USER MANUAL

RF2TTC 2.0

RF to TTC VMEbus Interface Card and S/W

Summary:

This document describes the functionality of the RF2TTC card as well as the generic S/W that has been developed for it.
Document Revision 2 – 12.03.07: the CR/CSR space has been transposed to the User Space.

Prepared by :	***Checked by :***	***Approved by :***
Sophie Baron, PH/ESS **Markus Joos, PH/ESS**		

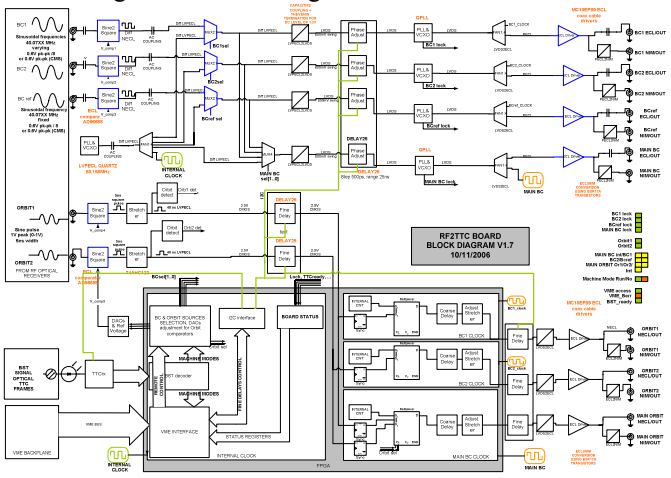
for information, you can contact :		Tel.	Fax.	E-Mail
	Sophie Baron	+41.22.7677339	+41.22.7678925	**sophie.baron@cern.ch**
	Markus Joos	+41.22.7672364	+41.22.7678925	markus.joos@cern.ch

Table of Contents

1. INTRODUCTION

The RF2TTC (RF to TTC VMEbus Interface Card) is an interface card between the optical receiver modules (receiving timing signals coming from the SR4 building in Echenevex), and the TTC electronics within the experiments.

The timing signals treated by the RF2TTC are the three 40.078MHz Bunch Clocks (BC1, BC2 and BCref) and the two orbit signals (Orb1 and Orb2) necessary to drive the 2 beam lines of the LHC. The RF2TTC module converts them into ECL signals, and performs various adjustments on each signal before making them available for the in-detector TTC electronics.



RF2TTC module diagram

The three Bunch Clocks (represented on the top part of the above diagram) are all treated in the following way:
A comparator with an adjustable threshold first converts the input signal into a PECL signal, before being multiplexed with an internal 40.078MHz clock in case of absence of the Bunch Clock on the front panel. The signal is then shifted by an adjustable delay with 0.5ns precision, before being cleaned by a QPLL and transmitted on the front panel via an ECL 50 Ohm coaxial cable driver with an AC-coupled output.

A global multiplexer allows selection between the three Bunch Clocks and the internal clock to generate a fourth Bunch Clock output, called Main BC, which can also be delayed.

The two orbit signals (middle and bottom right parts of the diagram) are first converted using the same adjustable comparator stage as for the Bunch Clocks. They are then lengthened to more than 25ns, finely delayed with 0.5ns steps, before going into an FPGA (grey block of the diagram), where they are synchronized to their corresponding clock, multiplexed with an internal orbit, and coarse delayed. Their length and polarity can be adjusted, and they are then again finely delayed before being transmitted by the ECL drivers.
A global multiplexer also allows selection between the two orbits and an internal one, synchronized to the Main Bunch Clock. This orbit signal is called Main Orbit and can as well be finely delayed before being transmitted.

The BST (Beam Synchronous Timing) optical signal (on the bottom left part of the diagram) is received, decoded and analyzed to recover the machine mode. This mode is useful to know when the timing signals are stable and can be used. In deed, neither the Bunch Clocks nor the Orbit signals are fully guaranteed out of the physics modes (flat top of the LHC energy curve). It is thus advised to use internal signals when the machine mode indicates that there is no beam.
All the adjustments are done using VME registers. Many status registers are available, as well as special configurations, for stand-alone or debugging work.

This document contains a description of all accessible registers of the RF2TTC card as well as description of the generic S/W that has been developed for this card. At the end of this document, some basic examples of configuration procedures are proposed.

2. RF2TTC HARDWARE

2.1. STANDARDS AND POWER SUPPLIES

The RF2TTC board is a VME64x 6U board. It requires the following power supplies:

- +3V3: 2.5A
- +5V: 0.3A
- -12V: 1.4A

Total power : 26W

2.2. VMEBUS INTERFACE

The VMEbus interface of the RF2TTC cards is implemented in its FPGA and based on the VME interface developed by Peter Lichard for the TRT-TTC board (ATLAS). It provides 2 types of VMEbus addressable resources as described in Table 1.

Resource	VMEbus access mode	Description
Control and status registers	A32/D32 with A19=0 and AM-code 0x09	These are the registers that control the behaviour of a RF2TTC card and provide information about its current status.
EPROM	A32/D32 with A19=1 and AM-code 0x09	The access to the EPROM is reserved for in-crate reconfiguration.

