A DISTRIBUTED TIMING SYSTEM FOR SYNCHRONIZING CONTROL AND DATA CORRELATION *

Matthew Stettler, Michael Thuot, Leo R. Dalesio, Roger Cole, Charles Fite,
Gale Slentz, David Warren
University of California, Los Alamos National Laboratory
P.O box 1663
Los Alamos, NM 87545

ABSTRACT

Synchronization is necessary in experimental physics machines to provide positive control over related events. The Ground Test Accelerator (GTA) timing system provides this function through a distributed control system, known as the Experimental Physics and Industrial Control System (EPICS). The EPICS timing system was designed to take advantage of a distributed architecture, and provides time stamping for synchronous data correlation as well as event control. The system has been successfully demonstrated on over a dozen controller nodes for operation and data analysis. The design of the hardware, software, and operational results are discussed.

INTRODUCTION

The Experimental Physics and Industrial Control System (EPICS) is a hierarchical, distributed system of controls [1]. Real time control is provided by a network of I/O controllers (IOCs), VME systems which directly control beamline hardware. Access to IOCs is provided by operator interface tools (X–window based) and other applications by channel access software [2]. All control system hardware nodes are interconnected by TCP/IP ethernet. Due to the asynchronous nature of TCP/IP ethernet, a dedicated timing system system is required to provide synchronization and data time stamping.

Architecture

The implementation of the EPICS timing system utilizes a central master timing device, which generates and distributes timing signals to subsystems via an optical distribution network. The distribution network is configured as a star, with dedicated fibers for each subsystem controller (IOC).

Four types of timing signals are defined, The system clock, system trigger, time tag clock, and time tag reset. The system clock is the reference for all timing events in the system. The system trigger synchronizes control and data acquisition. The time tag clock and time tag reset provide a means to tag acquired data with a unique event number and time stamp.

User timing events are defined as programmable one–shots retriggerable by every system trigger, or multiples of the system trigger. Both the delay from the system trigger to

the user event and the duration of the user event are programmable. The programming of user timing events of this type is directly supported by timing hardware and EPICS software tools.
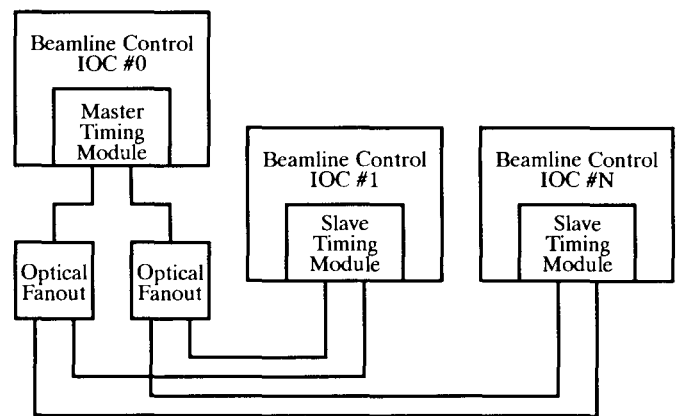

Figure 1 EPICS Timing System Architecture

HARDWARE IMPLEMENTATION

The EPICS master and slave timing modules implement a distributed timing system. Each device is packaged as a standard VME module, allowing timing functions to be implemented in a single slot in each IOC. Since many user timing requirements can be met by a system with 200nS resolution, 10 channels of user programmable timing channels of this resolution are provided on each device. Higher resolution is achievable with external hardware. Additional VME timing modules can be added to provide large numbers of user timing channels.

Hardware Internals

Four signals are distributed, a 5MHz system clock, a time tag clock, a time tag reset, and a system trigger. The generation rate of all signals except the 5MHz clock are programmable. All timing signals originate on the master timing module, and are distributed to multiple slave modules through optical fanout modules (star topology). The 5MHz clock is distributed on one fiber, while the remaining signals

---

are multiplexed over a second fiber. In addition, each timing module provides 10 channels of programmable counter/timer hardware for use in generating user timing signals. All timing signals are provided on the front panel through a 50 pin ribbon cable. The clock and system trigger are also provided through Lemo coax connectors.

The 5MHz clock is intended as the global system time base. The user programmable counter/timer hardware is directly connected to it, and the system trigger is synchronized to it on the slave module. All other timing signals are synchronized to the falling edge of the clock at the master. This insures 100nS of propagation delay tolerance between the two distribution fibers. The limiting factor on the system clock at this time is the speed rating of the counter/timer chips which generate the user timing signals (7MHz). This signal is available as a high drive TTL signal on both a front panel 50 pin header and coax lemo.

The time tag clock and time tag reset signals control two 16 bit counters intended for time stamping. One counter is free running, and one is latched at each system trigger. The source of these signals is programmable at the master between the 5MHz system clock and an external source. The timing modules can be programmed to interrupt upon counter reset. Both the time tag clock and reset are synchronized at the master to the falling edge of the system clock, but due to the asynchronous nature of the timing signal decoder, 50nS of jitter is to be expected on the slave. The frequency limitations of these signals are directly related to the design of the timing signal decoder, a conservative estimate is 100KHz.

The system trigger is used as the gate signal to the user programmable counter/timers, and is intended to be the main synchronizing event in the system. It is synchronized to the rising edge of the system clock at each slave to < 20nS. The synchronizing process delays the signal by 400nS (two clock ticks). The timing modules can be configured so that the system trigger causes an interrupt. This signal is available as a high drive TTL signal on both a front panel 50 pin header and coax lemo.

