HB0088
Handbook
Core1553BRT_APB 4.4

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

About Microsemi

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Learn more at www.microsemi.com.

51700088. 6.0 9/20

Table of Contents

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 6.0

The core version was updated from v4.3 to v4.4. PolarFire® SoC support information was also added.

1.2 Revision 5.0

The core version was updated from v4.1 to v4.3. PolarFire® support information was also added.

1.3 Revision 4.0

Update a new section "Precompiled Libraries" section.

1.4 Revision 3.0

The core version was updated from v3.3 to v4.1. SmartFusion®2, IGLOO®2, and RTG4™ support information was also added.

1.5 Revision 2.0

The version of the core has been updated from v3.2 to v3.3.

The INITLASTSW and EXTERNAL_BIST parameters were added to Table 4-1 • Core1553BRT_APB Parameters.

The RSTn signal was replaced by RSTINn in Table 4-3 • Control and Status Signals.

- The following signals were added:
- PURSTN
- BITINEN
- BITIN
- INTERRUPT

Table 3-5 • BIT Word was revised to update the description for the VERSION function.

The "Verification Testbench" section and "Verification Testbench Procedure and Function Calls" section were removed.

1.6 Revision 1.0

Revision 1.0 was the first publication of this document. Created for Core1553BRT_APB v3.1.

Preface

About this Document

This handbook provides details about the Core1553BRT_APB DirectCore module and how to use it.

Intended Audience

Designers using Libero® System-on-Chip (SoC) or Libero Integrated Design Environment (IDE).

1 – Introduction

General Description

Core1553BRT_APB provides a complete, dual-redundant MIL-STD-1553B remote terminal (RT), apart from the transceivers required to interface to the bus. A typical system implementation using Core1553BRT_APB is shown in Figure 1-1 and Figure 1-2 on page 4.

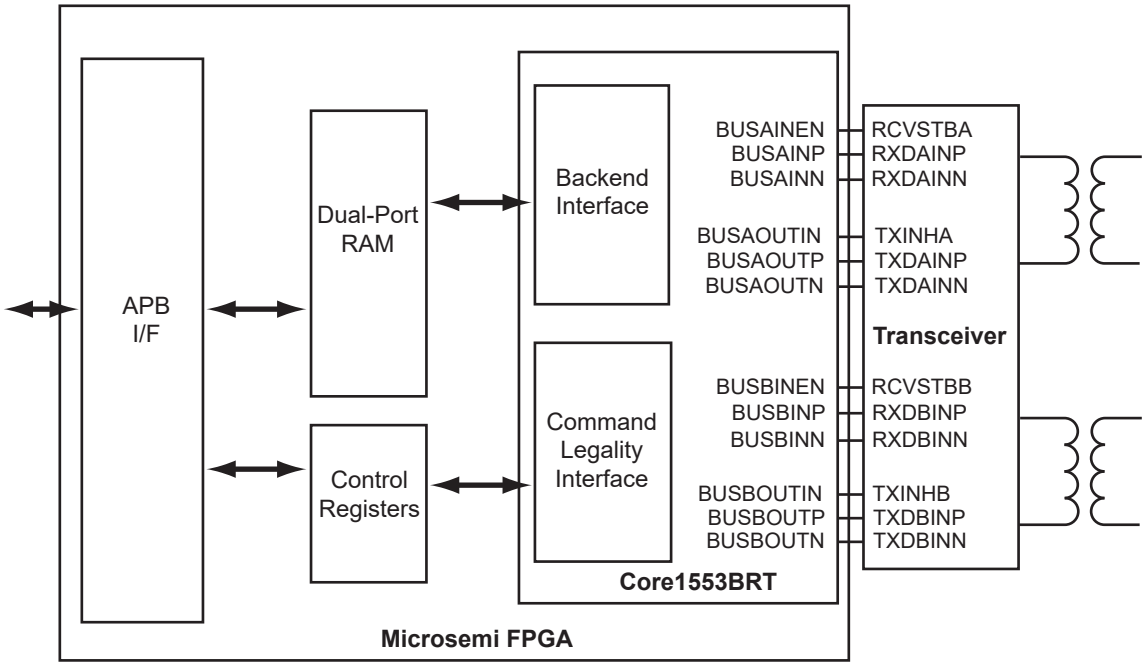

Figure 1-1 • **Typical Core1553BRT_APB System**

At a high level, Core1553BRT_APB simply provides a set of memory-mapped subaddresses that "receive data written to" or "transmit data read from." The core has an APB3 slave interface and internal memory buffer to allow simple connection to an AMBA-based processor system.

The core supports all 1553B mode codes and allows the user to designate as illegal any mode code or any particular subaddress for both transmit and receive operations. The command legalization can be done within the core or in an external command legality block via the command legalization interface.

The core consists of six main blocks: 1553B encoder, 1553B decoders, the APB interface, a command decoder, RT controller blocks, and a command legalization block (Figure 1-2 on page 4).

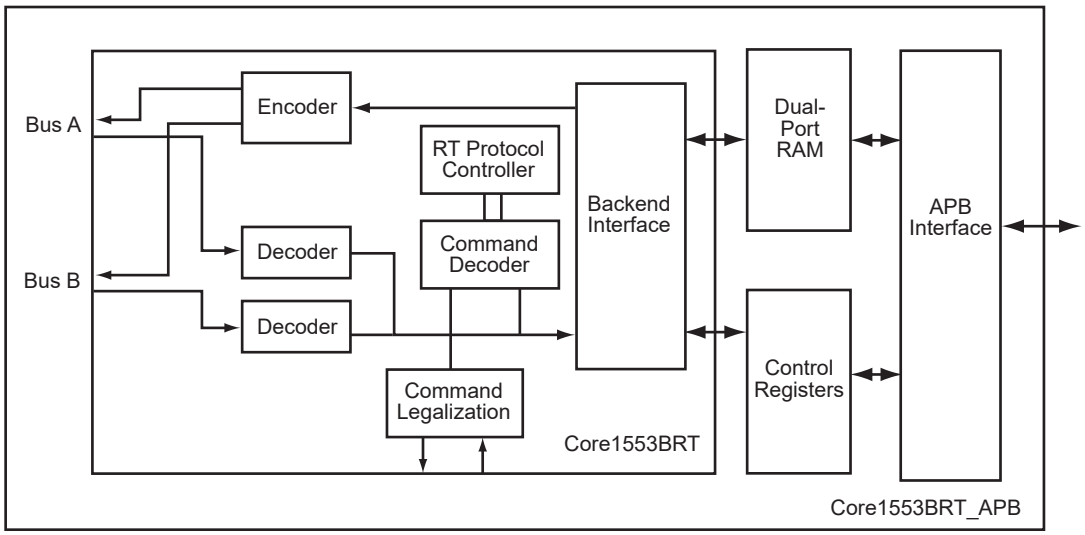

Figure 1-2 • **Remote Terminal Block Diagram**

In Core1553BRT_APB, a single 1553B encoder is used. This takes each word to be transmitted and serializes it, after which the signal is Manchester-encoded. The encoder also includes logic to prevent the RT from transmitting for longer than the allowed period, and loopback fail logic. The loopback logic monitors the received data and verifies that the core has correctly received every word that it transmits.

The output of the encoder is gated with the bus enable signals to select which buses the RT should use to transmit.

The core includes two 1553B decoders. A decoder takes the serial Manchester data received from the bus and extracts the received data words. A decoder requires a 12, 16, 20, or 24 MHz clock to extract the data and the clock from the serial stream.

The decoder contains a digital PLL that generates a recovery clock used to sample the incoming serial data. The data is then deserialized and the 16-bit word decoded. The decoder detects whether a command or data word is received and also performs Manchester encoding and parity error checking.

The APB interface for Core1553BRT_APB allows a simple connection to an APB-based processor system. The core contains an internal memory buffer to store incoming and outgoing 1553 messages. The access rates to this memory are slow, with one read or write every 20 µs. At 12 MHz operation, this is one read or write every 240 clock cycles.

The command decoder and RT controller blocks decode the incoming command words, verifying their legality. Then the protocol state machine responds to the command, transmitting or receiving data or processing a mode code.

Core1553BRT_APB has an internal command legality block that verifies every 1553B command word. A separate interface is provided that, when enabled, allows the command legality decoder to be implemented outside Core1553BRT_APB.

Verification and Compliance

Core1553BRT_APB functionality has been verified in simulation and hardware. Full functional verification against the RT test plan, as defined in MIL-HDBK-1553A, has been carried out using Core1553BRT v4.0.

Fail-Safe State Machines

The main 1553B logic implements fail-safe state machines. All state machines include illegal state detection logic. If a state machine should ever enter an illegal state, the core will assert its FSM_ERROR output and the state machine will reset. If this occurs, Microsemi® recommends that the external system reset the core and also assert the TFLAG input to inform the bus controller (BC) that a serious error has occurred within the remote terminal.

The FSM_ERROR output can be left unconnected if the system is not required to detect and report state machines entering illegal states.

Core Version

This handbook applies to Core1553BRT_APB v4.4 and later.

Version Compatibility

Core1553BRT_APB v4.4 adds a new top-level wrapper to the previous Core1553BRT release. The base VHDL and Verilog code used in the APB version for the 1553 function is identical[1] to the code in the standard Core1553BRT release. The new top-level wrapper adds an APB interface and a dual-ported memory. It also converts some of the standard core inputs to parameters, rather than the user tying the inputs to GND or VCC. SmartDesign flows allow this use of parameterized configurable RTL code. This simplifies using the core in the user's design.

1. *Two small changes, not affecting functionality, have been made. First, a hardwired constant, which was set to FALSE in the command word legality module, is now a generic defaulted to FALSE on the top level of the v4.4 core; this is to enable testing of the legality logic in the verification environment. Second, the version number encoding in the BIT word has been updated.*

Device Requirements

Core1553BRT_APB can be implemented in several Microsemi FPGA devices. Table 1-1 gives the utilization and performance figures for the core implemented in these devices.

