Notes on meeting to discuss the Eurogam Readout Controller

at: Centre d'Études Nucléaires de Bordeaux Gradignan

29-30 October 1990

Present: MM Aleonard
J Alexander
JC Gouillaud
JL Pedroza
C Ring

It was agreed that, conceptually, the Readout Controller (ROCO) consists of 5 main sections:

1) An interface to the Master Trigger module, which stores validated event numbers until they are required by the:

2) VXI Readout section, which reads data from VXI modules and puts it into the:

3) Data FIFO, which acts as a de-randomising data buffer, and passes the data to the:

4) DT32-bus interface section.

5) The final section is a VXI slave, through which the Readout Controller can be set-up and controlled.

It was also agreed that the Readout Controller will be designed to handle parallel and pipe-lined trigger modes, so that a re-design will not be required when Eurogam is upgraded from common dead-time mode.

1. Interface to Master Trigger module

Inputs (to ROCO): "Validation-1" on VXI line ECLTRG0
4-bit "Event Number" on VXI lines LBUS-8 to LBUS-11

Output (from ROCO): "Inhibit_Request" on VXI line ECLTRG2

(These signals from/to the Master Trigger module are communicated via the indicated VXI lines, the Resource Manager, and the cabling between Resource Managers and the Master Trigger module.)

This interface contains an "Event Number FIFO" which can store up to 16 4-bit Event Numbers. An Event Number is clocked into the FIFO by the leading edge of "Validation-1".

When an Event Number reaches the output of the FIFO it is passed to the VXI Readout section, where it triggers a readout cycle for that event.

NOTE: readout of an event will *not* start before the trailing edge of the Validation-1 pulse (for that event), but *may* start then.

If the FIFO becomes full, "Inhibit_Request" is asserted. This prevents the Master Trigger module from generating any more Validation pulses until Inhibit_Request is released (*even if* this means not validating one or more fast triggers which would otherwise be validated).
– This is the mechanism which ensures that there can never be more than 16 events waiting for readout in any VXI crate, which is necessary when using a 4-bit Event Number to label events within data acquisition modules (Ge, BGO, etc.).
NOTE: it is assumed that the Master Trigger module issues validation numbers in sequence, so that there are never two events with the same number waiting to be read out.
NOTE: The time taken for the Readout Controller to assert Inhibit_Request and for this to reach the Master Trigger module (via the Resource Manager) limits the minimum allowable time between Validations, and must therefore be minimised.

NOTE: For correct operation of the Readout Controller, the Master Trigger module must not issue any Validation pulses after the Go*/Stop signal has changed to "stop" (see also section 4).

2. VXI Readout section

It was agreed that this will use the "REN-Readout" cycle defined by Alexander in his document "Eurogam Project: Specification of the VXI Readout Mechanism". The following changes to version 0.9 of this document were agreed:

1) Change signal names: REN-out* to PASS*
REN-in* to REN*
Readout* to BLTACK* (Block Transfer Acknowledge).

2) Increase the time within which a slave must assert BLTACK* (Rule 3) from 100 ns to 200 ns.

3) Increase the time between REN-Readout cycles (i.e. AS* deassertion time) from 40 nS (the VMEbus minimum) to 100 nS; thus increasing the time allowed for a slave to deassert PASS* and BLTACK* at the end of a cycle from 15 nS to 50 nS.

Change (1) is for clarity and familiarity (c.f. REN and PASS on FERA-bus); and it was agreed that Alexander would send e-mail to the Eurogam community seeking agreement† . The other changes are to ease circuit design, particularly of the Ge, BGO and Master Trigger modules; and it was agreed that Alexander would consult the module designers about these timings, and about the suitability of *all* timings in his document.

† Changes agreed, as no objections received by 23 Nov '90. JA

It was agreed that the Readout Controller will use the VMEbus ROR (Release on Request) bus-release algorithm.

3. Data FIFO

This will be 36 bits wide, to store each 32-bit parameter (read from a Eurogam module by the VXI Readout section) together with its associated Event Number.

An "EndSubEvent Token" will be generated and inserted into the FIFO, following the data, for each event. This will also have the associated Event Number appended.
NOTE: when there is no data for an event, the EndSubEvent Token will appear in the FIFO immediately following the previous event's EndSubEvent Token.

It was agreed that it *is* necessary to use EndSubEvent Tokens in the FIFO (even though the Event Number can sometimes serve to delineate events) for the case when an event is being read out of the FIFO at the *same time* as data for that event is still being written into the FIFO. In this case, the DT32 interface *cannot* use the Event Number to recognise the end of an event's data. (Note that, in this case, there may be pauses in the data flow on the DT32-bus: to allow for data to be read from data acquisition modules and then to pass through the FIFO.)

When the Data FIFO is full, it prevents the VXI Readout section from starting to read a new event. However, the Data FIFO will be constructed so that it *always* has space to accept a complete event, once readout has started (like the HSM module in the Event Builder).

The Data-FIFO-full condition will *not* cause the assertion of Inhibit_Request. (Of course, when the Data FIFO is full, it is likely that the Event-Number FIFO will quickly fill – and *this* will assert Inhibit_Request.) However, the Data-FIFO-full condition will be available as a front-panel output, in case it is needed for some use in the future.

It was agreed that there is no reason to limit the Data FIFO to any maximum number of events. Therefore we did not define the number of words which the Data FIFO will hold, but a number of around 4K (36-bit words) is expected.

4. DT32 Interface section

It was agreed that this will use the protocol defined by Alexander in his document "Eurogam Project: DT32-bus Specification", with the modifications agreed at the Eurogam meeting in Strasbourg during October. These modifications are:

1) The addition of a "DTSTOP" line to replace the function of the Stop Token.

