R/W/'0'	A '1' on this bit sets RT status bit 3 to '1'. Busy bit.
3	R/W/'0'	A '1' on this bit sets RT status bit 2 to '1'. Subsystem flag bit.

1	R/W/'0'	A '1' on this bit sets RT status bit 0 to '1'. Terminal Flag bit.
0	Read/'0'	Message in Progress. Is set by the core to '1' on SOM and reset to '0' on EOM.

2.3.2.3 CONFIGURATION REGISTER #2 ADDRESS 0X2

Bit #	Read/Write/Default	Description
15	Assumed 1	'1' – Enhanced Interrupts mode enabled.
14	R/W/'0'	'0' – No Memory Parity bit use. '1' – Enable Memory Parity for systems that have parity bit support.
13	Assumed 0	'0' – Disables BUSY lookup table.
12	R/W/'0'	'0' – Double buffer for Rx messages is disabled. '1' – Double buffer for Rx depends on bit 1 of this register.
11	Assumed 1	'1' – Overwrite Invalid data Enabled.
10	Assumed 0	'0' – Rollover at 256-word boundaries
9..7	R/W/'000'	Time Tag resolution: "000" – 64 usec, "001"– 32 usec, "010"– 16 usec, "011" – 8 usec, "100"- 4 usec, "101"- 2 usec, "110" – Increment each time a "1" is written to Start Reset Register address 0x03 bit 4. "111" – Increments on Rising edge of TAG_CLK.
6	R/W/'0'	'0' – Time Tag counter controlled by Host only. '1' – clears Time Tag counter when synchronize without data mode command is received.
5	R/W/'0'	'0' – Time Tag counter controlled by Host only. '1' – Loads Time Tag counter when synchronize with data mode command is received.
4	R/W/'0'	'1' – Auto clear of interrupt status registers #1 and #2 as a result of CPU reading their value. Clears only register read. '0' – clear by reset register 0x03 only.
3	R/W/'0'	'0' – generates a 500ns low pulse on the INTn signal. '1' – Level mode. INTn stays low until the host reads status register(s).
2	R/W/'0'	'0' – Service Request Status bit only set by Host. '1' – Service request set by Host, is cleared by 1553 Transmit Vector Word mode command.
1	R/W/'0'	'0' – If bit 12 is '1', globally sets double buffer for all Rx commands. If bit 12 is '0', globally sets single buffer. '1' – Each SA can have different buffering mode depending on sub-address lookup table (see below).
0	R/W/'0'	'0' – BCST pointers are the same as Receive pointers. '1' – Separate BCST data pointers from receive data pointers can be defined in the sub-address LUT.

2.3.2.4 START RESET REGISTER WRITING TO ADDRESS 0X3

This is a write only register. Reading back will return '0's.

Bit number	Read/ Write/ Default	Description
10	Write	'1' clear self test register at address 0x1C.
9	Write	'1' initiate RAM self test.
7	Write	'1' initiate Protocol self test.
6	Write	'1' Stop the Message Monitor from recording.
4	Write	'1' Increments the Time Tag register if such mode set in configuration register #2.
3	Write	'1' resets the Time Tag counter. The time tag counter is also controlled by 'Synchronize' and 'Synchronize with data mode' commands from the 1553 bus.
2	Write	'1' resets the interrupt. Resets the FF that latches the interrupt condition. If the condition for interrupt persists, the relevant FF would be set again until the condition for causing this interrupt is cleared. Reset of Interrupt Register #1 and #2.
1	Write	'1' Starts the Message Monitor (MM) after it was stopped by writing '1' to bit 6 or it is in the Message Monitor only mode. In RT/MM mode a start is not required.
0	Write	'1' Reset core. Reset all registers, FFs in core. Memories are not reset, and should be reset by CPU. If '1', all other bits have no effect.

2.3.2.5 STACK POINTER REGISTER READING FROM ADDRESS 0X3

This is a read only register. Writing to it is explained above.

Bit number	Read/ Default	Description
15..0	Read 0x0	Copy of the Stack pointer value. After the EOM bit (15) is written to the first word of the stack entry, the stack pointer is incremented by 4. This pointer value can either be read from the memory in address 0x100, or from this register.

2.3.2.6 RT SUB-ADDRESS CONTROL WORD ADDRESS 0X4

Read back of the currently used RT Sub address control word.

2.3.2.7 TIME TAG REGISTER READING FROM ADDRESS 0X5

The time tag counter can be reset by writing to this register the value 0x0 or be loaded to any other value by writing the value into this register.

The time tag counter counts in clock steps defined by configuration register #2.

The time tag counter is also managed by the 1553 bus controller by two mode commands:

- Synchronize without data, will reset this counter, and
- Synchronize with data, will load this counter to the value sent on the bus.

The counter will rollover every 2^16 * resolution. Resolution is controlled by configuration register #2.

The value of this counter is also reset by the hardware raw_reset line, or the software reset done by writing '1' to the LSB of the reset register.

The value of the time tag counter is placed in the stack of each newly arrived message. Please see "Stack contents" section for details.

Sital Unique Mode: For monitor only mode, if configuration register 6 bit 3 is set high, the time tag register is also updated by all RT time tag mode commands that are selected to be recorded.

Bit number	Read/ Write/ Default	Description
15..0	R/W/'0'	Read the Time Tag Counter. Write a new value to the time tag counter.

2.3.2.8 INTERRUPT STATUS REGISTER #1 ADDRESS 0X6

This register indicates the cause of an interrupt.

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (recommended robust method) if configuration register #2 bit 4 is '1'.
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit number	Read/Write/ default	Description
15	Read	This bit is an OR of all interrupts bits of this register.
14	Read	'1' – Wiring Fault was detected using SmartWiring, when enabled
13	Read	'1' – Transmitter fail safe timeout. If set, this bit it is cleared only after receipt of a new valid command from the 1553 bus.
12	Read	'1' – Indicates that the Stack pointer has rolled over.
11	Read	'1' – Indicates that the Monitor command stack has rolled over.
10	Read	'1' – Indicates that the Monitor data stack has rolled over.
9	Read	'1' – Indicates Data Pointer points to a protected area where data blocks should not be present. Data was not written and memory was not corrupted. (Handshake Error in DDC devices)
7	Read	'1' – indicates RT Terminal Address Parity Error.
6	Read	'1' – indicates Time Tag Rollover from 0xFFFF to 0x0000.
4	Read	'1' – indicates an EOM occurred and the appropriate EOM bit was enabled in the control word.
2	Read	'1' – Message Error has been detected OR transmitted data or status hasn't been echoed back correctly.
1	Read	'1' – Mode command has been received and it was enabled by its individual interrupt enable bit.
0	Read	'1' – indicates that end of message has occurred.

2.3.2.9 CONFIGURATION REGISTER #3 ADDRESS 0X7

Bit #	Read/Write Default	Description
15	W/R/'0'	'1' always in Enhanced mode.
14..13	W/R/"00"	"00" RT Stack size is 256 words (64 message entries) "01" RT Stack size is 512 words (128 message entries) "10" RT Stack size is 1024 words (256 message entries) "11" RT Stack size is 2048 words (512 message entries)
12..11	W/R/"00"	Monitor Command Stack size in words, number of messages and the bits updated by the core in the pointer at address 0x0102*: "00" - 256 words, 64 message entries, 8 LSBs. "01" - 1024 words, 256 message entries, 10 LSBs. "10" - 4096 words, 1024 message entries, 12 LSBs. "11" - 16384 words, 4096 message entries, 14 LSBs. *Base value written to 0x0102/0x0106 should be >= 0x0300. *Base address should be aligned with the size.
10..8	W/R/"000"	Monitor Data stack size in words and number of bits updated by the core in the pointer at address 0x0103*: "111" - 512 words, 9 LSBs. "110" - 1024 words, 10 LSBs. "101" - 2048 words, 11 LSBs. "100" - 4096 words, 12 LSBs. "011" - 8192 words, 13 LSBs. "010" - 16384 words, 14 LSBs. "001" - 32768 words, 15 LSBs. "000" is illegal for message monitor and will be treated as "111". *Base value written to 0x0103/0x0107 should be >= 0x0400. *Base address should be aligned with the size.
7	W/R/'0'	'0' enables illegalization. The RT will return Message Error for messages selected by the table in addresses 0x0300 to 0x03FF, please refer to illegalization section. '1' will disable illegal message function. Addresses 0x0300 to 0x03FF can be used for data blocks.
6	Assumed 1	'1' Respond to reserved mode commands.
5	W/R/'0'	'0' - Only the Dynamic Bus Control Acceptance, Busy, Service Request, Subsystem Flag, and Terminal Flag RT Status Word bits are under control of the host processor, by means of bits 11 through 7 of Configuration Register #1. '1' – All status response bits of the RT are under the control of the Host processor by means of setting bits 1 to 11 in configuration register #1.
4	Assumed 1	'1' Illegal receive commands' data is dumped.
3	Assumed 1	'1' When Busy receive command data is dumped.
2	W/R/'0'	'0' – The host processor, via Configuration Register #1, controls Terminal flag status word bit. '1' – In addition to the host control, If transmitter timeout, or echo validity fail the terminal flag is set automatically.
1	W/R/'0'	'0' Mode code is detected for sub-address 00000 and 11111 (1553B) '1' Mode code is detected for sub-address 00000 (1553A), no response for mode messages with data words, and no data words transmitted. "Transmit status" and "transmit last command" mode commands will update status bits.
0	W/R/'0'	'0' – All mode command data words are saved in the same location in the RAM, based on the pointers of the sub address lookup table. '1' – The data word associated with transmit or receive mode words will be stored or taken from address 0x110 ... 0x13F. Please refer to "Mode Code Data" section below.

	The data associated with receive mode commands with data is also stored in the 3rd word in the stack.

2.3.2.10 CONFIGURATION REGISTER #4 ADDRESS 0X8

Bit #	Read/ Write/ default	Description
15	R/W/'0'	'0' – The BIT word transmitted as a response to "Transmit BIT word" is an internal BIT register. '1' – The BIT word transmitted is the one stored by the Host in address 0x0123.
14	R/W/'0'	'0' – BIT word transmitted even when RT is busy. '1' – No BIT word transmitted when BUSY bit is set.
13	R/W/'0'	Affects mode commands with mode code x16 to x1F '0' – Mode code data word isn't transmitted when BUSY. '1' – Mode code data word is transmitted when BUSY.
12	R/W/'0'	See BC manual. The core works in the Expanded Control Word and all its 15 bits are meaningful.
3	R/W/'0'	'1' – Enables latching of RT Address and parity by writing to configuration register #5. '0' – No latching of RT Address nor its parity. Refer to configuration register #5 for more options on RT Address source.
2..0	R/W/0	Write "010" to enable the loopback test. Refer to chapter 6 for detailed description. Write 0 to disable the test.

2.3.2.11 CONFIGURATION REGISTER #5 ADDRESS 0X9

Bit #	Read/ Write/ default	Description
15	R/W/'0'	This bit has no effect (*).
14	Read '0'	Single Ended select read port. Not supported. Reads '0'.
13	Read	External Tx Inhibit A line value
12	Read	External Tx Inhibit B line value
11	Assumed '1'	'1' – Expanded zero-xing is always enabled.
10..9	Read/Write "00"	Monitor (and BC) timeout: '00' – response timeout is set to approximately 18.5 us. '01' – response timeout is set to approximately 22.5 us. '10' – response timeout is set to approximately 53.5 us. '11' – response timeout is set to approximately 130 us. RT response time is set in HW to approximately 7 us.
8	Assumed '0'	'0' – The RT and MM do not check for a minimum of dead bus time between words.
7	Read/Write	'0' – Address 31 is detected as broadcast. '1' – Address 31 is used as normal RT address. Required for 1553A that does not support Broadcast. For 1553B it is hazardous to set this bit high.
6	Read	Reads the value of the RT_ADR_LAT hardware signal. '0' – RT address is constantly read from the hardware lines of RTAD(4..0) and RTADP. (As needed by MIL-STD-1760 standard). Rising from '0' to '1' not during hardware reset, the last RTAD(4..0) and RTADP wire values are latched as RT Address. '1' and Configuration register #4 bit 3 is '1' RT Address & parity is latched when writing to this register. The source for the RT address to be latched depends on Configuration register #6 bit 5.
5	Read/Write	Read the actual RT Address 4 Used in RT mode.
4	Read/Write	Read the actual RT Address 3 Used in RT mode.
3	Read/Write	Read the actual RT Address 2 Used in RT mode.
2	Read/Write	Read the actual RT Address 1 Used in RT mode.
1	Read/Write	Read the actual RT Address 0 Used in RT mode.
0	Read/Write	Read the actual RT Address Parity Used in RT mode.

(*) In DDC MiniAce BU65178 and in Sital's OCT-65178 devices, this bit specifies the input frequency:

'0' – 16 Mhz.
'1' – 12 Mhz

For other OCT-xxxxx devices this bit is ignored and Configuration Register #6 bits 1 and 0 select between 10, 12, 16, and 20 Mhz.

2.3.2.12 RT/MONITOR DATA STACK POINTER ADDRESS 0XA

Not implemented at this time. Will return 0.

2.3.2.13 RT LAST COMMAND REGISTER ADDRESS 0XD

Bit number	Read/ Write/ Default	Description
15..0	Read 0x0000	Value of the Last command processed by the Core. This register is updated at the same time SOM bit is set. Transmit Last command and Transmit status mode commands will never be "Last command" as defined by MIL-STD-1553B standard.

2.3.2.14 RT STATUS WORD ADDRESS 0XE

Bit number	Read/ Write/ Default	Description
15	Read/ '0'	'1' – Core works in limited mode. RT address limited to "11000" and only messages of 0 to 15 words, and 32 words are supported. Mode words without data.
10..0	Read 0x0000	Value of the status word of the core. Only 11 LSBs are shown.
10	0	Message Error
9	0	Instrumentation
8	0	Service Request
7	0	Reserved
6	0	Reserved
5	0	Reserved
4	0	Broadcast Received
3	0	Busy
2	0	Subsystem Flag
1	0	Dynamic bus Control Acceptance
0	0	Terminal Flag

2.3.2.15 RT BIT WORD ADDRESS 0XF

This is a read only register.

This register gathers errors found in the terminal. The content of this register is transmitted to the bus controller upon request by the "transmit BIT word" mode command.

Bit number	Read/ Write/ default	Description
15	Read	Transmitter Timeout
14	Read	Loop test failure B
13	Read	Loop test failure A
12	Read	SmartWiring detected a bus wiring fault, when enabled
11	Read	Transmitter Shutdown B
10	Read	Transmitter Shutdown A
9	Read	Terminal Flag Inhibit
8	Read	'0'
7	Read	High Word Count
6	Read	Low Word Count
5	Read	Incorrect Sync received
4	Read	Parity\Manchester Error received
3	Read	RT-RT Gap/Sync/Address error
2	Read	RT-RT No response Error
1	Read	RT-RT 2nd command word Error
0	Read	Command Word Contents Error

2.3.2.16 BC ENCODER TEST REGISTER 0X11

It is possible to perform an encoder test between bus A and B using this register only if IP supports the BC and word monitor. Please refer to Chapter 6 for details.

Bit number	Read/ Write/ default	Description
15	R/W/0	'1' would transmit on Bus A, '0' on Bus B
11	R/W/0	Writing '1' would trigger transmission
10	R/W/0	'1' - save the echo from the transmitting bus to monitor. '0' – do not save echo, only word received from opposite bus.

2.3.2.17 PASSIVE TDR LEVEL 0X12

pTDR level is indicated in bits 3..0.

Bits 15..4 should read 0x321.

Basic pTDR capabilities would read 0x3211. Advanced, fault location data vector should read 0x3218. No pTDR should read 0x3210.

2.3.2.18 CONSTANT VALUE 0X13

Read constant 0x7654. This constant can be used to verify proper Endian setup of system.

2.3.2.19 LATCHED TIME TAG REGISTER 0X15

The value of the time tag register address 0x5, is latched into this register each time a rising edge is detected on the Strobe_TT input line.

This allows the SW to read a very accurate Time Tag value when the hardware event occurred, such as when working with GPS systems.

2.3.2.20 CONFIGURATION REGISTER #6 ADDRESS 0X18

Bit number	Read/ Write/ default	Description
14	Read '0'	Enhanced CPU Access – Unlike in DDC device, the host access time for the IP core is 1 clock cycle (see DDC documentation for further information on DDC's CPU access). Therefore the host interface of the Core supports 'read' and 'write' bursts.
13	Assumed '1'	'1' – Command Stack increment on EOM, not SOM. The stack is incremented after end of message by 4.
12	Read/Write '1'	'1' – Global circular buffer enable. '0' – Global circular buffer disabled.
11..9	Write/Read 0x0	These three bits define the size of the Global Circular buffer. 0 – No global circular mode enabled. 1 – 128 words. 2 – 256 words. 3 – 512 words. 4 – 1024 words. 5 – 2048 words. 6 – 4096 words. 7 – 8192 words.
6	Assumed '0'	'0' – interrupt status Q disable.
5	Read/Write '0'	'0' – The RT address source is from the hardware pins RTAD(4..0) and RTADP (parity) bits. '1' – The RT address source is from configuration register 5. **Note**: in IP Cores and components, a hardware pin RTAD_SW_EN must be set to '1' in order to allow SW control of RT address. If this pin is '0', no SW control is available. See relevant Hardware Interface document.
4	Assume '1'	In RT/MT concurrent mode, The MT records all commands including those sent to the RT.
3	Read/Write '0'	Sital Unique mode: '1' – In MT only mode, the time tag is reset or updated based on selectively recorded mode messages. Only valid messages will load the time tag. '0' – MT only does not update the Time Tag counter, it is controlled only by host.
1..0	Read/Write "00"	These bits have no effect on IP cores. They specify the input frequency for OCTAVA and MINUET devices. "00" – 16 Mhz. "01" – 12 Mhz "10" – 20Mhz. "11" – 10 Mhz. For OCT-65178 these bits are ignored, and Configuration Register #5 bit 15 selects between 12 and 16 Mhz.

2.3.2.21 CONFIGURATION REGISTER #7 ADDRESS 0X19

Bit #	Read/Write/default	Description
9	R/W/0	Version of IP core. Used by DDC API for version testing. If '1' is written to this bit, it is read in bit 5 and 9.
5	R/0	Version of IP core feedback. If '1' is written to bit 9, it is read here.

This register is used to by the software determine the DDC chip to which the core is compatible to. A MiniACE chip will not return any feedback, while some versions of MicroACE will return feedback on bit 5.

Sital's IP core is compatible to the latest DDC chip, and therefore bit 5 returns '1' if written to bit 9.

2.3.2.22 VERSION OF CORE REGISTER ADDRESS 0X1A

This register returns the version of the BRM1553D core.

2.3.2.23 SELF TEST STATUS REGISTER 0X1C

This register indicates the state of the internal protocol and RAM tests

Writing to reset register at address 0x3 clears the content of this register.

Bit #	Description
15	Protocol Built-in-test complete
14	Protocol Built-in-test in progress, should be '1' for less than 1.1 ms(*).
13	Protocol Built-in-test passed
11	Protocol Built-in-test in progress or complete
7	RAM Built-in-test complete
6	RAM Built-in-test in progress, should be less than 1.1 ms.
5	RAM Built-in-test passed

(*) After hardware power up, the core initiates the protocol test automatically. Allow 1.1 ms for this test to complete.

Note: Please refer to the introduction section of this document for a description of the actual test coverage.

2.3.2.24 INTERRUPT MASK REGISTER #2 ADDRESS 0X1D

If the Host enables one of the bits below to '1' and the specified event occurs, an interrupt will be generated to the Host CPU. The status bit will be set regardless of the mask bit.