The core can operate with a clock of up to 24 MHz. This clock rate is easily met in all silicon families noted in Table 1-1.

Table 1-1 • **Device Utilization and Performance**

Family	Combinatorial	Sequential	Total	RAM Blocks	Device	Utilization	Performance (CLK and PCLK)
IGLOO®	1,577	697	2,274	8	AGL600	16%	> 70 MHz
IGLOOe				8	AGLE600		
ProASIC®3				8	A3P600		
ProASIC3E				8	A3PE600		
SmartFusion®					A2F200	50%	
Fusion®					AFS600	16%	> 70 MHz
ProASICPLUS®	2,314	731	3,045	16	APA450	25%	> 50 MHz
Axcelerator®	1,039	728	1,767	XX	AX500	21%	> 70 MHz
RTAX-S				XX	RTAX250S	42%	
SmartFusion®2	1272	792	2064	2	M2S050	1.8%	118 MHz
IGLOO®2	1271	788	2059	2	M2GL050	1.82%	120 MHz
RTG4™	1350	790	2140	2	RT4G150	0.7%	118 MHz
PolarFire®	1280	788	2068	2	MPF100T	1%	185 MHz

Utilization data was generated using standard Libero tool flows with typical core parameter settings. Utilization data will vary slightly with different parameter settings and tool usage.

Note: FPGA resources and performance data for the PolarFire SoC family is similar to PolarFire family.

2 – MIL-STD-1553B Bus Overview

The MIL-STD-1553B bus is a differential serial bus used in military and space equipment. It comprises multiple redundant bus connections and communicates at 1 Mbps.

The bus has a single active BC and up to 31 RTs. The BC manages all data transfers on the bus using the command and status protocol. The bus controller initiates every transfer by sending a command word and data if required. The selected RT will respond with a status word and data if required.

The 1553B command word contains a 5-bit RT address, transmit or receive bit, 5-bit subaddress, and 5-bit word count. This allows for 32 RTs on the bus. However, since RT address 31 is used to indicate a broadcast transfer, only 31 RTs can be connected. Each RT has 30 subaddresses reserved for data transfers. The other two subaddresses (0 and 31) are reserved for mode codes used for bus control functions. Data transfers contain up to thirty-two 16-bit data words. Mode code command words are used for bus control functions such as synchronization.

Message Types

The 1553B bus supports 10 message transfer types, allowing basic point-to-point and broadcast BC-to-RT data transfers, mode code messages, and direct RT-to-RT messages. Figure 2-1 on page 8 shows the message formats.

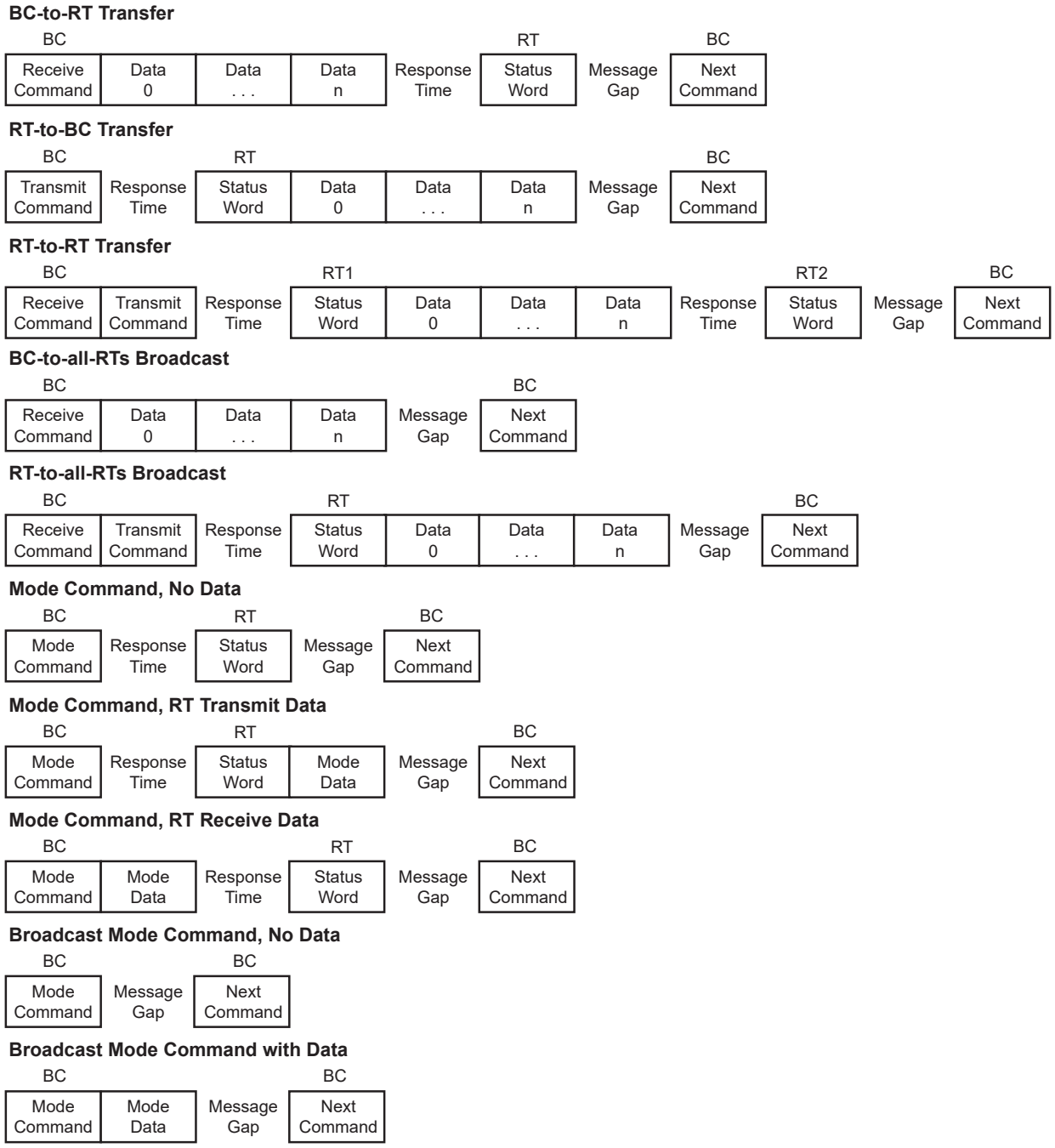

Figure 2-1 • **1553B Message Formats**

Word Formats

There are only three types of words in a 1553B message: a command word (CW), a data word (DW), and a status word (SW). Each word consists of a 3-bit sync pattern, 16 bits of data, and a parity bit, providing the 20-bit word (Figure 2-2).

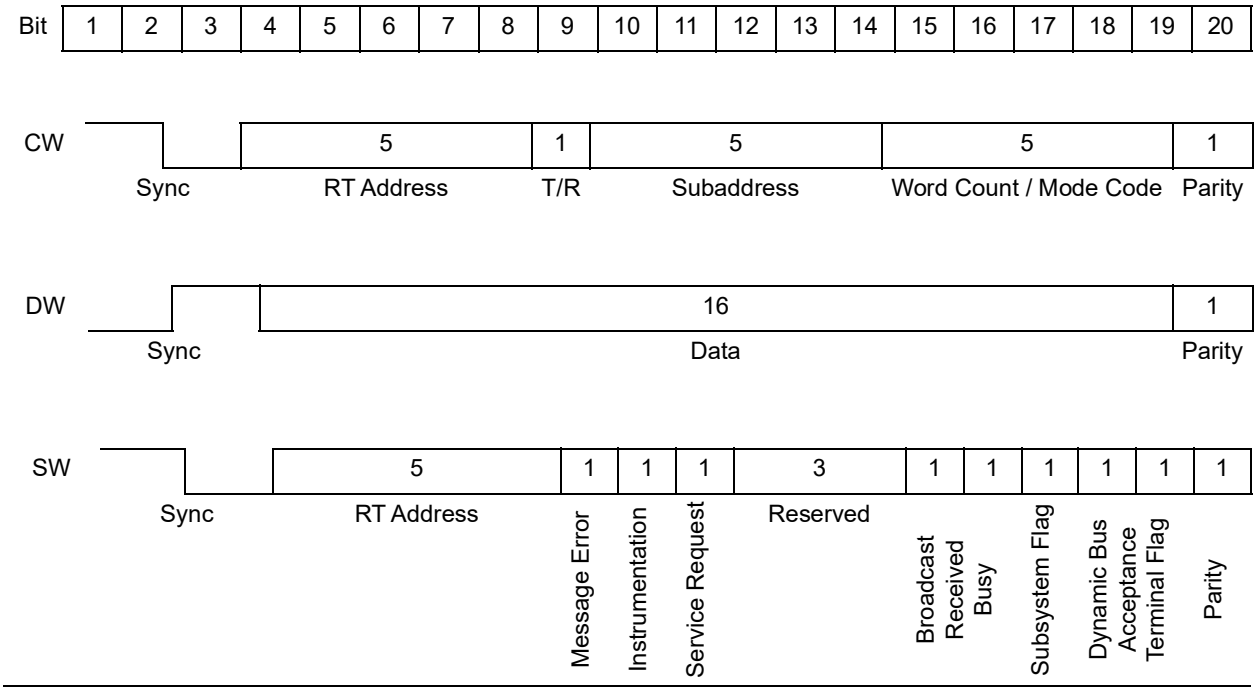

Figure 2-2 • **1553B Word Formats**

3 – Operation

Status Word Settings

Core1553BRT_APB sets bits in the 1553B status word in compliance with MIL-STD-1553B. This is summarized in Table 3-1.

