The Trigger-Time-Event System for the W7-X Experiment

Jörg Schacht, Helmut Niedermeyer, Christian Wiencke, Jens Hildebrandt and Andreas Wassatsch

Abstract-- All control and data acquisition systems of the fusion experiment W7-X need to be perfectly synchronised with an accuracy of ≤ 10 nanoseconds. Another essential requirement is to process and record events and trigger signals in such a way that they can be used for real-time control.

This paper describes the architecture of a Trigger-Time-Event (TTE) system of the W7-X experiment and gives an overview of the characteristics of the special board of the local TTE units. The main components of the board are a local oscillator synchronised by a central timing unit, and a field programmable gate array (FPGA). The FPGA comprises an event and trigger command receiver/decoder, a local 64 bit time counter, time capture registers, delay devices, a programmable state machine, trigger generators, and a programmable switch matrix, which allows to interconnect output and input signals of the devices. The time, fast trigger, and event information are distributed from the central timing unit to the local TTE boards via a tree type optical network. For hardware trigger signals an electrical network may be used. The board has configurable I/O ports for trigger signals and a 64 bit time port.

I. INTRODUCTION

The stellarator experiment WENDELSTEIN 7-X (W7-X) is a large experimental device, presently being constructed in Greifswald at the Max-Planck-Institut für Plasmaphysik. W7-X is integrated in the European Fusion Programme and aims to explore and demonstrate the potential of the stellarator principle for a nuclear fusion power reactor. Unlike most fusion devices, W7-X can operate in a steady state. This means that the experiment can produce discharges with a duration of up to 30 minutes. The experiment has the following main parameters [1]:
- major radius of the vacuum chamber is 5.5 m,
- coil system with 50 non-planar and 20 planar superconducting coils,
- maximum magnet field on axis of 3 Tesla,
- installation of 10 MW ECR, 4 MW ICR and 5 MW NBI heating.

The W7-X control system consists of a master control unit and local controllers for all subsystems such as diagnostics, data acquisition, magnetic system, heating units, and cryogenic units. All periods of operation will be subdivided into segments of variable duration. A segment programme defines the operational rules and parameters of each unit in use. Well in advance the information for each segment is stored in a database. Before the start of an experiment cycle all defined segments for this cycle are sorted in a segment-sequence. After each unit has loaded it's specific information and generated the necessary control objects, the master control system can start the cycle. Segment processing and fast feedback control will run under the real time operation system VxWorks. Programmable Logic Controllers (PLCs) will be used mainly to control the technical components and diagnostic systems. Some parts of the device have to be monitored continuously in order to control the segment processing. All acquired data from the diagnostic and control systems are matched by precise timestamps and will be stored in a centralised data archive. A distribution system for trigger, timing and synchronisation information, event messages and the accurate time therefore is a key component of the control system.

II. STRUCTURE OF THE TRIGGER-TIME-EVENT SYSTEM

The Trigger-Time-Event system (TTE) is an independent system, which closely interacts with the control systems of the experiment. It consists of one central and many local TTE systems. The requirements to the response time and precision of timestamps of the local systems are very different. Short response times with data processing in real time and a time resolution in a range of 10 ns are essential for data acquisition systems, segment processing, and fast feedback control. Most control systems for the technical components and diagnostic systems, based on PLCs, are less demanding. Because the cycle time of a PLC often exceeds 10-50 ms, time resolution and time accuracy can be much lower as for real time computer systems. The local TTE system, consisting of a computer board and software components for trigger, time and event message tasks, is integrated into the control systems of the technical components. The time, trigger and event related messages are broadcast from the central TTE system to the local units through a unidirectional fibre network. Units not equipped with a local TTE board, like PLCs or computers with low requirements for the precision of time stamps, can receive the same information.
through a dedicated Ethernet. Trigger signals with high requirements for the response time or the reliability will be distributed with hardware trigger lines. The number of connected TTE units is flexible and not limited. Figure 1 shows the structure of the TTE system.

The central TTE system operate as follows:
A very stable oscillator produces a 100 MHz reference clock signal, which will be used to clock the central 64 bit time counter. If necessary, the central time counter can be synchronized with the time information from a Global-Position-System (GPS) receiver or other time information systems. Redundancy is necessary within the central TTE-unit in order to achieve the very high reliability necessary for the central clock of the experiment. For synchronisation of the local components the clock signal and the time information are distributed over the fiber-optic network, the time information additionally over the Ethernet. Event messages such as information about the current status of the experiment (e.g. plasma events, control status, safety status) are received from the central control system and distributed via the fiber-optic network to all local systems. Across the lines of the trigger network the central timer can receive or transmit fast trigger signals.