Table 1: VMEbus resources of the RF2TTC

All the registers of the board are accessible using 0x09 AM (A32, D32). The board address is the geographical address of the module if the manual rotary switches are set to 0x00. The address used to access the user space is hence defined as follows:

	A31-A28	A27-A24	A23-A20
SW(7..0)=0x00	0	GEOG ADD	
SW(7..0)≠0x00	0	SW1[7..4]	SW2[3..0]

2.2.1. Reset registers

Register	Offset	Purpose	Access
BSET	0x00010	Set reset actions	R/W
BCLEAR	0x00014	Clear reset actions	R/W

BSET: Board Set Register assignment

Description:

This register is declared in the VME64x as a User-defined Bset register. It is used here to define partial reset functions (QPLL only, Delay25 chips only, TTCrx only). The bit definition is as follows:

Bit	Value	Write	Read
0	1	place Delay25 chips in reset mode	Delay25 chips in reset mode
	0	no effect	Delay25 chips not in reset mode
1	1	place QPLL chips in reset mode	QPLL chips in reset mode
	0	no effect	QPLL chips not in reset mode
2	1	place TTCrx chips in reset mode	TTCrx chips in reset mode
	0	no effect	TTCrx chips not in reset mode
3	1	place board in reset mode	Board in reset mode
	0	no effect	Board not in reset mode

BCLEAR (User-defined BCLEAR) Register assignment

Description:

This register is declared in the VME64x as a User-defined Bclear register. It is used here to remove partial reset functions (QPLL only, Delay25 chips only, TTCrx only). The bit definition is as follows:

Bit	Value	Write	Read
0	1	remove Delay25 chips from reset mode	Delay25 chips in reset mode
	0	no effect	Delay25 chips not in reset mode
1	1	remove QPLL chips from reset mode	QPLL chips in reset mode
	0	no effect	QPLL chips not in reset mode
2	1	remove TTCrx chips from reset mode	TTCrx chips in reset mode
	0	no effect	TTCrx chips not in reset mode
	1	remove Board from reset mode	Board in reset mode
	0	no effect	Board not in reset mode

2.2.2. Board Identification read only registers

Register	Address	Value	Access
MANUFACTURER ID	0x00000	0x00080030 (CERN)	R
BOARD ID	0x00004	0x0000016B	R
REVISION ID	0x00008	Hardware version Prototype = 0x2	R
PROGRAM ID	0x0000C	Firmware date number Ex: 0x07032007 (07 of March 2007)	R

2.2.3. Board configuration registers

BCx_MAN_SELECT, BCx_BEAM_SELECT & BCx_NOBEAM_SELECT

Name	Offset	Size	Access
BC1_MAN_SELECT	0x7FBFC	1 bit	R/W
BC2_MAN_SELECT	0x7FBCC		
BCref_MAN_SELECT	0x7FBAC		
BC1_BEAM_SELECT	0x7FBF8		
BC2_BEAM_SELECT	0x7FBC8		
BCref_BEAM_SELECT	0x7FBA8		
BC1_NOBEAM_SELECT	0x7FBF4		
BC2_NOBEAM_SELECT	0x7FBC4		
BCref_NOBEAM_SELECT	0x7FBA4		
BCmain_MAN_SELECT	0x7FB8C	2 bits	
BCmain_BEAM_SELECT	0x7FB88		
BCmain_NOBEAM_SELECT	0x7FB84		

Description:

These registers select the sources of the BC outputs. Only one set of registers is active at any time. The BCx_MAN_SELECT registers are active when the RF2TTC is operating in manual mode. If the card is in automatic mode and the beam is on the BC outputs are controlled by the BCx_BEAM_SELECT registers. The BCx_NOBEAM_SELECT registers control the BC outputs when the RF2TTC is in automatic mode and the beam absent.

Bit definition for BC1, BC2 and BCref registers

Value	Description
0	Output taken from internal 40.078MHz clock
1	Output follows the respective BC input

Bit definition for BCmain registers

Value	Description

0	Output taken from internal 40.078MHz clock
1	Output follows BCref input
2	Output follows BC2 input
3	Output follows BC1 input

BCx_QPLL_MODE

Name	Offset	Size	Access
BC1_QPLL_MODE	0x7FBF0	1 bit	R/W
BC2_QPLL_MODE	0x7FBC0		
BCref_QPLL_MODE	0x7FBA0		
BCmain_QPLL_MODE	0x7FB80		

Description:

These registers define the QPLL locking mode.

Value	Description
0	Re-lock only after a reset
1	Re-lock automatically if the lock gets lost

BCx_DAC

Name	Offset	Size	Access
BC1_DAC	0x7FBEC	8 bits	R/W
BC2_DAC	0x7FBBC		
BCref_DAC	0x7FB9C		

Description:

These registers define the threshold of the input comparator for the respective BC input channel in a range from -1.25V to +1.25V. The threshold is linked to the value of the register by the formula

Threshold = -1.25 + value * 2.5 / 255

BCx_QPLL_STATUS

Name	Offset	Size	Access
BC1_QPLL_STATUS	0x7FBE8	2 bits	R
BC2_QPLL_STATUS	0x7FBB8		
BCref_QPLL_STATUS	0x7FB98		
BCmain_QPLL_STATUS	0x7FB7C		

Description:

These registers contain the status of the QPLLs of the BC channels. Bit 1 indicates that the QPLL detected an error and bit 0 indicates the locking status.