***Table 3-1* • Status Word Bit Settings**

Bit(s)	Function	Setting
15:11	RT Address	Equals the set RT address.
10	Message Error	Set whenever the RT detects a message error.
9	Instrumentation	Always 0
8	Service Request	Controlled by Control register bit 0.
7:5	Reserved	Always 000.
4	Broadcast Received	Set whenever a broadcast message is received.
3	Busy	Controlled by Control register bit 1.
2	Subsystem Flag	Controlled by Control register bit 2.
1	Dynamic Bus Acceptance	Always 0. Core1553BRT_APB does not operate as a bus controller.
0	Terminal Flag	Controlled by Control register bit 3. If an “inhibit terminal flag” mode code is in effect, will be 0.

Transfer Status Words

At the end of every 1553B bus transfer, a transfer status word is written to the RAM in locations 000–76C for receive operations and F80–FFC for transmit operations. The address used is as follows:

- TSW location, RX commands: '000000' and SA and '00'
- TSW location, TX commands: '011111' and SA and '00'

As an example, the TSW address for a transmit command with subaddress 24 would be '0111111010000' (FD0h). The TSW contains the information in Table 3-2.

***Table 3-2* • Transfer Status Word**

Bit(s)	Name	Description	
15	USED	Set to 1 at the end of the transmit or receive command.	
14	OKAY	Indicates that no errors are detected; i.e., bits 11 to 5 are all 0.	
13	BUSN	Indicates on which bus the command was received: 0: BUSA 1: BUSB	
12	BROADCAST	Indicates a broadcast command.	
11	LPBKERRB	Indicates that the loopback logic detected an error in the transmitted data for bus B.	
10	LPBKERRA	Indicates that the loopback logic detected an error in the transmitted data for bus A.	
9	ILLEGAL CMD	The command was illegal. A request to transmit from either an illegal subaddress or an illegal mode code was received.	
8	MEMIFERR	Indicates that the DMA memory access failed to complete quickly enough.	
7	MANERR	Indicates that a Manchester encoding error was detected in the incoming data.	
6	PARERR	Indicates that a parity error was detected in the incoming data.	
5	WCNTERR	Indicates that an incorrect number of words was received.	
4:0	COUNT	SA1 to SA31	Indicates the number of words received or transmitted for that subaddress. If WCNTERR is 0, '00000' indicates 32 words. Otherwise, '00000' indicates zero words transferred.
		SA0 or SA31	Indicates which mode code was received or transmitted per the 1553B specification.

RT-to-RT Transfer Support

The core supports RT-to-RT transfers. If a transmitting core does not start transferring data within the required time, the core will detect this and set the WCNTERR bit in the transfer status word.

Mode Codes

When the core receives a mode code, it first checks its command validity. If the command is valid, it is processed in accordance with the specification. Otherwise, the message error bit will be set in the 1553B status word. Table 3-3 lists the supported mode codes.

Two mode codes, (1) transmit a vector word and (2) synchronize with data, require external data. The vector word value is set by the VWORD input, and the synchronize with data word is discarded.

***Table 3-3* • Supported Mode Codes**

T/R Bit	Mode Code	Function and Effect	Data Word	Core Supports	Broadcast Allowed
1	00000 0	Dynamic Bus Control The core does not support bus controller functions, so it will set the Message Error and Dynamic Bus Control bits in the status word.	No	No	No
1	00001 1	Synchronize The core will assert its SYNCNOW output after the command word has been received.	No	Yes	Yes
1	00010 2	Transmit Status Word The core retransmits the last status word.	No	Yes	No
1	00011 3	Initiate Self-Test The core does not support self-test. Since the core supports the transmit BIT word mode code, this command is treated as legal and will not set Message Error.	No	Yes	Yes
1	00100 4	Transmitter Shutdown The core will disable the encoder on the other bus.	No	Yes	Yes
1	00101 5	Override Shutdown The core will re-enable the encoder on the other bus.	No	Yes	Yes
1	00110 6	Inhibit Terminal Flag The core will mask the TFLAG input, and the Terminal Flag bit in the status word will be forced to zero.	No	Yes	Yes
1	00111 7	Override Inhibit Terminal Flag The core will re-enable the TFLAG input.	No	Yes	Yes
1	01000 8	Reset Remote Terminal The core will assert its BUSRESET output after the command word has been received. It will also reset itself.	No	Yes	Yes
1	10000 16	Transmit Vector Word The core will transmit a single data word that contains the value on the VWORD input.	Yes	Yes	No
1	10010 18	Transmit Last Command Word The core will transmit a single data word that contains the last command word received.	Yes	Yes	No
1	10011 19	Transmit BIT Word The core will transmit a single data word that contains the extended core status information. The value of this word is defined in Table 3-5 on page 15.	Yes	Yes	No
0	10001 17	Synchronize with Data The core will assert its SYNCNOW output after the data word has been received.	Yes	Yes	Yes

***Table 3-3* • Supported Mode Codes (continued)**

T/R Bit	Mode Code	Function and Effect	Data Word	Core Supports	Broadcast Allowed
0	10100 20	Selected Transmitter Shutdown The core only supports two buses. Hence, this command is illegal. The Message Error bit in the status word will be set.	Yes	No	Yes
0	10101 21	Override Selected Transmitter Shutdown The core only supports two buses. Hence, this command is illegal. The Message Error bit in the status word will be set.	Yes	No	Yes

Loopback Tests

Core1553BRT_APB performs loopback testing on all of its transmissions. The transmit data is fed back into the receiver, and each transmitted word is compared. If an error is detected, the loopback fail bit is set in the TSW and also in the BIT word.

Error Detection

Table 3-4 • **Error Detection**

Error Condition	Action
Command Word 1. Parity or Manchester encoding errors 2. Incorrect SYNC waveform	Command is ignored. No interrupt generated.
Mode Codes 1. Illegal mode code or invalid subaddress (from internal or external legality block)	MSGERR in SW is set, and the SW is transmitted. Message Failure interrupt generated.
Broadcast Data Commands 1. TX bit set in command word	Data transfer is aborted. MSGERR in SW is set, and the SW is not transmitted. Message Failure interrupt generated.
Data Word 1. Parity or Manchester encoding errors 2. Incorrect number of words received 3. Data words are not continuous/contiguous. 4. Incorrect SYNC waveform	Data transfer is aborted. MSGERR in SW is set, and the SW is not transmitted. Message Failure interrupt generated.
RT-to-RT 1. First command word must be RX. 2. Second command word must be TX and non-broadcast. 3. RX RT checks the TX SW and verifies the SYNC pattern, RT address, MSGERR, and BUSY fields. 4. The first data word sync must be received within 57 µs of the command word parity bit.	Data transfer is aborted. MSGERR in SW is set, and the SW is not transmitted. Message Failure interrupt generated.
Transmit Data Error 1. The RT monitors its transmissions on the bus through its decoder and verifies that the correct data is transmitted with no Manchester or parity errors.	Data transfer is aborted. MSGERR in SW is set, and the SW is not transmitted. Message Failure interrupt generated.
Backend Failure 2. The RT makes sure that the backend responds to read and write cycles within the required time.	Data transfer is aborted. MSGERR in SW is set, and the SW is not transmitted. Message Failure interrupt generated.
BUSY 1. Backend RTBUSY input is active at any point during the message.	Data transfer is aborted. BUSY in SW is set, and the SW is transmitted. Message Failure interrupt generated.
Transmitter Overrun 1. Transmits for greater than 668 µs. The internal state machines prevent this from happening, but the core includes the required timer and functionality. This is implemented separately to the encoder to provide complete protection.	Core shuts down transmissions on bus.

Built-In Test Support

Core1553BRT_APB provides a BIT word. This is used to communicate fail information back to the bus controller. The BIT word contains the information in Table 3-5.

Table 3-5 • **BIT Word**

Bit(s)	Function	Description
15	BUSINUSE	Indicates on which bus the transmit BIT word command was received: 0: Bus A 1: Bus B
14	LPBKERRB	Indicates that the loopback logic detected an error on the transmitted data for bus B. This bit is cleared by the CLRERR input.
13	LPBKERRA	Indicates that the loopback logic detected an error on the transmitted data for bus A. This bit is cleared by the CLRERR input.
12	SHUTDOWNB	Indicates that bus B is shut down. This occurs after a transmitter shutdown mode code is received or the hardware timer detected that the core transmitted for longer than 668 µs on bus B.
11	SHUTDOWNA	Indicates that bus A is shut down. This occurs after a transmitter shutdown mode code is received or the hardware timer detected that the core transmitted for longer than 668 µs on bus A.
10	TFLAGINH	Terminal flag inhibit setting
9	WCNTERR	A word count error has occurred. This bit is cleared by the CLRERR input.
8	MANERR	A Manchester encoding error has occurred. This bit is cleared by the CLRERR input.
7	PARERR	A parity error has occurred. This bit is cleared by the CLRERR input.
6	RTRTTO	The transmitting RT did not provide data on RT-to-RT transfer. This bit is cleared by the CLRERR input.
5	MEMFAIL	The backend memory interface failed to complete an access within the required time. This bit is cleared in the CLRERR input.
4:0	VERSION	Indicates the core version: '01110': Version 4.4.

Command Legalization Interface

1553B commands can be legalized in three ways with Core1553BRT_APB. For RTL versions, one of the modules in the source code can be edited to legalize or make illegal command words based on the subaddress, mode code, word count, or broadcast fields of the command word. For Obfuscated and RTL versions, external logic can be used to decode the legal/illegal command words (Figure 3-1). Also, the APB legalization registers can be used to allow software legalization.

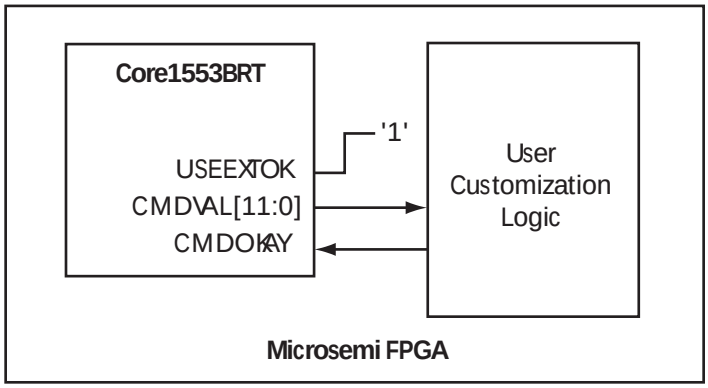

***Figure 3-1* • Command Legalization Logic**

The user customization logic block takes in CMDVAL and simply sets CMDOKAY for all legal command words. The CMDVAL encoding is given in Table 3-6. The external logic must implement this function within 3 µs.