2) The addition of a "DTPAUSE" line to allow the data sink (Event Builder) to temporarily halt data transmission.

3) An increase in the size of one of the two DT32-bus cables to accomodate these extra signals.

In addition, it was agreed to change signal names: DTREN-out to DTPASS
DTREN-in to DTREN

Alexander will incorporate these changes into a new version of his document.

During readout of an event (over the DT32-bus) each Readout Controller will compare the Event Number contained in the StartEvent Token with the Event Number of the next event to be read from its Data FIFO.
(NOTE: the StartEvent Token is the first word of an event to be transmitted over the DT32-bus, and must be captured from the bus by all Readout Controllers. Also note that the first Readout Controller *transmits* the StartEvent Token, which it reads from the Master Trigger module, and could therefore capture it internally.)
If a Readout Controller finds that the two Event Numbers differ, it will recognise an "Event Number Fault" condition – and report it to the system in two ways:

1) By setting a bit in its status register, which can cause a VMEbus interrupt (see section 5).
2) By setting a bit in the EndSubEvent Token (see below).

There is currently no facility for error reporting in the EndSubEvent Token, so the meeting recommends the following change to its format: replace the two most significant bits (currently set to 1) with two error bits (value 0 when no errors are detected). This accords with the use of these two bits as qualifier bits in the normal Data Word format. Alexander agreed to circulate this proposal to the Eurogam community for agreement[†]. The Readout Controller will set the most significant error bit as a result of an Event Number Fault.

The operation of the (new) DTSTOP line was discussed. It will be asserted by the first Readout Controller when:

1) The VXI backplane signal Go*/Stop is high (indicating "stop")
2) The first Readout Controller's FIFOs are both empty
3) Its DTREN input is true
4) Its DTPASS output is false

(The last two conditions signify that the Event Builder is trying to read a new event, after the last one has been fully read out.) These conditions are the same as for the transmission of a Stop Token in Alexander's original proposal; the difference is that the data and DTPASS signals are *not* asserted. The DTSTOP signal "handshakes" with the first Readout Controller's DTREN; this handshake will occur repeatedly if the Event Builder continues to try to read events while the first two conditions are true.

NOTE: To ensure that the stop sequence operates correctly, the Master Trigger module *must not* issue any Validation pulses after driving the Go*/Stop signal high.

NOTE: The meeting recommends that the HSM-Interface module should implement a time-out between receipt of a StartEvent Token and receipt of DTPASS, – so as to detect faulty Readout

[†] Change agreed, as no objections received by 23 Nov '90. JA

Controller operation. This should be reported to the Event Builder's CPU via an interrupt. The time-out period should be large: of the order of 1 second.

5. VXI Control of the Readout Controller

The Readout Controller will be a dynamically configured, register-based, A16 device. Four device-dependent registers are required:

5.1 Readout-Controller-Number register

This 16-bit read/write register provides the 16 most significant bits of the EndSubEvent Token, i.e. the 2-bit error field, the 6-bit Readout-Controller-Number field, and the 8-bit field which identifies a token or trigger-data-word (value "FF" for Eurogam). It is expected that system software will set the error field to "00", as the Readout Controller will set the most significant bit to 1 for events in which it detects an Event Number Fault.

5.2 Control/Status register

We identified 11 device-dependent bits required in this register. Their position in the register, and whether they will be incorporated into the VXI Control/Status register (at offset 04), was not decided. All control bits can be read back (except the Reset bit, which will always be read as 0). The "Event Number Fault" and "Timeout during VXI readout" status bits can be written (as well as read): so that software can clear them, and set them (for testing).

Bit		Control	Status
a	R/W	Enable Readout Controller	
b	R/W		Event Number Fault (ENF)
c	R/W	Enable Interrupt on ENF	
d	R/W	Assert Inhibit_Request on ENF	
e	W	Reset	(Always 0 when read)
f	R/W	Test Mode	
g	R		FIFOs not empty
h	R		Trigger FIFO full
i	R		Data FIFO full
j	R/W		Timeout during VXI readout
k	R/W	Enable Interrupt onTimeout	

Enable Readout Controller: When disabled, the Readout Controller will not:

a) store event validations
b) initiate VXI readout cycles
c) participate in DT32-bus activity.

NOTE: During the system initialisation process, the Master Trigger module should be enabled *after* the Readout Controller.

Reset: When the Reset bit is written with a 1, the Event-Number and Data FIFOs will be cleared.
(The "Enable Readout Controller" bit will *not* be changed from the value written during the same write operation.)

Test Mode: When this bit is set to 1, the Readout Controller will not store event validations or initiate VXI readout cycles. It will enter data written to its test register into its data FIFO, and will participate in DT32-bus activity (to read this data out over the DT32-bus).

Timeout during VXI readout: This bit will be set if, during a VXI readout cycle, a data transfer is terminated by the assertion of BERR* (by the bus-timer in the Resource Manager). The readout cycle will also be terminated.

5.3 Data test register

This 32-bit register is write only. When the Readout Controller is in Test Mode, writing to this register will cause the data to be loaded into the Data FIFO, together with the Event Number specified by the Event Number test register.

When the Readout Controller is not in test mode, writing this register will have no effect, and a bus timeout will occur.

5.4 Event Number test register

This 4-bit register is write only. It defines the Event Number to be loaded into the Data FIFO when the Readout Controller is in Test Mode.

5.5 Action of the VMEbus SYSREST signal

The Event-Number and Data FIFOs will be cleared, the "Enable Readout Controller" bit will be set to 0, and interrupts will be disabled.