PATTERN UNIT FOR HIGH THROUGHPUT DEVICE TESTING

G.M. Dallavalle, I. D'Antone, I. Lax, A. Montanari and F. Odorici (email: odorici@cern.ch)
I.N.F.N. Bologna

Abstract

A high speed and high throughput (1.3 Gbyte/s) programmable Pattern Unit has been implemented in a single 6U VME board. Such a device represents a basic tool for testing the high throughput devices that are adopted in trigger and readout systems of LHC experiments.

1. MOTIVATIONS

The current phase of research and development of trigger electronics for the CMS experiment at LHC includes the design of complex electronics systems, based on custom-made digital ASIC devices. This implies the production and subsequent tests of ASIC prototypes. The main requirement, in order to perform standalone test of such prototypes, is to dispose of a Pattern Unit, i.e. an input-output device, fulfilling the following constraints:

- large number of I/O channels (>100);
- clock rate multiple of LHC rate (40/80...MHz);
- clock – I/O synchronisation;
- pre-storage/readout of I/O patterns.

At the INFN section of Bologna, when we started working on this project, we immediately realised that with a small additional design effort a more general purpose tools, allowing to test trigger electronics and readout systems in a real-time set-up (typically a test beam), would be achieved. For such a device some of the additional constraints are:

- external clock in the range 0÷80 MHz;
- external control signals: Start/Abort/Clear...
- mixed standard ext. signals: NIM/ECL/TTL...
- event counting;
- interrupt generation.

With this target in mind we developed a programmable 6U VME board (one slot) which is able to generate and readout up to 16 Kwords on 128 I/O channels at 80 MHz.

2. FUNCTIONALITY

2.1 Main features

I/O channels are based on 8 independent FIFOs (CY7C42x5), each chip handling 16-bit words. All FIFOs work in parallel, i.e. with the same clock, and can be configured as input or output-FIFOs individually. Read and write operations are allowed in fully asynchronous and simultaneous access through front-panel connectors and VME bus. Therefore the Pattern Unit can work either as a Pattern Generator or a Readout Buffer of "events", or perform both functions simultaneously in a single board. The latter configuration gives advantages and is typically suitable when testing prototypes of synchronous devices. A pattern unit event represents a word containing up to 128-bits, as multiple of 16-bits. The total number of events to handle is controlled through a pre-settable event counter ranging from 0 to 16 Kevents. The event counting begins when a Start signal is issued and stops when the countdown reaches zero or when an abort signal is generated. This working mode is called "data sequence". Another feature, useful for testing purposes, is to make the event counter working in "continuos loop" mode: the FIFO address pointer is automatically reset to zero after reaching the end of transmission condition (full/empty FIFO), i.e. input-FIFOs are virtually never full and output-FIFOs are never empty. As a general feature, input-FIFOs ignore events when they are full, while output-FIFOs don't forward events when they are empty. A Clear signal is necessary to empty input or output-FIFOs. The event flow has a maximum I/O clock rate of 80 MHz.

The board clock can be provided either internally or by an external signal. Some high-resolution delay lines are adopted in order to distribute the clock signal, differently for input and output-FIFOs. This allows keeping synchronization among I/O patterns when and intermediate device, for example a chip under test, is placed between input and output-FIFOs.

Since the Pattern Unit has been conceived not only as a laboratory-testing tool but also as a real-time device, it can be adopted in experimental environments like a test beam set-up. For this reason all asynchronous control signals necessary for a "run" (Start, Abort, Clear) can be issued via front-panel LEMO-00 type connectors. In addition, a Gate signal, representing a temporal window of valid clocks, is also available.

Most configuration options for the pattern unit working-mode, as well as the board status, can be fully controlled through VME. A dedicated JTAG bus (4 lines) is also available for external chip programming.