For real time computer systems a special card is being developed. These local TTE units have the following properties:
A quartz-stabilised oscillator of the local TTE system drives a 64-bit time counter. By means of the clock signal distributed with the fibre network it can be phase locked to the oscillator of the central timer. In regular intervals the value of the local counter is compared to that of the central counter transmitted on the fibre network and set to this values if both differ. The transmission time on the network is compensated by a special technique. Event messages will be received and decoded. The reaction of the card on an event message is defined by the user. The local TTE unit has several ports to transmit signals. Hardware ports can be configured as inputs or outputs of trigger. A 64 bit time port allows the transmission of the actual time counter value. The local TTE unit consists of special units for digital signal processing. These units are, among other things, timer/counter, time capture registers, pulse generators, a programmable state machine, and I/O triggers. The properties of these signal processing units and the structure of the signal processing are programmable.

### III. TTE CARD OVERVIEW

The prototype of the TTE card has been developed as a PCI bus computer card. Figure 2 shows the main components of the card. AMCC’s S5933 is a powerful flexible PCI controller and serves as a PCI bus target. Bus transactions between the host and the TTE-core are executed across the S5933 pass through interface. All TTE system functions are integrated into a FPGA Xilinx Virtex 1000E. This combination in one chip allows a simple update for other bus interface types in connection with further developments. By implementing the complex function of the device in a FPGA, production time and costs for the integrated circuit are minimized. Design may be carried out very fast, because the Virtex FPGA is re-programmable. The TTE devices are functional units for special time, trigger or event message tasks. They can be configured through registers and can be controlled both with device hardware signals and/or software commands through the PCI interface. All TTE devices types and their functions are summarized in table I.

Figure 3 depicts a generic TTE device model. Over the bus interface the user chooses the desired device behaviour by setting flags of the device configuration register.

The configuration register values determine, for example, the active input and output signals, the pulse frequency, the trigger mode, and the trigger signal length.
Either device can initiate a bus interrupt, when its status is active. The input status of a device is determined both by the hardware input signal and the control flags.

The device transfer function defines the type of TTE device (alarm timer, pulse generator,) and is not changeable. If the conditions for an active output signal are fulfilled, the output signal switches from inactive to active. The status of output signals (device active, output signal active) is determined by the output register status. For tests the user can set both the input and output status by the bus interface. This permits to evaluate the complex TTE card functions in a simple manner.

<table>
<thead>
<tr>
<th>device type</th>
<th>number of devices</th>
<th>function</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 bit time counter</td>
<td>1</td>
<td>local timer with 10 usec resolution (prototype: 20 usec)</td>
</tr>
<tr>
<td>alarm timer</td>
<td>1</td>
<td>generates alarm signals in a absolute or relative alarm time mode</td>
</tr>
<tr>
<td>time capture</td>
<td>4</td>
<td>latch time value for an event (signal or set a control flag)</td>
</tr>
<tr>
<td>pulse generator</td>
<td>2</td>
<td>generates pulses with variable rates</td>
</tr>
<tr>
<td>delay timer/counter</td>
<td>4</td>
<td>delay timer mode: generates a delayed trigger pulse with programmable shapes; counter mode: count pulses</td>
</tr>
<tr>
<td>I/O trigger</td>
<td>8</td>
<td>input or output pulses with programmable shapes</td>
</tr>
<tr>
<td>programmable logic</td>
<td>4</td>
<td>resolves a free programmable boolean function with 4 input and 1 output signals</td>
</tr>
<tr>
<td>finite state machine</td>
<td>1</td>
<td>a free programmable state machine with max. 16 states</td>
</tr>
<tr>
<td>event trigger</td>
<td>8</td>
<td>received event messages can generate max. 8 trigger signals</td>
</tr>
</tbody>
</table>

The alarm timer device generates a signal at the output Alarm Q/Qneg, if the current time value equals the preset alarm time value. The alarm time has to be written in the register Alarm Set Value. The alarm timer works with two different time modes.

1. **Alarm Mode 0:** The Alarm Set Value is an absolute 64 bit value for the alarm time.
2. **Alarm Mode 1:** The Alarm Set Value is a relative 64 bit value for the alarm time. When the alarm timer is starting, the absolute alarm value is calculated by adding Alarm Set Value to the current time counter value.

After starting the alarm timer the user can set a new value for alarm time for the next alarm cycle without interfering with the current alarm function.

**Time Port**

The local time value can be written out through this 65 bit port (64 bit time value and 1 bit for control: Time Port Ready). The update frequency is variable.

