An Emulator of Timing, Trigger and Control (TTC) System for the ATLAS End cap Muon Trigger Electronics

Y. Ishida, C. Fukunaga, K. Tanaka and N. Takahata
Tokyo Metropolitan University, 1-1 Minami-Osawa, Hachioji, Tokyo, 192-0397 Japan
M. Ikeno and O. Sasaki
KEK, 1-1 Oho, Tsukuba, Ibaraki, 305-0801 Japan

Abstract

We have developed a stand-alone TTC emulator system. This system simulates relevant TTC signals and their sequences needed for the ATLAS TGC electronics. Almost all functionalities are packed in an FPGA chip, which is mounted on the same board as one of TTCrx test board developed by CERN EP/mic group. The signal pin allocation is also the same as the TTCrx test board. Hence, instead of the test board, if the emulator board is mounted, TTC signals are generated and distributed consistently with this board without any modification of the mother board electronics system.

I. Introduction

In general a facility for TTC signal generation and distribution system is indispensable for development of electronics for an LHC experiment [1]. We need two VME modules [2,3], TTCrx chip [4], fiber optical/electronics converter, and software in order to implement full functionality of the TTC signal distribution system.

Electronics development of a particular sub-detector of an LHC experiment will be collaborated with several institutes of universities and laboratories. Although each institute will need more-or-less TTC signals, to facilitate such the TTC signal distribution system in every institute is unthrifty and inefficient.

The ATLAS Thin Gap Chamber (TGC) electronics development team consists of seven institutes from two countries [5]. More-or-less all the institutes will need at least a few restricted TTC signals for their electronics development.

The group has then developed a TTC emulator. The emulator has a functionality to emulate relevant TTC signals, which the TGC electronics system will use. Timing sequences among the signals are also emulated. In the final TGC system, a TTCrx chip will be mounted as a mezzanine card. As we have installed the emulator system into a small circuit mountable on the same size daughter board, physical consistency is also satisfied with the final TTCrx board. We discuss the structure of the emulator in the section 2, and show some emulator performance of several signals in the section 3. Finally in the section 4 we summarize the results.

II. Structure and Functionality

A. Hardware structure

The main functionality of the emulator is wholly packed in a Xilinx-FPGA of SPARTAN-2 (XC2S150). Beside the FPGA, a 40.08MHz clock generator, a PROM for storage of the FPGA firmware, a variable delay, several jumpers and a dip switch, Lemo connectors for external signal inputs and a JTAG connector for the FPGA configuration are mounted on a PC board. The size and the footprint of this PC board are the same as the TTCrx carrier test board (TTCrx test board) developed by the CERN EP/mic group [4]. The top face of the emulator board is shown in Fig.1.

Figure 1: TTC emulator board

We have used about 61% of the maximum number of available gates for the FPGA, which is 150 000. In order to extend the emulation behavior beyond the firmware contents, some signals can be inputted through the surface mounted Lemo connectors. One can operate an own electronics with specially adjusted TTC signals by inputting

them through the connectors. The jumpers and the DIP switch are used to switch signal sources (external or internal) or the emulation mode.

The 40.08MHz quartz oscillator is mounted to supply the default clock. The variable delay is used to emulate the signal called Clock40Des1. The skew of the clock signal is adjusted with this variable delay.

B. Emulation Firmware

Situation of Emulation is summarized in Table 1. Since the pin assignment of the emulator board is the same as the TTCrx test board, the situation of the emulation for the actual board is listed in the same way as the TTCrx board in the table. Since even if pins, which are not emulated, are connected to the FPGA, we can utilize these pins in the future firmware upgrade. In this subsection we discuss how the firmware emulates and handles the signals, and give emulation recipes of some signals.

Table 1: Situation of Emulation for TTCrx board

The emulated signals are indicated with "X" in the third column.