2.2 Usage examples

As already said the Pattern Unit can work either as a Pattern Generator of events or a Readout Buffer, or perform both functions simultaneously. Let's consider the first application. In this case some or all FIFOs are configured as Output, and a given number of patterns can be loaded into the FIFOs via VME. After pre-setting the event counter (VME register), the Start command (either VME or external) allows a given pattern sequence to be sent to the output connector, at a given clock rate. Only an Abort or a Clear (either VME or external) can interrupt the sequence. The Abort stops the event counter (if it is running) and (normally) re-initialize it to zero. If a Clear is issued also the internal FIFO pointers are reset to zero, unless the "Continue" mode is set. In this case, at the next Start, the readout of the FIFO continues from where it previously ended. The Continue mode allows therefore to readout the FIFO content as "data packets". When Continue mode is not set, Clear resets the FIFO pointers but the data content is still valid and readable. If the event counter runs for a number of events greater than the number of words loaded into the FIFO, then the output word will always be the last pattern loaded.

When the board is used as a Readout Buffer some or all FIFOs are configured as Input. After presetting the event counter, the Start command allows a given pattern sequence to be written into the FIFOs through the front-panel input connectors. An Abort or a Clear command can interrupt the writing sequence before the event counter reaches the preset number of events. While the Abort simply stops the event counter and reset it to zero, the Clear command acts as the Abort but also re-initializes the FIFO pointers to zero and cancel the FIFO data content. The Continue mode doesn't affect the input-FIFOs behavior. The FIFO readout is done via VME: the same number of words written on the FIFO can also be read, until the FIFO is empty. If the VME readout exceeds the number of words written into the FIFO, always the last written word will be received.

As a general feature, if a Start is issued during a run (i.e. before the event counter is arrived at zero), it is ignored. Input-FIFOs that are full ignore next events. Output-FIFOs that are empty always give the last loaded event. After a Clear all output-FIFOs give zero on the output bus. For the Start and Clear commands an enable-mask allows to decide which FIFO will be affected by the above signals. The event counting can work in "Loop" mode: the FIFO address pointer is automatically reset to zero after reaching the end of transmission condition (full/empty FIFO), i.e. input-FIFOs are never full and output-FIFOs are never empty.

2.3 Synchronisation

Several synchronization aspects have been taken into account in the design of the board:

- clock-*vs*-data;
- clock-*vs*-control signals;
- data-*vs*-data;
- board-*vs*-board.

In practice all these aspects have been overcome by including on board programmable delay lines, which allow adjusting the phase among different signals. Two delay lines are inserted on the clock paths associated to the input- and output-FIFOs respectively. In this way clock-*vs*-data synchronization can be tuned within a 5-bit range, i.e. with a granularity of about 7% per step.

A third delay line is inserted on one of the front-panel control signals, typically the Start, to adjust the phase with respect to the clock.

The necessity to obtain a given data-*vs*-data synchronization could occur for example when some FIFOs are used as control lines of a given (external) device, while other FIFOs are used as data receiver or transmitter. In this case, control lines will probably change at a lower rate than I/O data. That can be easily achieved via software, by writing the same control patterns into some consecutive FIFO locations.

When two or more pattern unit boards are chained together, a board-*vs*-board synchronization can be obtained on each board by taking as input the output clock provided by the previous board of the chain, and also by adjusting the on board delay lines associated to input- or output-FIFOs.

For an application that needs more than 128 I/O channels, two or more boards can be configured to work in parallel. This requires a board-*vs*-board synchronization that can be easily achieved by feeding the same clock to all boards. Moreover, all boards' delay lines are programmed to add some extra delay, so that a fine phase tuning can be obtained on each board.

3. CONTROL AND CONFIGURATION

3.2 VME control registers

Most of the board controls and setting, as well as the VME interface, are implemented or handled through an FPGA device (XC3195) so that further functional features can be added in future. The 4 JTAG lines correspond to individual VME registers. The VME access is a standard 32-bit addressing with 16-bit data. The list of VME control registers is reported in Table 1.

Table 1:VME control registers

Registers	Description
1 (w-only)	FIFO Configuration: Input/Output
1 (r-only)	FIFO Status: I/O; empty/full
8 (r/w)	FIFO data access
1 (r/w)	Run Mode: Sequence/Loop/Data-blocks
1 (r/w)	Run Status: running/paused
1 (r/w)	Event counter
4 (r/w)	VME signals: Start,Abort,Clear,HWReset
3 (r/w)	VME signal enable-mask (1-bit/FIFO)
1 (r/w)	Front-panel I/O enable: 3-state/active
1 (r/w)	Front-panel signal enable
1 (r/w)	Delay lines programming
4 (r, w)	JTAG lines

3.3 Front-panel signals

The 128 I/O signals are TTL levels (or TTL differential by inserting adapters), arranged on 4 x 40-pin connectors. All control signals are fed through LEMO-00 type connectors, as TTL, NIM and ECL logic levels. The input clock can be accepted also as a PECL level. The input signals are Clock, Start, Abort, Clear and Gate. The only output signal is the Clock, which can be used to synchronize further boards. An 8-pin connector is provided for handling JTAG signals. Some LEDs are placed on the front panel for a visual check of the FIFOs I/O configuration.