Bit 1	Description
0	QPLL OK
1	QPLL has error

Bit 0	Description
0	QPLL not locked
1	QPLL locked

ORBx_MAN_SELECT, ORBx_BEAM_SELECT & ORBx_NOBEAM_SELECT

Name	Offset	Size	Access
ORB1_MAN_SELECT	0x7FB6C	1 bit	R/W
ORB2_MAN_SELECT	0x7FB2C		
ORB1_BEAM_SELECT	0x7FB68		
ORB2_BEAM_SELECT	0x7FB28		
ORB1_NOBEAM_SELECT	0x7FB64		
ORB2_NOBEAM_SELECT	0x7FB24		
ORBmain_MAN_SELECT	0x7FAEC	2 bits	
ORBmain_BEAM_SELECT	0x7FAE8		
ORBmain_NOBEAM_SELECT	0x7FAE4		

Description:

These registers select the sources of the orbit outputs. Only one set of registers is active at any time. The ORBx_MAN_SELECT registers are active when the RF2TTC is operating in manual mode. If the card is in automatic mode and the beam is on the orbit outputs are controlled by the ORBx_BEAM_SELECT registers. The ORBx_NOBEAM_SELECT registers control the orbit outputs when the RF2TTC is in automatic mode and the beam absent.

Bit definition for ORB1 and ORB2 registers

Value	Description
0	Output follows the respective orbit input
1	Output from internal BC synchronized orbit generator

Bit definition for ORBmain registers

Value	Description
0	Output follows the orbit 1 input

1	Output follows the orbit 2 input
2	Output from internal BCmain synchronized orbit generator

ORBx_POLARITY

Name	Offset	Size	Access
ORB1_POLARITY	0x7FB60	1 bit	R/W
ORB2_POLARITY	0x7FB20		
ORBmain_POLARITY	0x7FAE0		

Description:

If set, this bit inverts the polarity of the orbit output with respect to the orbit input (i.e. the orbit output is negative active).

ORBx_COARSE_DELAY

Name	Offset	Size	Access
ORB1_COARSE_DELAY	0x7FB5C	12 bits	R/W
ORB2_COARSE_DELAY	0x7FB1C		
ORBmain_COARSE_DELAY	0x7FADC		

Description:

This register allows the orbit output signal to be shifted by multiples of 25 ns with respect to the input. If set to 0, the output is shifted by the minimum intrinsic delay induced by the board itself. Values above 0xDEB (3563) are illegal because they would result in a shift longer than the LHC orbit period (88.93 μs)

ORBx_LENGTH

Name	Offset	Size	Access
ORB1_LENGTH	0x7FB58	8 bits	R/W
ORB2_LENGTH	0x7FB18		
ORBmain_LENGTH	0x7FAD8		

Description:

This register allows the orbit pulse to be stretched in steps of 25 ns. If set to 0 the width of the orbit pulse is stretched by 75 ns. The largest pulse width (with all 8 bits set to 1) is 6.4 μs. The original width of the internally generated orbit pulse is 75ns.

ORBx_INT_PERIOD_SET

Name	Offset	Size	Access
ORB1_INT_PERIOD_SET	0x7FB54	12 bits	R/W

ORB2_INT_PERIOD_SET	0x7FB14		
ORBmain_INT_PERIOD_SET	0x7FAD4		

Description:

This register allows setting the period of the internally generated orbit signal in units of 25 ns. The default value is 0xDEB, which corresponds to 3563 bunch clocks between two orbits.

ORBx_INT_PERIOD_COUNTER

Name	Offset	Size	Access
ORB1_INT_PERIOD_COUNTER	0x7FB50	12 bits	R
ORB2_INT_PERIOD_COUNTER	0x7FB10		
ORBmain_INT_PERIOD_COUNTER	0x7FAD0		

Description:

This register is provided for debugging purposes. It holds the value of the BC counter that is used to generate the internal orbit signal. This can be reset by the ORB_INT_RESET register.

ORBx_COUNTER

Name	Offset	Size	Access
ORB1_COUNTER	0x7FB4C	32 bits	R
ORB2_COUNTER	0x7FB0C		
ORBmain_COUNTER	0x7FACC		

Description:

This register holds the number of orbit pulses that have been received since the counter was reset/enabled. At an orbit period of 89 μs this counter will overflow after approximately 106 hours, and will be reset.

ORBx_PERIOD_RD

Name	Offset	Size	Access
ORB1_PERIOD_RD	0x7FB48	12 bits	R
ORB2_PERIOD_RD	0x7FB08		
ORBmain_PERIOD_RD	0x7FAC8		

Description:

This register holds the time, in units of 25 ns BC ticks, that has elapsed between the last two orbit output pulses.

ORBx_PERIOD_FIFO_STATUS

Name	Offset	Size	Access

ORB1_PERIOD_FIFO_STATUS	0x7FB44	2 bits	R
ORB2_PERIOD_FIFO_STATUS	0x7FB04		
ORBmain_PERIOD_FIFO_STATUS	0x7FAC4		

Description:

This register holds the status of the FIFO that contains the most recent 128 orbit periods of the respective orbit output channel. Bit definitions:

Bit	Value	Description
0	0	Fifo not empty
	1	Fifo empty
1	0	Fifo not full
	1	Fifo full

ORBx_PERIOD_FIFO_RD

Name	Offset	Size	Access
ORB1_PERIOD_FIFO_RD	0x7FB40	16 bits	R
ORB2_PERIOD_FIFO_RD	0x7FB00		
ORBmain_PERIOD_FIFO_RD	0x7FAC0		

Description:

These registers provide access to three 256 word deep FIFOs which contain the most recent 256 orbit periods of the respective orbit output channel in bits 0..13. Reading the last period stored in the FIFO, or from an empty FIFO results in reading a 1 in bit 14 (FIFO empty) . Bit 15, if set to 1, indicates that the FIFO is full. For the moment it is not possible to read these FIFOs with a constant address block transfer.