Bit #	Read/Write/Default	What event triggers the interrupt when enabled by '1'.
7	R/W/'0'	Monitor command stack passed the half way mark
6	R/W/'0'	Monitor data stack passed the half way mark
1	R/W/'0'	BIT test complete

2.3.2.25 INTERRUPT STATUS REGISTER #2 ADDRESS 0X1E

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (if clear by read enabled in configuration register #2).
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit #	Read/Write/Default	What event triggers the interrupt when enabled by '1'.
15	Read	This bit is an "OR" of bits 0 to 14 of this register. When any of the bits 0 to 14 is '1' then this bit will be '1'.
7	Read	Monitor command stack passed the half way mark
6	Read	Monitor data stack passed the half way mark
1	Read	BIT test complete
0 (note)	Read	'1' indicates that interrupt register #1 is requesting an interrupt. It is recommended to read Interrupt status register #2 if its bits are required and if bit 0 is '1' then read Interrupt status register #1.

2.3.3 MEMORY DATA STRUCTURE AND MODES OF OPERATION

The data reception and transmission is controlled on a message-by-message basis. Each message is stored in the memory or read from the memory based on mapping defined by the CPU during offline state. The CPU sets up the mapping and modes of operations in the dual port RAM and in the configuration registers. The mapping options are defined in the following sections.

2.3.3.1 MEMORY MAPPING

The following table suggests a memory mapping for the RT & MT modes.

Selective Message Monitor mode (SMM) is optional by SW setup and described in bold red color.

Address in Hex	Description	Comments
0000 to 00FF	RT Stack	Updated by Core
0100	RT Command Stack Pointer	Updated by Core
0101	RT Global Circular Pointer	Updated by Core
0102	Reserved OR **SMM command Stack Pointer**	**For SMM: Should be 0x0300 or bigger.** **Initiated by CPU, updated by core.**
0103	Reserved OR **SMM data Stack Pointer**	**For SMM: Should point to area for Monitor data** **Initiated by CPU, updated by core.**
0104 to 0107	Area B	Same as 100 to 103 but for Area B. NOT BUS B!!
0108 to 010F	Mode Commands Interrupt Enable Lookup table	Updated by CPU
0110 to 013F	Mode Code Data	Updated by CPU
0140 to 01BF	RT Lookup Table Area A	Updated by CPU. Area A NOT BUS A!!
01C0 to 023F	RT Lookup Table Area B	Updated by CPU. Area B NOT BUS B!!
0240 to 025F	Reserved.	Busy LUT is not implemented
0260 to 027F	Data Block	32 or 64 words per block.
0280 to 02FF	Data Blocks OR **SMM Lookup table**	32 or 64 words per block.
0300 to 03FF	Illegal LUT OR Data Blocks	See configuration register #3 bit 7. 32 or 64 words per block
0400 to 04FF	Data blocks OR **SMM command stack**	32 or 64 words per block OR **When SMM: 256 words, 64 messages stack.**
0500 to 06FF	Data blocks	32 or 64 words per block.
0600 to 07FF	**SMM data stack**	**When SMM: Cyclic Data stack of 512 words.**
0800 to 0FFF	Data blocks	32 or 64 words per block.
1000 to EOF	Data blocks	As much memory as the core was programmed to.

Note: All stack length bits should be 0 at base. Example: If SMM stack length is 512 then 9 LSBs of base address should be 0. 0x500 would **not** be a good choice SMM data stack base - prefer base to be 0x400 or 0x600.

2.3.3.2 RT STACK

The RT Stack is 256 words deep by default. The stack is a set of records, each built of 4 words. The RT Stack pointer points to the first word of the next empty record, ready for the next message to be stored.

The RT Stack pointer is incremented by 4 each time a valid message was serviced. The stack pointer is incremented and updated after the completion of a message.

All 4 words are updated in the stack for every message. The host CPU should read these words in order to understand what message has been stored, and treat the data payload according to the ICD.

The CPU SW should maintain an internal pointer, and compare its value against the Stack Pointer. The difference between the two pointers should indicate how many messages have been received from the bus and require SW attention. When all stack entries process is complete, the SW pointer should be equal to the Stack Pointer's value.

The RT stack does not save messages with errors. Only valid messages exist in the stack. The error bits in the block status word should all be '0'.

Hardware note for the IP core:
Both sides update the shared memory with the same clock. This assures data consistency when reading the data to either side – the host CPU, or the core. There is no need to read the stack pointer more than once.

2.3.3.3 MESSAGE MONITOR STACK

Monitor command stack should be handled the same way as the RT Stack. First the software should determine the size of both stacks by writing to the configuration registers.

The next step is to enable the messages that need to be monitored by writing '1' to the appropriate memory location in the lookup table in address 0x0280 to 0x02FF.

Then enable the Monitor by configuration register #1.

Once the Monitor starts functioning, the command stack is updated in a cyclic manner endlessly, or until disabled through configuration register #1.

The 4 words in command stack entry elements are similar to the RT command stack.

The MT data stack however is somewhat different. Messages in the data stack include the command words, status words, and data words as they are received from the bus.

The command stack will hold the monitored command.

The various message formats in Stack are listed below:

- CMDtx – Transmit Command
- STS – Status word
- STStx – Status word transmitted
- STSrx – Status word received
- DAT – Data word
- BCST - Broadcast
- { DAT } – replication 1 to 32 Data words

Data Stack contents:

Type	Data stack contents
BC => RT	{ DAT } STS
RT => BC	STS { DAT }
RT => RT (1) *see note below	CMDtx STStx { DAT } STSrx
RT => RT (2) *see note below	STStx { DAT } STSrx
BC => Mode w/o data	STS
BC => Mode transmit data	STS DAT
BC => BCST	{ DAT }
RT => BCST(1) *see note below	CMDtx STStx { DAT }
RT => BCST(2) *see note below	STStx { DAT }
BC => Mode receive data	DAT STS
BC => BCST Mode w/o data	<no words>
BC => BCST Mode data	DAT

Note:

In RT=>RT transfer, there are two options, which reflect the monitored data, whether both RTs that are involved in the message are monitored or only one of them:

If both the transmitting and receiving RTs are monitored, then the TX command would be placed in the command stack as seen above.

If only the 'Tx Command' is monitored, it will reside in the command stack, and the Rx command would be dropped.

In both cases the status of the Rx command would be the last word in the data stack. The host SW can determine that it is an RT=>RT transfer by checking the block status word where RT-RT transfer bit is set to '1'.

In RT=>BCST command the same behavior occurs, except for the last status word that does not exist.

2.3.3.4 STACK CONTENTS

The following Stack contents are valid for the RT stack or the Message Monitor (MM) stack.
The RT stack does not contain error messages, thus several bits are redundant.

Each stack record is composed of the following words:

Offset	Name	Bit	Description
0	Block Status Word	15	End of message flag (EOM). Set after completion of message.
		14	Start of message Flag (SOM). Set after receipt of valid command.
		13	Channel B. When High indicates command received on bus B. The default bus is Bus A, and then this bit is '0'.
		12	'1' - Error was detected in the message. This is an OR of bits 9 and 10 in the MM, and bit 8, 9 and 10 in RT stack.
		11	RT-to-RT message format. Set in RT stack only when it is the receiving RT.
		10	'1' indicates a 1553 violation in the message. Either wrong Sync, Manchester, parity, bit count, word count, etc… See RT stack bits 5..0 for further details.
		9	'1' indicates a 'no response' timeout in the message. In RT stack this bit is valid for RT-to-RT message, when the other RT did not respond on time.
		8	RT stack => '1' is Loop-back failed. MT stack => '1' is Message ended OK.
		7	'1' indicates RT global circular buffer rolled over during this message, or a MM data stack rollover has occurred during the write of one of the data words of the message.
		6	RT - '1' indicates an illegal message. MT – indicates wiring fault using SmartWiring, when enabled
		5	'1' indicates a word count error occurred in the message.
		4	'1' indicates an incorrect Sync in the message.
		3	'1' indicates an invalid word was detected in the message. Errors could be sync format error, Manchester error, parity error and bit count error.
		2	'1' indicates that an RT responded too fast, or a responding RT in RT2RT responded with an invalid status word, or the status RT address does not correspond with the RT address I the command.
		1	'1' indicates an error in the 2nd command in RT2RT message. It was either a receive command, or a mode command, or both commands relate to the same RT.
		0	'1' indicates an erroneous 1553 message such as a non-mode BCST transmit command.
1	Time Tag Word		16 bit real time counter incremented every 64 us. Maximum count is 4.2 sec (2^16*64us). Auto rollover.
2	Data Block Pointer		Points to the memory location of the first message word that was either stored or was transmitted.
3	Command		16 bits of the command word received.

Note:
Only valid messages are stored in RT stack.

When a command is received, then the core updates bit 14 of the current stack pointer address and writes the command into memory address of the stack pointer + 3. The Time Tag is set at the beginning of the Status word.

The Data Pointer is set to the address of the beginning of data for this message in the data memory.

When the message is finished, then bit 15 of the current stack is set to '1' (End of Message), bit 14 (Start of Message) is set back to '0' and the stack pointer is forwarded by 4 addresses.

The following diagram indicates when each of the fields being updated in relation to the 1553 message.

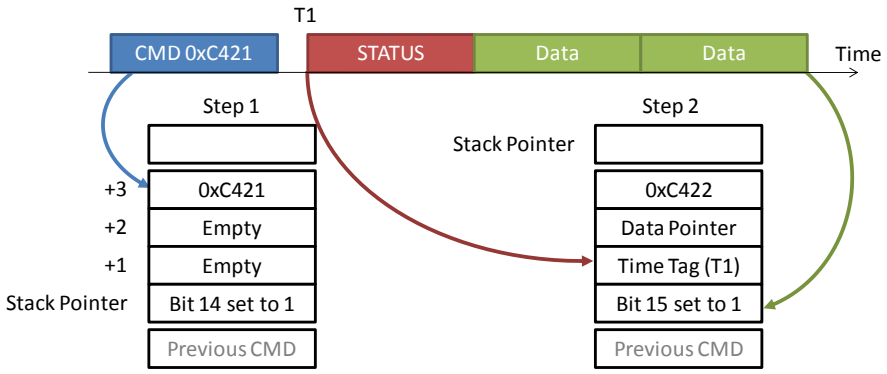

Figure 2: Stuck Filling Up

2.3.3.5 MODE CODE DATA

This look up table at addresses 0x110 to 0x13F is a storage place for all data words of mode commands that are either received or transmitted to the Bus Controller.

The following locations are relevant mode commands data:

Mode command Data	Address	Updated by	Comments
Synchronize with Data	0x0111	Bus controller	From BC
Selected Transmitter shutdown	0x0114	Bus controller	From BC
Override Selected Transmitter shutdown	0x0115	Bus controller	From BC
Transmit Vector Word	0x0120	Host CPU	Transmitted to BC
Transmit Last Command	0x0122	Core	Host data ignored
Transmit BIT Word	0x0123	Host CPU	Transmitted to BC
Broadcast Synchronize with Data	0x0131	Bus controller	From BC
Broadcast Selected Transmitter shutdown	0x0134	Bus controller	From BC
Broadcast Override Selected Transmitter shutdown	0x0135	Bus controller	From BC

Other locations in the table are RESERVED and UNDEFINED mode commands and are supported by the core and saved or read for the relevant address between 0x0110 and 0x013F.

2.3.3.6 MODE COMMANDS INTERRUPT ENABLE LOOKUP TABLE

This look up table at addresses 0x108 to 0x10F is a table for enabling an interrupt at the end of a mode command. In weapon bus protocols, typically, the synchronize mode codes are programmed to initiate an interrupt. Each bit in the table enables with a '1' one of the 64 non BCST and 32 BCST mode codes.

7 bits derived from (BCST & T/R & Mode Code Number) determine the bit-wise offset from 0x0108.

Mode command	Mode Code	Address	Bit Number	Bit Mask
Synchronize with Data	R 10001	0x0109	1	0x0002
Selected Transmitter Shut Down	R 10100	0x0109	4	0x0010
Override Selected Tx Shut Down	R 10101	0x0109	5	0x0020
Dynamic Bus Control	T 00000	0x010A	0	0x0001
Synchronize without Data	T 00001	0x010A	1	0x0002
Transmit Status Word	T 00010	0x010A	2	0x0004
Initiate Self Test	T 00011	0x010A	3	0x0008
Transmitter shutdown	T 00100	0x010A	4	0x0010
Override transmitter shutdown	T 00101	0x010A	5	0x0020
Inhibit terminal flag bit	T 00110	0x010A	6	0x0040
Override inhibit terminal flag bit	T 00111	0x010A	7	0x0080
Reset remote terminal	T 01000	0x010A	8	0x0100
Transmit vector word	T 10000	0x010B	0	0x0001
Transmit last command	T 10010	0x010B	2	0x0004
Transmit BIT word	T 10011	0x010B	3	0x0008
BCST Synchronize with Data	R 10001	0x010D	1	0x0002
BCST Selected Transmitter Shut Down	R 10100	0x010D	4	0x0010
BCST Override Selected Tx Shut Down	R 10101	0x010D	5	0x0020
BCST Synchronize without Data	T 00001	0x010E	1	0x0002
BCST Initiate Self Test	T 00011	0x010E	3	0x0008
BCST Transmitter shutdown	T 00100	0x010E	4	0x0010
BCST Override transmitter shutdown	T 00101	0x010E	5	0x0020
BCST Inhibit terminal flag bit	T 00110	0x010E	6	0x0040
BCST Override inhibit terminal flag	T 00111	0x010E	7	0x0080
BCST Reset remote terminal	T 01000	0x010E	8	0x0100

Note:
The core mechanism supports all possible mode commands. The appropriate bits can be set or reset for those reserved mode commands. It is a good practice to enable interrupts for all unused mode codes.

2.3.3.7 SUB-ADDRESS LOOKUP TABLE

The Sub-address lookup table (LUT) is composed of 4 sub LUTs dedicated for receive, transmit and broadcast data pointers, and a 4th LUT dedicated for sub-address control word.

Hex Address	Points to data for	Purpose	Comments
0x0141 **:** **0x015E**	Rx command SA1 : Rx command SA30	Receive Lookup table	Pointer for mode codes (SA0 and SA31) depends on configuration register setting.
0x0161 **:** **0x017E**	Tx command SA1 : Tx Command SA30	Transmit Lookup Table	Pointer for mode codes (SA0 and SA31) depends on configuration register setting.
0x0181 **:** **0x019E**	BCST SA1 : BCST SA30	Broadcast Lookup Table	Pointer for mode codes (SA0 and SA31) depends on configuration register setting.
0x01A1 **:** **0x01BE**	SA1 : SA31	Sub Address control words	Refer to next section for details.

For example: Assume the words received with Rx SA2 needed to be saved to memory location 0x0400 – Load address 0x0142 with value 0x0400.

Example 2: Assume the words received with Broadcast Rx SA2 needed to be saved also in location 0x0400 – Load address 0x0182 with 0x0400.

Tip:
It is wise to reserve one of the data blocks in memory for SA that are not covered by the ICD, and load all of those data pointers to that location. In case such a SA is executed by mistake or during testing, the data saved will not overrun the stack, required data or tables.

Warning:
If the pointers are set to a wrong location, like 0x0100, data received from the bus will be dropped and the core's protection circuits would set bit 9 of the interrupt status register to high.

2.3.3.8 SUB-ADDRESS CONTROL WORD

The Sub-address control word (SACW) is intended to specify the data management and interrupt behavior for each Sub-address. As seen in the following sections, the SACW is accessed for each valid command, its value defines the type of memory management requested by the Host CPU for that specific Sub Address and Transmit / Receive / Broadcast commands.

With the appropriate setup of the 30 Sub-address Control words the Host CPU can choose between Single buffer memory mode, Double buffer memory mode, Circular buffer memory mode and Global Circular buffer memory mode.

Note that broadcast (BCST) commands are always considered as receiving commands with no status reply, but have their unique memory management bits.

The SACW are stored in a fixed memory location in address 0x01A1 to 0x1BE. 0x01A0 and 0x1BF are offsets 0 and 31 which designate mode commands sub address value. For these two registers, only the interrupt enable bits enable or disables interrupt generations.

Bit number	Memory management reference
15	For Rx/BCST command : Double Buffer or Global Circular Enable(1)(2) *See notes below
14	Tx : End Of Message Interrupt enable.
13	Reserved
12	Tx : Memory Management bits. These bits define the size of circular buffer for Tx. Please see next table to sizes.
11	
10	
9	Rx : End Of Message Interrupt enable.
8	Reserved
7	Rx : Memory Management bits. These bits define the size of circular buffer for Rx. Please see next table to sizes.
6	
5	
4	BCST : End Of Message Interrupt enable.
3	Reserved
2	BCST Rx : Memory Management bits. These bits define the size of circular buffer for BCST. Please see next table to sizes.
1	
0	

Notes:

(1) A '1' will enable double buffer for receive commands if Memory Management Bits are 000. If they are bigger than Zero (equal 011 preferred) than Global Circular Buffer mode is used for that sub-address.

(2) A '0' will enable single buffer mode for receive commands if Memory Management Bits are 000, else local circular buffer mode with size specified in the next table.

2.3.4 MEMORY MANAGEMENT MODES:

Memory Mode Detailed memory mode descriptions follow in next sections.	Command type	Buffer size for bits 12...10 Tx command 7...5 Rx command 2...0 BCST command	Bit 15
Single buffer memory mode	Tx Rx BCST	0	0
Double buffer memory mode.	Rx BCST	0	1
SA specific Circular buffer – 128 words	Tx Rx BCST	1	0
SA specific Circular buffer – 256 words	Tx Rx BCST	2	0
SA specific Circular buffer – 512 words	Tx Rx BCST	3	0
SA specific Circular buffer – 1024 words	Tx Rx BCST	4	0
SA specific Circular buffer – 2048 words	Tx Rx BCST	5	0
SA specific Circular buffer – 4096 words	Tx Rx BCST	6	0
SA specific Circular buffer – 8192 words	Tx Rx BCST	7	0
Global Circular buffer * see notes below.	Rx BCST	Bigger than 0 (use 3)	1

Notes on Global Circular buffer size:

Global Circular buffer mode is used for 'receive' and 'broadcast receive' messages only. Configuration register #6 (register address 0x0018) defines the Global Circular Memory size. Bits 11, 10, 9 define the size as in the table above - Memory management modes – if bit 12 is '1'. Else Global Circular buffer is not enabled and single buffer mode will be used.

2.3.5 ILLEGALIZATION LOOK UP TABLE

From memory address 0x0300 to 0x03FF, functionality is under user's control. Configuration register #3 bit 7 defines this space as regular data blocks or as a table that defines the messages to be regarded as illegal.

If illegalization LUT mode is used (Configuration register #3 bit 7 is set to '0'), each bit in the table corresponds to a specific message, whether it is BCST, Tx or Rx, Sub-Address and Word Count.

When a bit is defined as '1' it will make that particular message illegal, and the core will respond with only a status word with Message Error (ME) bit set. For BCST messages, no status is replied, and the BC has other means of finding the ME.

Each valid command received will be examined by the LUT contents. There are 256 words of 16 bits. The 16 bits of address used by the core to access the LUT is composed of the following bits: (& operator attaches bits to a vector)

00000011	$\overline{\text{BCST}}$	Tx/ Rx	Sub-Address (4..0)	WC (4)

- 0x03 is the upper 8 bits of the address;
- Bit 7 is the BCST detection inverted (i.e. – '0' for Broadcast, '1' for no-broadcast);
- Bit 6 is '1' for transmit command and '0' for receive;
- SA is the 5 bits of the Sub-Address;
- WC (4) is the MSB of the word count (or mode code). When the particular address is fetched, its 16 bits data define either word count 0 to 15 or 16 to 31.
- The data itself is a bit-map of the illegal words, 16 at the time, depending on the MSB of WC.

