Escape Sequence Detection with 29C93A

Escape Sequence Detection

The main goal of the "Escape Sequence Detection" feature is to help the designer of a terminal adapter to implement protocol like the Hayes Standard AT Command Set Enhanced for ISDN ((c) Hayes Microcomputer Products, Inc., widely used in PC communications). "Detect escape sequence" means that we are able to recognize one character sent by the local DTE in data transfer phase (i.e. when no control normally occurs on the data that go through the Terminal Adapter (TA)).

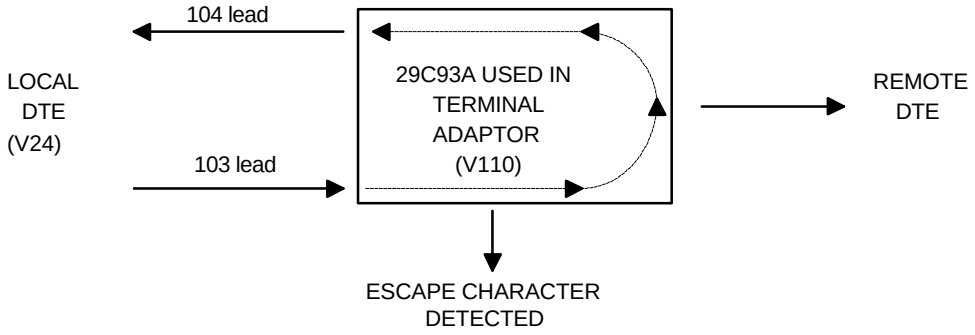

Figure 1. Test Configuration

Structure of an Escape Sequence for the 29C93A

An Escape Sequence begins with a special character (Escape character). This sequence may be 1, 2 or 3 character long but only the first character can be detected by the 29C93A. The length of the sequence is used to define when the chip has to send one (or many) interrupt signal(s).

Actions After an Escape Sequence Detection

Interrupts

Depending of the length selected, we can have interrupts (and status information) generated after the first character (Escape character) is detected or (and) when the last character is received.

Data path

When the Escape character is detected, the followings bytes of the sequence are stored in a 3 (max.) character length FIFO. The sequence may be removed (filtered) from the output data stream or not. It is also possible to automatically switch from data transfer mode ("line" mode) to command mode ("buffer" mode).

Local DTE Interface Programming

In the following, we will consider that the TRAC is connected with a Local asynchronous DTE (**CONF**) that send data on lead 103 at 9600 b/s (**ASCLK**). The data format is 8 bit of data without parity plus 2 stop bits (**FASYNC**). The programming of the other V24 command lines is not important regarding the Escape detection process. All the lines will be active and given by **CMOD** register content.

CMOD @ 00H

1	TMT1	HF	ECTS	CTS	DSR	DCD	RING
1	0	1	0	1	1	1	0

CONF @ 01H

AT	AS	EPA	SPA	LOCAL	APRIM	TNIC	RNIC
1	1	0	0	0	0	0	0

ASCLK @ 04H

-	TRSEL	RRSEL	AR4	AR3	AR2	AR1	AR0
0	1	1	0	1	1	0	0

FASYNC @ 05H

H0	NS1	NS0	ND1	ND0	P2	P1	P0
1	0	1	1	1	0	0	0

The Escape detection features are selected in register **ESCMOD**. Only one bit is mandatory to enable the detection, the other bits (x) depend of the sequence expected and of the actions to take.

ESCMOD @ 0DH

0	FSEQ	AUTO	EIEND	EIBEG	ENESC	LONG1	LONG0
0	X	X	X	X	1	X	X

Remote DTE Interface

Because the Escape detection mainly concerns the Local side of the DTE, the programming of the remote end interface has been simplified as much as possible. Thus, we are using the Local loopback mode (register **CONF**, DOUT internally connected to DIN), it is then not necessary to provide or emulate a V110 line. It also will allow to show us when the filtering is active (when the sequence is removed or not from the data stream). Concerning the timings, an IOM1 (TE) type of interface is selected (**CLKSEL**). Thus we have : NREF=1536 Khz, BCLK=512 Khz. The synchronous terminal speed (clock for RA1 step adaptation process) will be 9600 Hz.

CFRM @ 00H

0	TMT0	-	SD	ED	EX	SA	SB
0	0	0	1	0	1	0	0

TFRM @ 02H

-	RFILT	SPRIM	TBL	BL	TBD	BD	B2
0	0	0	1	1	0	0	0

CLKSEL @ 03H

BTYP1	BTYP0	REF1	REF0	V3	V2	V1	V0
1	0	0	1	0	1	1	1

Microcontroller Interface

The only register directly concerned is **MASK** that is used to enable or disable the interrupts. Note that bit *EIEND* and *EIBEG* (register **ESCMOD**) can also disable the interrupts involved in the detection (X shows bits not directly concerned).

