St. Petersburg State University of Aerospace Instrumentation

Distributed Interrupts in SpaceWire networks.

Addition to the ECSS-E-50-12A. Draft A.

*Yuriy **Sheynin**,*
*Sergey **Gorbachev***

St. Petersburg
2006

7.3 Control characters and control codes[1]

a. A control character shall be formed from a parity bit, a data-control flag and a two-bit control code with the data-control flag set to one to indicate that the current character is a control character.

NOTE The different control characters and control codes are illustrated in Figure 16.

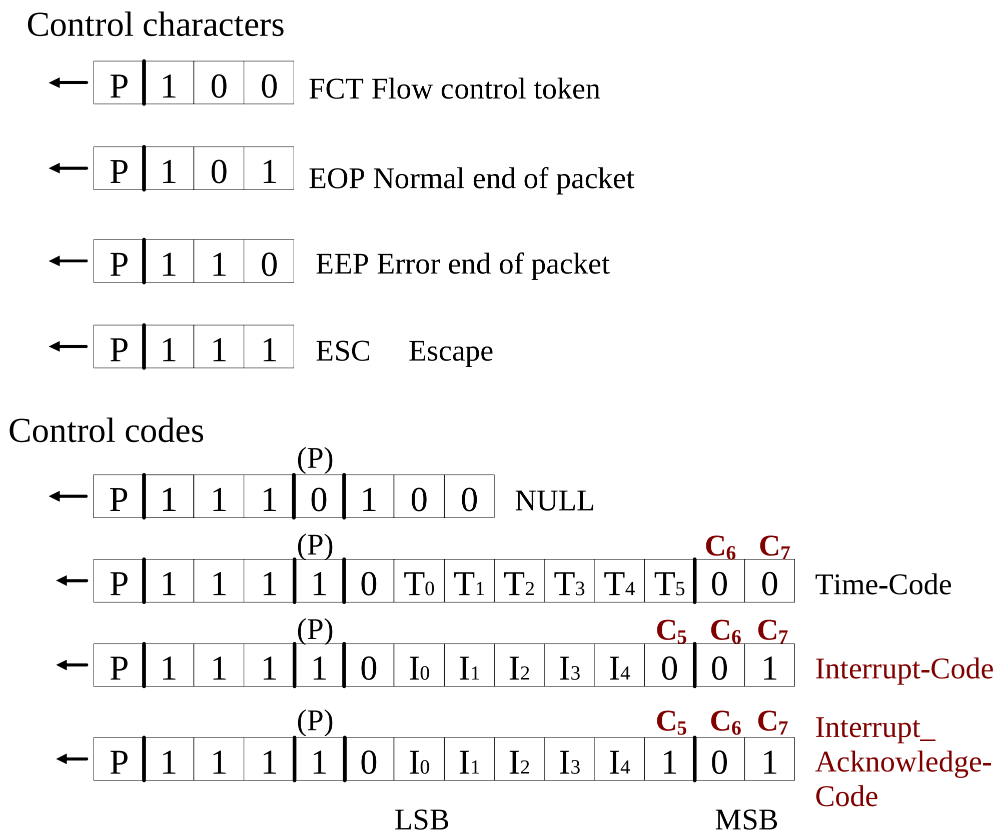

Fig. 16: SpaceWire control characters and control codes

b. The NULL control code shall be formed from ESC followed by the flow control token (FCT).

NOTE 1 The parity bit (P) in the middle of the control code is zero, in accordance with subclause 7.4 b.

NOTE 2 NULL is transmitted whenever a link is not sending data or control tokens, to keep the link active and to support link disconnect detection (see clause 8).

---

[1] Document clause changes in respect to the correspondent ECSS-E-50-12A clauses are highlighted in dark red

c. The other three control codes (Time-Code, Interrupt-Code and Interrupt_Acknowledge-Code) shall be formed from ESC followed by a single data character.

NOTE 1. The parity bit (P) in the middle of these control codes is one (in accordance with subclause 7.4 b.).