***Table 3-6* • CMDVAL Encoding**

Bit(s)	Function	Description
11	Broadcast	'1' indicates broadcast; i.e., the RT address was set to 31 in the 1553B command word.
10	Transmit or Receive	TX/RX field from the 1553B command word. '0' indicates receive and '1' transmit.
9:5	Subaddress	Subaddress field from the 1553B command word
4:0	Word Count Mode Code	Word count field from the 1553B command word. When the subaddress is 0 or 31, this contains the 1553B mode code.

4 – Interface Descriptions

Parameters on Core1553BRT_APB

The parameters given in Table 4-1 are used to configure the core.

Table 4-1 • **Core1553BRT_APB Parameters**

Parameter	Range	Description
FAMILY	2 to 27	Must be set to the required FPGA family: 11: Axcelerator 12: RTAX-S 14: ProASIC^{PLUS} 15: ProASIC3 16: ProASIC3E 17: Fusion 18: SmartFusion 19: SmartFusion2 20: IGLOO 21: IGLOOe 24: IGLOO2 25: RTG4 26: PolarFire 27: PolarFire SoC
CLKSPD	12, 16, 20, or 24	Sets the clock (named CLK) frequency of the 1553B portion of the core to 12, 16, 20, or 24 MHz.
CLKSYNC	0 or 1	When the CLK and PCLK inputs are asynchronous, this should be 0. When CLK and PCLK are synchronous and driven from the same source, this should be set to 1. Setting this to 1 will remove additional internal clock synchronization logic, lowering tile counts.
BCASTEN	0 or 1	This input enables broadcast operation. When 1, broadcast operations are enabled. When 0, broadcast messages (i.e., RT address 31) are treated as normal messages. If the RTADDR input is set to 31, the RT will respond to the message.
SA30LOOP	0 or 1	This input alters the backend memory mapping so that subaddress 30 provides automatic loopback. When 0, the RT does not loop back subaddress 30. Separate memory buffers are used for transmit and receive data buffers. When 1, the RT maps the transmit memory buffer for subaddress 30 to the receive memory buffer for subaddress 30; i.e., the upper address line is forced to 0.
INTENBBR	0 or 1	When active (1), the core generates interrupts when both good and bad 1553B messages are received. When inactive (0), the core only generates interrupts when good messages are received.

Table 4-1 • **Core1553BRT_APB Parameters (continued)**

Parameter	Range	Description
TESTTXTOUTEN	0 or 1	This enables the TESTTXTOUT input; it is for test use only. This parameter should be set to 0 if it is not required to be able to force transmission overrun for testing the internal transmit timer.
ECC	0 or 1	When 1, enables the ECC operation in memory. When 0, ECC is not enabled for memory. Option is available for RTG4 family.
LOCKRT	0 or 1	When set to 1, the RT address is set by the RTADDR input and cannot be overridden by the internal registers.
INT_POLARITY	0 or 1	When 0, the INTERRUPT output is active low; when 1, active high.
LEGMODE	0 to 3	This controls the RT subaddress and mode code legalization logic: 0: (Internal) Core uses the internal legalization logic inside the 1553B RT core. 1: (External) Core uses the external legalization interface, CMDVAL output, and CMDOK input. 2: (APB registers) Legislation is controlled via software using the Legalization registers, implemented using registers. This option will increase logic elements by approximately 750 tiles. 3: (APB registers, RAM-based) Legislation is controlled via software using the Legalization registers, implemented using memory.
INITLASTSW	0 or 1	This input controls the last status word. When 0, the first received command is a transmit last status word. The core will respond with an undefined status word since no status word has previously been sent (same function as previous core versions). When 1, the first received command is a transmit last status word. The core will respond with a valid RT address and the all other status bits zero even though no status word was previously sent. It requires PURSTN to be asserted at power-up.
EXTERNAL_BIST	0 or 1	This parameter controls the mode code 19 support. When 0, the internal BIST value is returned in response to the transmit BIST mode code as shown in Table 4-3 on page 20. When 1, the input BITIN [15:0] is returned in response to the transmit BIST mode code. The default value of EXTERNAL_BIST is 0.

I/O Signal Descriptions

Table 4-2 and Table 4-3 on page 20 describe the I/O signals for the bus interface, control, and status.

Table 4-2 • **1553B Bus Interface**

Port Name	Type	Description
RTADDR[4:0]	In	Sets the RT address; must not be set to '11111'.
RTADDRP	In	RT address parity input. This input should be set High or Low to achieve odd parity on the RTADDR and RTADDRP inputs. If RTADDR is '00000', the RTADDRP input should be 1.
RTADERR	Out	Indicates that the RTADDR and RTADDRP inputs have incorrect parity, or broadcast is enabled, and the RT address is set to 31. When active (High), the RT is disabled and will ignore all 1553B traffic.
BUSAINEN	Out	Active high output that enables for the A receiver
BUSAINP	In	Positive data input from the A receiver
BUSAINN	In	Negative data input from the A receiver

***Table 4-2* • 1553B Bus Interface (continued)**

Port Name	Type	Description
BUSBINEN	Out	Active high output that enables for the B receiver
BUSBINP	In	Positive data input from the bus to the B receiver
BUSBINN	In	Negative data input from the bus to the B receiver
BUSAOUTIN	Out	Active high transmitter inhibit for the A transmitter
BUSAOUTP	Out	Positive data output to the bus A transmitter (held High when no transmission)
BUSAOUTN	Out	Negative data output to the bus A transmitter (held High when no transmission)
BUSBOUTIN	Out	Active high transmitter inhibits the B transmitter
BUSBOUTP	Out	Positive data output to the bus B transmitter (held High when no transmission)
BUSBOUTN	Out	Negative data output to the bus B transmitter (held High when no transmission)

Table 4-3 • **Control and Status Signals**

Port Name	Type	Description
CLK	In	Master 1553B clock input (12, 16, 20, or 24 MHz)
RSTINn	In	Reset input asynchronous (active low)
BUSY	Out	Indicates that the 1553BRT is either receiving or transmitting data or handling a mode command
SYNCNOW	Out	Pulses High for a single clock cycle when the RT receives a synchronize (with or without data mode) command. The pulse occurs just after the 1553B command word (sync with no data) or data word (sync with data mode code) has been received.
BUSRESET	Out	Pulses high for a single clock cycle whenever the RT receives a reset mode command. The core logic will also automatically reset itself on receipt of this command.
FSM_ERROR	Out	This output will go High for a single clock cycle if any of the internal state machines enter an illegal state. This output should not go High in normal operation. Should it go High, it is recommended that the core be reset.
PURSTN	In	Asynchronous active low power-up reset.
BITINEN	In	Transmit bit word enable signal. Valid when EXTERNAL_BIST = 1.
BITIN	In	16-bit signal for transmit bit word. Valid when EXTERNAL_BIST = 1.
INTERRUPT	Out	Core interrupt output signal.
SB_CORRECT	Out	1-bit error correction flag indicator, enabled when ECC is selected for RTG4.
DB_DETECT	Out	2-bit error detection flag indicator, enabled when ECC is selected for RTG4

Note: All control inputs are synchronous and sampled on the rising edge of the clock. All status outputs are synchronous to the rising edge of the clock.

Command Legalization Interface

The core checks the validity of all 1553B command words. In RTL and Obfuscated versions of the core, the logic may be implemented externally to the core. The command word is provided, and the logic must generate the command-valid input. The command legalization interface also provides two strobes that are used to latch the command value to enable it to be used for address mapping and interrupt vector extension functions (Table 4-4).

Table 4-4 • **Command Legalization Interface**

Port Name	Type	Description
CMDVAL[11:0]	Out	**Active**Command 11:0: Non-broadcast 1: Broadcast 10:0: Receive 1: Transmit 9:5:Subaddress 4:0:Word count / mode code These outputs are valid throughout the complete 1553B message. They can also be used to steer data to particular backend devices. In particular, bit 11 allows non-broadcast and broadcast messages to be differentiated, as required by MIL-STD-1553B, Notice 2.
CMDSTB	Out	Single-clock-cycle pulse that indicates CMDVAL has changed
CMDOKAY	In	Command word is okay (active high). The external logic must set this within 2 µs from the CMDVAL output changing.

APB Interface

The APB interface supports both synchronous operation (to the core clock) and asynchronous operation (Table 4-5).

Table 4-5 • **APB Signals**

Port Name	Type	Description
PCLK	In	APB clock
PRESETN	In	APB reset (active low and asynchronous)
PSEL	In	APB select
PENABLE	In	APB enable
PWRITE	In	APB write
PADDR	In [12:0]	APB address
PWDATA	In [15:0]	APB write data
PRDATA	Out [15:0]	APB read data

5 – Interface Timing

Specifications

APB Interface Timing

Figure 5-1 and Figure 5-2 depict typical write cycle and read cycle timing relationships relative to the APB system clock, PCLK.