ORBx_DAC

Name	Offset	Size	Access
ORB1_DAC	0x7FB3C	8 bits	R/W
ORB2_DAC	0x7FAFC		

Description:

These registers allow setting the threshold voltage of the orbit input comparator in a range from -1.25 V to +1.25 V. The threshold is linked to the value of the register by the formula:

Threshold = -1.25 + value * 2.5 / 255

TTCrx_status

Name	Offset	Size	Access
TTCrx_status	0x7FAA0	1 bit	R

Description:

This register reflects the status of the on-board TTCrx chip. Bit definitions:

Value	Description
0	TTCrx not ready
1	TTCrx ready – the BST message is correctly decoded (at least a 40MHz clock is sent over the optical fibre connected to the TTCrx)

BST_Machine_Mode

Name	Offset	Size	Access
BST_Machine_Mode	0x7FA9C	32 bits	R

Description:

This register holds the LHC machine mode as decoded from the BST messages received by the TTCrx..

Each number (here in hexadecimal) corresponds to one machine mode, as transmitted by the BST:

Value	Description
0	No Beam
1	Filling
2	Ramping
3	Physics
4	
5	
6	
7	
8	

BEAM_NO_BEAM_DEF

Name	Offset	Size	Access
BEAM_NO_BEAM_DEF	0x7FA7C	32 bits	R/W

Description:

This register controls the operation of the RF2TTC in automatic mode. Each bit controls one machine mode. A bit that is set to 0 causes the RF2TTC to use the NOBEAM_SELECT registers for BC and orbit to be active when the machine is in the mode that corresponds to that bit. If a bit is set to 1 the RF2TTC applies the settings in the BC and orbit BEAM_SELECT registers for as long as the machine is in the respective mode.

Bit	MODE Name	Description	Default
0	No Beam	0= NOBEAM mode, 1= BEAMMODE when automatic mode is activated.	0
1	Filling		0
2	Ramping		0
3	Physics		1
4			
5			
6			

WORKING_MODE

Name	Offset	Size	Access
WORKING_MODE	0x7FA78	7 bits	R/W

Description:

The bits in this register control the operational modes of the outputs of the RF2TTC. Each bit corresponds to one signal.

Bit number	Related output	Bit value	Selected mode
0	BC1	0	Manual
		1	Automatic
1	BC2	0	Manual
		1	Automatic
2	BCref	0	Manual
		1	Automatic
3	BCmain	0	Manual
		1	Automatic
4	ORB1	0	Manual
		1	Automatic
5	ORB2	0	Manual
		1	Automatic
6	ORBmain	0	Manual
		1	Automatic

ORB_INT_ENABLE

Name	Offset	Size	Access
ORB_INT_ENABLE	0x7FA6C	3 bits	R/W

Description:

This register controls the status of the BC counters that generate the internal orbit pulses.

Bit number	Related orbit	Bit value	Counter mode
0	Orbit 1 (counts BC1 ticks)	0	Disabled
		1	Enabled
1	Orbit 2 (counts BC2 ticks)	0	Disabled
		1	Enabled
2	Main orbit (counts BCmain ticks)	0	Disabled
		1	Enabled

ORB_COUNTER_ENABLE

Name	Offset	Size	Access
ORB_COUNTER_ENABLE	0x7FA68	3 bits	R/W

Description:

This register controls the status of the orbit pulse counters. Once a channel has been enabled the registers ORBx_COUNTER count the orbit pulses of that channel

Bit number	Related orbit	Bit value	Counter mode
0	Orbit 1	0	Disabled
		1	Enabled
1	Orbit 2	0	Disabled
		1	Enabled
2	Main orbit	0	Disabled
		1	Enabled

PERIOD_COUNTER_ENABLE

Name	Offset	Size	Access
PERIOD_COUNTER_ENABLE	0x7FA64	3 bits	R/W

Description:

This register controls the status of the orbit period counters. Once a channel has been enabled the FIFOs and ORBx_PERIOD_FIFO_RD start measuring and storing the duration of orbit signals.

Bit number	Related orbit	Bit value	Counter mode
0	Orbit 1	0	Disabled
		1	Enabled
1	Orbit 2	0	Disabled
		1	Enabled
2	Main orbit	0	Disabled
		1	Enabled

Counter RESET registers

Name	Offset	Access	Function
ORB_INT_RESET	0x7FA4C	W	Reset the three counters that generate the internal orbits 1, 2 and Main. One bit per counter. Same definition than the ORB_INT_ENABLE register.
PERIOD_COUNTER_RESET	0x7FA48		Reset the counters that measure the period of the orbit pulses 1, 2 and Main. At the same time the period FIFOs are cleared. One bit per counter. Same definition than the PERIOD_COUNTER_ENABLE register.
ORB_COUNTER_RESET	0x7FA44		Reset the orbit pulse counters 1, 2 and Main

Description:

A reset is triggered by writing a 1 to the address of the respective register. The 3 bits of the counter_reset register can reset the counters of ORB1, ORB2 and/or ORBmain, by writing various patterns:

Bit number	Related orbit	Bit value	Counter mode
0	Orbit 1	0	No effect
		1	reset
1	Orbit 2	0	No effect
		1	reset
2	Main orbit	0	No effect
		1	reset

DELAY25_REG, TTCrx_REG

Name	Offset	Size	Access
DELAY25_REG	0x7D200	8 bits	R
TTCrx_REG	0x7E200		

Description:

These registers are required to read values from the TTC and Delay25 registers described below. Due to delays introduced by the I2C bus it is not possible to read these registers directly. Instead a sequence of three steps is required.