The table initializes as all '0', meaning that all commands are legal.

Please see table below.

Examples:

In order to illegalize all Broadcast (BCST) commands for example, one has to write 0xFFFF to all addresses from 0x0300 to 0x037F.

If the user wants to illegalize a non-BCST transmit command, with Sub Address 3, and word count 30, one would write 0x4000 to address 0x03C7 (0x03 and '11000111'). Data is '0100000000000000' (0x4000) because bit 14 corresponds to word 30. If, for example, word 8 is illegal, then bit 8 on address 0x3C6 would be set to '1'.

There are entries to the LUT which will never be accessed because they are not valid commands, like BCST transmit commands.

Hex Address	Illegalizes
0x0300	BCST Rx SA0 Mode commands 0 to 15 (Example: bit 15 – word count 15)
0x0301	BCST Rx SA0 Mode commands 16 to 31
0x0302	BCST Rx SA1 words count 0 to 15
0x0303	BCST Rx SA1 words count 16 to 31
:	
:	
0x033E	BCST Rx SA31 Mode commands 0 to 15
0x033F	BCST Rx SA31 Mode commands 16 to 31
0x0340	BCST Tx SA0 Mode commands 0 to 15
0x0341	BCST Tx SA0 Mode commands 16 to 31
	0x0342 to 0x037D are not relevant for MIL-STD-1553B. Core will not access these addresses for illegalization anyway.
0x037E	BCST Tx SA31 Mode commands 0 to 15
0x037F	BCST Tx SA31 Mode commands 16 to 31
0x0380	RT Address, Rx SA0 Mode commands 0 to 15
0x0381	RT Address, Rx SA0 Mode commands 16 to 31
0x0382	RT Address, Rx SA1 words count 0 to 15
0x0383	RT Address, Rx SA1 words count 16 to 31
:	
:	
0x03BE	RT Address, Rx SA31 Mode commands 0 to 15
0x03BF	RT Address, Rx SA31 Mode commands 16 to 31
0x03C0	RT Address, Tx SA0 Mode commands 0 to 15
0x03C1	RT Address, Tx SA0 Mode commands 16 to 31
:	
:	
0x03FC	RT Address, Tx SA30 words count 0 to 15
0x03FD	RT Address, Tx SA30 words count 16 to 31
0x03FE	RT Address, Tx SA31 Mode commands 0 to 15
0x03FF	RT Address, Tx SA31 Mode commands 16 to 31

2.3.6 MESSAGE MONITOR LOOK UP TABLE

When a valid command is picked up by the Message Monitor, its state machine access a fixed area of the shared memory in order to determine if the message that follows needs to be monitored or not.

The fixed memory location of the Message Monitor Lookup table is 0x0280 to 0x02FF.

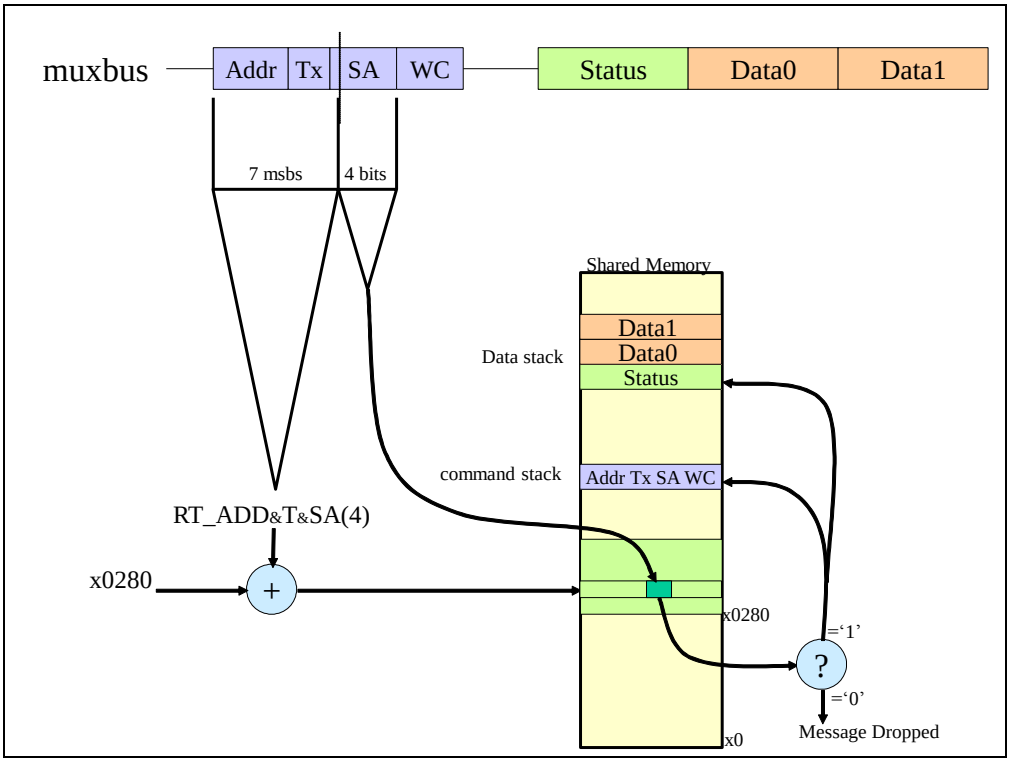

Figure 3: Message Monitor lookup table

When a command is received, the Terminal Address with Transmit receive bit plus the MSB of the sub-address serve as an offset from base address 0x0280 to read a 16 bit word. Each of the 16 bits marks a specific sub-address. If the bit is '1' than the command and it's following message elements will be monitored and stored in memory. If it is '0' than all following message element will be dropped.

2.3.7 SINGLE TRANSMIT MESSAGE MODE

Single message transmit mode reads the messages from the same memory location over and over again. The host CPU is expected to be able to write new data for transmits, before a new message transmitted on the 1553 bus.

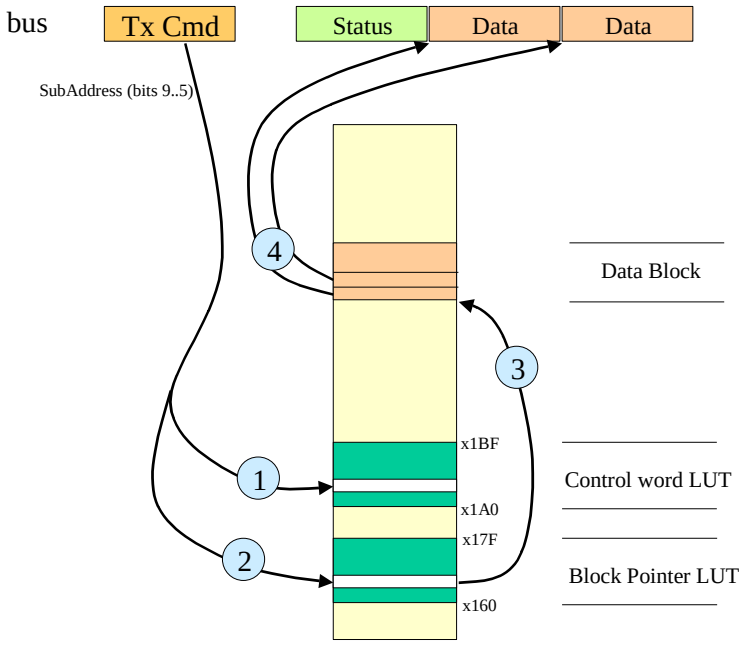

Figure 4: Single Message mode transmit command

The above diagram illustrates the actions taken by the Core when a transmit command is received and analyzed:

1. Initially the sub-address (SA) is striped from the command, and with this 0 to 31 (1 to 30 non mode) offset value, the lookup table of address 0x1A0+SA is accessed. If bits 10 to 12 of the fetched control word is 000 a single message transmit mode is selected.
2. With the same SA offset, a block pointer Lookup table is accessed, and the data pointer is fetched.
3. The data pointer points to the base address of the words to be sent.
4. When the data words are needed for transmission, the core fetches these words, from the base address, one by one, up to the full 32 words.

Comments:

1) The data pointer's 5 LSBs of the block pointer are ignored. That will assure blocks are spaced 32 words apart.
2) Sub-Address field is 5 bits of the command word. Sub-Address value 0 and 31 are mode commands. Please refer to the mode command section for description of operation.

2.3.8 SINGLE RECEIVE MESSAGE MODE

Single receive message mode stores the messages in the same memory location over and over again. The host CPU is expected to be able to analyze the data received before a new message is received.

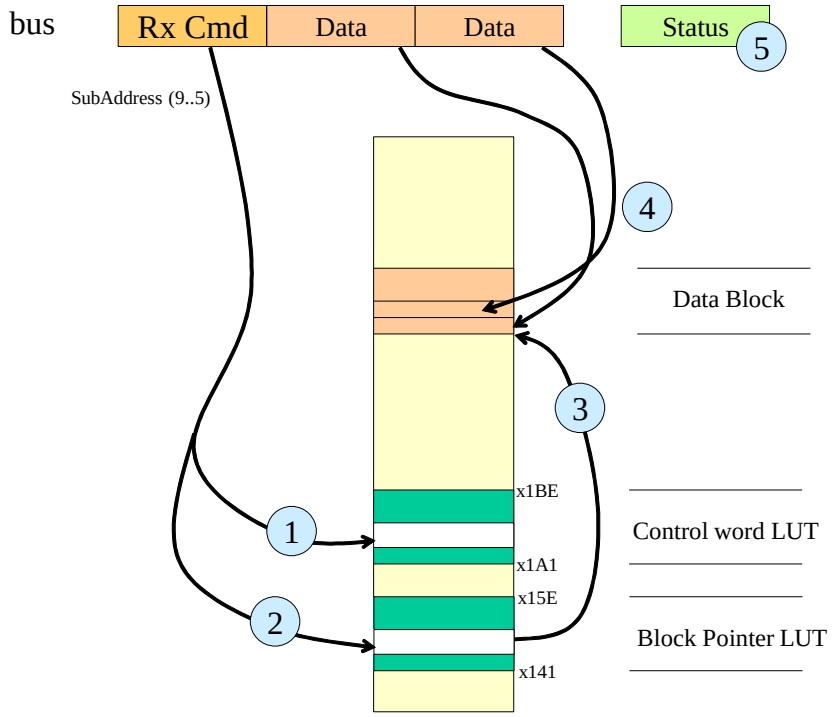

Figure 5: Single message mode receive command

The above diagram illustrates the procedures taken by the Core when a valid receive command is analyzed:

1. Initially the sub-address (SA) is striped from the command, and with its value 1 to 30 (non mode), a control word offset from base address 0x1A0 is accessed. If bits 5 to 7 of the fetched control word is 000 a single message transmit mode is selected.
2. With the same SA offset from base address 0x0140, a data pointer Lookup table is accessed, and the data pointer is fetched.
3. The data pointer points to the base address of the words to be sent. The 5 LSBs of the data pointer are ignored and generated for each message word by the core. That way the first word received is loaded to address xxxxxx00000, the second to xxxxxxx00001, then to xxxxxxx00010 and so on…
4. Each data word received error-free, is stored by the core, to the memory pointed by the incrementing data pointer.
5. Finally, after a valid message has been received, a status word is returned to the BC to confirm operation.

Comments:
Sub-Address field is 5 bits of the command word. Sub-Address value 0 and 31 are mode commands. Please refer to the mode command section for description of operation.

2.3.9 RECEIVE DOUBLE BUFFERING MESSAGE MODE

In this mode of operation two buffers are used for storing the data for a specific SA. The two buffers are located continuously on 64 addresses. There are 32 addresses used for the first buffer and 32 for the second. The buffer is aligned to 6 LSBs. Such that the first buffer would be saved in address x..x**0**yyyyy, and the second buffer would be saved at addresses x..x**1**yyyyy. The core will ignore the 5 LSBs of the data pointer and will assemble it by itself.

The Host CPU programs the block pointer for that SA into the block pointer lookup table at addresses 0x0160 to 0x017F with bit 15 (MSB) set high and the three memory management bits set to 0.

In addition Configuration Register #2 bit 12 must be '1', and Configuration Register #2 bit 1 allows for individual selection of double buffering mode.

The first valid message will be stored in the lower block, the second would be stored in the upper block, and the third would be stored in the lower block again, and so on…

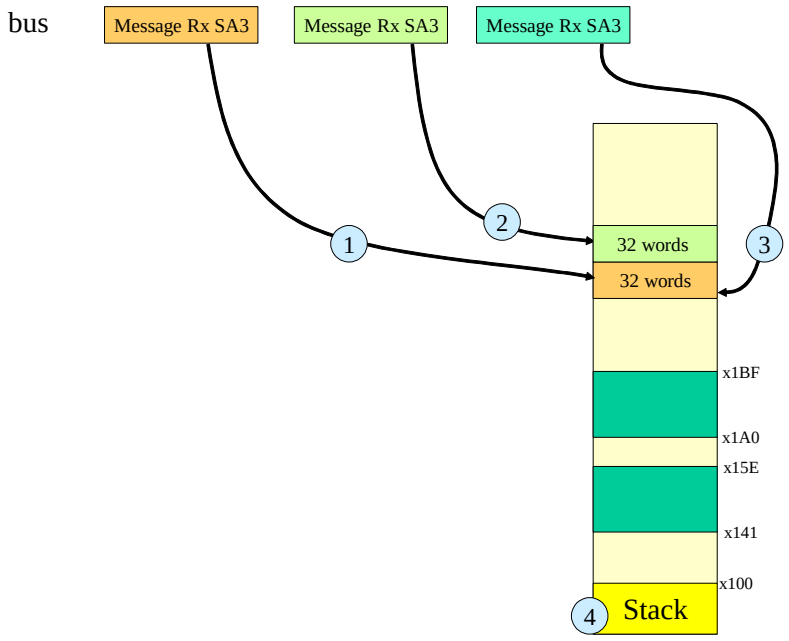

Figure 6: Receive Double Buffering Message Mode

Once the message has been completed successfully, the block pointer that was used to save the data is written into both the stack, and into the data block lookup table. This will assure that the Host CPU will know the location of the received data, and the core will know the proper location for future data.

Double buffer mode allows for a relaxed software response time. If the messages for a specific SA arrive every 20ms, then the buffer is written every 40ms (once every two consecutive messages).

1. The first message received for the specific SA is stored in the lower half of the dual buffer area designated by the Host CPU using the block pointer lookup table.
2. The second message received for that same SA is stored in the upper half of the dual buffer.

3. The third message received for the specific SA is stored in the lower half of the dual buffer area, and so on.
4. The 4-word stack entry will have its third word data pointer updated to the appropriate base address of one of the blocks.

2.3.10 CIRCULAR BUFFER MESSAGE MODE

In SA specific circular buffer mode, messages are either written or read from an array allocated by the Host CPU.

For Transmit commands the CPU stores an array of data (programmable to sizes of 128, 256 … 8196 words) in the shared memory, and the 1553 messages will read this buffer in slices, each slice defined by the number of words in the transmit command.

For Receive commands, the CPU allocates an array of data in the shared memory for storing the data received in message slices.

This mode allows high optimization of memory usage. In the single and double buffer modes, 32 word buffers are allocated for messages even if only 10 words are used in the ICD. The 22 tailing words in the buffer are not used. In this circular mode, data is stored or read with no gaps. The first data word of the current message is stored one address after the last word of the previous message. For example: If a message has 4 words then 128/4 = 32 messages can be stored in the 128 words array, before the first message is overwritten by the 33rd message.

The Stack Data Pointer for each message points to location of the first data word that was either received or transmitted depending on the command.

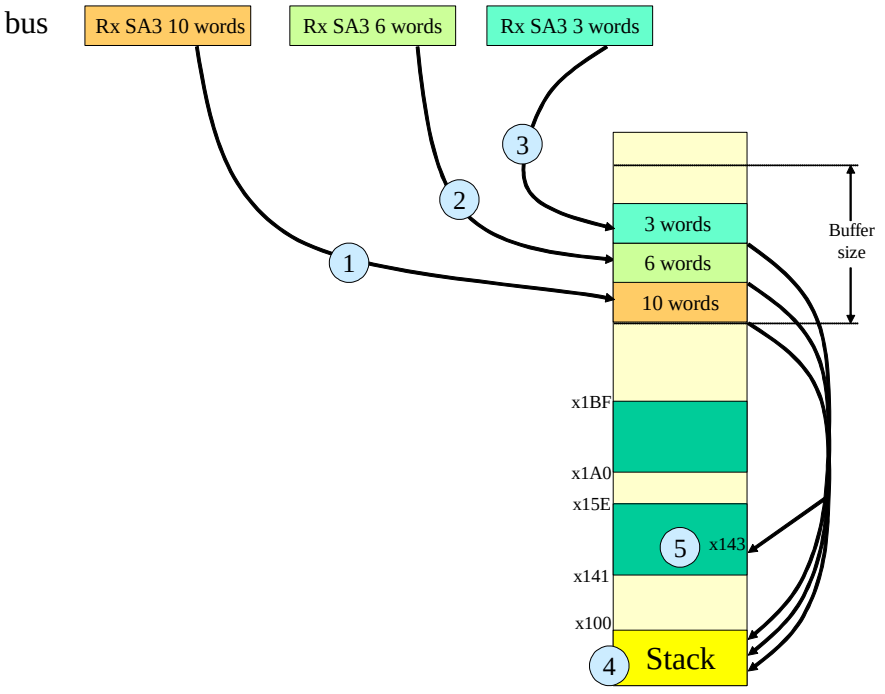

Figure 7: Circular Buffer Message Mode

1. A message with 10 words (word count = 10) is received and stored into the circular buffer from its base + 0 to base + 9.
2. A second message for the same SA with 6 words is received and stored into address base + 10 to base + 15.
3. A third message with 3 words to the same SA is stored into the buffer from base + 16 to base + 18.
4. For each message a stack entry of 4 words is generated. The data pointer of the stack entry is base + 0 for the first, base + 10 for the second, and base + 16 for the third message.
5. The updated base address is also saved to the SA data pointer lookup table by the core after each valid message. Invalid messages are thus overwritten.

2.3.11 GLOBAL CIRCULAR BUFFER MESSAGE MODE

Global Circular buffer Message mode is similar to the SA circular message mode, but it allocates a single circular buffer for all receive and BCST commands SA that are programmed as global circular mode.

The global circular data pointer cannot be stored in the SA lookup table because its common to a number of SA, thus it is stored in a fixed address location 0x0101.

Configuration register 6 bits 11, 10, and 9 define the circular buffer size. Bit 12 enables this mode of operation.

Note: If the SA control word is programmed to global circular mode, and bit 12 is low, single buffer mode is used for that SA.

PART II:
LEGACY BUS CONTROLLER MODE - BRM1553D-BC

3 BUS CONTROLLER INTRODUCTION

3.1 OVERVIEW

The Mil-Std-1553 Bus Controller (BC) allows a subsystem to control the 1553 bus. Control is achieved by sending instructions to RTs connected to the bus. Each command either instructs the RT to transmit words, receive words from the BC or perform some special actions, known as mode commands.

The Host programs the device with the relevant messages data alongside some configuration registers. Many such messages are loaded into the shared memory by the host.

The Host normally wants to attend the device only once every period of time. This period of time is referred to as a frame. During a frame, many messages are transferred between the BC and the RTs and between the RTs themselves (RT to RT commands). The Host needs to load the device with a complete set of messages for the entire frame.

When all commands of the frame have been loaded to the device's memory, the host initiates the frame. From this point the device will sequence the messages one after the other until all messages are done. No need for Host intervention is required during the frame run.

When the frame is finished, all data received is waiting in the device's memory for Host analysis.

Typically for real-time operation, frames would be initiated every 20 ms.