The 10 user programmable counter/timer outputs are intended as general purpose timing outputs. They are provided by two AMD 9513A system timing controller chips. These devices contain 5 programmable 16 bit counters with numerous operating modes. A complete description of the operating modes is beyond the scope of this document (reference the Am9513A technical manual for details). Each timing chip has access to all timing signals and an external input as source inputs. All gate inputs of each chip are connected to the system trigger. Buffered outputs and external inputs are available on a front panel 50 pin header. Six of the 10 timing channels (1-3, 6-8) can be configured individually to interrupt. The counter/timer chips and their associated control/interrupt hardware are logically separate from the time stamp/system trigger logic and can be treated as separate devices.

SOFTWARE IMPLEMENTATION

The EPICS software tools provide easy user access to the features of the timing system. Users define database entries that control timing parameters. All user timing signals are produced relative to the system trigger, but may be specified as relative to another user timing signal. The delay from the reference event, the pulse width, and the active sense (logic "1" or "0") are entered into a database configuration tool and the resulting database is downloaded into the IOC at boot-up. The fields of these database records can be changed during operation by a network access using "channel access" routines. These routines allow database records to be accessed by name across the network, and are generally called by operator interface software.

Record Types

The most basic timing signal is a simple pulse referenced to the system trigger. Entries in the database are made for channel #, time units (eg. microseconds), delay until trigger, and pulse width. In the following example, timer_22:0 is specified to occur 3uS after the start of system trigger.

Process Variable: timer_22:0 *Type: timer Node: ioc 22*

Field Name	Entry	Specifies
SCAN	I/O Intr	*Scan on interrupt*
EVNT	0	
TORG	0.00	*Source: system Trigger*
TYPE	8309	*Card Type*
OUT	#C0 S0	*Card and channel #*
TIMU	microseconds	
DUT1	3.000	*Delay until trigger*
OPW1	1.000	*Pulse width*

Figure 2 Database Record for Simple Pulse

Timing signals may also be referenced to other user defined timing signals. The absolute delay from the system trigger will be calculated and loaded into the hardware. In this way, related timing signals may be shifted in time with the relative timing intact. Entries in the database are similar to the simple pulse, with the addition of a non-zero trigger origin. In the following example, timer_22_1 is specified to occur 3uS after the start of timer_22:0.

Process Variable: timer_22:1 *Type: timer Node: ioc 22*

Field Name	Entry	Specifies
SCAN	I/O Intr	*Scan on interrupt*
EVNT	0	
TORG	timer_22:0	*Trigger origin*
TYPE	8309	*Card Type*
OUT	#C0 S0	*Card and channel #*
TIMU	microseconds	
DUT1	3.000	*Delay until trigger*
OPW1	1.000	*Pulse width*

Figure 3 Database Record for Referenced Pulse

Timing signals may also be generated at multiples of the system trigger. Software synchronizes the phasing using the time tag counters. Entries in the database are similar to the previous examples, with the addition of an event number.

Process Variable: timer_22:2 *Type: timer Node: ioc 22*

Field Name	Entry	Specifies
SCAN	I/O Intr	*Scan on interrupt*
EVNT	2	*System trigger/2*
TORG	timer_22:0	*Trigger origin*
TYPE	8309	*Card Type*
OUT	#C0 S0	*Card and channel #*
TIMU	microseconds	
DUT1	3.000	*Delay until trigger*
OPW1	1.000	*Pulse width*

Figure 4 Database Record for Different Rep Rate

A planned enhancement to the software is single shot event support. This event type requires that a arming signal be distributed before the system trigger. Entries in the database are similar to the previous examples, with the addition of a new scan type, "one shot".

Limitations

The hardware places several constraints on the timing signals generated in the manner described above. The user timing delay and pulse width is implemented by a 16 bit counter running at 5MHz, so the limit on both is 13.1 mS. The time tag clock and time tag reset should also be chosen so that the time tag counter (also 16 bit) does not overflow during normal operation. The specification of the Am9513A system timing controller chips states a maximum delay of 20nS between terminal count (pulse start or stop) and output change of state, so user timing signals utilizing the local timing channels are subject to this accuracy limitation. However, operational results have shown that the actual value is a constant and does not contribute measurably to signal jitter.

OPERATIONAL RESULTS

The EPICS timing system has been in operation since June '91, and found to be flexible and reliable. Over 95% of the GTA timing requirements have been met by the 200nS resolution user timing channels, the few requiring greater resolution have been implemented with external instruments triggered by the system trigger directly.

Several revisions have been made to the hardware to improve time tag accuracy and functional density. The hardware described above is the latest version, combining improved timing signal distribution and accuracy with user programmable timing outputs. Tests on the installed system have shown a worst case trigger jitter between nodes of +/-1nS with a standard deviation of 700pS. The system has proved to be economical as well, with both master and slave modules costing under $1500 each to construct.

REFERENCES

[1] L.R Dalesio, M. Anderson, R. Cole, C. Eaton, J.Hill, D. Kerstiens, A. Kozubal, F. Lenkszus, R. Zieman "EPICS Architecture", these procedings.

[2] J. O. Hill "Channel Access: A Software Bus for the LAACS", International Conference on Accelerator and Large Experimental Physics Control Systems, Vancouver B.C., Canada, October 30, 1989.