1) Read a dummy data word from the address of the TTC or Delay25 register that is to be read out
2) Wait for at least 2 ms
3) Read the data value from the DELAY25_REG or TTCrx_REG (FIFO contents of the read access to delay25 and TTCrx chips)

If multiple registers are to be read one can group the dummy reads (step 1) and data reads (step 3) such that they are only separated by one 2 ms delay. This pipelining however works for up to 256 read requests.

BC_ DELAY25_x

Name	Offset	Used	Size	Access
BC_DELAY25_GCR	0x7D014	Yes	8 bits	(R)/W
BC_DELAY25_BCmain	0x7D00c	Yes		
BC_DELAY25_BCref	0x7D008	Yes		
BC_DELAY25_BC2	0x7D004	Yes		
BC_DELAY25_BC1	0x7D000	Yes		

Description:

These registers control the configuration of the Delay25 chips for the BC signals. These chips ensure the BC signal to be shifted by steps of 0.5ns with a jitter lower than 19ps rms. For details about the read protocol see above. Bit definition (from Delay25 manual):

The bit allocation of each channel control register is as given in the following table. Bits Del<5:0> control the delay for each channel and the ***Enable bit enables the channel output***. Upon a reset, bit Enable and bits Del<5:0> are cleared.

Control registers (CR0 to CR4) bit allocation

B7	B6	B5	B4		B3	B2	B1	B0
n.u.	**Enable**	Del<5>	Del<4>	Del<3>	Del<2>	Del<1>	Del<0>	Function
n.u.	**0**	0	0	0	0	0	0	Reset State

The general control register GCR controls the operation of the Delay-Locked Loop (DLL) and allows to reset the DLL or the ASIC via the I2C interface. The bit allocation for this register is given in Table 4

General Control Register (GCR) bit allocation

B7	B6	B5	B4	B3		B2	B1	B0
reserved	IDLL	n.u.	n.u	n.u.	n.u.	M<1>	M<0>	Function
0	0	-	-	-	-	Not cleared	Not cleared	Reset State

The ASIC can operate with for different clock frequencies (32, 40, 64 and 80 MHz). For this application, the M<0> and M<1> bits must be set to 0 (40MHz).

IDLL: bit IDLL is used to force the resynchronization of the DLL without resetting the chip. Writing a "1" to this bit forces the resynchronization of the DLL. This bit always reads as a "0"

ORBIN_DELAY25_x

Name	Offset	Used	Size	Access
ORBIN_DELAY25_GCR	0x7D034	yes	8 bits	(R)/W
ORBIN_DELAY25_ORB2	0x7D024	Yes		
ORBIN_DELAY25_ORB1	0x7D020	Yes		

Description:

These registers control the configuration of the Delay25 chips for the orbit input signals. Finely adjusting the delay of the orbit inputs allows centralising the orbit pulse with the rising edge of the corresponding bunch clock to ensure a good synchronisation of the 2 signals. For details about the read protocol see above. Bit definition: see BC_DELAY_x register.

ORBOUT_DELAY25_x

Name	Offset	Used	Size	Access
ORBOUT_DELAY25_GCR	0x7D054	Yes	8 bits	(R)/W
ORBOUT_DELAY25_ORBmain	0x7D048	Yes		
ORBOUT_DELAY25_ORB2	0x7D044	Yes		
ORBOUT_DELAY25_ORB1	0x7D040	Yes		

Description:

These registers control the configuration of the Delay25 chips for the orbit output signals. This adjustment is to allow the experiments fine tuning the orbit for its use in their trigger electronics. For details about the read protocol see above. Bit definition: see BC_DELAY_x register.

TTCrx registers

Name	Offset	Size	Access
TTCrx_pointer to the register	0x7E000	8 bits	R/W
TTCrx_pointer to the data	0x7E004	8 bits	W

Description:

These are the two registers used to read and write all the internal registers of the TTCrx used to receive the BST message (see TTCrx user manual). Only one register requires to be accessed for the purpose of receiving the BST message: the control register (internal address= 0x03). Its value should be 0xB3 instead of 0x93 (its default value). The '1' added on bit 5 allows enabling the Dout bus of the TTCrx, which contains the broadcast data, and hence the Machine Mode.

Register access protocol:

The TTCrx chip needs to be ready (ie, the optical fibre needs to deliver a correct encoded 40MHz clock), in order to access the internal registers.

The way to access the TTCrx registers is described in the TTCrx manual, p30:

"I2C_pointer register and the **I2C_data** register. The **I2C_pointer** register is five bits wide and contains the address of the internal register as defined in Table 3 (page 16). When reading the **I2C_data** register, the content of the TTCrx register *being addressed by the pointer register* is transferred. Conversely, writing a byte to the **I2C_data** register in fact writes to the TTCrx register *addressed by* the **I2C_pointer** register. Hence, each I2C access is performed in two steps:
1) Write the register number in the **I2C_pointer** register
2) Read or write the **I2C_data** register
According to the I2C bus specification, each device on the bus is addressed by a 7-bit wide I2C device address. Each TTCrx chip occupies two consecutive positions in the 7-bit I2C address space. Hence, it is possible to address 64 devices in the system. The 7-bit I2C address is derived from the content of the ID_I2C<5:0> base address register in the following way:

I2C access register name	Resulting 7 bit I2C address
I2C_pointer	ID_I2C<5:0> * 2
I2C_data	ID_I2C<5:0> * 2 + 1

Table 12 I2C address calculation."