The device can be programmed to repeat its frames automatically, thus off loading the Host from sequencing the 1553 bus even further.

3.1.1 NON MIL-STD-1553B SETTINGS

The IP core is set to support different dialects than 1553B, by means of configuration.

These different settings affect both the BC and eBC modes. The RT and Message Monitor modes are also affected, please refer to the appropriate section.

The configuration register 3 bit 1, when set high, sets SA 31 to be non-mode, but rather a regular SA. In addition, this setting will assume only mode commands without data, so the Bc and eBC will not transmit nor expect data words for messages designated to it with SA=0 with WC(4)='1'.

If the RT responds to a message with ME bit in the status response, it is not illegal message as in 1553B, but rather an indication that one of the data words in the message had a 1553 error. The BC and eBC can be enabled to perform a retry in that condition using a high in bit 3 of the BC Control word and Configuration register #4 bit 10.

Using Configuration register 5 bit 7, it is possible to turn Broadcast address 31 to a standard RT as defined by 1553A.

3.2 BLOCK DIAGRAM

The host processor controlling the core accesses the memory and registers located in the host interface block. By setting the values of the configuration registers and setting up the transmission stack in memory, the host processor defines which commands would be transmitted to the 1553 bus.

The following drawing shows the BRM1553D-BC Block Diagram:

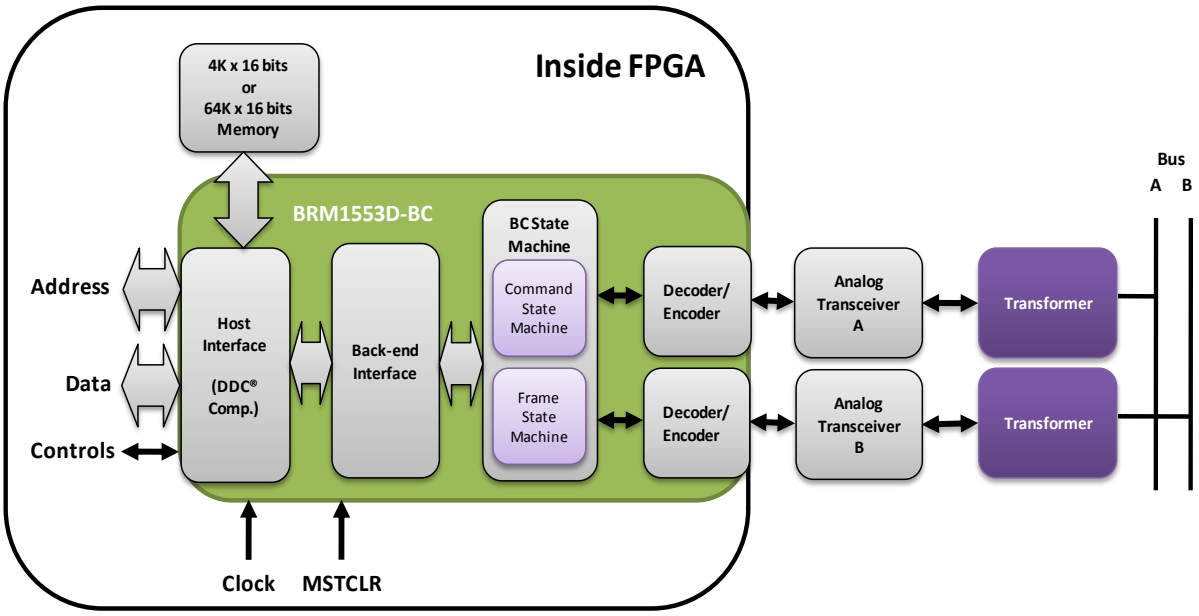

Figure 8: BRM1553D-BC Block Diagram

3.2.1 THE STATE MACHINES

When transmission is initiated the frame state machine manages the sequencing of a whole frame of messages as defined by the configuration registers and fixed memory locations in the memory.

The frame is composed of a set of individual messages being transmitted and managed over the 1553 bus one after the other. The host also points the beginning of the first message in memory and defines how many messages to transact through 2 fixed location memory words.

When all data has been loaded, and the state machines are idle, the host sends a START command. As a response, the frame state machine starts the frame transmission. The frame state machine fetches the messages from memory and forwards the requested message information to the command state machine which in turn sequences the command data and status words for a complete legal 1553 message.

When the message is complete the frame state machine accesses the next command, and so on until all messages have been completed.

The command state machine either transmits words through the encoders or receives RT responses through the decoders. The encoders and decoders interface between the core's 16 bits parallel internal buses and the MuxBus serial bus.

3.3 BUS CONTROLLER OPERATION MODES

3.3.1 BASE OPERATION

The host manages the 1553 bus through the shared memory and several registers. The host should load these registers and memory prior the START command, and then analyze the results in memory after the messages transactions have been finished.

The memory data defines which messages need to be transmitted and when. The host stores the messages table in the stack area of the memory.

The stack defines a set of 4-word message descriptors. Each message descriptor describes a message that needs to be transmitted and managed by the core.

The typical BC operation is organized in frames. Each frame includes several messages. Once the BC is triggered, the core's internal state machines start to manage each message, one after the other until finished.

When the BC core is triggered by the START command (R03B02) it looks for two memory locations in the shared memory.

Address M0100 is initially set by the host and points to the first stack entry from which the core starts to run. The 2 LSBs are ignored because each message occupies 4 memory words.

Using address M0101 the host defines the number of stack entries that the core is required to run until it stops the frame. NOTE that the host should load the 1's compliment of the number of messages to run. If, for example, there is just one message in frame, then load 0xFFFE.

3.3.2 MULTI FRAME SETUP

In most avionics systems, not all messages are transmitted every frame. There are messages that need to be transmitted once every 20ms, others every 40ms, 80ms…

The above mechanism supports this type of management. The host should load the stack with the messages for the first frame, then the messages for the second frame, and so on until all frame types are described in stack. Then, before the start of each frame, the host should write the stack new start point to address M0100, and define its length in M0101, and then order the START command (R03B01).

3.3.3 AUTO-REPEAT MODE

In order to off load the host from the burden of initializing each frame with the START command, the core supports the auto-repeat mode. By defining the frame length (Register 0D) and enabling a configuration register, the core would restart the same frame it finished all over again automatically. In order to work in auto repeat mode, the host should set the appropriate configuration register and load M0102 with the repeated start point, and M0103 with the repeated length. The core will read these values if auto repeat is on (R01B08) and copy to M0100 and M0101 and update them continuously during the frame. When the frame is finished, the core will wait for the frame counter zero (if zero occurred BEFORE the frame has been completed, the core will restart the frame with minimum gap time after the last message).

3.3.4 MULTI-FRAME & AUTO-REPEAT

If multi frame mode is required and also auto repeat is required then the above multi frame setup requires different settings before each frame, which cannot be accomplished by the host in auto repeat mode.

For this challenge, a programmable message-to-message gap can be enabled. Each stack entry's third word defines the number of microseconds from the start of this message until the start of the next message. Up to 65.5 ms can be programmed with the third word.

The host should load all of the messages of the first frame, consecutively load all messages of the second frame, and so on. The groups of messages that define a frame in stack are called **minor frames**. All of the minor frames together, from the first message of the first frame until the last message of the last frame, are called the **major frame**.

If each message in the minor frame has a pre defined gap time, the total minor frame transmission period can be summed, and the gap time of the last message in the minor frame should be programmed to compensate for the gap to the next frame.

For example, if frames are separated by 20ms, and a frame is composed of 5 messages, each programmed to 1 ms gap, then the total message activity will take 5 ms. In this case the gap time to the fifth message should be set to 15 ms.

Calculating the gap time for each message requires 1553 know-how as to how each message is spread over time. Rule of the thumb would be = (number of data words + 3) * 20 us. For RT 2 RT format = (number of data words + 6) * 20 us (where 3 and 6 are the number of extra control words per message).

The frame counter should be loaded with major frame time. The major frame time is sum of all minor frames.

3.3.5 RETRY

Sometimes in life, things don't work, they just don't. There are sometimes failures. This also happens in 1553 serial communication during operation. In this case the core allows for retries. Retry can be enabled globally through the configuration registers (R01B04) and if enabled, each message can be set to retry on failure using the message control word.

Retry can also be set to perform on the same bus or the opposite bus from the one that failed.

If the optional first retry fails, an optional second retry can be initiated using the message control word.

If the BC is set to work in 1553A dialect, so it is anticipated that 1553 RTs would reply with ME bit set if they find 1553 errors in the data words, it is possible to retry if that ME is set.

3.3.6 STOP ON ERROR

When things do go wrong and messages fail, the core can be programmed to stop.

The core state machine can be stopped at the end of a failed message, or at the end of the frame that contained a failed message.

A failed message is a message that even optional retries didn't solve the error. If retry passes, the message is not considered a failure.

3.3.7 STOP ON STATUS BITS

The core can also be programmed to stop its frame or stop frames in auto repeat mode if bits of the RT's status word are not as expected. Each of these bits can be masked, so even if the bit is set, it is not considered as a reason for stopping the frame.

The broadcast bit has a special treatment. It could either be masked OR it could be compared to either '1' or '0'.

4 BC PROGRAMMING AND SETUP

4.1 REGISTERS

There are several registers mapped to the register section (when MEM_REG signal is kept low) during a read or write operation.

The following section describes the registers and the operational bits of these registers.

Note:
If a bit is defined as read only, then the core acts according to the specified level, and the read operation is available for software compatibility with DDC® reasons. However, these bits may be written with values which will be ignored by the core.
If the bit is Write/Read, then the default value is as defined but could be changed to desired level for the required operation.

Note for the way this section is written:
If a reference is made to a particular **register** and a specific bit the notation is R01B12 for Register 01 bit 12.
If a reference is made to a particular **memory** location the notation is M0100 for memory address 0x100.

4.2 REGISTERS MAP

The software interface of the core to the host processor consists of 20 internal operational registers for normal operation. These registers determine the device configuration, modes of operation, memory structure, interrupt control and status, etc.

The address mapping for the registers is detailed in the following table:

Address Lines					Register Description (Read/Write)
A4	A3	A2	A1	A0	
0	0	0	0	0	Interrupt Mask Register #1 (RD/WR)
0	0	0	0	1	Configuration Register #1 (RD/WR)
0	0	0	1	0	Configuration Register #2 (RD/WR)
0	0	0	1	1	Start/Reset Register (WR)
0	0	0	1	1	Stack Pointer Register (RD)
0	0	1	0	0	RT Sub address Control Word Register (RD) – Not used in BC mode
0	0	1	0	1	Time Tag Register (RD/WR)
0	0	1	1	0	Interrupt Status Register #1(RD)
0	0	1	1	1	Configuration Register #3 (RD/WR)
0	1	0	0	0	Configuration Register #4 (RD/WR)
0	1	0	0	1	Configuration Register #5 (RD/WR) – Not used in BC mode
0	1	0	1	0	(RD) – all '0'
0	1	0	1	1	BC Frame Timing Remaining (RD)
0	1	1	0	0	BC Message Timing Remaining (RD)
0	1	1	0	1	BC Frame Time Register (WR)
0	1	1	1	0	RT Status Word Register (RD) – Not used in BC mode
0	1	1	1	1	RT BIT Word Register (RD)
1	0	0	0	0	Not implemented
1	0	0	0	1	Test register. Supports external loopback test in Word Monitor mode.
1	0	0	1	0	Bus A coupling to bus information.
1	0	0	1	1	Bus B coupling to bus information.
1	0	1	0	0	Not implemented
1	0	1	0	1	Latched Time Tag Register.
1	0	1	1	0	Not implemented
1	0	1	1	1	Not implemented
1	1	0	0	0	Configuration Register #6 (RD/WR) – Not Used
1	1	0	0	1	Configuration Register #7 – not used.
1	1	0	1	0	version of Core (RD)
1	1	0	1	1	eBC Conditions register (RD) – eBC mode only.
1	1	0	1	1	eBC General Purpose Flag setting (WR) – eBC mode only.
1	1	1	0	0	BIT Test Status Register (RD)
1	1	1	0	1	Interrupt Mask Register #2 (RD/WR)
1	1	1	1	0	Interrupt Status Register #2 (RD)
1	1	1	1	1	eBC General Purpose Queue pointer (WR/RD) – eBC mode only.

Note:
Not all of the bits of the DDC® MiniACE® registers are implemented. Please see "1553 Core compatibility" document for details on all differences between the 1553 Core and DDC® MiniACE® and Enhanced MiniACE.

4.3 REGISTER'S SHORT FORM LOOK UP TABLE

Reg / Bit	Int Reg 0x0	Configuration 1 0x1	Configuration 2 0x2	Start Reset 0x3	Time 0x5	INT Status 0x6	Conf 3 0x7	Configuration 4 0x8	Frame 0xD
15		0 for BC	1 (enhanced Int)			OR of below	'1'		
14							Stack Size		
13	Tx Timeout					Tx Timeout			
12	Stack Rollover	Error Stop EOM				Stack Rollover		'1'	
11		Error Stop EOF						Mask BCST	
10		Sts Stop EOM	No 256 Boundaries						
9	Mem Protect	Sts Stop EOF	Time Tag Resolution			Mem Protect		Retry STS	
8	BC Retry	Auto Repeat				BC Retry		1st Retry Opposite	Frame Length LSB is 100 us
7	RT Addr Parity	Ext Trigger			Time Tag Value	RT Addr Parity		2nd Retry Opposite	
6	Time Rollover	Int Trigger		Stop EOM		Time Rollover		ME valid	
5		Gap Enable	Sync Internal	Stop EOF				Busy Valid	
4	Filtered EOM	Retry Enable	Clear INTv on Read			Filtered EOM			
3	EOF	Retry Twice	1 – Level INT	Reset Time		EOF			
2	1553 Error	BC is IDLE		Reset INTv		1553 Error			
1	RT statused	Frame Active		Start BC		RT statused			
0	EOM	Message Active		SW reset		EOM			
Init value	0x0000	0x0000	0x0000	Reads Stack point		0x0000	0x0000	0x0000	

4.3.1 INTERRUPT ENABLE REGISTER #1 ADDRESS 0X0

If the Host enables one of the bits below by setting its value to '1', and the specified event occurs, an interrupt will be generated to the Host CPU.

Bit number	Read/ Write/ Default	What event triggers the interrupt when enabled by '1'.
13	R/W/0	Transmitter timeout occurred
12	R/W/0	Stack Rollover occurred
9	R/W/0	Memory protection fault **
8	R/W/0	BC Retry – following incorrect RT response has been initiated.
7	R/W/0	RT Address parity error found
6	R/W/0	Time tag counter rollover
4	R/W/0	BC End Of Message.
3	R/W/0	BC End Of Frame.
2	R/W/0	1553 message error occurred.
1	R/W/0	Wrong RT status or unexpected status bits set in status respond.
0	R/W/0	End Of Message.

*All other bits are not used and will be read as zero. Default Value: 0x0000 – all events masked.

** This bit is used in DDC® MiniACE® as memory handshake error. Since this cannot occur with the core, bit #9 is used for data protection. When Data pointer points to a non data block area (M0000 to M0103) due to wrong memory setup by the host, the core detects that, and asserts bit 9 of the interrupt status register interrupt to '1'. If this bit is not masked than an interrupt will occur.

Memory overrun is not prevented in BC mode, only an interrupt is set.

4.3.2 CONFIGURATION REGISTER #1 ADDRESS 0X1

This configuration register works in BC function if bit 15 is '0'.

Bit number	Read/ Write/ Default	Description
15	R/W/1	'0' For legacy BC. All default values below are in BC mode. '1' For Enhanced RT (see RT/MT documentation) following values assume this bit is set to '0'
14	Assumed '0'	'0' is preset for this bit.
13	R/W/0	'0' area A, '1' area B.
12	R/W/0	'1' – Stop at end of Message error. If optional retry succeeded => messages continue!
11	R/W/0	'1' – Stop at end of frame if error. If optional retry succeeded => frames continue!
10	R/W/0	'1' – Stop messages if unexpected, non-masked status bits are set. If optional retry succeeded => messages continue!
9	R/W/0	'1' – Don't process additional frames if message unexpected, non-masked status bits are set and auto frame enabled. If optional retry succeeded => frames continue!
8	R/W/0	'1' – Auto Frame Repeat. Will start next frame if none of the problems in bits 9-12 caused a halt. Refer to setting up auto repeat presets. Trigger first frame with START (R03B01). '0' – Host has to trigger every frame with START.
7	R/W/0	'1' – Enables external trigger mode – the trigger is sensitive to rising edge. '0' – External trigger disable.
6	R/W/0	'1' – Internal frame counter (64K steps of 100 us) re-trigger the frame in case auto frame repeat is enabled (bit 8). '0' – Internal trigger disable.
5	R/W/0	'1' – Message gap timer enable. If enable will start next message after gap*1us. Gap is defined in 3rd word in BC message block descriptor in stack. '0' – Message gap set to default (~10 us).
4	R/W/0	'1' – Global Retry enable. If enabled, each message can be enabled for retry with its BC control word bit 8.
3	R/W/0	'1' – Retry twice after 2 failures. '0' – Retry only once after failure.
2	Read	'1' – BC state machine has been started, and has not finished its messages transactions. In Auto repeat mode this signal will stay high until halted by host or error. '0' – BC is idle (after frame finished or "stop on error").
1	Read	'1' – bus active – from start of first message in frame till end of last message in frame.
0	Read	'1' - Message active – from start till end of message.

4.3.3 CONFIGURATION REGISTER #2 ADDRESS 0X2

Bit number	Read/ Write/ Default	Description
15	Assumed '1'	'1' – Enhanced Interrupts mode enabled. Interrupt status register will set its bit flags high if condition occur even if it is masked by the interrupt enable register. An interrupt will be generated if a bit is set and also enabled.
14	R/W/0	'0' – No Memory Parity bit.
13	R/W/0	RT mode.
12	R/W/0	RT mode.
11	R/W/0	RT mode.
10	Assumed '1'	256-word boundaries disable. '0' for RT mode. '1' in BC mode.
9..7	R/W/'000'	Time Tag resolution: "000" – 64 usec, "001" – 32 usec, "010" – 16 usec, "011" – 8 usec, "100" – 4 usec, "101" – 2 usec, "110" – Increment by writing to "Start Reset" register bit 4. "111" – External Tag Clock increments Time Tag (< 10Mhz).
6	R/W/0	RT mode.
5	R/W/0	'1' – in BC mode – transmits the internal time tag word in "synchronize with data" mode command.
4	R/W/0	'1' – Auto clear of interrupt status registers #1 and #2 as a result of CPU reading its value.
3	R/W/0	'0' – generates a 500ns low pulse on the INTn signal. '1' – Level mode. INTn stays low until the host reads interrupt status register.
2	R/W/0	RT mode.
1	R/W/0	RT mode.
0	R/W/0	RT mode.

4.3.4 START RESET REGISTER WRITING TO ADDRESS 0X3

This is a write only register. Reading back will return Stack Pointer Register (same as reading address 0x0100).

Bit number	Read/ Write/ Default	Description
10	Write	'1' clear self test register at address 0x1C.
9	Write	'1' initiate RAM self test.
7	Write	'1' initiates Protocol self test.
6	Write	'1' stops BC operation at end of message.
5	Write	'1' Stops BC operation at end of frame.
4	Write	'1' Increments Time Tag by one when Time Tag resolution is set to "110".
3	Write	'1' resets the Time Tag counter.
2	Write	'1' resets the interrupt. Resets the FF that latches the interrupt condition. If the condition for interrupt persists, the relevant FF would be set again until the condition for causing this interrupt is cleared. Reset of Interrupt Register #1.
1	Write	'1' BC START command. Starts the BC state machine.
0	Write	'1' Reset core. Reset all registers, FFs in core. Memories are not reset, and should be reset by CPU. If '1', all other bits have no effect.