MASK @ 06H

AIB	AIESC	AISX	AIPAR	AITD	AIRL	AITL	AIRD
X	1	X	X	X	X	X	X

Results

Three different tests have been made :
sequence length = 3, interrupt on both first and last character, sequence filtering disabled
sequence length = 3, interrupt on both first and last character, sequence filtering enabled
sequence length = 2, interrupt on the last character, sequence filtering enabled

On all the tests, the data sent on lead 103 by the data pattern generator are :

....[20H][2DH][20H][4DH][48H][53H]....
in ASCI I : ... - M H S - ....

The Escape Character is fixed to [M] (this value has to be written in register **ESCVAL**).

ESCVAL @ 0FH

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
0	1	0	0	1	1	0	1

First Test

The sequence length that we expect to detect is 3 (i.e. the whole sequence is MHS. Bits *LONG0* and *LONG1*). The interrupts will be generated after the reception of the first and the last character of the sequence (i.e. after the reception of [M] then [S], bits *EIBEG* and *EIEND*) and the sequence will not be filtered (*FSEQ*). The 29C93A will not switch to flow control mode after the detection (*AUTO*)

ESCMOD @ 0DH

0	FSEQ	AUTO	EIEND	EIBEG	ENESC	LONG1	LONG0
0	0	0	1	1	1	1	1

We can see on the picture below that, as mentioned earlier, the whole Escape sequence is transmitted towards the remote end (visible on RxD thanks to the loopback). The first interrupt is caused by the reception of the escape character (bit *BEG* in **ESCSTA**), the second is caused by the reception of the last character of the sequence (bit *END* in **ESCSTA**).

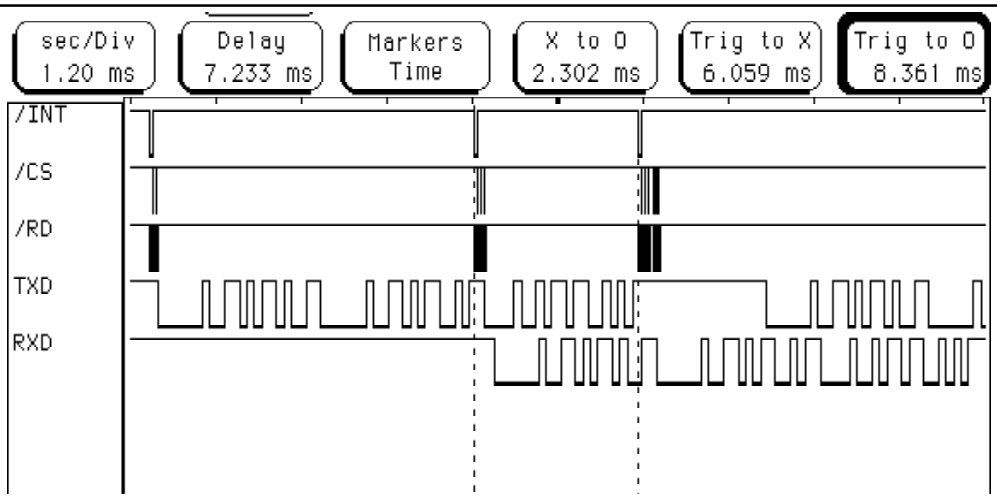

Figure 2.

On the trace below, we can see in detail 3 read operations (**INT0**, **INT1** then **ESCSTA**) to acknowledge the first interrupt (*BEG*). For the second interrupt (*END*), these 3 read operations are followed by 3 more read operations to read the received sequence in FIFO **ESCR**.

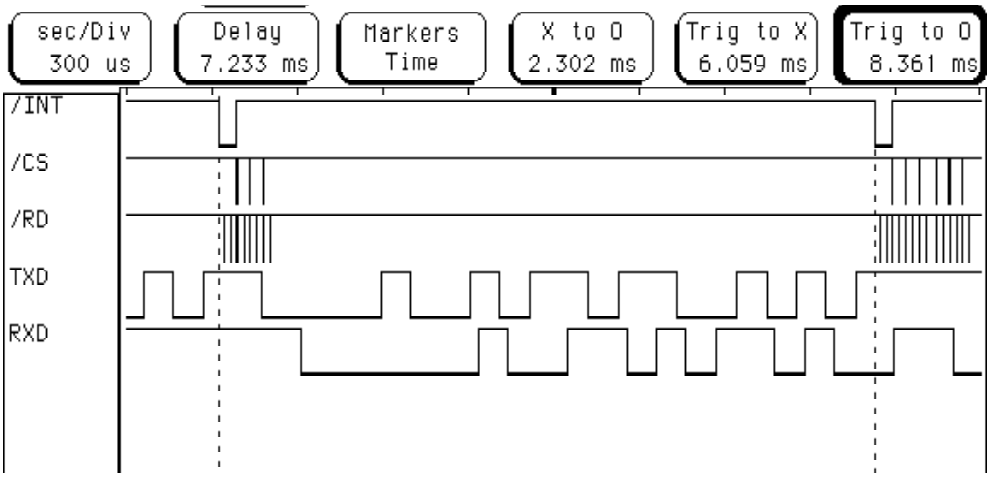

Figure 3.