The registers accessible via I2C are the following:

I2C reg. address (decimal)	Register name	Default content (After reset)
0	Fine Delay 1	00000000
1	Fine Delay 2	00000000
2	Coarse Delay	00000000
3	Control	10010011
8	Single error count<7:0>	00000000
9	Single error count<15:8>	00000000
10	Double error count<7:0>	00000000
11	SEU error count <15:8>	00000000
16	ID<7:0>	00000000
17	MasterModeA<1:0>, ID<13:8>	00000000
18	MasterModeB<1:0>, I2C_ID <5:0>	00000000
19	Config 1	00011010
20	Config 2	10000100
21	Config 3	10100111
22	Status	11100000
24	Bits <7:0>	00000000
25	Bits <15:8>	00000000
26	Bits <7:0>	00000000
27	Bits <15:8>	00000000
28	Bits <23:16>	00000000

Example of registers read and write via VME access:

Read control register (I2C address 3):

1- VME WRITE, AM=0x09, RegOFFSET=0xE000, Data=0x3 (register I2C address)

2- VME READ, AM=0x09, RegOFFSET=0xE000, (the Data read has no meaning)

3- VME READ, AM=0x09, RegOFFSET=0xE200, Data should be 0x00000003

Write Fine Delay Register (I2C address 1):

1- VME WRITE, AM=0x09, RegOFFSET=0xE000, Data=0x1 (register I2C address)

2- VME WRITE, AM=0x09, RegOFFSET=0xE004 (offset of the data register), 0x0000YOURDATA

Read fine delay register (I2C address 1):

1- VME WRITE, AM=0x09, RegOFFSET=0xE000, Data=0x1

2- VME READ, AM=0x09, RegOFFSET=0xE000, (the Data read has no meaning)

3- VME READ, AM=0x09, RegOFFSET=0xE200, Data should be 0x000XYOURDATA, with X=0 if the fIFO is not empty, X=1 if you are reading the last word stored in a FIFO

Successively read fine delay registers 1 and 2 (I2C address 1 and 2):

4- VME WRITE, AM=0x09, RegOFFSET=0xE000, Data=0x1

5- VME READ, AM=0x09, RegOFFSET=0xE000, (the Data read has no meaning)

6- VME WRITE, AM=0x09, RegOFFSET=0xE000, Data=0x2

7- VME READ, AM=0x09, RegOFFSET=0xE000, (the Data read has no meaning)

8- VME READ, AM=0x09, RegOFFSET=0xE200, Data should be 0x0000YOURDATA

9- VME READ, AM=0x09, RegOFFSET=0xE200, Data should be 0x0001YOURDATA

2.3. CALIBRATION PROCEDURES

For a proper functioning of the RF2TTC in its environment a number of coarse and fine grained pulse delay and stretch registers have to be tuned by the user. A description of this procedure will follow once if that been analyzed to what extent the RF2TTC can perform FPGA-based auto-calibrations.

2.4. BOARD CONFIGURATION

Element	Description
LSB rotary switch	TBD
MSB rotary switch	TBD

2.5. FIBRE / CABLE CONNECTIONS

Connector name	To be connected to
BST	TTC encoded signal. One of the BST optical fibres (two are normally available, one per ring). The optical power level should be between -5dBm and -25dBm.
BC INPUTS – BC1	ECL AC coupled signal. Should be connected to the BC1 output of the RF_Rx_D
BC INPUTS – BC2	ECL AC coupled signal. Should be connected to the BC2 output of the RF_Rx_D
BC INPUTS – BC_REF	ECL AC coupled signal. Should be connected to the BCref output of the RF_Rx_D
ORB INPUTS – ORB1	ECL AC coupled signal. Should be connected to the ORB1 output of the RF_Rx_D
ORB INPUTS – ORB2	ECL AC coupled signal. Should be connected to the ORB2 output of the RF_Rx_D
BC OUTPUTS – BC1	ECL AC coupled signal. Experiments electronics.
BC OUTPUTS – BC2	ECL AC coupled signal. Experiments electronics.
BC OUTPUTS – BC_REF	ECL AC coupled signal. Experiments electronics.
BC OUTPUTS – BCmain	ECL AC coupled signal. Experiments electronics.
ORB OUTPUTS – ORB1	ECL AC coupled signal. Experiments electronics.
ORB OUTPUTS – ORB2	ECL AC coupled signal. Experiments electronics.
ORB OUTPUTS – ORBmain	ECL AC coupled signal. Experiments electronics.