4.3.5 STACK POINTER REGISTER READING FROM ADDRESS 0X3

This is a read only register. Writing to it is explained above.

Bit number	Read/ Default	Description
15..0	Read 0x0	Copy of the Stack pointer value. After the EOM bit (15) is written to the first word of the stack entry, the stack pointer is incremented by 4. This pointer value can either be read from the memory in address M0100, or from this register.

4.3.6 CURRENT BC CONTROL WORD. ADDRESS 0X4

The last used BC message control word may be read through this register.

4.3.7 TIME TAG REGISTER – ADDRESS 0X5

The time tag counter can be reset by writing to this register value 0x0 or loaded to any other value by writing it here.

The time tag counter counts in clock steps of 2us, 4us, 8us, 16us, 32us or 64us or by external clock, or by writing to a bit 4 of the Start Reset Register R03B04.

The counter will rollover every 64us * 2^16.

The value of this counter is also reset by the hardware raw_reset line, or the software reset done by writing '1' to R03B00 or R03B03 register.

The value of the time tag counter is placed in the stack of each transmitted command. Please see "Stack contents" section for details.

The BC Host may write to this register any time.

When performing synchronize with data command, the time tag value can be sent instead of the data word in the stack memory. The BC control word bit 15 controls the source of this data word.

Bit number	Read/ Write/ Default	Description
15..0	Read/Write 0x0	Read the Time Tag Counter. Write a new value to the time tag counter.

4.3.8 INTERRUPT STATUS REGISTER #1 ADDRESS 0X6

This register indicates the cause of an interrupt.

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (if configuration register #2 bit 4 is '1'.).
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit number	Description
15	This bit is an OR of all interrupts bits that are not masked by interrupt mask register. It reflects the HW interrupt line in level mode, and indicates there is still an interrupt request in the pulse mode.
14	'1' – Wiring Fault was detected using SmartWiring, when enabled
13	'1' – Transmitter fail safe timeout.
12	'1' – Stack Rollover has occurred
9	'1' – Data Pointer pointed to a protected area where data blocks should not be present. Data was not written and memory was not corrupted. (Handshake Error in DDC® devices)
8	'1' - BC retry has occurred. Each message has a bit in status word in stack that indicates if retry was done.
7	'1' - RT Terminal Address Parity Error. RT mode.
6	'1' - Time Tag Rollover from 0xFFFF to 0x0000.
4	'1' – Message was sent. This interrupt can be enabled or not for each message with BC control word bit 4.
3	'1' – BC has reached end of frame.
2	'1' – RT has not replied OR replied with error OR loop-back error of transmitted words.
1	'1' – Wrong RT address response or one of the (non BCST) status word bits were set high. BCST bit sets this bit high if configurations register 4 bit number 11 is set low (compare mode) AND the BCST bit is different from the BC control word of the message bit #5.
0	'1' – End of message has occurred.

*All other bits are not used and will be read as zero.

Default Value: 0x0000

4.3.9 CONFIGURATION REGISTER #3 ADDRESS 0X7

Bit #	Read/Write/ Default	Description
15	Assumed '1'	'1' always in Enhanced mode.
14..13	Write/Read/'00'	BC stack size selected as follows: (bits 14,13) "00"- 256 words / 64 messages "01"- 512 words / 128 messages "10"- 1024 words / 256 messages "11"- 2048 words / 512 messages See note 1 below.

Note 1: The core will update the 8 LSBs of the stack pointer in "00" mode, 9 LSBs in "01" mode, 10 LSBs in "10" mode and 11 LSBs in "11" mode. The host defines the Stack pointer, and the core will affect only the relevant lower bits.

4.3.10 CONFIGURATION REGISTER #4 ADDRESS 0X8

Bit number	Read/ Write/ default	Description
12	Assumed '1'	Expanded BC control word always on. All 15 bits of BC control word are functional.
11	R/W/0	Mask / Compare the BCST bit in returned status bit. ** '1' – Mask. The relevant bit in the BC control word operates as mask for the BCST bit of the received status. '0' – Compare. The relevant bit in the BC control word is compared with the BCST bit of the received status.
10	R/W/0	'1' – Retry if 1553A dialect and RT status' word ME bit is set. 1553A dialect is set by setting BC control word bit 3.
9	R/W/0	'0' – No retry if status bits that are not masked are set. '1' – Retry a message if retry enabled and one of the unmasked status bits are set high in the returned RT status word. BCST bit pass/fail has a special treatment as seen above in bit 11 setup.
8	R/W/0	'0' – First retry on same bus if message failed. '1' – First retry on opposite bus if message failed.
7	R/W/0	'0' – Second retry on same bus as original failed message. '1' – Second retry on opposite bus of original message.
6	R/W/0	'1' - An RT can respond to a transmit command with Message Error (ME) bit set followed by the data words OR without any data words (as in illegal command) and no format error is reported for either response. Both are valid responses. '0' – Format error is posted if no data words follow a ME status.
5	R/W/0	'1' - An RT can respond to a transmit command with Busy bit set followed by the data words OR without any data words (as in busy RT state) and no format error is reported for either response. Both are valid responses. '0' – Format error is posted if no data words follow a busy status.
2..0	R/W/0	Write "010" to enable the loopback test. Refer to chapter 6 for detailed description. Write 0 to disable the test.

*All other bits are not used and will be read as zero.

** Note that the BCST bit in the status return is only set by the RT in the proceeding message's status word after a broadcast message FOR a transmit status or transmit command mode commands. Otherwise the BCST bit should be '0'.

4.3.11 CONFIGURATION REGISTER #5 ADDRESS 0X9

Bit number	Read/ Write/ default	Description
10..9	R/W/"00"	'00' – response timeout is set to approximately 18.5 us. '01' – response timeout is set to approximately 22.5 us. '10' – response timeout is set to approximately 53.5 us. '11' – response timeout is set to approximately 130 us.
7	R/W/'0'	'0' – Address 31 is used as broadcast. '1' – Address 31 is used as normal RT address. Required for 1553A that does not support Broadcast. For 1553B it is hazardous to set this bit high.

4.3.12 BC FRAME TIMING REMAINING ADDRESS 0XB

16 bits that define the number of 100us left until end of frame, or zero.

This is a count-down register that is read only.

4.3.13 BC MESSAGE TIMING REMAINING ADDRESS 0XC

16 bits that define the number of 1us left until the start of the next message, or zero.

The value is relevant only when this mode is enabled with the configuration registers.

This is a count-down register that is read only.

This register starts from the value of the 3rd word in the BC command descriptor in Stack.

4.3.14 BC FRAME TIME REGISTER ADDRESS 0XD

These 16 bits define the number of 100 us that is the frame length.

The frame length can be set from 100 us (0x0001) to 6.55 seconds (0xFFFF).

The running value can be read from register address 0xB.

4.3.15 BC ENCODER TEST REGISTER 0X11

It is possible to perform an encoder test between bus A and B using this register only if IP supports the BC and word monitor. Please refer to Chapter 6 for details.

Bit number	Read/ Write/ default	Description
15	R/W/0	'1' would transmit on Bus A, '0' on Bus B
11	R/W/0	Writing '1' would trigger transmission
10	R/W/0	'1' - save the echo from the transmitting bus to monitor. '0' – do not save echo, only word received from opposite bus.

4.3.16 PASSIVE TDR LEVEL 0X12

pTDR level is indicated in bits 3..0.

Bits 15..4 should read 0x321.

Basic pTDR capabilities would read 0x3211. Advanced, fault location data vector should read 0x3218. No pTDR should read 0x3210.

4.3.17 CONSTANT VALUE 0X13

Read constant 0x7654. This constant can be used to verify proper Endian setup of system.

4.3.18 LATCHED TIME TAG REGISTER 0X15

The value of the time tag register address 0x5, is latched into this register each time a rising edge is detected on the Strobe_TT input line.

This allows the SW to read a very accurate Time Tag value when the hardware event occurred, such as when working with GPS systems.

4.3.19 CONFIGURATION REGISTER #6 ADDRESS 0X18

Bit number	Read/ Write/ default	Description
15	R/W/0	'1' – Enhanced Bus Controller (eBC). Please refer to the eBC Mode section. '0' – legacy bus controller, relevant for this manual.

4.3.20 CONFIGURATION REGISTER #7 ADDRESS 0X19

Bit number	Read/ Write/ default	Description
9	R/W/0	Version of IP core. Used by DDC API for version testing. If '1' is written to this bit, it is read in bit 5 and 9.
5	R/0	Version of IP core feedback. If '1' is written to bit 9, it is read here.

This register is used to by the software determine the DDC chip to which the core is compatible to. A MiniACE chip will not return any feedback, while some versions of MicroACE will return feedback on bit 5.

Sital's IP core is compatible to the latest DDC chip, and therefore bit 5 returns '1' if written to bit 9.

4.3.21 VERSION OF CORE REGISTER ADDRESS 0X1A

This register returns the revision of core.

4.3.22 BC CONDITION CODE READING FROM REGISTER ADDRESS 0X1B

This register is used in the Enhanced BC (eBC) mode. Please refer to the eBC Section for details.

The reset value should be 0x8000.

4.3.23 SELF TEST STATUS REGISTER 0X1C

This register indicates the state of the internal protocol and RAM tests.

Writing to reset register at address 0x3 clears the content of this register.

Bit number	Description
15	Protocol Built-in-test complete
14	Protocol Built-in-test in progress, should be less than 2 ms.
13	Protocol Built-in-test passed
11	Protocol Built-in-test in progress or complete
7	RAM Built-in-test complete
6	RAM Built-in-test in progress, should be less than 2 ms.
5	RAM Built-in-test passed

After hardware power up, the core initiates the protocol test automatically. Allow 2 ms for this test to complete.

Note: Please refer to the introduction section of this document for a description of the actual test coverage.

4.3.24 INTERRUPT MASK REGISTER #2 0X1D

If the Host enables one of the bits below to '1' and the specified event occurs, an interrupt will be generated to the Host CPU. The status bit will be set regardless of the mask bit.

Bit number	Read/ Write/ default	Description
1	R/W/0	BIT test complete

4.3.25 INTERRUPT STATUS REGISTER #2 0X1E

This register indicates the cause of an interrupt.

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (if clear by read enabled in configuration register #2).
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit number	Description
15	This bit is high when an interrupt request is pending.
1	BIT test complete.
0	'1' indicates that interrupt register #1 is requesting an interrupt. It is recommended to read Interrupt status register #2 if its bits are required and if bit 0 is '1' then read Interrupt status register #1.

4.4 MEMORY

4.4.1 MEMORY MAPPING

The host should prepare the frames of messages in the memory prior to initiating the START command.

The following table suggests a memory mapping for the BC core.

Note that BC mode uses the same fixed addresses as the RT and MT, thus BC cannot operate in the same time RT/MT is used.

Hex Address	Description	Comments
0000 to 00FF	Stack (Note 1)	Initialized by host, Updated by core.
0100 (fixed)	Stack Pointer	Initialized by host, Updated by core.
0101 (fixed)	Frame length (1's complement)	Initialized by host, Updated by core.
0102 (fixed)	Initial Stack Pointer	Initialized by host. For Auto-repeat.
0103 (fixed)	Initial Frame length (1's complement)	Initialized by host. For Auto-repeat.
0104 (fixed)	Stack Pointer – Area B	Initialized by host, Updated by core.
0105 (fixed)	Frame length (1's complement) – Area B	Initialized by host, Updated by core.
0106 (fixed)	Initial Stack Pointer – Area B	Initialized by host. For Auto-repeat.
0107 (fixed)	Initial Frame length (1's complement) – Area B	Initialized by host. For Auto-repeat.
0108 to 012D	Message Block 0	
012E to 0153	Message Block 1	
:		
0ED6 to 0EFB	Message Block 93	

Shown for 4Kx16 memory array.

More memory can be loaded with additional Message Blocks for larger memory devices.

Note 1 – The stack can be placed in other places in the memory by setting the stack pointer address (100 or 104) to that base address. For example if 0x100 is set to 0x2000 and the stack depth is set to 1024 words (configuration register 3), then the stack would be from 0x2000 to 0x2400-1.

If auto-repeat mode is ***not*** used please set:

1. The **Stack pointer** (M0100) points to the currently stack entry. The host sets the initial value prior to START command. After START command the core updates it constantly. The host should NOT change it after START but it could be read.

2. The **Frame Length** (M0101) is the number of stack entries (messages) that the host wants to transact. ***The one's compliment*** of the desired frame length should be loaded to this memory location. If 5 messages are required, load the value 0xFFFA to M0101.

If auto-repeat mode is used please set:

1. The **Initial Stack pointer** (M0102) points to the first stack entry. The host sets the initial value prior to START command. After START command the core updates the stack pointer (M0100) constantly. The host should NOT change it after START but it could be read.

2. The **Initial Frame Length** (M0103) is the number of stack entries (messages) that the host wants to transact. The one's compliment of the desired frame length should be loaded. The core updates frame length M0101, the host can read M0101 to see how many more messages are left.

4.4.2 STACK CONTENTS

The BC stack contains up to 64 groups of 4-word command descriptors. Each command descriptor is composed of the following words:

	Name	Bit	Description
0	Block Status Word. Updated by Core*	15	End Of Message – Set to '1' by the core when the message has been complete.
		14	Start Of Message - Set to '1' by the core when the message has been started. In most cases, this bit is '1' after end of message if there is a 1553 bus-coupling problem.
		13	'0' – This message has been processed on Bus A. '1' – This message has been processed on Bus B.
		12	'1' – Error was found in the message. Bits 10, 9, 8, 3, 2, 1, 0 indicate cause of error. If all are zero, the error might be one of: 1. The BC is trying to transmit on a busy line (another BC is active). 2. The first message in RT2RT is a transmit command 3. The second RT2RT command is a receive command.
		11	Status Set. One of the status bits (excluding BCST bit) of the status return was '1'. BCST bit works in either mask mode or compare mode. In mask mode it works like other mask bits on the BCST bit. In compare mode, Status set occurs if BCST bit is different from bit 5 of BC control word.

		10	Format Error. The returned echo from the RT contained 1553 violations. See bits 3, 2, 1, 0 for a more accurate guess of the source of the problem.
		9	Response timeout. The RT responded too late or didn't respond at all.
		8	Loop back failed. The nature of 1553 bus is that every word transmitted, is also echoed back. The core verifies that the echo is correct and equal to the transmitted word. If not, this bit is set to '1'. Tip: The source of this type of error could be transceiver fault, or bus coupling problem.
		7	Unmasked Status bit set. This bit will be set to '1' if one of the status bits are set high and its appropriate mask bit in the BC control word is unmasked ('0'). BCST bit influences only in mask mode. See registers section for description of BCST bit.
		6	'1' indicates that two retries were performed.
		5	If bit 6 is '0' then a '1' indicates one retry was performed.
		4	Good data block received by BC. '1' – after an RT-BC, RT2RT, and Transmit Mode code with data commands if the message ended OK. '0' – after other message types, or if the above type of message was invalid. Loop back test failure does not cripple this bit result.
		3	'1' – The RT responded with wrong RT address.
		2	'1' – The RT transmitted a wrong number of words.
		1	'1' – Incorrect sync type response by RT.
		0	'1' – Invalid word. Indicates that the RT responded with a word containing 1553 errors.
1	Time Tag Word	15..0	16 bit real time counter. Written by core when the message started.
2	Gap to next message	15..0	Number of microseconds until start of next command. If gap shorter than message (0 for example), gap to next message would be the default as if this feature is off.
3	Message Pointer *	15..0	Points to the starting memory location of a message to be transmitted – the Message Block. The first word of the Message Block is the BC control word. The host sets this value. Core does not change it.

* The core has a built-in memory protection mechanism. If the Message Pointer (4th word) pointes to an address lower than 0x0108 than it is a wrong address. In that case, interrupt status register #1 bit 9 will be set.

The following diagram describes the usage of the inter-message gap feature provided by the 3rd word in the stack entry.

If the gaps are shorter than the message length (0 for example), gap to next message would be the default (~10 us) as if this feature is off.

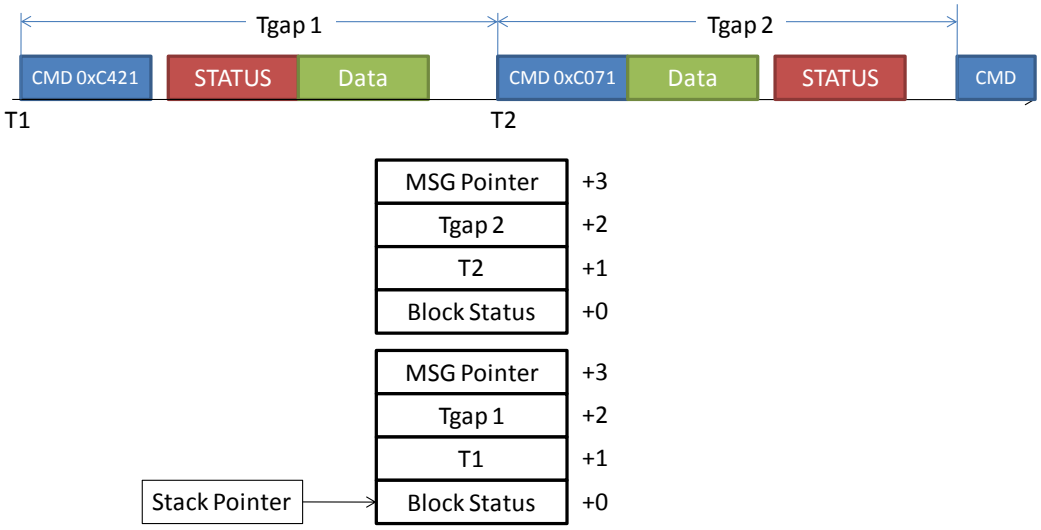

Figure 9: 3rd Word Gap Time

4.4.3 THE MESSAGE BLOCK

The Message Pointer points to the message block. The host should load the message block with the relevant words for the message.

The first word is not transmitted. The first word is the BC control word. This control word describes various modes of operation for the specific message as detailed ahead.

The structure of the message block varies according to the settings in the BC control word and the message type and length. The following table defines the message block contents types. Each entry is a 16bit word.

CONT – control word (not transmitted).
CMD – Command
STS – Status word
DAT – Data word
BCST - Broadcast
{DAT} – replication 1 to 32 Data words depending on word count.
XXX**LB** – Echo loop back of last transmitted word XXX

Color-coding:

Blue – loaded by Host before START command.
Green Italic – written by core after message transferred.

Type	Data Stack Contents
BC => RT	CONT CMD { DAT } *DATLB STS*
RT => BC	CONT CMD *CMDLB STS { DAT }*
RT => RT	CONT CMDrx CMDtx *CMDtxLB STStx { DAT } STSrx*
BC => Mode w/o data	CONT CMD *CMDLB STS*
BC => Mode transmit data	CONT CMD *CMDLB STS DAT*
BC => Mode receive data	CONT CMD DAT *DATLB STS*
BC => BCST	CONT CMD { DAT } *DATLB*
RT => BCST	CONT CMDrx CMDtx *CMDtxLB STStx { DAT }*
BC => BCST Mode w/o data	CONT CMD *CMDLB*
BC => BCST Mode data	CONT CMD DAT *DATLB*

The maximum length is 38 words. RT2RT command is composed of => 32 data + 2 commands + 2nd command loop-back + 2 status' + control word. For this reason the data area of the memory is divided into 38 word slices, but this is optional to be defined by the user.

4.4.4 THE BC CONTROL WORD

The first word in a message block is the BC control word. This word is not transmitted to the 1553 bus. This control word describes various modes of operation for the specific message as detailed.