Second test

The only difference with the previous test is the filtering of the escape sequence (not transmitted to the remote DTE. bit *FSEQ*).

ESCMOD @ 0DH

0	FSEQ	AUTO	EIEND	EIBEG	ENESC	LONG1	LONG0
0	1	0	1	1	1	1	1

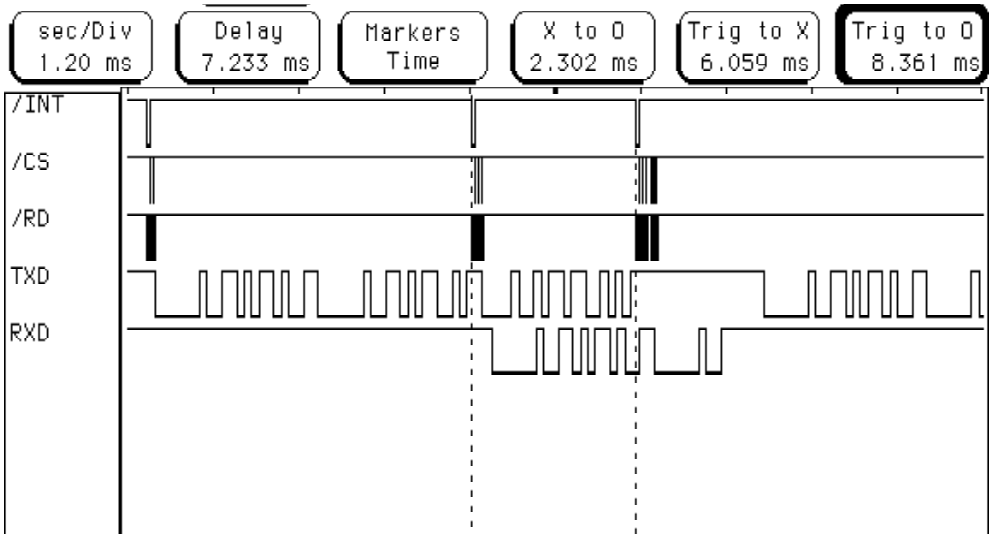

Figure 4.

Third test

The length of the sequence is then 2 (bits LONG0 and LONG1), with interrupt enabled for the reception of the last character only (bit *EIBEG* and *EIEND*).

ESCMOD @ 0Dh

0	FSEQ	AUTO	EIEND	EIBEG	ENESC	LONG1	LONG0
0	1	0	1	0	1	1	0

On RXD we can see that the sequence has been filtered ([M] and [H] removed from the output data stream) while the character [S] has been transmitted.

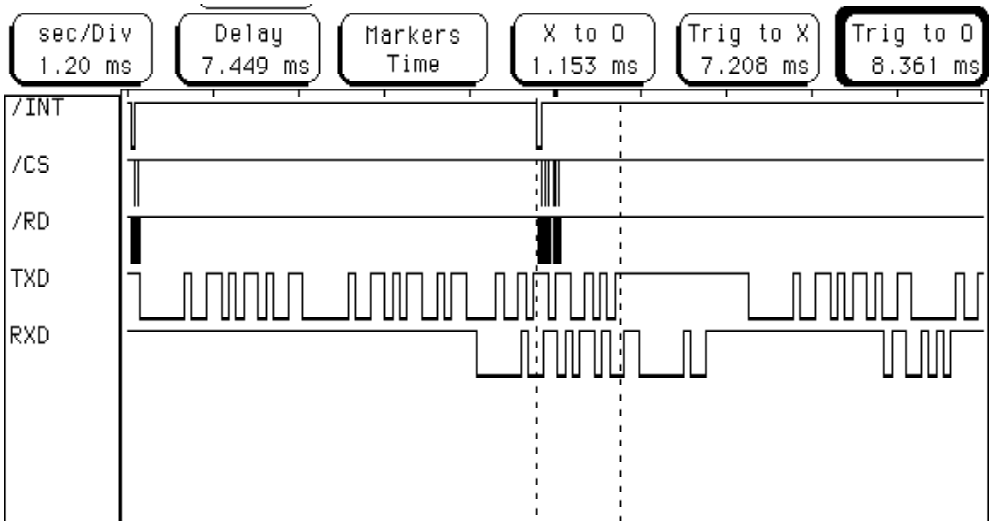

Figure 5.