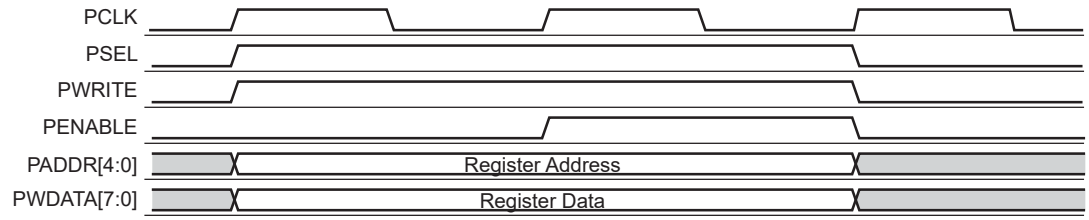

Figure 5-1 • **Data Write Cycle**

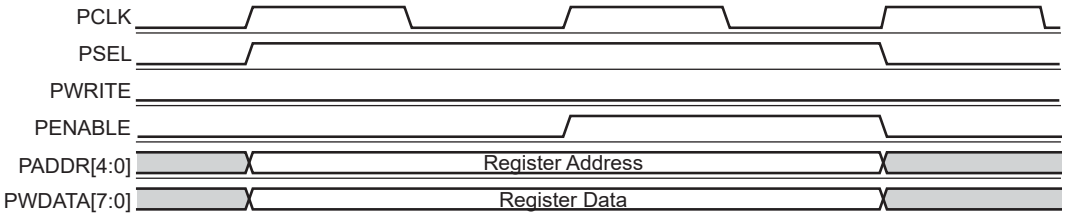

Figure 5-2 • **Data Read Cycle**

Command Word Legality Interface Timing

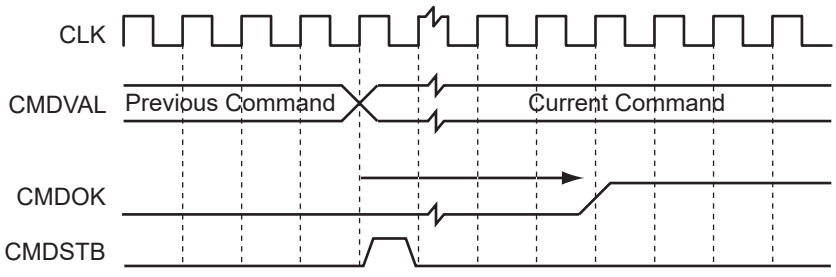

Figure 5-3 • **Command Word Legality Interface Timing**

Table 5-1 • **Command Word Legality Interface Timing**

Name	Description	Time
$T_{pdCMDOK}$	Maximum external command word legality decode delay	3 µs

RT Response Times

RT response time is from the midpoint of the parity bit in the command word to the midpoint of the status word sync (Table 5-2).

Table 5-2 • **RT Response Times**

Spec	Description	At 12 MHz	At 16 MHz	At 20 MHz	At 24 MHz
T_{rtresp}	RT response time	4.75 to 7.0 µs	4.75 to 7.0 µs	4.75 to 7.0 µs	4.75 to 7.0 µs
T_{rtrtto}	RT-to-RT timeout	57 µs	57 µs	57 µs	57 µs
T_{xxto}	Transmitter timeout	704 µs	668 µs	691 µs	693 µs

The RT-to-RT timeout is from the first command word parity bit to the expected sync of the first data word.

Transceiver Loopback Delays

Core1553BRT_APB verifies that all transmitted data words are correctly transmitted. As data is transmitted by the transceiver on the 1553B bus, the data on the bus is monitored by the transceiver and decoded by Core1553BRT_APB. The core requires that the loopback delay—the time from BUSAOUTP to BUSAINP—be less than the values given in the Table 5-3.

Table 5-3 • **Transceiver Loopback Requirements**

Clock Speed	Maximum Loopback Delay
12 MHz	2.3 µs
16 MHz	2.3 µs
20 MHz	2.3 µs
24 MHz	2.3 µs

The loopback delay is a function of the internal FPGA delay, PCB routing delays, and internal transceiver delay as well as transmission effects from the 1553B bus. Additional register stages can be inserted on either the 1553B data input or output within the FPGA, providing the loopback delays in Table 5-3 on page 23 are not violated. This is recommended if additional gating logic is inserted inside the FPGA between the core and transceiver to minimize skew between the differential inputs and outputs.

Clock Requirements

To meet the 1553B transmission bit rate requirements, the Core1553BRT_APB clock input (CLK) must be 12 MHz, 16 MHz, 20 MHz, or 24 MHz ± 0.01%.

6 – Tool Flows

Licensing

Core1553BRT_APB is licensed in three ways. Depending on your license tool flow, functionality may be limited.

Evaluation

Pre-compiled simulation models are provided, allowing the core to be instantiated with SmartDesign. Simulation can be performed within Libero.

Obfuscated

Complete RTL code is provided for the core, allowing the core to be instantiated with SmartDesign. Simulation, Synthesis, and Layout can be performed within Libero. The RTL code for the core is obfuscated[1] and some of the testbench source files are not provided; they are precompiled into the compiled simulation library instead.

RTL

Complete RTL source code is provided for the core and testbenches.

Precompiled Libraries

Core1553BRT_APB supports the following precompiled libraries:

- **IDE**
  - Precompiled libraries are built with ModelSim 10.2c for v4.4
- **SoC**
  - Precompiled libraries are built with ModelSim 10.7c for v4.4

1. *Obfuscated means the RTL source files have had formatting and comments removed, and all instance and net names have been replaced with random character sequences.*

SmartDesign

Core1553BRT_APB is preinstalled in the SmartDesign IP deployment design environment.

The core can be configured using the configuration GUI within SmartDesign, as shown in Figure 6-2.

For more information on using SmartDesign to instantiate and generate cores, refer to *Libero SoC documents page* and use the latest SmartDesign user guide..

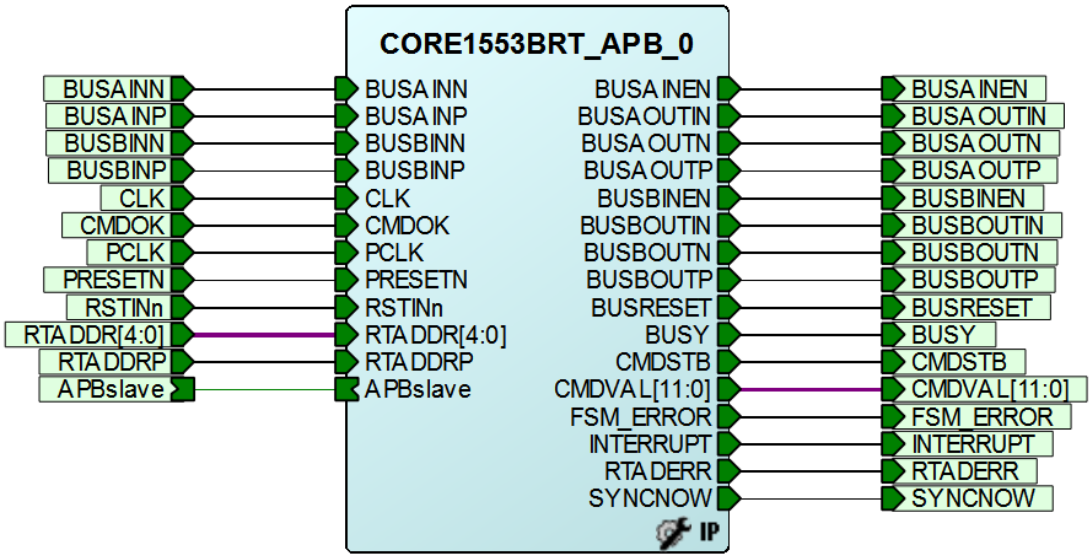

Figure 6-1 • **Core1553BRT_APB Full I/O View**

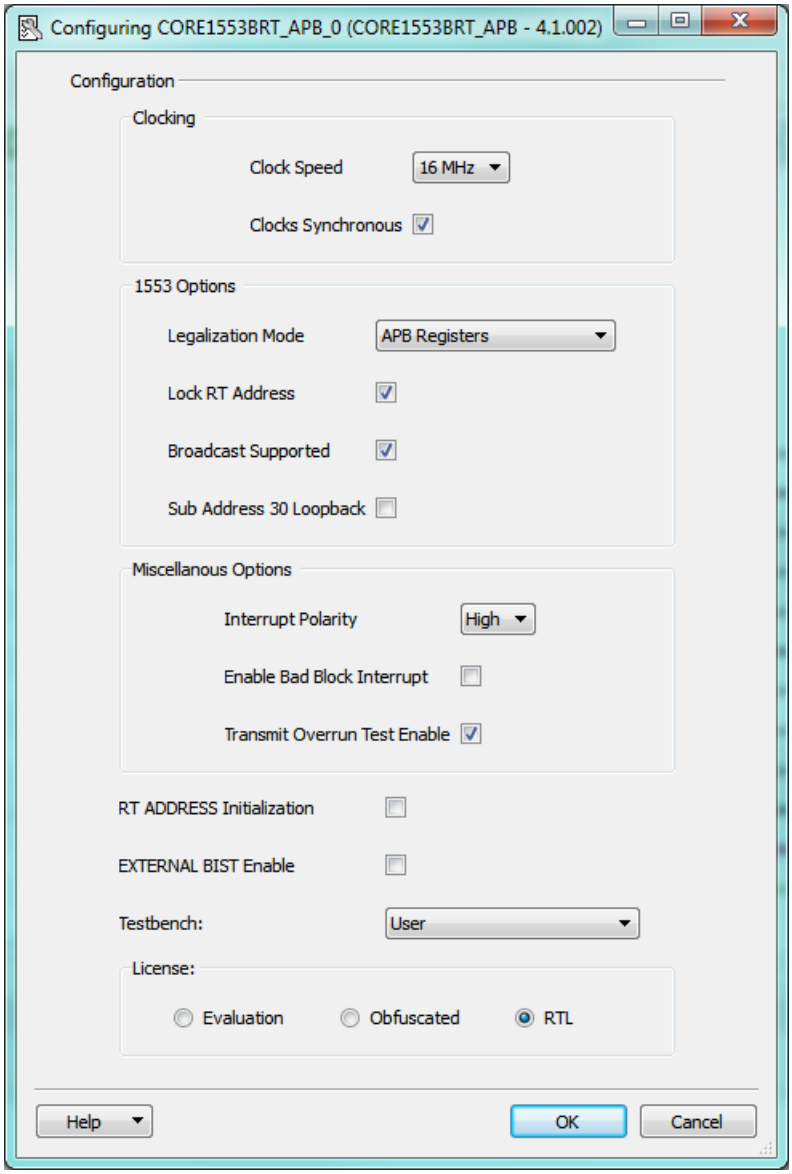

Figure 6-2 • **Core1553BRT_APB SmartDesign Configuration**

Simulation Flows

The User Testbench for Core1553BRT_APB is included in all releases.