NOTE 2 The Time-Code is used to distribute system time information (see subclause 8.12) and control flags isochronous with the time-code distribution.

NOTE 3. The Interrupt-Code and Interrupt_Acknowledge-Code are used to distribute real-time interrupt signals from nodes that are sources of interrupts to nodes that can do interrupt processing procedures (see subclause 8.13). Interrupt/ Interrupt_Acknowledge-codes can eliminate system-wide sideband signals for low latency control signals distribution.

d. Six bits of time information shall be held in the least significant six bits of the Time-Code (T0-T5) and the two most significant bits (C6=0, C7=0) shall contain control flags that are distributed isochronously with the Time-Code.

e. Five bits of interrupt information shall be held in the least significant five bits of the Interrupt-Code (I0-I4) and the three most significant bits (C5=0. C6=0, C7=1) shall contain control flags that are distributed isochronously with the Interrupt-Code.

NOTE The Interrupt-Code is used to distribute interrupt request information and control flags (C5=0, C6=0, C7=1) isochronous with the Interrupt-Code distribution.

f. Five bits of interrupt acknowledge information shall be held in the least significant five bits of the Interrupt_Acknowledge-Code (I0-I4) and the three most significant bits (C5=1, C6=0, C7=1) shall contain control flags that are distributed isochronously with the Interrupt_Acknowledge-Code.

NOTE The Interrupt_Acknowledge-Code is used to distribute interrupt acknowledge information and control flags (C5=1, C6=0, C7=1) isochronous with the Interrupt_Acknowledge-Code distribution.

g. An escape character (ESC) followed by ESC, EOP or EEP is an invalid sequence and shall be noted as an escape error (see subclause 8.9.2.3).

8.13 Interrupts distribution (normative)

8.13.1 General

a. Interrupts distribution provides a low latency signalling among nodes in SpaceWire networks.
It has independent to data packets transfer latency and can run even on blocked by data links.

b. As defined in subclause 7.3, the interrupt control codes (Interrupt-Code and Interrupt_Acknowledge-Code) comprise the ESC character followed by a single 8-bit data character. The data character contains three bit control flags (C5, C6, C7) and a five-bit interrupt source identifier.

c. Interrupt-Code represents an interrupt request. It shall be issued by a node link that will be considered as the source node for this interrupt (Interrupt Source).

d. One of 32 interrupt request signals (interrupt source identifiers) could be identified by the Interrupt-Code.

e. An Interrupt-Code should be accepted for handling in some node of the SpaceWire network. The node will be called the Interrupt Handler. The host of the node is supposed to implement some interrupt processing routine for the accepted interrupt.

f. Interrupt_Acknowledge-Code represents a confirmation that the Interrupt-Code has reached some Interrupt Handler and has been accepted by it for processing.

8.13.2 Handling

a. Each link controller of a node shall manage the Interrupt-Codes and Interrupt_Acknowledge-Codes .

b. Each link controller of a node and each router shall contain one 32-bit Interrupt Source Register (ISR) for Interrupt-Codes and Interrupt_Acknowledge-Codes distribution.

NOTE There shall be one 32-bit ISR per link controller of a node and a single 32-bit ISR in a router.

c. On acceptance of an Interrupt-Code the Interrupt Handler node should send an Interrupt_Acknowledge-Code with the same five-bit interrupt source identifier as in the accepted Interrupt Code.

d. The interrupt-source link controller interface shall have an INTR_IN input, which can be asserted by its host system in case of an interrupt event, accompanied by an interrupt source identifier.

e. When the interrupt-source link interface receives from the host an interrupt request (INTR_IN is asserted) with a five-bit interrupt source identifier it shall set appropriate bit to '1' in the 32-bit ISR. Then it shall send out an Interrupt-Code with the five-bit interrupt source identifier field of the data character and the other three control bits set to the C5=0, C6=0, C7=1 values.