Bit	Description
15	Transmit internal time tag in Synchronize with data mode command: '1' - Use time tag counter as data. '0' - Use data word in message block.
14	'1' – Mask Message Error (ME) bit in status word. '0' – No masking. If ME is set, this will cause "Status Bit Set" condition.
13	'1' – Mask Service Request (SR) bit in status word. '0' – No masking. If SR is set, this will cause "Status Bit Set" condition.
12	'1' – Mask busy bit in status word. '0' – No masking. If busy is set, this will cause "Status Bit Set" condition.
11	'1' – Mask Sub-System Flag (SSF) bit in status word. '0' – No masking. If SSF is set, this will cause "Status Bit Set" condition.
10	'1' – Mask Terminal Flag (TF) bit in status word. '0' – No masking. If TF is set, this will cause "Status Bit Set" condition.
9	'1' – Mask Reserve bits (RSRV) in status word. '0' – No masking. If RSRV is set, this will cause "Status Bit Set" condition.
8	'1' – Retry enable. Perform retry if message fails. The number of retries and the retry bus is defined in configuration registers 1 and 2.
7	'1' – Transmit on bus A. '0' – Transmit on bus B.
6	'1' – Perform internal loop back test. Message does not go out to 1553 bus but rather loop backed just like the normal echo of the transceiver. Of course the RTs do not reply. The host can check that the Loop back word has arrived correctly and that the loop back error flag is not set. Of course, other error bits should be masked because most likely they will be set.
5	If R08B11='1', i.e., Mask mode for BCST bit then this bit is the mask bit: '1' - will mask checking BCST bit '0' - will verify BCST is not '1', otherwise "Status Set" condition found. ELSE in compare mode: This bit should be equal to BCST bit, else "Status Set" condition found.
4	'1' – End OF Message Interrupt Enable (if not masked by interrupt mask register).
3	'1' – Validate RT response as in 1553A '0' – Validate RT response as in 1553B
2	'1' – The message is a mode command format (ignored by core)**.
1	'1' – The message is a BCST message format (ignored by core)**.
0	'1' – The message is an RT2RT command format.

** Core detects Mode command type and BCST command type from the command bits value regardless of the setting of these bits.

PART III:
ENHANCED BUS CONTROLLER MODE - (eBC)

5 ENHANCED BUS CONTROLLER

5.1 INTRODUCTION

The 1553 Bus Controller (BC) allows a subsystem to control the 1553 bus. Control is achieved by sending commands to RTs connected to the bus. Each command either instructs the RT to transmit words, receive words from the BC or perform some special actions, known as mode commands.

The Host programs the core with the relevant messages data alongside some configuration registers. Several such messages can be loaded into the shared memory by the host.

The Host normally wants to attend the core only once every time slice. This time slice is referred to as a frame. During a frame, many messages are transferred between the BC and the RTs and between the RTs (RT to RT commands). The Host needs to load the core with a complete set of messages for the entire frame.

When all commands per frame have been loaded to the core's memory, the host initiates the frame. From this point the core will sequence the messages one after the other until it is all done. No Host intervention is required during the frame run.

When the frame is finished, all data received is waiting in the core's memory for Host analysis.

Typically for real-time operation frames would be initiated every 20 ms.

The core can be programmed to repeat its frames automatically, thus off loading the Host from sequencing the 1553 bus even further.

The BRM1553D core can work in two different modes – "Legacy BC" and "Enhanced BC". In this section it is assumed that configuration register 6 bit 15 is set to high.

5.1.1 NON MIL-STD-1553B SETTINGS

The IP core is set to support different dialects than 1553B, by means of configuration.

These different settings affect both the BC and eBC modes. The RT and Message Monitor modes are also affected, please refer to the appropriate section.

The configuration register 3 bit 1, when set high, sets SA 31 to be non-mode, but rather a regular SA. In addition, this setting will assume only mode commands without data, so the BC and eBC will not transmit nor expect data words for messages designated to it with SA=0 with WC(4)='1'.

If the RT responds to a message with ME bit in the status response, it is not illegal message as in 1553B, but rather an indication that one of the data words in the message had a 1553 error. The BC and eBC can be enabled to perform a retry in that condition using a high in bit 3 of the BC Control word and Configuration register #4 bit 10.

Using Configuration register 5 bit 7, it is possible to turn Broadcast address 31 to a standard RT as defined by 1553A.

5.2 BUS CONTROLLER OPERATION MODES

5.2.1 BASIC OPERATION

The host manages the 1553 bus through the shared memory and several registers. The host should load the memory and registers prior the START command, and then analyze the results in memory after the messages transactions have been finished.

The memory data defines which messages need be transmitted and when. The host stores a set of up to 512 instructions into the memory. There are 16 different instructions that allow very complex and autonomous message sequencing.

The typical BC operation is organized in frames. Each frame includes several messages. Once the BC is triggered, the core's internal state machines start to execute the instruction list, one after the other until finished.

When the BC core is triggered by the START command (R03B02) instruction execution is initiated from the address in memory pointed by register address 0x000D. The instructions are executed one after the other until the last instruction which stops the operation.

The value of the current instruction pointer can be read from register 0x0003.

5.3 PROGRAMMING AND SETUP

5.3.1 REGISTERS

There are several registers mapped to the register section during a read or write operation.

The following section describes the registers and the operational bits of these registers.

Note for relevant bits:

- If the bit is read only, then the core acts according to the specified level. However for compatibility reasons, these bits may be written values that are different from their functionality.
- If the bit is Write/Read, then the default value is as defined but could be changed to desired level for preferred operation.
- If a reference is made to a particular register and a specific bit the notation is R01B12 for Register address 0x0001 bit 12.
- If a reference is made to a particular memory location the notation is M0100 for memory address 0x100.

5.3.1.1 INTERRUPT ENABLE REGISTER #1 ADDRESS 0X0

If the Host enables one of the bits below by setting its value to '1', and the specified event occurs, an interrupt will be generated to the Host CPU.

Bit number	Read/Write/Default	What event triggers the interrupt when enabled by '1'.
13	R/W/0	Transmitter timeout occurred
9	R/W/0	Memory protection fault **
8	R/W/0	BC Retry – following incorrect RT response has been initiated.
7	R/W/0	RT Address parity error found
6	R/W/0	Time tag counter rollover
4	R/W/0	BC End Of Message.
3	R/W/0	Enhanced BC Watchdog. If Configuration register #7 bit 1 is asserted.
2	R/W/0	Loopback error or message error or response timeout.
1	R/W/0	Wrong RT status or unexpected status bits set in status respond.
0	R/W/0	End Of Message.

Default Value: 0x0000 – all events masked.

** This bit is used in DDC Mini-ACE as memory handshake error. Since this cannot occur with the core, bit #9 is used for data protection. When Data pointer points to a non data block area (M0000 to M0103) due to wrong memory setup by the host, the core detects that, and asserts bit 9 of the interrupt status register interrupt to '1'. If this bit is not masked than an interrupt will occur.

Memory overrun is not prevented in BC mode, only an interrupt is set.

5.3.1.2 CONFIGURATION REGISTER #1 ADDRESS 0X1

This configuration register description assumes bits 14 and 15 are "00" – BC mode.

For RT/MT bit description and other registers value please refer to Remote Terminal and Monitor Mode section.

Bit number	Read/Write/Default	
15	R/W/0	'0' For legacy BC. Rest of description assume this bit is '0'.
14	R/W/0	'0' is preset for this bit.
13	R/W/0	Area A/B. Not relevant for eBC.
12	R/W/0	'1' – Stop at end of Message error. If optional retry succeeded => messages continue!
11	R/W/0	'1' – Stop at end of frame if error. If optional retry succeeded => frames continue!
10	R/W/0	'1' – Stop messages if unexpected, non-masked status bits are set. If optional retry succeeded => messages continue!
9	R/W/0	'1' – Don't process additional frames if message unexpected, non-masked status bits are set and auto frame enabled. If optional retry succeeded => frames continue!
8	R/W/0	'1' – Auto Frame Repeat. Will start next frame if none of the problems in bits 9-12 caused a halt. Refer to setting up auto repeat presets. Trigger first frame with START (R03B01). '0' – Host has to trigger every frame with START.
7	R/W/0	'1' – External trigger mode – sensitive to rising edge. '0' – External trigger disable.
6	R/W/0	'1' – Internal frame counter (64K steps of 100 us) re-trigger the frame in case auto frame repeat is enabled (bit 8). '0' – Internal trigger disable.
5	R/W/0	'1' – Message gap timer enable. If enable will start next message after gap*1us. Gap is defined in 3rd word in BC message block descriptor in stack. '0' – Message gap set to default (~10 us).
4	R/W/0	'1' – Global Retry enable. If enable, each message can be enabled for retry with its BC control word bit 8.
3	R/W/0	'1' – Retry twice after 2 failures. '0' – Retry only once after failure. See also Configuration register 4.
2	R/0	'1' – BC state machine has been started, and has not finished its messages transactions. In Auto repeat mode this signal will stay high until halted by host or error. '0' – BC is idle (after frame finished or "stop on error").
1	R/0	'1' – bus active – from start of first message in frame till end of last message in frame.
0	R/0	'1' - Message active – from start till end of message.

5.3.1.3 CONFIGURATION REGISTER #2 ADDRESS 0X2

All bits can be written and read. However they are not programmable, and their used value is indicated by the Assume comment.

Bit number	Read/Write/Default	
15	Assumed '1'	'1' – Enhanced Interrupts mode enabled. Interrupt status register will set its bit flags high if condition occur even if it is masked by the interrupt enable register. An interrupt will be generated if a bit is set and also enabled.
14	Assumed '0'	'0' – No Memory Parity bit for IP core.
10	Assumed '1'	256-word boundaries disable. '0' for RT mode. '1' in BC mode.
9..7	R/W/000	Time Tag resolution: "000" – 64 usec, "001"– 32 usec, "010"– 16 usec, "011"– 8 usec, "100"- 4 usec, "101"- 2 usec, "110" – test mode, "111" – External Tag Clk.
5	Assumed '1'	'1' – in BC mode – transmits the internal time tag word in "synchronize with data" mode command.
4	Assumed '1'	'1' – Auto clear of interrupt status registers #1 as a result of CPU reading its value.
3	R/W/0	'0' – generates a 500ns low pulse on the INTn signal. '1' – Level mode. INTn stays low until the host reads interrupt status register.

5.3.1.4 START RESET REGISTER WRITING TO ADDRESS 0X3

This is a write only register. Reading back will return Stack Pointer Register (same as reading address 0x0100).

Bit number	Read/Write/Default	
10	Write	'1' clear self test register at address 0x1C.
9	Write	'1' initiate RAM self test.
7	Write	'1' initiate Protocol self test.
6	Write	'1' stop BC operation at end of message.
5	Write	'1' Stop BC operation at end of frame.
4	Write	'1' increments Time Tag counter when resolution is "110".
3	Write	'1' resets the Time Tag counter.
2	Write	'1' resets the interrupt. Resets the FF that latches the interrupt condition. If the condition for interrupt persists, the relevant FF would be set again until the condition for causing this interrupt is cleared. Reset of Interrupt Register #1 and #2 and INTn to '1'.
1	Write	'1' BC START command. Starts the BC state machine.
0	Write	'1' Reset core. Reset all registers, FFs in core. Memories are not reset, and should be reset by CPU. If '1', all other bits have no effect.

5.3.1.5 STACK POINTER REGISTER READING FROM ADDRESS 0X3

This is a read only register. Writing to it is explained above.

Bit number	Read/Default	
15..0	Read 0x0	Image of the instruction list pointer.

Default Value: 0x0

5.3.1.6 LAST\CURRENT BC MESSAGE CONTROL WORD. ADDRESS 0X4

The last used BC message control word may be read through this register.

5.3.1.7 TIME TAG REGISTER – ADDRESS 0X5

The time tag counter can be reset by writing to this register value 0x0 or loaded to any other value by writing it here.

The time tag counter counts in clock steps according to the resolution set in configuration register #2.

The counter will rollover every 2^16 * resolution.

The value of this counter is also reset by the hardware raw_reset line, or the software reset done by writing '1' to register 3 bit 3 or 0.

The value of the time tag counter is placed in the stack of each transmitted command. Please see "Stack contents" section for details.

The BC Host may write to this register any time.

When performing synchronize-with-data command, the time tag value can be sent instead of the data word in the stack memory. The BC control word bit 15 controls the source of this data word.

Bit number	Read/Write/Default	
15..0	R/W/0	Read the Time Tag Counter. Write a new value to the time tag counter.

Default Value: 0x0.

5.3.1.8 INTERRUPT STATUS REGISTER #1 ADDRESS 0X6

This register indicates the cause of an interrupt.

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (if configuration register #2 bit 4 is '1'.).
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit number	
15	This bit is an OR of all interrupts bits that are not masked by interrupt mask register. It reflects the HW interrupt line in level mode, and indicates there is still an interrupt request in the pulse mode.
14	'1' – Wiring Fault was detected using SmartWiring, when enabled
13	'1' – Transmitter fail safe timeout.
9	'1' – Data Pointer pointed to a protected area where data blocks should not be present. Memory overrun has happened. Need to reset IP core.
8	'1' - BC retry has occurred. Each message has a bit in status word in stack that indicates if retry was done.
7	'1' - RT Terminal Address Parity Error. RT mode.
6	'1' - Time Tag Rollover from 0xFFFF to 0x0000.
4	'1' – Message was sent. This interrupt can be enabled or not for each message with BC control word bit 4.
3	'1' – Enhanced BC Watchdog. If Configuration register #7 bit 1 is asserted.
2	'1' – Loopback error or message error or response timeout.
1	'1' – Wrong RT address response or one of the (non BCST) status word bits were set high. BCST bit sets this bit high if configurations register 4 bit number 11 is set low (compare mode) AND the BCST bit is different from the BC control word of the message bit #5.
0	'1' – End of message has occurred.

*All other bits are not used and will be read as zero.

Default Value: 0x0000

5.3.1.9 CONFIGURATION REGISTER #3 ADDRESS 0X7

Bit #	Read/Write/Default	
15	Assumed '1'	'1' - always in Enhanced mode.
14..13	R/W/"00"	"00" – BC stack size is 256 words "01" – BC stack size is 512 words "10" – BC stack size is 1024 words "11" – BC stack size is 2048 words

5.3.1.10 CONFIGURATION REGISTER #4 ADDRESS 0X8

Bit #	Read/Write/ default	
12	Assumed '1'	Expanded BC control word always on. All 15 bits of BC control word are functional.
11	R/W/0	Mask / Compare the BCST bit in returned status bit. ** '1' – Mask. The relevant bit in the BC control word operates as mask for the BCST bit of the received status. '0' – Compare. The relevant bit in the BC control word is compared with the BCST bit of the received status.
10	R/W/0	'1' – Retry if 1553A dialect and RT status' word ME bit is set. 1553A dialect is set by setting BC control word bit 3.
9	R/W/0	'0' – No retry if status bits that are not masked are set. '1' – Retry a message if retry enabled and one of the unmasked status bits are set high in the returned RT status word. BCST bit pass/fail has a special treatment as seen above in bit 11 setup.
8	R/W/0	'0' – First retry on same bus if message failed. '1' – First retry on opposite bus if message failed.
7	R/W/0	'0' – Second retry on same bus as original failed message. '1' – Second retry on opposite bus of original message.
6	R/W/0	'1' - An RT can respond to a transmit command with Message Error (ME) bit set followed by the data words OR without any data words (as in illegal command) and no format error is reported for either response. Both are valid responses. '0' – Format error is posted if no data words follow a ME status.
5	R/W/0	'1' - An RT can respond to a transmit command with Busy bit set followed by the data words OR without any data words (as in busy RT state) and no format error is reported for either response. Both are valid responses. '0' – Format error is posted if no data words follow a busy status.
2..0	R/W/0	Write "010" to enable the loopback test. Refer to chapter 6 for detailed description. Write 0 to disable the test.

** Note that the BCST bit in the status return is only set by the RT in the proceeding message's status word after a broadcast message FOR a transmit status or transmit command mode commands. Otherwise the BCST bit should be '0'.

5.3.1.11 CONFIGURATION REGISTER #5 ADDRESS 0X9

Bit #	Read/Write/default	
15	Read/Write/'0'	This bit has no effect on IP cores. They specify the input frequency for OCT-65178 devices. '0' – 16 Mhz. '1' – 12 Mhz For other OCT-xxxxx devices this bit is ignored, and Configuration Register #6 bits 1 and 0 select between 10, 12, 16, and 20 Mhz.
10..9	Read/Write	'00' – response timeout is set to approximately 18.5 us. '01' – response timeout is set to approximately 22.5 us. '10' – response timeout is set to approximately 53.5 us. '11' – response timeout is set to approximately 130 us.
7	Read/Write	'0' – Address 31 is used as broadcast. '1' – Address 31 is used as normal RT address. Required for 1553A that does not support Broadcast. For 1553B it is hazardous to set this bit high.
6	Read	Reads the value of the RT_ADR_LAT hardware signal.
5	Read/Write	Read the actual RT Address 4 Used in RT mode.
4	Read/Write	Read the actual RT Address 3 Used in RT mode.
3	Read/Write	Read the actual RT Address 2 Used in RT mode.
2	Read/Write	Read the actual RT Address 1 Used in RT mode.
1	Read/Write	Read the actual RT Address 0 Used in RT mode.
0	Read/Write	Read the actual RT Address Parity Used in RT mode.

5.3.1.12 BC FRAME TIMING REMAINING ADDRESS 0XB

16 bits that define the number of 100 us left until end of frame, or zero.

This is a count-down register that is read only.

The eBC instructions may alter the value of this register.

5.3.1.13 BC MESSAGE TIMING REMAINING ADDRESS 0XC

16 bits that define the number of 1 us left until the start of the next message, or zero.

This is a count-down register that is read only.

This register starts from the value of the 4th word in the Message block.

5.3.1.14 EBC INITIAL INSTRUCTION REGISTER ADDRESS 0XD

These 16 bits define initial address in RAM for the instruction list to be executed.

The running value can be read from register address 0x3.

5.3.1.15 TEST REGISTER ADDRESS 0X11

It is possible to perform an encoder test between bus A and B using this register only if IP supports the BC and word monitor. Please refer to Chapter 6 for details.

Bit number	Read/ Write/ default	Description
15	R/W/0	'1' would transmit on Bus A, '0' on Bus B
11	R/W/0	Writing '1' would trigger transmission
10	R/W/0	'1' - save the echo from the transmitting bus to monitor. '0' – do not save echo, only word received from opposite bus.

5.3.1.16 PASSIVE TDR LEVEL 0X12

Read constant 0x7654. This constant can be used to verify proper Endian setup of system.

5.3.1.17 CONSTANT VALUE 0X13

Read constant 0x7654. This constant can be used to verify proper Endian setup of system.

5.3.1.18 LATCHED TIME TAG REGISTER 0X15

The value of the time tag register address 0x5, is latched into this register each time a rising edge is detected on the Strobe_TT input line.

This allows the SW to read a very accurate Time Tag value when the hardware event occurred, such as when working with GPS systems.

5.3.1.19 CONFIGURATION REGISTER #6 ADDRESS 0X18

Bit #	Read/Write/default	
15	R/W/0	'1' - Enhanced Bus Controller '0' – legacy bus controller (see legacy BC manual).
1..0	Read/Write "00"	These bits have no effect on IP cores. They specify the input frequency for OCTAVA devices. "00" – 16 Mhz. "01" – 12 Mhz "10" – 20Mhz.

	"11" – 10 Mhz. For OCT-65178 these bits are ignored, and Configuration Register #5 bit 15 selects between 12 and 16 Mhz.