To run simulations, select the User Testbench flow within SmartDesign and click **Save & Generate** on the Generate pane. The User Testbench is selected through the Core Testbench Configuration GUI.

When SmartDesign generates the Libero project, it will install the user testbench files.

To run the user testbench, set the design root to the **Core1553BRT_APB instantiation i**n the Libero design hierarchy pane and click the **Simulation** icon in the Libero Design Flow window. This will invoke ModelSim® and automatically run the simulation.

Synthesis in Libero

After setting the design root appropriately, click the **Synthesis** icon in Libero. The Synthesis window appears, displaying the Synplicity® project. Set Synplicity to use the Verilog 2001 standard if Verilog is being used or VHDL 2008 if VHDL. To run Synthesis, select the **Run** icon.

Place-and-Route in Libero

After setting the design root appropriately and run Synthesis, click the **Layout** icon in the Libero to invoke Designer. Core1553BRT_APB requires no special place-and-route settings.

7 – Testbench Operation and Modification

User Testbench

The user testbench (Figure 7-1) is intended to be used as a starting point for the verification of the core when used within the customer application. It consists of an APB master bus functional model (BFM) that creates APB bus cycles, a 1553B bus controller to generate 1553B command sequences, and multiple Core1553BRT_APB blocks. Having multiple RTs allows RT to RT messages to be tested.

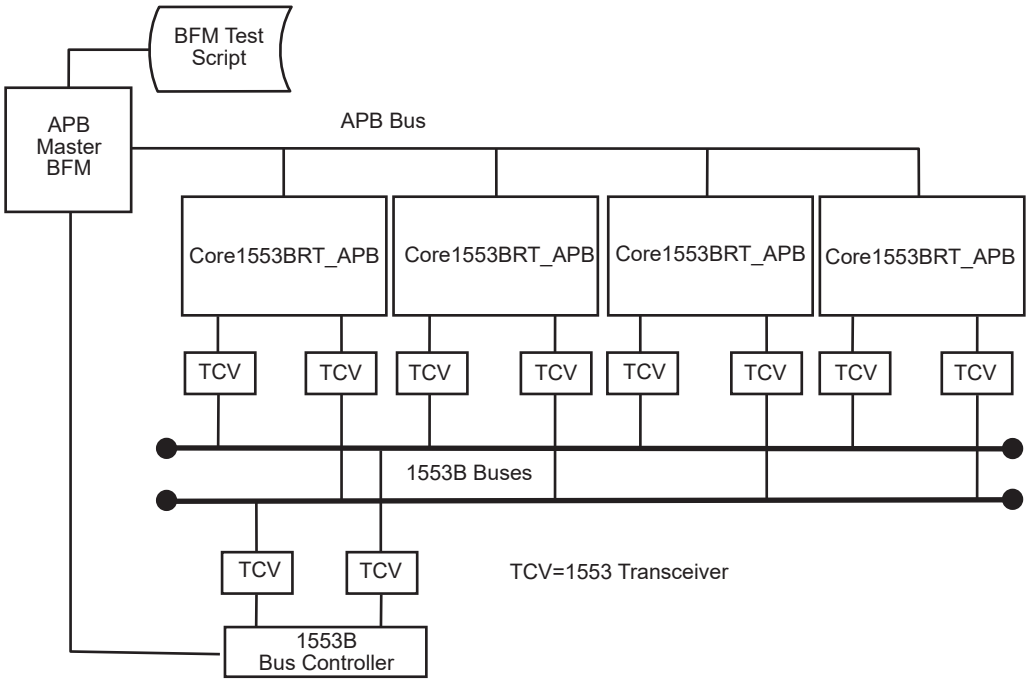

Figure 7-1 • **User Testbench**

The APB master BFM generates APB cycles based on command file core1553brt.bfm. The syntax of these commands is documented in the *DirectCore AMBA BFM User's Guide*.

The 1553B bus controller is connected to AMBA APB BFM via its external interface. The BFM script can initiate 1553B messages by writing to registers within the bus controller model. The register set is described in the"Bus Controller Model" section on page 29.

The four Core1553BRT_APB cores are connected via transceiver models to 1553B buses and to the bus controller.

Bus Controller Model

The bus controller model is configured though an internal 4K memory array. The contents are changed by using the BFM *extwrite* and *extread* commands. The APB BFM creates a descriptor list and then writes the address of the starting descriptor to the descriptor pointer (at 0x0001); then starts the bus controller by writing to the command/status register (at 0x0000).

***Table 7-1* • Bus Controller Register Map**

Address	Registers	Function (Bit)
0x0000	Command/Status	Bit 0: Writing a '1' starts the bus controller Bit 1: 0: Single 1553B Command 1: Multiple 1553B Commands
0x0001	Descriptor Pointer	Pointer to first descriptor address

***Table 7-2* • Command/Status Bit Definitions**

Descriptor Offset	Function	Description
0	BC Command	Bits 3:0 0x0= Stop 0x1 = Do message and stop 0x2 = Do message and continue at descriptor address plus 8 Bit 4: RT to RT message Bit 5: Bus 0 = A; 1 = B
1	Command Word 1	Normal CW or RTRT RX CW
2	Command Word 2	RTRT TX CW
3	Status Word 1	Normal SW or RTRT TX SW
4	Status Word 1	RTRT RX SW
5	Data	Data pointer or mode code data Points to address containing data to be transmitted or received, or storage location for mode code data.
6	BC Status	Bits 7:0 = Number data words received Bit 8: Done Bit 9: Error Detected Bit 10: Got SW 1 Bit 11: Got SW 2
7	Reserved	

The following BFM scriptlet creates two descriptors at addresses 0x0100 and 0x0108. The first message is a BC to RT message and the second is an RT to BC message.

```
# Addr  Control    CW      CW2     SW     SW2   DP   Status Reserved
  extwrite 0x0100 0x0002  1.0.1.4  0.0.0.0 0x0000 0x0000 0x0200 0       0   # BC to RT
  extwrite 0x0108 0x0002  1.1.1.3  0.0.0.0 0x0000 0x0000 0x0300 0       0   # RT to BC
  extwrite 0x0110 0x0000   # Stop
  extwrite 0x0200  1 2 3 4 5 6 7 8 9 10        # Test Data to transmit
  extwrite 0x0001 0x0100                          # Point at first descriptor
  extwrite 0x0000 0x0003                          # Start the bus controller
  extwait                                                  # BFM script will
pause to bus controller completes
  # Check returned data
  extcheck 0x0300 0x1000                          # The RT was expected to send back
data 0x1000 etc
  extcheck 0x0301 0x1001
  extcheck 0x0302 0x1002
```

8 – Software Interface

Memory Map

The 8k words of APB address space are used to address the memory buffer and the control registers; the address map is DWORD-aligned (Table 8-1). The control registers overlay the top 32 DWORD locations.

***Table 8-1* • APB Address Map**

APB Address	RAM Contents (16-bit-wide data)	Action
0x0000	RX transfer status words	The core only writes to these addresses (except when SA30LOOP is HIGH).
0x0080	RX subaddress 1	
0x0100	RX subaddress 2	
0x0180	RX subaddress 3	
…	…	
0x0E00	RX subaddress 28	
0x0E80	RX subaddress 29	
0x0F00	RX subaddress 30	
0x0F80	TX transfer status words	
0x1000	Not used	The core only reads from these addresses.
0x1080	TX subaddress 1	
0x1100	TX subaddress 2	
0x1180	TX subaddress 3	
…	…	
0x1E00	TX subaddress 28	
0x1E80	TX subaddress 29	
0x1F00	TX subaddress 30	
0x1F80	Registers	Control and Status registers

If the SA30LOOP parameter is set, the RT maps transmit subaddress 30 to receive subaddress 30; that is, the upper address bit is forced to 0. This provides a loopback subaddress, as per MIL-STD-1553B, Notice 2.

The TSW is still written to address 0x0F78. It should be noted that this is not strictly compliant with the specification, since the transmit buffer will contain invalid data if the received command fails; e.g., with a parity error. The transmit buffer should only be updated if the receive command had no errors. To implement this function in full compliance, the SA30LOOP parameter should be set to zero, and the RT backend system processor should copy the receive memory buffer to the transmit memory buffer only after the RT signals that the message was received with no errors.

When using ProASIC3 devices, the memory buffer is implemented within the FPGA using dual-port RAMs (two write and two read ports); in this case, the complete buffer is readable and writable by the APB.

When using ProASIC^{PLUS}, Axcelerator, and RTAX-S devices, the memory buffer is implemented within the FPGA using two-port RAMs (one read and one write port); in this case, separate receive and transmit RAM blocks are used (each as 1k words). The APB can only read locations 0x0000–0x0FFF and write to locations 0x1000–0x1FFF. The SA30LOOP parameter must be zero in this case.

Register Set

There are twenty 16-bit registers used for controlling the core; these are DWORD-address-aligned (Table 8-2). The registers overlap the memory address space, and addresses 0x1F80–0x1FBF actually access internal registers rather than the memory block. When LEGMODE = 3, the last 16 memory locations are used to hold the legalization registers.