3.4 Jumpers' setting

The board has several jumpers on it which, during an application, are normally configured as a "definitive" setup. Some of them are worth to be mentioned:

- the internal or external clock selection;
- the "routing" of the clock lines, which has to be defined according to the needed delay lines;
- control signals level selection (TTL, NIM ...);
- each input line can be terminated with on-board resistors (pull-down and/or pull-up) in order to properly match the cable impedance;

3.5 Software programming

The aid of a control program is necessary for several reasons. Firstly in order to read/write the VME registers dedicated to the board configuration and in order to get access to the FIFO data locations; secondly to test the board functioning and also to determine the proper delay lines configuration, for example through automatic algorithms based on reference sequences.

4. CURRENT APPLICATIONS

4.1 Standalone test of ASIC prototypes

The first application of the Pattern Unit was for a standalone test of the "Track Sorter Slave" (TSS) ASIC prototype (Bologna, May 1998, see A. Montanari's talk in these proceedings). For this custom-made digital CMOS device the nominal clock rate is 40 MHz. The chip needs to get 4 input words of 10-bits each and 10 control signals. It gives a 10-bit output word. We therefore used 4 FIFOs to provide the ASIC input data, one FIFO to handle the control lines and one FIFO to get the ASIC output data. We mounted the ASIC device on a very simple "connection board" which has been located as close as possible to the Pattern Unit. The link between the two boards was done via standard 0.5 pitch flat cables (100 Ohm) 30 cm long, properly terminated at their ends. We used an external 40 MHz TTL clock, connected to both the Pattern Unit and the ASIC device. Transmitting and receiving FIFOs were phased with the clock through two on board delay lines (2nsec/step, 5-bit range).

The chip behavior was verified with input test patterns and the output compared to the wanted data. Once the right behavior of the ASIC was proved, the chip processing speed was determined by increasing the clock rate until wrong data started to appear (about 70 MHz in this case). By using the JTAG registers available on the Pattern Unit, also the JTAG logic that is part of this ASIC prototype has been verified.

4.2 Real time test of ASIC prototypes

The next application of the Pattern Unit was a real time test of the "Bunch and Track Identifier" (BTI) ASIC prototypes (CERN, August 1998, see P. Zotto's talk in these proceedings). Also these CMOS digital devices are custom-made and the nominal clock rate is 80 MHz. In this case the chip test was combined with the test of other objects involved in a quite complex trigger system of a test-beam data taking. Inputs to the ASICs were real data and only the ASIC output words were recorded. All 128 channels of a Pattern Unit were used to get data in phase with an external PECL clock at 80 MHz. The clock was left free running and a start to the data acquisition was given by an external NIM signal, provided by the trigger logic. The start was properly synchronized to the clock through an on board delay line, while clock-*vs*-data synchronization used a second delay line (both lines with 1 nsec/step, 5-bit range). Between two starts signals a bunch of about 4000 words were read out and a FIFO clear issued. Also in this case the BTIs were mounted on a dedicated connection board, linked to the Pattern Unit through standard flat cables.

At high clock rates, short connection lines between the Pattern Unit and the chip under test, for example through a piggy-board, would improve signal propagation. A pin grid array, present on the Pattern Unit board and conceived for connecting arrays of resistors (to adapt the I/O impedance), can be used for connecting a piggy-board.

5. SUMMARY

A programmable 6U VME board, with the functionality of a high speed and high throughput Pattern Unit has been developed. Already the first version of the Pattern Unit board permitted to fulfil the project goal, that was to design a general tool for testing digital ASIC prototypes, in standalone and real time applications, up to 80 MHz.