2.6. FRONT-PANEL LEDS

LED	Description
BC1_LOCK	Displays the state of the QPLL chip connected to the selected BC1 signal. (ON=locked/OFF=not locked).
BC2_LOCK	Displays the state of the QPLL chip connected to the selected BC1 signal. (ON=locked/OFF=not locked).
BCREF_LOCK	Displays the state of the QPLL chip connected to the selected BC1 signal. (ON=locked/OFF=not locked).
BCmain_LOCK	Displays the state of the QPLL chip connected to the selected BC1 signal. (ON=locked/OFF=not locked).
ORB1_OK	Monitors the presence of the external orbit after the comparator. (ON=signal present, OFF= no signal). When OFF, it can mean, either that the orbit is not present, or that the DAC setting the threshold at the input does not deliver an adapted threshold.
ORB2_OK	Monitors the presence of the external orbit after the comparator. (ON=signal present, OFF= no signal). When OFF, it can mean, either that the orbit is not present, or that the DAC setting the threshold at the input does not deliver an adapted threshold.
BEAM	Monitors if the current machine mode corresponds to a "BEAM" mode or a "NO BEAM" mode. (ON=BEAM/OFF=NO BEAM).
BST ready	Monitors the state of the TTCrx in charge of receiving and transmitting the BST message to the FPGA. (ON=TTCrx ready, received frame is consistent and can be decoded/ OFF=no consistent BST message).
BERR	Flashes when the RF2TTC generates a BERR. Not implemented yet.
VME	Flashes if the RF2TTC has replied to a VMEbus cycle

2.7. IMPROVEMENTS FOR THE VERSION 3

2.7.1. Initialisation procedure

The state of the Delay25 chips after a reset is not satisfactory, as the outputs are all disabled. The same for the TTCrx, which does not allow by default the transmission of the broadcast words. Finally, the DAC in charge of the threshold adjustments are set to -1.25V by default after power-up. Hence, the Delay25, TTCrx and DAC chips need to be initialised first, and it requires using some internal protocols (I2C or other), controlled by VME accesses.

A solution will be provided to ensure the board initialisation without using a crate processor. This initialisation will ensure that:

- All the delay25 chips are enabled (i.e. transmit the signals present at their inputs)
- All the DACs are configured correctly to allow latching the input signals if any
- The TTCrx chips is configured to transmit the BST message to the FPGA

2.8. REGISTERS SUMMARY

Name	Offset	Size (bits)	Access
BC1_MAN_SELECT	0x7FBFC	1	R/W
BC1_BEAM_SELECT	0x7FBF8	1	R/W
BC1_NOBEAM_SELECT	0x7FBF4	1	R/W
BC1_QPLL_MODE	0x7FBF0	1	R/W
BC1_DAC	0x7FBEC	8	R/W
BC1_QPLL_STATUS	0x7FBE8	2	R/W
BC2_MAN_SELECT	0x7FBCC	1	R/W
BC2_BEAM_SELECT	0x7FBC8	1	R/W
BC2_NOBEAM_SELECT	0x7FBC4	1	R/W
BC2_QPLL_MODE	0x7FBC0	1	R/W
BC2_DAC	0x7FBBC	8	R/W
BC2_QPLL_STATUS	0x7FBB8	2	R/W
BCref_MAN_SELECT	0x7FBAC	1	R/W
BCref_BEAM_SELECT	0x7FBA8	1	R/W
BCref_NOBEAM_SELECT	0x7FBA4	1	R/W
BCref_QPLL_MODE	0x7FBA0	1	R/W
BCref_DAC	0x7FB9C	8	R/W
BCref_QPLL_STATUS	0x7FB98	2	R/W
BCmain_MAN_SELECT	0x7FB8C	2	R/W
BCmain_BEAM_SELECT	0x7FB88	2	R/W
BCmain_NOBEAM_SELECT	0x7FB84	2	R/W
BCmain_QPLL_MODE	0x7FB80	1	R/W
BCmain_QPLL_STATUS	0x7FB7C	2	R/W
ORB1_MAN_SELECT	0x7FB6C	1	R/W
ORB1_BEAM_SELECT	0x7FB68	1	R/W
ORB1_NOBEAM_SELECT	0x7FB64	1	R/W
ORB1_POLARITY	0x7FB60	1	R/W
ORB1_COARSE_DELAY	0x7FB5C	12	R/W
ORB1_LENGTH	0x7FB58	8	R/W
ORB1_INT_PERIOD_SET	0x7FB54	12	R/W
ORB1_INT_PERIOD_COUNTER	0x7FB50	12	R
ORB1_COUNTER	0x7FB4C	32	R

Name	Offset	Size (bits)	Access
ORB1_PERIOD_RD	0x7FB48	12	R
ORB1_PERIOD_FIFO_STATUS	0x7FB44	2	R
ORB1_PERIOD_FIFO_RD	0x7FB40	16	R
ORB1_DAC	0x7FB3C	8	R/W
ORB2_MAN_SELECT	0x7FB2C	1	R/W
ORB2_BEAM_SELECT	0x7FB28	1	R/W
ORB2_NOBEAM_SELECT	0x7FB24	1	R/W
ORB2_POLARITY	0x7FB20	1	R/W
ORB2_COARSE_DELAY	0x7FB1C	12	R/W
ORB2_LENGTH	0x7FB18	8	R/W
ORB2_INT_PERIOD_SET	0x7FB14	12	R/W
ORB2_INT_PERIOD_COUNTER	0x7FB10	12	R
ORB2_COUNTER	0x7FB0C	32	R
ORB2_PERIOD_RD	0x7FB08	12	R
ORB2_PERIOD_FIFO_STATUS	0x7FB04	2	R
ORB2_PERIOD_FIFO_RD	0x7FB00	16	R
ORB2_DAC	0x7FAFC	8	R/W
ORBmain_MAN_SELECT	0x7FAEC	2	R/W
ORBmain_BEAM_SELECT	0x7FAE8	2	R/W
ORBmain_NOBEAM_SELECT	0x7FAE4	2	R/W
ORBmain_POLARITY	0x7FAE0	1	R/W
ORBmain_COARSE_DELAY	0x7FADC	12	R/W
ORBmain_LENGTH	0x7FAD8	8	R/W
ORBmain_INT_PERIOD_SET	0x7FAD4	12	R/W
ORBmain_INT_PERIOD_COUNTER	0x7FAD0	12	R
ORBmain_COUNTER	0x7FACC	32	R
ORBmain_PERIOD_RD	0x7FAC8	12	R
ORBmain_PERIOD_FIFO_STATUS	0x7FAC4	2	R
ORBmain_PERIOD_FIFO_RD	0x7FAC0	16	R
TTCrx_status	0x7FAA0	1	R
BST_Machine_Mode	0x7FA9C	32	R
BEAM_NO_BEAM_DEF	0x7FA7C	32	R/W