Connector J1			Connector J2		
Pin#	Name	Emulation	Pin#	Name	Emulation
1	Clock40	-	1	BrcstStr2	X
2	Clock40Des1	X	2	ClockL1A	-
3	Brcst<5>	X	3:4	Brcst<6:7>	X
4:6	Brcst<4:2>	-	5	EvCntRes	X
7	Clock40Des2	-	6	L1Accept	X
8	Brcststr1	X	7	EvCntLStr	X
9	DbErrstr	-	8	EvCntHStr	X
10	SinErrStr	-	9	BcntRes	X
11:12	Subaddr<0:1>	X	10	GND	X
13:18	Subaddr<2:7>	X	11:22	Bcnt<0:11>	X
19:22	DQ<0:3>	X	23:26	JTAG	-
23	DoutStr	X	27	I2C	-
24	GND	X	28	JTAG	-
25:32	Dout<0:7>	X	29	BCntStr	X
33	Reset_b	X	30	Serial_B	-
34	TTCReady	X	31:34	GND	X
35:50	GND	X	35:38	PIN Vcc	-
			39	N.C.	X
			40	I2C	-
			41:42	GND	X
			43:46	TTCrx Vdd	X
			47:50	GND	X

The bunch-crossing signal (BX), which is, although, not outputted from the emulator, is simulated in the firmware. The emulator generates the timing structure of BX as exactly the same as one that the actual LHC will supply [6]. The Level1 Accept (L1A) signal must be made coincidence with BX even if the signal is supplied externally.

L1A is generated in the firmware or can be inputted externally through the Lemo connector. One can select the trigger source with one of the jumpers mounted on the board. For the internal generation, seven different generation modes are prepared, which are the average frequencies of 100K, 10K, 1K, 100 and 1Hz of the random mode and the ones of 75K and 1Hz of the regular mode. The random mode means the L1A is generated with the time interval of Poisson distribution based on the given average rate. Bunch crossing ID (BCID<11:0>) is outputted together with L1A, and one (two) clock later from L1A, the Event Identification Number EVID<11:0> (EVID<23:12>) is outputted. Furthermore the Trigger Type<7:0>, EVID<23:16>, EVID<15:8> and EVID<7:0> are outputted on the Dout<7:0> with each clock after 4.4μs from the L1A generation. The SubAddr<1:0> is used to distinguish which data fragment is on Dout bus.

Although Clock40des1 is one of the total three LHC 40.08MHz clock signals in the actual TTCrx, this is only the one that the emulator will output. The clock deskew is emulated through the on-board variable delay with either the internal or external clock signal. The variable delay can adjust the delay timing of maximum 20ns in 40 steps. As we foresee the necessity of a variable delay of 25ns at most, we will implement such a delay in future. The clock signal can be generated by the 40.08MHz oscillator installed on the board or inputted externally through the Lemo connector. One can select the internal clock or external input with the jumper connection. The clock of different frequency rather than 40.08MHz can be supplied externally. But in this case the BX timing signal is scaled with the input frequency.

Since the emulator will not emulate Clock40 and Clock40des2, Brcst<7:6> are also synchronized with only Clock40des1 as Brcst<5:0>, whereas the actual TTCrx can set the synchronization of Brcst<7:6> optionally with the Clock40des2.

Among the Brcst<7:0> signals, the emulator simulates Brcst<1:0> and Brcst<7:5>. All these five signals are inputted through the Lemo connectors. A Lemo-input called as RST is shared with both Brcst<5> and Brcst<7>. We

have two kinds of the reset signal, one is for the whole electronics system except DCS (Detector Control System), and the other one is for the DCS reset. Although we should prepare two different Lemo connectors for these two reset signals, there is no more room to install one more connector on the board. It is foreseen to emulate hardly the DCS reset signal with this emulation board.

III. Performance

In Figure 2, the L1A signals of the frequency 100KHz randomly generated by the emulator are shown. In the scope image of this figure, one division of the horizontal axis corresponds to 10μs. The time interval between signals will be Poisson distributed with the average interval of 10μs.

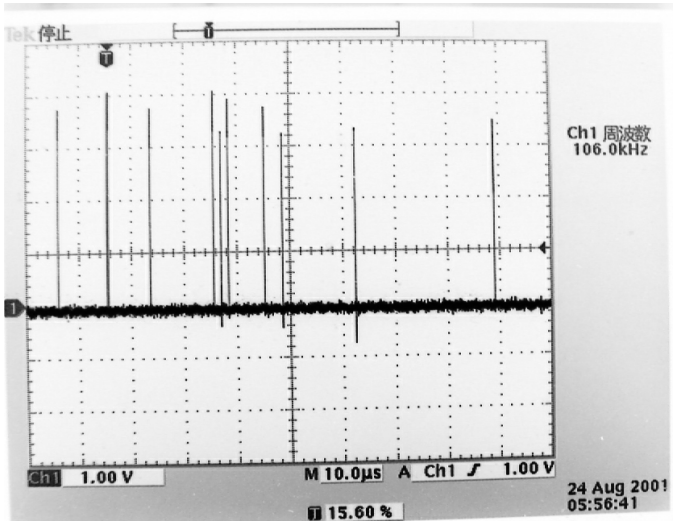

Figure 2: Internal L1A signal generation (100KHz random mode)