   NOTE If the correspondent bit in the ISR is in ‘1’ state already then the Interrupt-Code shall not be sent out.

f. The Interrupt-Code shall be sent out as soon as the current character or control code would be transmitted. However, the Time-Code shall have priority for transmission over the Interrupt-Code.

g. Interrupt-Code shall not be sent out until a link interface is in the Run state (see subclause 8.5).

h. A subsequent Interrupt-Code with the same interrupt source identifier can be sent by the link either after receipt of an or Interrupt Acknowledge (INTR_ACK_IN asserted) with the correspondent five-bit interrupt source identifier, or after recovery procedure in case of a failure (see subclause 8.13.3).

i. .When a link interface in a node receives an Interrupt-Code it shall check the correspondent bit in the 32-bit ISR to be reset to '0'. If the bit is '0' then the link controller shall set the correspondent bit in the ISR to '1' and assert its INTR_OUT output signal at the interrupt-source interface. It can be accompanied by the five-bit interrupt source identifier of the incoming Interrupt-Code.

   NOTE. In a node, the INTR_OUT signal goes to the host, where it can be used as a hardware interrupt request.

j. If a link interface of a node receives an Interrupt-Code and the correspondent bit in the 32-bit ISR is equal to '1' the Interrupt-Code shall be ignored and the link controller should not issue the output signal INTR_OUT at the host system interface.

NOTE. In this case the Interrupt-Code could have arrived as a result of a malfunction, an error in router or node operation in the network and shall be ignored.

k. When a link interface in a router receives an Interrupt-Code it shall check the correspondent bit in the 32-bit ISR. If the bit is '0' it shall set the ISR bit to '1' and the input port asserts its INTR_OUT output signal at the link controller interface. It shall be accompanied by the five-bit interrupt source identifier of the incoming Interrupt-Code.

l. In a router the INTR_OUT signal propagates to all the output ports of the router (except the port that have issued the INTR_OUT signal) so that they all shall emit the Interrupt-Code with the same five-bit interrupt source identifier field of the data character, which was received by the router.

m. If a link interface in a router receives an Interrupt-Code and the correspondent bit in the 32-bit ISR is equal to '1' the Interrupt-Code shall be ignored and the input port should not issue the INTR_OUT output signal and the router shall not retransmit the Interrupt-Code to the output ports of other link interfaces.

NOTE. In this case the Interrupt-Code shall be ignored to prevent repeated Interrupt-Code propagation in networks with circular connections.

n. .In a node, which host system can be an Interrupt Handler, the link controller shall have a INTR_ACK_IN input at the host system interface that can be asserted by the interrupt handler in response to an Interrupt-Code receipt.

o. When the link controller in a node receives from the host an input signal of Interrupt Acknowledge (INTR_ACK_IN asserted) with a five-bit interrupt source identifier it shall reset appropriate bit in the 32-bit ISR to '0' and then send out a Interrupt_Acknowledge-Code with the five-bit interrupt source identifier field of the data character. The other three control bits shall be set to the value of the control flags C5=1, C6=0, C7=1.

NOTE. The five-bit interrupt source identifier in the Interrupt_Acknowledge-Code should be the same to the five-bit interrupt source identifier in the Interrupt –Code it acknowledges.

p. The Interrupt_Acknowledge-Code shall be sent out as soon as the current character or control code is transmitted. The Time-Code shall have priority for transmission over the Interrupt_Acknowledge-Code. The Interrupt_Acknowledge-Code shall have priority for transmission over an Interrupt-Code.

q. Interrupt_Acknowledge-Codes shall not be sent out until a link interface is in the Run state (see subclause 8.5).

r. When a link controller in a router receives an Interrupt_Acknowledge-Code it shall check the correspondent bit in the 32-bit ISR. If the bit is '1' then it shall be reset to '0' and input port assert its INTR_ACK_OUT output signal at the link controller interface. It shall be accompanied by the five-bit interrupt source identifier of the incoming Interrupt_Acknowledge -Code.

s. In a router the INTR_ACK_OUT signal propagates to all the output ports of the router (except the port that have issued the INTR_ACK_OUT signal) so that they all emit the Interrupt_Acknowledge-Code with the same control flags (C5=1, C6=0, C7=1) and the same five-bit interrupt source identifier field of the Interrupt_Acknowledge-Code, which was received by the router.

t. If a router receives an Interrupt_Acknowledge-Code and the correspondent bit in the 32-bit ISR is equal to '0' the Interrupt_Acknowledge-Code shall be ignored and the link interface should not issue the INTR_ACK_OUT output signal; router output ports will not emit this Interrupt_Acknowledge-Code further.