**Time Capture**

The use of the time capture devices allows to generate a timestamp when an event occurs. An event can be a hardware signal or flag set operation. The current time value is written in a 64 bit register named TCValue.
Delay Timer/Counter
The delay timer/counter supports two modes:

1. delay timer mode:
   When the delay timer is started, a single trigger signal or a periodic pulse chain will be generated after a delay time depending on the setting of a flag. The 48 bit value DTCSetValue defines the delay time.

2. counter mode:
   Incoming pulses on the input port are counted. The number of pulses is written in the register DTCCountValue. The value DTCSetValue in counter mode is the value for the overflow of a counter. A trigger signal is generated when an overflow occurs.

Pulse Generator
The pulse generator provides a periodic pulse signal. The frequency can be programmed with the value PGSetValue. It is possible to synchronise the phase of this generator with a time mark signal from the master time unit by setting a flag PGPhaseSychCtr.

I/O device
The connections between the TTE devices and external electronics can be using the I/O devices. The flag I/ODir sets the signal direction of the device. The user can change between 5 modes of input signal processing. Table 2 describes these modes. When desired, the input signal can be filtered. The filter value is stored in register I/OFilterValue. In mode 2 an I/O device can work with or without re-trigger attributes.

<table>
<thead>
<tr>
<th>mode</th>
<th>description</th>
<th>means</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/OMode0</td>
<td>PassThrough Mode</td>
<td>Input signals are put through directly from input to output</td>
</tr>
<tr>
<td>I/OMode1</td>
<td>Latch Mode</td>
<td>Active input signals set the output signal to active. The status of the output can only be reset by setting flags 10 to Reset or I/OCmdStat Reset.</td>
</tr>
<tr>
<td>I/OMode2</td>
<td>Trigger Mode</td>
<td>When an active input signal occurs the output signal is set to the active status for a predefined time (register I/OTriggerSignLength).</td>
</tr>
<tr>
<td>I/OMode3</td>
<td>Start/Stop Mode</td>
<td>The first active signal edge set the output to the active state. The next active input signal reset this signal.</td>
</tr>
<tr>
<td>I/OMode4</td>
<td>Edge Detector</td>
<td>Every signal edge sets the output signal for a duration of one clock cycle.</td>
</tr>
</tbody>
</table>

Event Trigger device
Special messages for control tasks, so called event messages, can be transmitted through a) the optical network or b) the Ethernet from the central TTE unit (variant a and b) or every technical component with an Ethernet port (only var. b). These messages use the simple protocol shown in figure 4. The event is characterised by a predefined event number (0-1023) and can include a maximum amount of 14 parameter bytes. Two stages for event message processing are provided on the TTE board as shown in figure 5. The user of a TTE card can define the reaction on each of the 1024 event messages. By setting flags in a 1024x9 bit transfer matrix one can enable up to 8 trigger devices and write the message into a FIFO. Via the bus interface all FIFO entries can be transferred into the host RAM so that very specific reactions depending on the parameters can be programmed in the host. Using the connection matrix (see below) the output of the event trigger devices can be connected with arbitrary TTE device inputs.

![Fig. 4. event message protocol](image)

![Fig. 5. event message processing](image)

Logic device
The TTE FPGA includes four devices for logical applications. This devices is characterised by 4 input and 1 output signals. The free programmable boolean transfer function is stored in a memory area of this devices.
Finite state machine
In addition to the logical devices the use of the Finite State Machine (FSM) facilitates a very flexible signal connection with time dependencies. The FSM processes input signals and information about the current state for determining a new state. The FSM works on the principle of a MOORE automate. The on board FSM allows to define a state machine with a maximum of 16 states. Figure 6 depicts the principle of the FSM on the TTE FPGA. The FSM uses a dual ported RAM area.

The connexion matrix
Potentially users of the TTE card have very different requirements for the signal flow between the devices on the board. A connection matrix permits to interconnect the output of the TTE devices with the input of another device in a flexible way making it possible to design complex structures for trigger, time and event based tasks. The matrix defining the connections between input and output has 48 rows for outputs and 58 columns for inputs (see figure 7). A device input can be connected to one device output only, but one output can have many connection to inputs.

V. STATUS
The TTE system for W7-X at the moment is in a state of ongoing development. The program for the FPGA runs faultlessly with a clock frequency of 50 MHz in a simulation and on a prototype board. A standalone version of a PCI board will be available by mid 2001. A basic driver for Vx-Works (Wind River Systems Inc.) and an interactive software tool for initialising the very complex unit were developed in co-operation with the University of Applied Sciences at Stralsund, Germany.