Figure 3 shows externally supplied L1A with the original pulse width of 350ns and the output one produced by the emulator. If an externally supplied pulse for L1A is longer or shorter than 25nsec, the emulator will adjust the width as 25 ns. The latency in the emulator from the input to the output pulse for the width modification is predicted as 53ns by the FPGA simulation. In Fig.3, the wider pulse indicated in the lower part is the input one, and the narrower one in the upper part is the output, and the horizontal division is 40ns. One can find the width of the output is 25ns, and the timing interval between the leading edges of both the input and output is measured as 53ns, although this actual value have uncertainty of a few 10ns due to the measurement setup. The emulator works anyway as the simulation indicated.

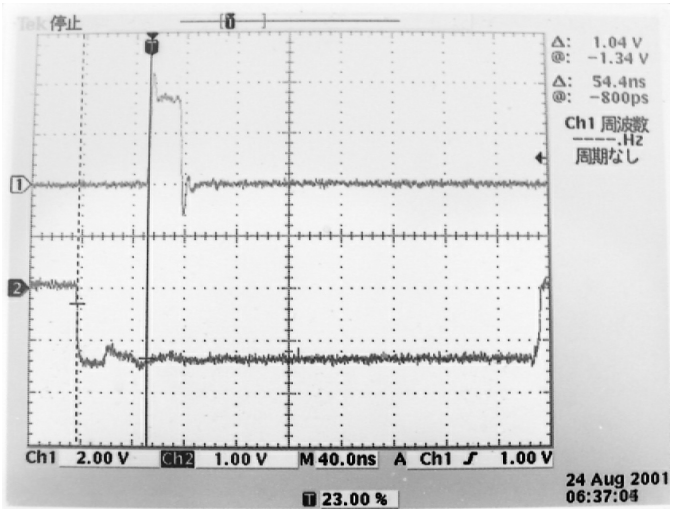

Figure 3: External L1A signal Input (Lower) and Output (Upper) through the emulator.

After L1A signal, BCID<11:0>, EVID<11:0> and EVID<23:12> are loaded on BCnt<11:0> lines periodically with the clock. BCntStr is generated when BCID<11:0> is loaded on BCnt. EvCntLStr and EvCntHStr are also generated when EVID<11:0> and EVID<23:12> are loaded on the BCnt respectively. The sequence of the signal generation has been observed with a logic analyzer, and we show its output in Fig.4. A typical timing sequence of relevant signals is shown for three L1A generations in the figure. From this figure we can confirm the emulation of BCID<11:0> and EVID<23:0> data loading on Bcnt<11:0> with three clock-intervals works fine.

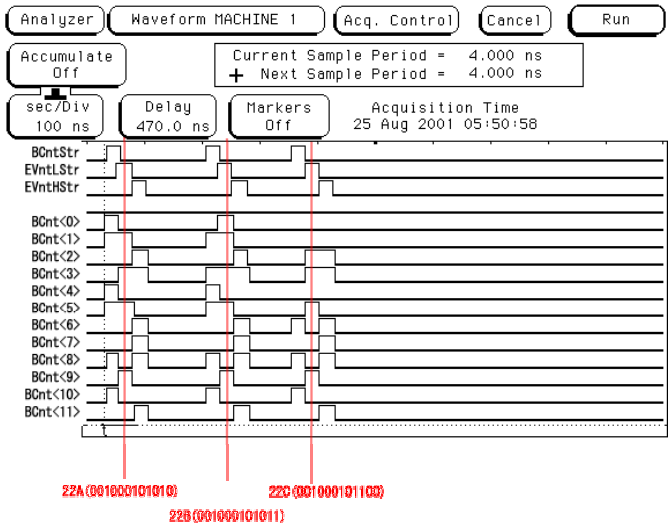

Figure 4: Data loading sequence on Bcnt<11:0>

There is another 24bit event/orbit counter, which is implemented in the TTCvi module. Together with the

trigger type parameter (8 bit), which is received from the Central Trigger Processor and stored in the module, these two data are broadcasted by the module through the B-channel of the TTC network to individual TTCrx after approximately 4.4μs of the L1A generation. The emulator also emulates this sequence because this emulator will be used for the trigger and timing generation at the developing stage of the ROD module.