NOTE. It prevents repeated Interrupt_Acknowledge-Code propagation in a network with circular connections.

u. When a link controller in a node receives an Interrupt_Acknowledge-Code it shall check the correspondent bit in the 32-bit ISR. If the bit is '1' then the link controller shall reset it to '0'. The link controller shall assert its INTR_ACK_OUT output signal at the host system interface. It shall be accompanied by the five-bit interrupt source identifier from the incoming Interrupt_Acknowledge-Code.

NOTE. A link interface in any node, not only in the Interrupt Source node, can notify its host about an Interrupt_Acknowledge-Code arrival.

v. If a node receives an Interrupt_Acknowledge-Code and the correspondent bit in the 32-bit ISR is equal to ‘0’ the Interrupt_Acknowledge-Code shall be ignored and the link interface should not issue the INTR_ACK_OUT output signal at the host interface.

w. After link in a node reset or disconnect-reconnect (state machine in *ErrorReset* state) the ISR bits shall be set to zero. After reset, the ISR bits in a router shall be set to zero.

8.13.3 Interrupt Codes distribution recovery in case of errors

a. Causes that can lead to an Interrupt -Code/Interrupt_Acknowledge-Code distribution malfunction are:

- link disconnect error that can cause an Interrupt -Code/Interrupt_Acknowledge-Code loss;
- spontaneous change of an ISR bit state as a result of intermittent faults in a node or in a router;
- parity error in Interrupt -Code/Interrupt_Acknowledge-Code transmission by a link can cause an Interrupt -Code/Interrupt_Acknowledge-Code loss.

NOTE. In SpaceWire networks with redundant links and circular connections (e.g. mesh, torus, fat tree) an error that cause an Interrupt -Code/Interrupt_Acknowledge-Code loss will not stop the Interrupt control codes distribution to network nodes.

b. Each ISR in a node or in a router should have a timer per ISR bit. A timer shall start at the receipt of an Interrupt-Code with correspondent five-bit interrupt source identifier and reset at receipt of an Interrupt_Acknowledge-Code with the same five-bit interrupt source identifier. In case of timeout Ti before the timer is reset, the ISR timeout event arises.

NOTE 1. ISR reset timeouts recover Interrupt –Codes distribution for following interrupt requests.

NOTE 2. ISR reset timeouts recover Interrupt –Codes distribution both after Interrupt –Code and Interrupt_Acknowledge-Code losses.

c. For an Interrupt Source link the reset timeout Ti=T1 shall be not less than double Interrupt-Code/Interrupt_Acknowledge-Code worst propagation time in the network with diameter D (dependent upon the SpaceWire network interconnection topology) plus delay in an Interrupt Handler node that should send an Interrupt_Acknowledge-Code (dependent upon implementation).

d. At an ISR timeout event in an Interrupt Source link, the link should reset the correspondent ISR bit to ‘0’ and send an Interrupt_Acknowledge-Code with the five-bit interrupt source identifier that corresponds to the ISR bit, which has aroused the ISR timeout event.

e. In each node and in each router at an ISR timeout event the correspondent ISR bit should be reset to ‘0’.

NOTE. Reset timeout Ti for routers and non-Interrupt Source nodes can be set Ti=T2, where T2 ≥ T1.

f. Some nodes in SpaceWire networks (e.g. network management nodes) may have rights to send Interrupt_Acknowledge-Codes while not being the sources for correspondent Interrupt–Codes.

g. After link in a node reset or disconnect-reconnect (state machine in *ErrorReset* state) the ISR timers shall be set to zero.

h. After reset, the ISR timers in a router shall be set to zero.