Table 8-2 • **Core1553BRT_APB Register Set**

Address	Name	Reset Value	Description
0x1F80	CONTROL	0000	Control and Status register
0x1F84	INTERRUPT	0000	Interrupt register
0x1F88	VWORD	0000	1553B vector word register
0x1FCO	LEG_REG0	0000	Receive SA 15:0
0x1FC4	LEG_REG1	0000	Receive SA 31:16
0x1FC8	LEG_REG2	0000	Transmit SA 15:0
0x1FCC	LEG_REG3	0000	Transmit SA 31:16
0x1FD0	LEG_REG4	0000	Broadcast Receive SA 15:0
0x1FD4	LEG_REG5	0000	Broadcast Receive SA 31:16
0x1FD8	LEG_REG6	FFFF	Broadcast Transmit SA 15:0
0x1FDC	LEG_REG7	FFFF	Broadcast Transmit SA 31:16
0x1FE0	LEG_REG8	FFFF	Mode Code Receive 15:0
0x1FE4	LEG_REG9	FFFD	Mode Code Receive 31:16
0x1FE8	LEG_REGA	FE01	Mode Code Transmit 15:0
0x1FEC	LEG_REGB	FFF2	Mode Code Transmit 31:16
0x1FF0	LEG_REGC	FFFF	Broadcast Mode Code Receive 15:0
0x1FF4	LEG_REGD	FFFD	Broadcast Mode Code Receive 31:16
0x1FF8	LEG_REGE	FE05	Broadcast Mode Code Transmit 15:0
0x1FFC	LEG_REGF	FFFF	Broadcast Mode Code Transmit 31:16

Control Register

The Control register is shown in Table 8-3.

***Table 8-3* • Control Register 0x1F80**

Bit	Name	Type	Reset	Description
0	SREQUEST	RW	0	Sets the Service Request bit in the 1553 status word.
1	RTBUSY	RW	0	Sets the Busy bit in the 1553 status word.
2	SSFLAG	RW	0	Sets the Subsystem bit in the 1553 status word.
3	TFLAG	RW	0	Sets the Terminal Flag bit in the 1553 status word.
4	TESTORUN	RW	0	When set, the RT will transmit an additional 16 words. This is for test use only to force the internal transmit overrun timer to trigger; it should not enabled during normal operation.
6:5	Reserved	RO	0	Reserved for future use.
7	BUSY	RO	0	Indicates that the RT is busy processing a 1553 message.
12:8	RTADDR	RW	0	Sets the RT address when the LOCK bit is 0 and if the LOCKRT core parameter is not set. On read, returns the current set RT address, either from this register or the external RTADDR inputs.
13	RTPARITY	RW	0	Sets the RT address parity bits; same conditions apply as to the RTADDR bits.
14	RTADERR	RO	?	Indicates that the RT address and parity settings are incorrectly set. At reset, this will reflect the external RTADDR/RTADDRP settings.
15	LOCK	RW	1	When set, forces the RT address and parity to be set by the external inputs. When reset (0) the RT address can be set by bits 12 to 8.

Interrupt Register

The Interrupt register is shown in Table 8-4.

Table 8-4 • **Interrupt Register 0x1F84**

Bit	Name	Type	Reset	Description
6:0	INTVECT	R	0000000	Interrupt vector
				6 0: Bad block received 1: Good block received 5 0: RX data 1: TX data 4:0 Subaddress
7	INTACT	RW	0	When set, indicates that an interrupt event is pending and that the INTVECT bits contain valid information. Cleared by writing 1 to this bit.
8	INTEN	RW	0	When set, the core will drive the INTERRUPT output when INTACT is set.
9	INTORUN	RW	0	When set, indicates that the 1553 core has generated a second interrupt before the processor acknowledged the previous interrupt. Cleared by writing 1 to this bit.
10	CLRERRR	RW	0	This allows the processor to clear internal error conditions stored and used for the 1553 BIT word. The processor must set and clear this bit; it must be left set for greater than two CLK clock periods.
15:11	Version	RO	01110	Indicates the core version. The same values are used as per the 1553 BIT word.

Vector Word Register

The Vector Word register is shown in Table 8-5.

Table 8-5 • **Vector Word Register 0x1F88**

Bit	Name	Type	Reset	Description
15:0	VWORD	RW	00000	Sets the vector word that will be transmitted by the 1553 "transmit vector word" mode code.

Illegalization Registers

The illegalization registers allow each subaddress and mode code message to be illegalized (Table 8-6); they are only implemented when the LEGMODE parameter is set to 2 or 3. A 0 in a bit indicates that the subaddress or mode code is legal, and a 1 that the subaddress or mode code is illegal.

Table 8-6 • **Subaddress and Mode Code Legalization Registers**

Address	Name	Reset Value	Description
0x1FCO	LEG_REG0	0000	Receive SA 15:0
0x1FC4	LEG_REG1	0000	Receive SA 31:16
0x1FC8	LEG_REG2	0000	Transmit SA 15:0
0x1FCC	LEG_REG3	0000	Transmit SA 31:16
0x1FD0	LEG_REG4	0000	Broadcast Receive SA 15:0
0x1FD4	LEG_REG5	0000	Broadcast Receive SA 31:16
0x1FD8	LEG_REG6	FFFF	Broadcast Transmit SA 15:0
0x1FDC	LEG_REG7	FFFF	Broadcast Transmit SA 31:16
0x1FE0	LEG_REG8	FFFF	Mode Code Receive 15:0
0x1FE4	LEG_REG9	FFFD	Mode Code Receive 31:16
0x1FE8	LEG_REGA	FE01	Mode Code Transmit 15:0
0x1FEC	LEG_REGB	FFF2	Mode Code Transmit 31:16
0x1FF0	LEG_REGC	FFFF	Broadcast Mode Code Receive 15:0
0x1FF4	LEG_REGD	FFFD	Broadcast Mode Code Receive 31:16
0x1FF8	LEG_REGE	FE05	Broadcast Mode Code Transmit 15:0
0x1FFC	LEG_REGF	FFFF	Broadcast Mode Code Transmit 31:16

When LEGMODE is 2, the registers are reset, as shown in Table 8-6, so that subaddresses are enabled and mode codes are legalized as per normal 1553 rules. When LEGMODE = 3, these registers are not reset and will contain undefined values after power-up. When using ProASIC^{PLUS}, Axcelerator, or RTAX-S devices and LEGMODE = 3, these registers cannot be read by the processor; they are write-only.

Memory Buffers

The 8 kbytes of APB addressable memory is actually organized as 2048×16 locations and DWORD-aligned; the address of each memory buffer is shown in Table 8-7.

Table 8-7 • **APB Memory Buffer Address Map**

APB Address	RAM Contents (16-bit-wide data)	Action
0x0000	RX transfer status words	The core only writes to these addresses (except when SA30LOOP is High).
0x0080	RX subaddress 1	
0x0100	RX subaddress 2	
0x0180	RX subaddress 3	
…	…	
0x0E00	RX subaddress 28	
0x0E80	RX subaddress 29	
0x0F00	RX subaddress 30	
0x0F80	TX transfer status words	
0x1000	Not Used	The core only reads from these addresses.
0x1080	TX subaddress 1	
0x1100	TX subaddress 2	
0x1180	TX subaddress 3	
…	…	
0x1E00	TX subaddress 28	
0x1E80	TX subaddress 29	
0x1F00	TX subaddress 30	
0x1F80	Registers	Control and Status registers

The memory is split into sixty-four 32-word buffers, one per transmit and receive subaddress. The unused subaddresses 0 and 31 are used to store the transfer status words and implement the core registers.

The transfer status words are stored at the locations shown in Table 8-8.

Table 8-8 • **Transfer Status Word Storage**

APB Address	Transfer Status Words	APB Address	Transfer Status Words
0x0000	RX Mode Code (SA = 0)	0x0F00	TX Mode Code (SA = 0)
0x0004	RX SA = 1	0x0F04	TX SA = 1
0x0008	RX SA = 2	0x0F08	TX SA = 2
0x000C	RX SA = 3	0x0F0C	TX SA = 3
0x0010	RX SA = 4	0x0F10	TX SA = 4
0x0014	RX SA = 5	0x0F14	TX SA = 5
0x0018	RX SA = 6	0x0F18	TX SA = 6
0x001C	RX SA = 7	0x0F1C	TX SA = 7
0x0020	RX SA = 8	0x0F20	TX SA = 8
0x0024	RX SA = 9	0x0F24	TX SA = 9
0x0028	RX SA = 10	0x0F28	TX SA = 10
0x002C	RX SA = 11	0x0F2C	TX SA = 11
0x0030	RX SA = 12	0x0F30	TX SA = 12
0x0034	RX SA = 13	0x0F34	TX SA = 13
0x0038	RX SA = 14	0x0F38	TX SA = 14
0x003C	RX SA = 15	0x0F3C	TX SA = 15
0x0040	RX SA = 16	0x0F40	TX SA = 16
0x0044	RX SA = 17	0x0F44	TX SA = 17
0x0048	RX SA = 18	0x0F48	TX SA = 18
0x0050	RX SA = 19	0x0F50	TX SA = 19
0x0054	RX SA = 21	0x0F54	TX SA = 21
0x0058	RX SA = 22	0x0F58	TX SA = 22
0x005C	RX SA = 23	0x0F5C	TX SA = 23
0x0060	RX SA = 24	0x0F60	TX SA = 24
0x0064	RX SA = 25	0x0F64	TX SA = 25
0x0068	RX SA = 26	0x0F68	TX SA = 26
0x006C	RX SA = 27	0x0F6C	TX SA = 27
0x0070	RX SA = 28	0x0F70	TX SA = 28
0x0074	RX SA = 29	0x0F74	TX SA = 29
0x0078	RX SA = 30	0x0F78	TX SA = 30
0x007C	RX Mode Code (SA = 31)	0x0F7C	TX Mode Code (SA = 31)

9 – Implementation Hints

External Command Word Legality Example

The core provides three ports (USEEXTOK, CMDVAL, and CMDOKAY) that allow the legal command word set to be modified. When USEEXTOK is LOW, the core internally decides which command words are legal (the legal command word set is defined in the Core1553BRT MIL-STD-1553B Remote Terminal datasheet). When USEEXTOK is HIGH, an external block decodes the CMDVAL output and generates a CMDOKAY input to indicate legal command words.