In Figure 5 the Trigger Type and Event/Orbit counter output sequence observed by a logic analyzer is shown. The contents of the data are loaded on Dout bus with a bunch of 8bit. The data are loaded together with the DoutStr and SubAddr<1:0>. If SubAddr is 00, Trigger Type is loaded on Dout. The next three numbers with these two bits of SubAddr are assigned to indicate the Dout loading of three bytes of the event/orbit counter from the most significant to least significant bytes.

In principle in the original system, the event counter to be loaded on BCnt and event/orbit counter to be loaded on Dout must be different ones. In this emulator system, the same number is used for both two operations. Hence, the content of event/orbit counter will be reset by EvCntRes signal.

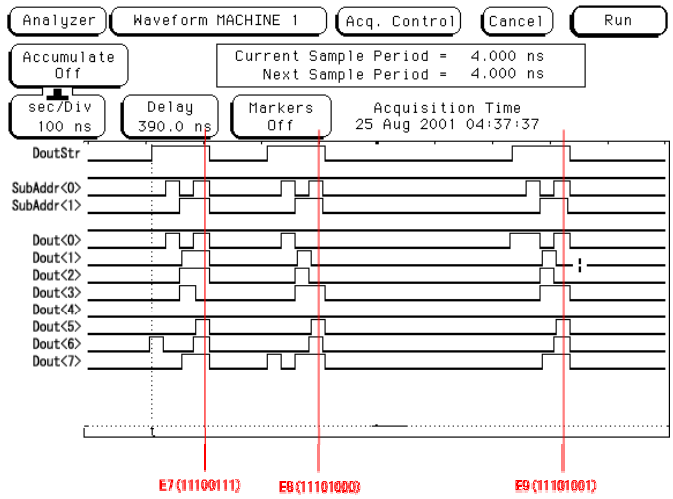

Figure 5: Data loading sequence on Dout<7:0>

IV. SUMMARY

We have made the TTC emulator. We have implemented the system on a board, which is the same dimension and same footprint as the TTCrx test board. In order to do that, most of the emulation logic is installed in an FPGA. With this emulator, signals needed for operation of TGC electronics are all generated. The signals successfully emulated are listed in Table 1.

Consequently we do not need a set of the whole TTC generation system at every development cite. As TTCrx boards are always used as mezzanine board in circuits in which the TTC signals are required in the current TGC electronics, instead to put TTCrx board, we can get all the necessary TTC signals by putting the emulator. With this emulator, we can save money, time as well as man power resource to prepare and maintain the whole TTC system.

Although the emulator has been customized for the TGC electronics development, it will be easily adopted by any other electronics systems if they use the TTCrx test board as a daughter one.

As all-in-one emulator system like the presently discussed one is easy to handle and easy to realize the complicated signal sequence, it will be useful for not only the development stage but also the production stage with various workshops.

IV. ACKNOWLEDGEMENT

This work was done in Japanese contribution framework of the ATLAS experimental project. We would like to acknowledge all of Japanese ATLAS TGC group as well as KEK electronics group for their supports and discussions throughout the work. We would like to express our gratitude to Prof. T. Kondo of KEK for his support and encouragement.

V. REFERENCES

[1] B.G.Taylor: “RD12 Timing, Trigger and Control (TTC) systems for LHC Detectors” Web page http://www.cern.ch/TTC/intro.html

[2] Ph.Farthouat and P.Gallno: “TTC-VMEbus Interface (TTCvi-MkII)” http://www.cern.ch/TTC/TTCviSpec.pdf

[3] P.Gallno: “TTCvx Technical Description and Users Manual” http://www.cern.ch/TTC/TTCvxManual1a.pdf”

[4] J.Christiansen et al.: “TTCrx Reference Manual ver3.2” http://www.cern.ch/TTC/TTCrx_Manual3.2.pdf”

[5] K.Hasuko et al.: “First-Level Endcap Muon Trigger System for ATLAS”, Proceedings of LEB2000, Cracow, Poland, Sept.,2000, pp328.

[6] P. Collier: “SPS Cycles for LHC Injection” http://sl.web.cern.ch/SL/sli/Cycles.htm