Name	Offset	Size (bits)	Access
WORKING_MODE	0x7FA78	7	R/W
ORB_INT_ENABLE	0x7FA6C	3	R/W
ORB_COUNTER_ENABLE	0x7FA68	3	R/W
PERIOD_COUNTER_ENABLE	0x7FA64	3	R/W
ORB_INT_RESET	0x7FA4C	3	W
PERIOD_COUNTER_RESET	0x7FA48	3	W
ORB_COUNTER_RESET	0x7FA44	3	W
TTCrx_REG (FIFO)	0x7E200	8	R
TTCrx_pointer to the data	0x7E004	8	(R)/W
TTCrx_pointer to the register	0x7E000	8	(R)/W
DELAY25_REG (FIFO)	0x7D200	8	R
ORBOUT_DELAY25_GCR	0x7D054	8	(R)/W
ORBOUT_DELAY25_ORBmain	0x7D048	8	(R)/W
ORBOUT_DELAY25_ORB2	0x7D044	8	(R)/W
ORBOUT_DELAY25_ORB1	0x7D040	8	(R)/W
ORBIN_DELAY25_GCR	0x7D034	8	(R)/W
ORBIN_DELAY25_ORB2	0x7D024	8	(R)/W
ORBIN_DELAY25_ORB1	0x7D020	8	(R)/W
BC_DELAY25_GCR	0x7D014	8	(R)/W
BC_DELAY25_BCmain	0x7D00c	8	(R)/W
BC_DELAY25_BCref	0x7D008	8	(R)/W
BC_DELAY25_BC2	0x7D004	8	(R)/W
BC_DELAY25_BC1	0x7D000	8	(R)/W
BCLEAR	0x00014	4	R/W
BSET	0x00010	4	R/W
PROGRAM ID	0x0000C	32	R
REVISION ID	0x00008	32	R
BOARD ID	0x00004	32	R
MANUFACTURER ID	0x00000	32	R

3. RF2TTC COMMON SOFTWARE

3.1. INTRODUCTION

Even though the RF2TTC performs the same task in each of the four LHC experiments it will be operated in H/W and S/W environments that are specific to the respective experiment. Therefore the common S/W is limited to the lowest level which consists of some diagnostic programs and a user library. This S/W is implemented in the rf2ttc package and can be found in CERN CVS repository at http://isscvs.cern.ch/cgi-bin/viewcvs-all.cgi/rf2ttc/?cvsroot=rf2ttc. For direct access from Unix use (e.g.) "setenv CVSROOT :kserver:isscvs.cern.ch:/local/reps/rf2ttc"

3.1.1. H/W Environment

In the ATLAS and ALICE experiments the RF2TTC is controlled by a VMEbus SBC from Concurrent Technologies (either a VP110 or a VP315). CMS uses a PCI-VMEbus link from CAEN and LHCb a USB-VMEbus link from the same manufacturer. In all four experiments the crate that houses the RF2TTC should be VME64X compatible as otherwise it is not possible use geographical addressing.

3.1.2. S/W Environment

On the low end ATLAS and ALICE will use the vme_rcc driver developed by ATLAS to communicate with the RF2TTC. CMS and LHCb will use S/W packages provided by CAEN for the respective interface. The common S/W will be programmed in a way that it is compatible with any of these bus access packages.

At the top end each experiment has to develop appropriate secondary libraries and applications to interface the RF2TTC to their respective control systems. The development of the interface to the DIP server is also up to the experiments.

3.2. TEST PROGRAMS

Currently there exists one program that comes in three flavours: rf2ttcscope_atlice (for ATLAS and ALICE), rf2ttsscope_cms and rf2ttcscope_lhcb. This interactive application can be used to read, decode and write any register of the RF2TTC in a (hopefully) intuitive way. If a users feels that some functionality is lacking he is welcome to contact the developer (M. Joos). It is e.g. possible to extend rf2ttcscope by additional command line parameters such that certain tests can be executed from scripts.

A tcltk graphical user interface (rf2ttc.tcl) is also available for test purpose. It has been written for SBCs from Concurrent Technologies (VP110 and VP315).

3.3. THE USER LIBRARY

This library consists of a common source file that implements the access to the registers of the RF2TTC in a generic way and a number of files to implement glue layers to the VMEbus access libraries from ATLAS and CAEN respectively.