5.3.1.20 CONFIGURATION REGISTER #7 ADDRESS 0X19

Bit #	Read/Write/default	
9	R/W/0	Version of IP core. Used by DDC API for version testing. If '1' is written to this bit, it is read in bit 5 and 9.
5	R/W/0	Version of IP core feedback. If '1' is written to bit 9, it is read here.
1	R/W/0	'1' - Enhanced BC watchdog timer enabled. If bit 3 of Interrupt mask register #1 is high, an End Of Frame interrupt will be issued when frame counter reaches 0.

5.3.1.21 BACK END VERSION OF CORE REGISTER ADDRESS 0X1A

This register returns the revision of the back-end logic.

5.3.1.22 BC CONDITION CODE READING FROM REGISTER ADDRESS 0X1B

This table describes the data read from this register. Please see next table for data write to this register. Message test conditions are set on each message, and stay valid until next message.

Bit #	
15	Constant Logic '1'
14	Retry MSB – '1' if 2 retries were performed on last message.
13	Retry LSB – '1' if 1 or 2 retries were performed on last message.
12	'1' when a loop back error, or format error, or no response in the last message.
11	Test if one of the bits in the RT response status word was set and its mask is off. Broadcast bit-testing works in two modes; refer to configuration register 4 bit 11.
10	Good data block - saved error free data in memory.
9	1553 format error in last RT response.
8	RT did not respond.
7	General purpose flag 7
6	General purpose flag 6
5	General purpose flag 5
4	General purpose flag 4
3	General purpose flag 3
2	General purpose flag 2
1	General-purpose flag 1 OR "Less than" results of last compare instruction.
0	General-purpose flag 0 OR "Equal" results of last compare instruction.

It is very important to be careful when working with GP0 and GP1 since they are updated from both the host CPU and the instructions engine.

5.3.1.23 BC GP FLAGS CONTROL BY WRITING TO REGISTER ADDRESS 0X1B

If data is written to this register, the following table describes the outcome of a '1' in any one of the bits. The default values of the GP flags are '0'. If a general purpose flag is both cleared and set, it will be toggled.

Bit #	
15	Clear General purpose flag 7 (GP7)
14	Clear General purpose flag 6 (GP6)
13	Clear General purpose flag 5 (GP5)
12	Clear General purpose flag 4 (GP4)
11	Clear General purpose flag 3 (GP3)
10	Clear General purpose flag 2 (GP2)
9	Clear General purpose flag 1 (GP1)
8	Clear General purpose flag 0 (GP0)
7	Set General purpose flag 7 (GP7)
6	Set General purpose flag 6 (GP6)
5	Set General purpose flag 5 (GP5)
4	Set General purpose flag 4 (GP4)
3	Set General purpose flag 3 (GP3)
2	Set General purpose flag 2 (GP2)
1	Set General purpose flag 1 (GP1)
0	Set General purpose flag 0 (GP0)

5.3.1.24 SELF TEST STATUS REGISTER 0X1C

This register indicates the state of the internal protocol and RAM tests

Writing to reset register at address 0x3 clears the content of this register.

Bit #	
15	Protocol Built-in-test complete
14	Protocol Built-in-test in progress, should be less than 2 ms.
13	Protocol Built-in-test passed
11	Protocol Built-in-test in progress or complete
7	RAM Built-in-test complete
6	RAM Built-in-test in progress, should be less than 2 ms.
5	RAM Built-in-test passed

After hardware power up, the core initiates the protocol test automatically. Allow 2 ms for this test to complete.

Note: Please refer to the introduction section of this document for a description of the actual test coverage.

5.3.1.25 INTERRUPT MASK REGISTER #2 ADDRESS 0X1D

If the Host enables one of the bits below to '1' and the specified event occurs, an interrupt will be generated to the Host CPU. The status bit will be set regardless of the mask bit.

Bit #	Read/Write/Default	What event triggers the interrupt when enabled by '1'.
14	R/W/0	Enhanced BC instruction parity error detected.
12	R/W/0	General Purpose queue rollover.
11	R/W/0	eBC – Call stack overflow or underflow. Max depth is 16.
10	R/W/0	eBC illegal Op-code or illegal parity or illegal constant.
5	R/W/0	eBC IRQ3
4	R/W/0	eBC IRQ2
3	R/W/0	eBC IRQ1
2	R/W/0	eBC IRQ0
1	R/W/0	BIT test complete

5.3.1.26 INTERRUPT STATUS REGISTER #2 ADDRESS 0X1E

This register indicates the cause of an interrupt.

The status bits will be cleared as a result of either of:

- Writing to reset register 0x3 bit 2 '1' will reset this register's bits.
- Reading this register will clear it contents (if clear by read enabled in configuration register #2).
- Writing to this register, write the bits just read assures clearing only bits that were handled. This also supports multiple interrupt services to work with a single register. This method is unique to Sital Technology IPs. Note that if interrupt causing bits are left high after write, an interrupt pulse will not be generated in pulse mode, but the interrupt line will stay active in level mode.

Bit #	What event triggers the interrupt when enabled by '1'.
15	This bit is high when an interrupt is pending.
14	Enhanced BC instruction parity error detected.
12	General Purpose queue rollover.
11	eBC – Call stack overflow or underflow. Max depth is 16.
10	eBC illegal Op-code or illegal parity or illegal constant.
5	eBC IRQ3
4	eBC IRQ2
3	eBC IRQ1
2	eBC IRQ0
1	BIT test complete.
0	'1' indicates that interrupt register #1 is requesting an interrupt. Its is recommended to read Interrupt status register #2 if its bits are required and if bit 0 is '1' then read Interrupt status register #1.

5.3.1.27 BC GENERAL PURPOSE QUEUE POINTER REGISTER ADDRESS 0X1F

The host loads this register with 16 bits that point to a memory queue of 64 words.

The enhanced BC engine stores data to the pointed address and increments the 6 LSBs of this register, and points to the next location where data will be saved. When the 6 LSBs are 3F and a new word is pushed to the queue, the next address will have the same 10 MSBs, but the 6 LSBs are 00.

Bits 15..6 are the base queue address.

Bits 5..0 point to the next empty space for data loading by the BC engine.

5.4 MEMORY

5.4.1 MESSAGE SEQUENCE CONTROL

The sequencing engine runs the instructions one after the other.

There are two instructions that sequences message transmission, XEQ and XQF. The following schematic illustrates the indirect memory data structure of pointers and databases that construct the message sequencing.

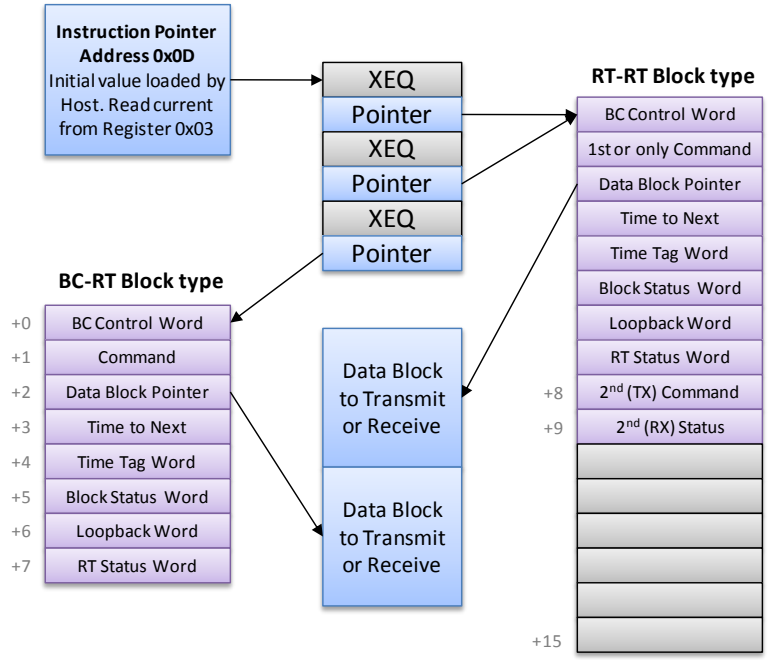

Figure 10: Execute BC sequencing

Register 0xD is loaded with the address of the start of the instruction list. Once a BC start command is initiated, this pointer is loaded to an internal register, but register 0xD does not change. The value of the instruction pointer can be read from register 0x3 when the IP is in BC mode.

If it is an XEQ or XQF instructions, the parameter points to the message block.

Two message blocks are available. The BC RT block type and the RT2RT block type. The BC RT clock type is 8 words long and always starts from an 8-aligned address. It describes messages of type BC to RT, RT to BC, Mode commands, and BC to Broadcast commands. For RT2RT messages the block is 10 words long and 6 unused extra words align this clock to 16 words.

Please refer to the Message Block paragraph for detailed description.

5.4.2 VARIABLES

There are 16 variables added by Sital to the instructions list.

Dedicated commands are added for variable manipulations and checking.

Variables 8 to 15 are used by Sital Technology Software Library, so the user should avoid accessing these variables.

User should first focus on a variable, and then use variable operations on it.

If an operation is desired on a different variable, it needs to be focused first.

After the user wrote his program of OpCodes and defined his frame, he calls on a frame resolve function in the SW library package. This frame resolve functions manipulates the OpCodes and inserts additional call OpCodes commands to optionally asynchronous commands.

User should assume that Sital SW Library changes focus, thus not assume focus survives Sital's SW library inserted commands.

The variable commands include:

Focus Variable, FSV, to specify which variable is being addressed.

Push Variable, PSV, to focus on and push a variable value to the GPQ.

Load Variable, LDV, to load 16 bit value to the focused variable.

Add to Variable, ADV, to add 16 bits value to the focused variable. The added value should be regarded as a 2's compliment value. For example, adding 0xFFFF is like decreasing by one.

Compare Variable, CMV, to compare the focused variable with a 16 bit parameter, and set GPF0 and GPF1 according to the results. The compare is unsigned on two 16-bit numbers.

These new commands are in a different instruction space. These instructions are enabled when bits 9..5 are set to "10010" rather than the normal "01010" code.

5.4.3 INSTRUCTIONS

This chapter list the instructions supported by the core.

All instructions are pairs of two 16-bit words.

The first word is an instruction and the second is a parameter.

Each instruction has a few bits that are constant, as a well as a parity bit (15). The instruction executer verifies that the constant bits and parity is valid prior to execution, else will halt the execution and return to idle, and optionally issue an interrupt (ISR2 bit 10).

Offset	Name	Bit	Description
0	Instruction	15	Parity. Odd parity of instruction.
		14..10	Op Code Field. Go to table.
		9..5	"01010" – mandatory for these bits, "10010" – Legal for Sital OpCodes.
		4	Invert condition. See condition code table.
		3..0	Condition codes. Go to table.
1	Parameter	15..0	16-bit parameter or pointer.

The following table describes the instructions:

Name	Short	Op Code Bits Hex	Parameter or Pointer	Condition met?	Description
Execute Message	XEQ	00001 0x1	Address of message block	Yes	Execute message pointed by the parameter. Conditions 0x8 to 0xE are not to be used.
				No	Continue to next instruction
Jump	JMP	00010 0x2	Address to jump to	Yes	Jump to pointed address.
				No	Continue to next instruction
Call Subroutine	CALL	00011 0x3	Address to jump to	Yes	Jump and push return address to stack. Call stack depth is 16. Conditions 0x8 to 0xE are not to be used. Halt upon stack depth violated.
				No	Continue to next instruction
Return from Subroutine	RTN	00100 0x4	Not used	Yes	Pop the last address and continue from the popped address + 2. Halt upon stack depth violated.
				No*	Continue to next instruction.
Interrupt Request	IRQ	00110 0x6	4 LSBs	Yes	If bit N (0 to 3) is set, bit N+2 of ISR#2 is set. If bit is reset, it will not reset the ISR#2 bit.
				No	Continue to next instruction
Halt	HLT	00111 0x7	Not used	Yes	Stop BC operation.
				No	Continue to next instruction
Delay	DLY	01000 0x8	Number of microseconds	Yes	Delay operation for the number of microseconds. This command uses the time to next message timer!
				No	Continue to next instruction
Wait EOF	WFT	01001 0x9	Not used	Yes	Wait until frame timer = 0
				No	Continue to next instruction
Compare to Frame Timer (FT)	CFT	01010 0xA	Value in 100 us resolution.	Null	Compare to frame timer (FT). Set flag (Less Than) = (FT > Value) Set flag (Equal) = (FT == Value)
Compare to Message Timer (MT)	CMT	01011 0xB	Value in 1 us resolution.	Null	Compare to message timer (MT). Set flag (Less Than) = (MT > Value) Set flag (Equal) = (MT == Value) After XEQ and XQF MT is always 0.
Change General Purpose Flags	FLG	01100 0xC	8 LSBs set GPx 8 MSBs clear GPx	Null	Bits n and n+8 affect General Purpose Flag n 0 0 – Keep value. 0 1 – Clear GPF n 1 0 – Set GPF n 1 1 – Toggle GPF n
Load Time Tag Counter	LTT	01101 0xD	Value for loading	Yes	Load the 16 bits of the Time Tag Counter. Resolution is determined by configuration register #2.
				No	Continue to next instruction
Load Frame register	LFT	01110 0xE	Value in 100 us resolution.	Yes	Load the 16 bits of register 0xD - frame time register.
				No	Continue to next instruction

Start Frame timer	SFT	01111 0xF	Not used	Yes	Start count down of frame counter.
				No	Continue to next instruction
Push Time Tag Counter	PTT	10000 0x10	Not used	Yes	Push the value of the time tag counter to the general-purpose queue.
				No	Continue to next instruction
Push Block Status word	PBS	10001 0x11	Not used	Yes	Push the current Block Status Word to the general-purpose queue. Bits 15 and 14 are always written as '0'!
				No	Continue to next instruction
Push value	PSI	10010 0x12	Value	Yes	Push the parameter value to the general-purpose queue
				No	Continue to next instruction
Push indirect Value	PSM	10011 0x13	Memory Address	Yes	Push the parameter value that is fetched from the address parameter to the general-purpose queue
				No	Continue to next instruction
Wait for trigger	WTG	10100 0x14	Not used	Yes	Wait for a rising edge in the external trigger. Be very careful not to hang the BC engine!
				No	Continue to next instruction
Execute and Flip	XQF	10101 0x15	Message block address	Yes	Execute the message. Toggle bit 4 of the parameter.
				No	Execute the message! (The condition is checked after message completion). Don't toggle bit 4 of parameter.
The following commands work with bits 9..5 code: "10010".					
Focus on Variable	FSV	00000 0x00	Variable bits 3..0	Yes	All variable actions are performed on the selected variable 0 to 15. Variables 8 to 15 are used by Sital's API library. Users using Sital API library should avoid focusing on these vars.
Push Variable	PSV	00000 0x00	Variable bits 3..0	No	Focus and push value of selected variable to GPQ.
Add to Variable	ADV	00001 0x01	16 bit signed parameter	Yes	Add the parameter to the variable on focus -1 is 0xFFFF, 1 is 0x0001, 2^15-1 is 0x7FFF
				No	Leave variable as is.
Load Variable	LDV	00010 0x2	16 bits value	Yes	Load the parameter value to the variable on focus
				No	Leave variable as is.
Compare Variable	CMV	00011 0x03	16 bit unsigned value	Yes	Compare to focus Variable (VARx). Set flag (GPF0, Less Than) = (VARx > Value) Set flag (GPF1, Equal) = (VARx == Value)
				No	Do not change GPF0 and GPF1

Note: Blue colored instructions from0x00 to 0x03 are non DDC instructions and need code "10010" in bits 9..5.

The following table describes the possible conditions for the instructions:

Cond. Bits 3..0	Name when Bit 4 = 0	Name when Bit 4 = 1	Functional description
0	Less than OR GP0	Not Less than AND not GP0	This bit is set as a result of the compare CFT or CMT or FLG instructions, or by the host through general-purpose register. Less Than is set high if parameter < Timer. Low if >=.
1	Equal OR GP1	Not Equal AND Not GP1	This bit is set as a result of the compare CFT or CMT or FLG instructions, or by the host through general-purpose register. Equal is set high if parameter = Timer. Low /=.
2	GP2	Not GP2	This bit is affected by FLG instructions, or by the host through general-purpose register.
3	GP3	Not GP3	This bit is affected by FLG instructions, or by the host through general-purpose register.
4	GP4	Not GP4	This bit is affected by FLG instructions, or by the host through general-purpose register.
5	GP5	Not GP5	This bit is affected by FLG instructions, or by the host through general-purpose register.
6	GP6	Not GP6	This bit is affected by FLG instructions, or by the host through general-purpose register.
7	GP7	Not GP7	This bit is affected by FLG instructions, or by the host through general-purpose register.
8	RT Respond timeout	Ok RT Responded	Test RT's response time compared with response timeout parameter in configuration register #5.
9	RT 1553 error	RT respond Ok	Test if there were errors in last message like sync, bi-phase, parity, bit count, word count or wrong RT address in status.
0xA	Good Data received	Bad or no Data received	Test if the transmitting RT sent data to the BC (or other RT), and it was valid.
0xB	Non masked status bits are set	Non masked status bits are not set	Test is one of the bits in the RT response status word was set and its mask is off. Broadcast bit testing works in two modes, refer to configuration register 4 bit 11.
0xC	Bad Message	Good Message	Test is there was a loop back error, or format error, or no response in the last message.
0xD	One retry	No retry	Test if there was one retry or no retry.
0xE	Two retries	Not relevant	Test if there was two retries or no retries.
0xF	True	False	Constant conditions. True condition expected to populate table.

5.4.4 MESSAGE BLOCK

The BC stack contains up to 512 groups of 8/16-word command descriptors. Each command descriptor is composed of the following words:

	Name	Bit	Description
0	BC Control Word	15	Transmit internal time tag in Synchronize with data mode command: '1' - Use time tag counter as data. '0' - Use data word in message block.
		14	'1' – Mask Message Error (ME) bit in status word. '0' – No masking. If ME is set, this will cause "Status Bit Set" condition.
		13	'1' – Mask Service Request (SR) bit in status word. '0' – No masking. If SR is set, this will cause "Status Bit Set" condition.
		12	'1' – Mask busy bit in status word. '0' – No masking. If busy is set, this will cause "Status Bit Set" condition.
		11	'1' – Mask Sub-System Flag (SSF) bit in status word. '0' – No masking. If SSF is set, this will cause "Status Bit Set" condition.
		10	'1' – Mask Terminal Flag (TF) bit in status word. '0' – No masking. If TF is set, this will cause "Status Bit Set" condition.
		9	'1' – Mask Reserve bits (RSRV) in status word. '0' – No masking. If RSRV is set, this will cause "Status Bit Set" condition.
		8	'1' – Retry enable. Perform retry if message fails. The number of retries and the retry bus is defined in configuration registers 1 and 2.
		7	'1' – Transmit on bus A. '0' – Transmit on bus B.
		6	'1' – Perform internal loop back test. Message does not go out to 1553 bus but rather loop backed just like the normal echo of the transceiver. Of course the RTs do not reply. The host can check that the Loop back word has arrived correctly and that the loop back error flag is not set. Of course, other error bits should be masked because most likely they will be set.
		5	If R08B11='1', i.e., Mask mode for BCST bit then this bit is the mask bit: '1' - will mask checking BCST bit '0' - will verify BCST is not '1', otherwise "Status Set" condition found. ELSE in compare mode: This bit should be equal to BCST bit, else "Status Set" condition found.
		4	'1' – End OF Message Interrupt Enable (if not masked by interrupt mask register).
		3	Ignored. Should be '0' for future compatibility.
		2	'1' – The message is a mode command format (detected from command, thus it is ignored by core).
		1	'1' – The message is a BCST message format (detected from command, thus it is ignored by core).
		0	'1' – The message is an RT2RT command format. 16 words block. '0' – The message is BC to/from RT message. 8 words block.
1	Command	All	1553 command. Tx, Rx, Mode or broadcast command. If the command code is 0xFFFF, the message is skipped. Typically skipping would take 5 us.
2	Data Block Pointer	All	Address pointer to the data block. Up to 32 words block.
3	Gap to next message	All	Time in microseconds from beginning of this message to the beginning of next message. If smaller than message length inter-message gap is 10 us.
4	Time Tag Word	All	Time Tag value when the message was executed.
5	Block Status Word.	15	End Of Message – Set to '1' by the core when the message has been complete .