The VHDL and Verilog code blocks below implement an external legality checker that does the following:

- Legalizes mode codes as per the Core1553BRT_APB Handbook
- Disables transmits from subaddresses 26 and 27
- Disables receives to subaddress 25
- Only enables word counts 1 to 9 and receives to subaddress 27

The source files for these modules are provided in the *source* directory.

The core allows 3 µs for the legality block to decode CWVAL and generate the CMDOKAY value; this can be implemented within the FPGA, as shown below.

VHDL Example

```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.std_logic_unsigned.all;

entity CWLEGALITY is
  port ( CWVAL      : in  std_logic_vector(11 downto 0);
            CMDOKAY : out std_logic
          );
end CWLEGALITY;

architecture RTL of CWLEGALITY is
signal BROADCAST    : std_logic;
signal ISMCODE      : std_logic;
signal TX           : std_logic;
signal SA           : std_logic_vector(4 downto 0);
signal WCMC         : std_logic_vector(4 downto 0);

begin

-- Decode incoming value
BROADCAST   <= CWVAL(11);
TX          <= CWVAL(10);
SA          <= CWVAL(9 downto 5);
WCMC        <= CWVAL(4 downto 0);
ISMCODE     <= '1' when ( SA="00000" or SA="11111") else '0';

-- This process decodes the command word and sets CMDOKAY for legal command words.
PLEGAL:
process(BROADCAST,TX,SA,WCMC,ISMCODE)
 variable OK     : std_logic;
 variable MUXSEL : std_logic_vector(5 downto 0);
 begin

    if (ISMCODE='0') then
        -- Data transfers
        MUXSEL := TX & SA;
```

```
            OK := '0';           -- Default is disabled
            ------------------------------------------------------------------------
            -- This case statement legalizes data transfers to certain subaddresses
            case MUXSEL is
              when "111010" => OK := '0';                    -- SA 26 Disabled for TX
              when "111011" => OK := '0';                    -- SA 27 Disabled for TX
              when "011001" => OK := '0';                    -- SA 25 Disabled for RX
             when "011011" => if WCMC>0 and WCMC <10 then   -- SA 27 Disabled for RX if WC>9
                                          OK := '1';
                                       end if;
              when others   => OK := '1';  -- Legalize all other subaddresses
            end case;
            ------------------------------------------------------------------------
            -- Broadcast transmits are not allowed; overrides above case statement
            if BROADCAST='1' and TX='1' then
              OK := '0';  -- Broadcast transmit is not allowed
            end if;
        else
            ------------------------------------------------------------------------
            -- This case statement legalizes mode codes
            MUXSEL := TX & WCMC;
            OK := '1';       -- Default is OKAY
            case MUXSEL is
              when "100000" => -- Dynamic Bus Control
                               OK := '0';        -- Since we can't do it, we Message Error
              when "100001" => -- Synchronise
              when "100010" => -- Transmit Status Word
                               OK := not BROADCAST;
             when "100011" => -- Initiate Self-Test; we set this because we provide BIT word
                               OK := '1';
              when "100100" => -- Transmitter Shutdown
              when "100101" => -- Override Transmitter Shutdown
              when "100110" => -- Inhibit Terminal Flag
              when "100111" => -- Override Inhibit Terminal Flag
              when "101000" => -- Reset Remote Terminal
              when "110000" => -- Transmit Vector Word
                               OK := not BROADCAST;
              when "010001" => -- Synchronise with Data
              when "110010" => -- Transmit Last Command
                               OK := not BROADCAST;
              when "110011" => -- Transmit BIT Word
                               OK := not BROADCAST;
              when "010100" => -- Selected Transmitter Shutdown
                               OK := '0';
              when "010101" => -- Override Selected Transmitter Shutdown
                               OK := '0';
              when others   => -- All other commands illegal
                               OK := '0';
            end case;
      end if;
      CMDOKAY <= OK;
end process;

end RTL;
```

Verilog Example

```
module CWLEGALITY (CWVAL, CMDOKAY);
   input[11:0] CWVAL;
   output CMDOKAY;
   reg CMDOKAY;
   wire BROADCAST, ISMCODE, TX;
   wire[4:0] SA, WCMC;

   assign BROADCAST = CWVAL[11] ;   // Decode incoming Value
   assign TX = CWVAL[10] ;
   assign SA = CWVAL[9:5] ;
   assign WCMC = CWVAL[4:0] ;
   assign ISMCODE = (SA == 5'b00000 | SA == 5'b11111) ? 1'b1 : 1'b0 ;

   always @(BROADCAST or TX or SA or WCMC or ISMCODE)
   begin : PLEGAL
      reg OK;
      reg[5:0] MUXSEL;
      if (ISMCODE == 1'b0)
      begin
         MUXSEL = {TX, SA};    // Data transfers
         OK = 1'b0;
         // This case statement legalizes data transfers to certain subaddresses
         case (MUXSEL)
            6'b111010 : OK = 1'b0;                    // SA 26 Disabled for TX
            6'b111011 : OK = 1'b0;                    // SA 27 Disabled for TX
            6'b011001 : OK = 1'b0;                    // SA 25 Disabled for RX
            6'b011011 : OK = (WCMC > 0 & WCMC < 10); // SA 27 Disabled for RX if WC>9
            default   : OK = 1'b1;                   // Legalize all other subaddresses
         endcase
        // Broadcast transmits are not allowed; overrides above case statement
         if (BROADCAST == 1'b1 & TX == 1'b1)
            OK = 1'b0; // Broadcast transmit is not allowed
      end
      else
      begin
         //-----------------------------------------------------------------
         // This case statement legalizes mode codes
         MUXSEL = {TX, WCMC};
         OK = 1'b1;
         case (MUXSEL)
            6'b100000 : // Dynamic Bus Control
                        OK = 1'b0; // Since we can't do it, we Message Error
            6'b100001 : // Synchronize
            6'b100010 : // Transmit Status Word
                        OK = ~BROADCAST;
            6'b100011 : // Initiate Self-Test; we set this because we provide BIT word
                        OK = 1'b1;
            6'b100100 : // Transmitter Shutdown
            6'b100101 : // Override Transmitter Shutdown
            6'b100110 : // Inhibit Terminal Flag
            6'b100111 : // Override Inhibit Terminal Flag
            6'b101000 : // Reset Remote Terminal
            6'b110000 : // Transmit Vector Word
                        OK = ~BROADCAST;
            6'b010001 : // Synchronise with Data
            6'b110010 : // Transmit Last Command
                        OK = ~BROADCAST;
            6'b110011 : // Transmit BIT Word
                        OK = ~BROADCAST;
            6'b010100 : // Selected Transmitter Shutdown
                        OK = 1'b0;
            6'b010101 : // Override Selected Transmitter Shutdown
                        OK = 1'b0;
```

```
                default   : // All other commands illegal
                            OK = 1'b0;
            endcase
        end
        CMDOKAY <= OK ;
    end
endmodule
```

Bus Transceivers

Core1553BRT_APB does not include the transceiver that is required to drive the 1553B bus. Core1553BRT_APB is designed to interface directly to MIL-STD-1553 transceivers. There are several suppliers of MIL-STD-1553 transceivers, such as Aeroflex (ACT4453).

When using Fusion, IGLOO/e, ProASIC3/E, ProASIC^{PLUS}, or Axcelerator FPGAs, level translators are required to connect the 5 V outputs of the 1553B transceivers to the 3.3 V inputs of the FPGA.

In addition to the transceiver, a pulse transformer is required for interfacing to the 1553B bus. Figure 9-1 shows the connections required from Core1553BRT_APB to the transceivers and then to the bus via pulse transformers.

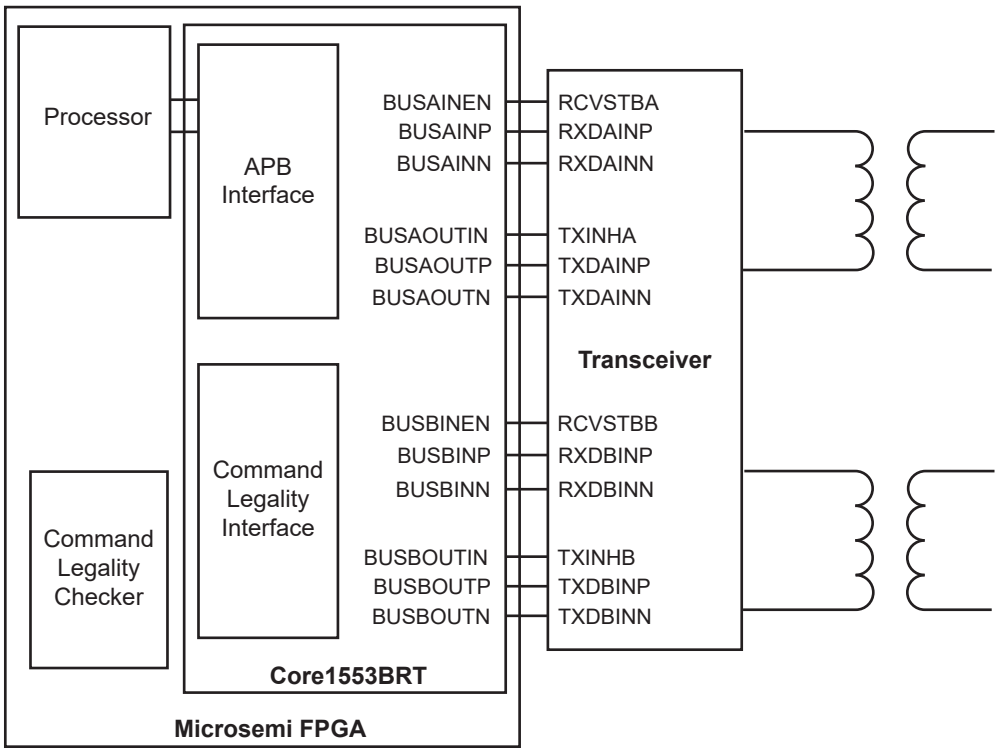

Figure 9-1 • **Typical CPU- and Memory-Based RT System**