	Updated by Core*		
		14	Start Of Message - Set to '1' by the core when the message has been started. In most cases, this bit is '1' after end of message if there is a 1553 bus-coupling problem.
		13	'0' – This message has been processed on Bus A. '1' – This message has been processed on Bus B.
		12	'1' – Error was found in the message. Bits 10, 9, 8, 3, 2, 1, 0 indicate cause of error. If all are zero, the error might be one of: 4. The BC is trying to transmit on a busy line (another BC is active). 5. The first message in RT2RT is a transmit command 6. The second RT2RT command is a receive command.
		11	Status Set. One of the status bits (excluding BCST bit) of the status return was '1'. BCST bit works in either mask mode or compare mode. In mask mode it works like other mask bits on the BCST bit. In compare mode, Status set occurs if BCST bit is different from bit 5 of BC control word.
		10	Format Error. The returned echo from the RT contained 1553 violations. See bits 3, 2, 1, 0 for a more accurate guess of the source of the problem.
		9	Response timeout. The RT responded too late or didn't respond at all.
		8	Loop back failed. The nature of 1553 bus is that every word transmitted, is also echoed back. The core verifies that the echo is correct and equal to the transmitted word. If not, this bit is set to '1'. Tip: The source of this type of error could be transceiver fault, or bus coupling problem.
		7	Unmasked Status bit set. This bit will be set to '1' if one of the status bits are set high and its appropriate mask bit in the BC control word is unmasked ('0'). BCST bit influences only in mask mode. See registers section for description of BCST bit.
		6	'1' indicates that two retries were performed.
		5	If bit 6 is '0' then a '1' indicates one retry was performed.
		4	Good data block received by BC. '1' – after an RT-BC, RT2RT, and Transmit Mode code with data commands if the message ended OK. '0' – after other message types, or if the above type of message was invalid. Loop back test failure does not cripple this bit result.
		3	'1' – The RT responded with wrong RT address.
		2	'1' – The RT transmitted a wrong number of words.
		1	'1' – Incorrect sync type response by RT.
		0	'1' – Invalid word. Indicates that the RT responded with a word containing 1553 errors.
6	Loop back word	All	The echo of the last word transmitted by the BC in the message.
7	RT status word	All	The (first) returned RT status.

For RT2RT message blocks 2 additional words are used, and 6 more words are null			
	Name	Bit	Description
8	2nd Command	All	1553 command. The Tx command for RT2RT message formats.
9	2nd RT Status	All	The receiving RT returned status in RT2RT message format.
10 to 15	Unused words		Serve as alignment words to assure all message blocks are 8 word aligned, i.e., their pointer address' 3 LSBs is always "000".

5.4.5 GENERAL PURPOSE QUEUE

The eBC instruction set includes 4 commands for managing a queue for mailbox with the Host CPU.

The eBC sequence program may contain instructions to push data on to the queue. These instructions may push the time tag, the block status word, direct values from their parameter word or indirectly push the contents of memory data.

The queue is 64 words deep and it can be located any place in the memory using a 16-bit pointer that is initialized by the host CPU.

The pointer is located in register 0x1F.

The host sets all 16 bits of this register. The eBC engine increments only the 6 LSBs of that register and the 10 MSBs stay static.

If for example 0 is loaded into register 0x1F, the queue will fill up from address 0x0 to 0x3F (63) and cycle back to 0x0.

If for example 0xF000 is loaded into register 0x1F, the queue will fill up from address 0xF000 to 0xF03F (63) and cycle back to 0xF000.

When the queue's 6 LSBs cycle from 0x3F to 0x00 a status bit is set in interrupt status register bit 12. If it is not masked off, it also generates an interrupt.

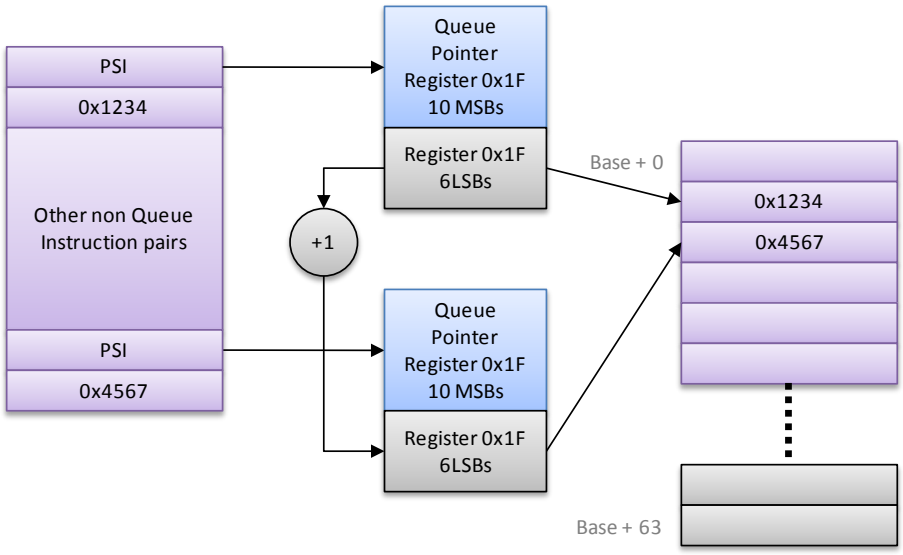

Figure 11: General purpose Queue

5.4.6 MINOR-MAJOR MESSAGE GAP CONTROL

In some systems it is desired to offload the host CPU and let the BC engine manage the message sequencing in a more autonomous mode. In such systems that some messages do not appear in every frame, but rather every second frame, or every forth frame, it is possible to load the BC with instructions to send multiple frames. Those multiple frames are said to be a single major frame. The major frame can be instructed by the host to be repeated automatically, or started every time.

The minor frame has to comply with the frame length, for example 20 ms, thus by using the time-to-next-message word, the sum of all messages' time-to-next-message should be the frame length.

The following diagram describes the usage of the time-to-next-message word provided by the 4th word in the message block.

If the time-to-next-message is shorter than the message length (0 for example), inter-message is the default (~10 us).

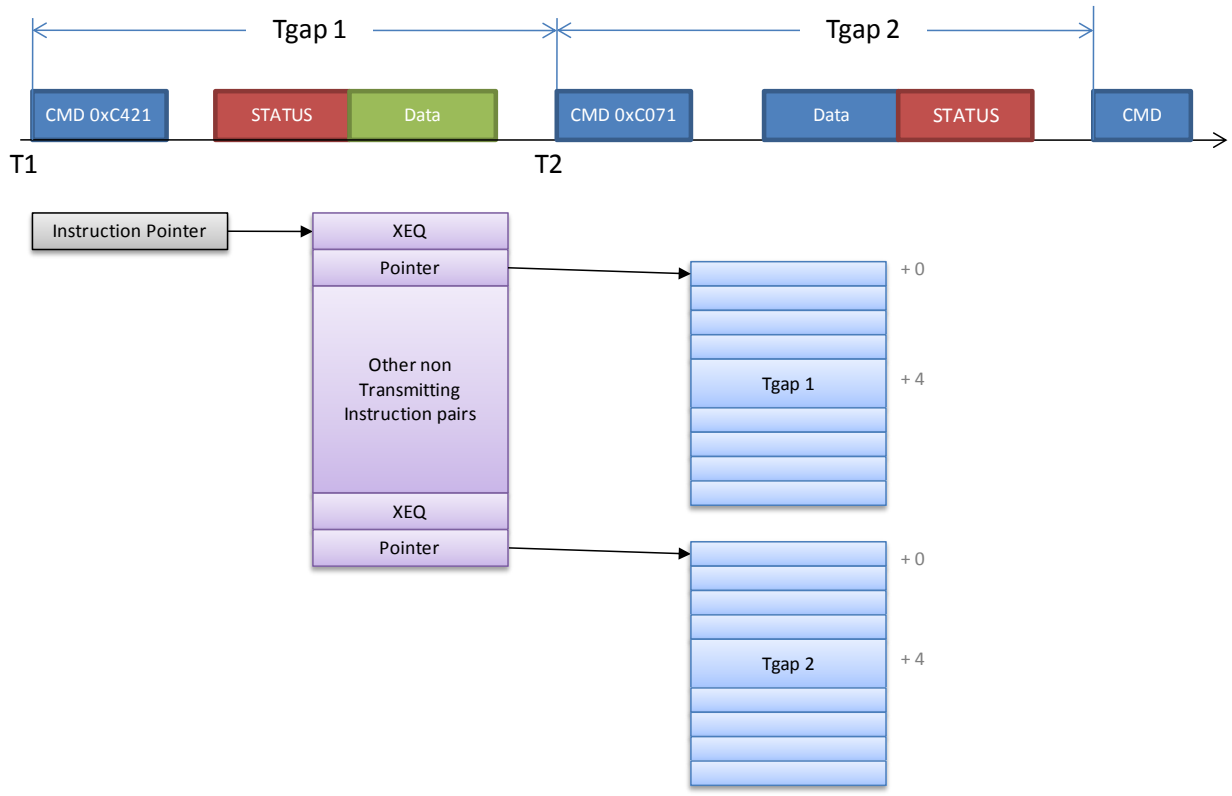

Figure 12: 4th word example time-to-next-message

PART IV:
OTHER MODES.

6 EXTERNAL LOOP-BACK

It is possible to perform an external Loop Back test from bus A to bus B or bus B to bus A with the BRM1553D. For EBR1553 users, it is impossible to use this loopback in HUB architecture. See EBR1553 section for suggested external loopback.

The loopback test is only supported for BRM1553D which incorporates the BC and Word Monitor.

The motivation to perform this test is to test the transceiver and transformer circuits connected to the BRM1553D without the need for an external 1553 tester.

The disadvantage of this test is that if the clock speed fed to the BRM1553D is incorrect or unstable, the test would pass; setting an allusion that everything is Ok.

This test is a very compact way of testing the transceivers and transformers and their connection to the BRM1553D.

It is possible to run this test for a very long time, such as when performing ATP for the unit under heat and other environmental conditions.

It is recommended to connect a resistor network bridge between channel B and A. The resistor network should attenuate the transmitting channel signal to the receiving channel such that the received signal would be just above the minimum detected signal (MDS) of the receiver. With that approach, a damaged transmitter, or receiver would not pass the test as opposed to simply loading the bridge with a 78 Ohm resistor.

Please consider the following resistor network bridge:

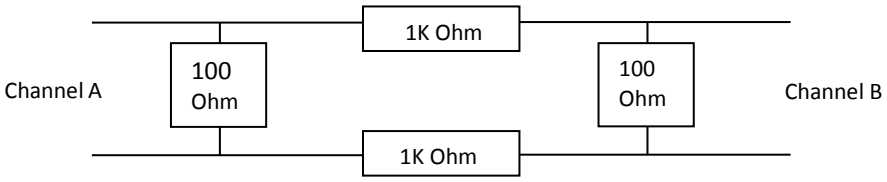

Figure 13: Resistors Network for Loop-Back

A transmission value of 24Vp-p on one side would yield 1Vp-p on the other side.

The minimum test would involve transmission from channel A to B and then from Channel B to A.

You may transmit as many as the length of the memory divided by 4. That is 1024 words in any direction for a 4Kx16 memory.

Please notice that each word transmitted would be monitored twice, once on the transmitting channel, and the second on the receiving channel. It is most likely that the echo of the transmitting channel would be received BEFORE the receiving channel, but theoretically it could swap.

Each word received is saved in the Word Monitor memory in 2 16-bit words. The even address holding the 16 bit pattern that was sent, along with 16 bits of tag word. Please refer to the word monitor document for a description of the tag word. In this test you would need to check a few bits of the tag word.

Please refer to the next table and follow its steps to perform the testing:

Software steps to run the test. Addresses are word offset not byte offset.

	Setup stages. Done once to setup the test.	
1	Configuration Register 4 (0x8 <= 2)	Write "010" to 3 LSBs.
2	Configuration Register 1 (0x1 <= x4000)	Write 0x4000 – enables word monitor mode. Make sure this mode is supported by BRM1553D IP that you have.
3	Memory Address 0x0100 (Memory 0x100 <= 0)	Write the monitor base – 0x0000. This is the place to look for the results.
4	Start / Stop Register (0x3 <= 2)	Write 0x0002 to start the word Monitor.
	Iterative stages. Loop for as long as needed.	
6	Register 0x000D (0xD <= xABCD)	Write the 16 bit word to transmit.
7	Test Register 0x0011 (0x11<=xBC20)	For A=>B write 0xBC20
		For B=>A write 0x3C20
		Bit 10 when '1' => first word is echo on same bus, second is from other bus. * When '0'=> first word is from second bus, no echo word stored.
8	Wait for at least 20 us	
9	Repeat	Repeat these stages until all transmitted words are sent.
		One word is also Ok.
	Stop the test	
10	Test Register 0x0011	Write 0x0000 to stop the transmission
11		
	Analyze the results. * 4/2 words are stored in the Word Monitor for each one transmitted.	
11	Memory address 0x0000	* Echo of 1st Data word of step 6 in same channel.
12	Memory address 0x0001	Tag word (see below) of 1st data: Verify error bit (4) is '0'. Bit 2 is '1' for bus B, '0' for bus A.
13	Memory address 0x0002	Echo of 1st Data word of step 6 in opposite channel.
14	Memory address 0x0003	Tag word (see below) of 1st data: Verify bit 4 is '0'. Bit 2 is opposite of its value in step 12.
15	Repeat checking	For each iteration in steps 6 to 9, check additional 4 addresses. (for iteration N, check addresses N*4 to N*4+3)
	Test end	**Write '1' to register 3 to Software reset the IP.**

Tag Word Contents:
Bit 15...8 – number of ½ microsecond from previous word. (can ignore)
Bit 7 – Always '1'.
Bit 6 – '0' is command word written to 0xD is for this RT address or broadcast.
Bit 5 – '0' if broadcast command (5 MSBs = x1F)
Bit 4 – '1' indicates an error in word (make sure it is always '0').
Bit 3 – '1' Command sync detected, '0' data sync detected.
Bit 2 – '1' received on channel B, '0' on A.
Bit 1 – '0' – There was a gap before this word. '1' back to back. Should be '0'.
Bit 0 – '0' - Mode command (bits 9...5 are "11111" or "00000")

Copy to command file to run the test:

```
WRITE 40008 0002  # Enable test mode
WRITE 40001 4000  # Word Monitor Enable
WRITE 00100 0000  # monitor base
WRITE 40003 0002  # Start monitor
WRITE 4000D 0021  # monitor base
WRITE 40011 BC20  # A=>B
WAIT_FOR 30 us    # WAIT for command to flow
WRITE 40011 0000  # stop test
#Check results
READ  00000 0021  # Read command
READ  00001 --A9  # Read tag word
WAIT_FOR 3 us
```

PART V:
EXTENDED BIT RATE - EBR1553 OPTION

7 EBR OPTION

7.1 INTRODUCTION

The BRM1553D IP was also adjusted to support EBR1553 mode.

EBR1553 (Enhanced Bus Rate MIL-STD-1553) is defined in the standard AS5652 by SAE Aerospace and supports a subset of the MIL-STD-1553 commands, for example excludes RT to RT messages, and is arranged in a star topology mode instead of the traditional multi-drop architecture of MIL-STD-1553. The Media that runs this faster communication is dependent on RS485 10Mhz transceivers rather than the traditional floating ground transformer style. In addition, Bus redundancy is not defined, so only one bus is used.

EBR1553 is intended for the weapons bus for new aerospace architectures, but can be used in existing platforms that have RS485 wiring present.

Sital's EBR1553 offering is based on the same BRM1553D functionality with the required changes in the front end transceiver interface. The EBR1553 mode was designed to be 100% SW compatible with existing SW developed for the BRM1553D. The obvious changes of speed might be required to be adjusted. Note that all counters timers and timeouts are 10 times faster than in the regular 1 Mhz MIL-STD-1553. Response timeout parameter for example can be set to 22.5 us, but in the EBR mode this would be 2.25 us.

To minimize the changes, modes like RT to RT were not deleted from the state machines which means that the SW designer should avoid setting RT to RT messages in the BC mode to prevent improper operation.

Note:
All other chapters of this manual apply also to the EBR Option.

7.2 EBR1553 ADDED PINS

Please refer to BRM1553D_HW_Interface_Verxx.PDF for additional EBR pins.

7.3 BC MODE AND ITS HUB

In EBR1553 the BC manages a HUB for star topology. This HUB is built of up to 31 RS485 transceivers, each having a single Tx and Rx as well as DE (Data Enable) line. The decoder and encoders inside the BRM1553D where upgraded to handle this single line bus, and assume that the transceivers are biased to '1'. This means that when no one transmits on the bus, the Rx line would read '1'. Modern RS485 transceivers are internally biased, thus the biasing resistors can be avoided.

For HUB lines that are not in use, and are not connected to a transceiver, the Rx line must be set to '1', otherwise the IP would not work properly.

There are three types of HUB functionality in the standard. All three types are supported and selected via a dedicated pair of input signals.

For the Link mode, which seems to be the most popular among our customers, The RT address selected for usage (see RT address in RT mode) is used by the BC for all transmissions. It should be set to 0 with parity 1 for a typical application.

The width of the HUB is a synthesis parameter that needs to be provided prior to the supply of the IP core. Care must be taken when the HUB width is very large and the BC transmits long messages. When such happens, the RS485 transceivers each consume 3.3v / 60 Ohm, which might overload the power supply.

Other than the star topology, the BRM1553D-EBR can be connected in a standard RS485 multi-drop configuration.

7.4 RT MODE

In RT mode, the HUB width should be defined as 1. The RT is not connected to its own HUB, but rather its line is connected to one of the BC HUB's ports.

7.5 MT MODE

In MT mode, there is no transmission, only bus monitoring. In that case the HUB Width can also be set to 1. The MT might also connect to a wider HUB, i.e. to multiple HUB stubs, but it can only record a single RT response (as EBR1553 defined).

7.6 RT / MON MODE

The BRM1553D EBR can function as an RT/MON as in regular 1553. This is a SW selectable mode.

7.7 EXTERNAL LOOPBACK

EBR bus topology is HUB and not multi-drop as in standard 1553. This HUB topology does not allow loop-backing between the links. In case an external loopback is required please consult with Sital Technology technical support.

APPENDIX A: CHANGES TRACKING

Date	Revision	Change	By
January '11	6.7	United manuals for BC/RT/MT Split manuals between HW and HW/SW interface	Duli
April '11	6.71	Added explanation on Configuration Register #7 Made format updates on the document	Duli
January '12	6.8	Added Extended Bit Rate Support	Ofer H.
April '12	6.9	Added Automatic Endian control in Hardware	Ofer H.
Feb '14	7.1	Added MIL-STD-1553A support	Ofer H.
Jan 11 '15	7.1	Monitor only mode update time tag register from bus.	Ofer H.
June 2016	8.0	Added Sital Smart Wiring. Removed coupling block.	Ofer H.

17 Atir Yeda St., Kfar-Saba, ISRAEL 44643

Email: info@sitaltech.com

Website: http://www.sitaltech.com

The information provided in this User's Guide is believed to be accurate; however, no responsibility is assumed by Sital Technology for its use, and no license or rights are granted by implication or otherwise in connection therewith. Specifications are subject to change without notice.

Please visit our Web site at www.sitaltech.com for the latest information.

© All rights reserved. No part of this User's Guide may be reproduced or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission by Sital Technology.