QuP Multiplexer command list

Date: May 2021

Firmware version 2.0

Authors: INTI

Command lists

COMMAND	ARGUMENT #1	ARGUMENT #2	DESCRIPTION
ENA	SLx CHx	ON | OFF	Enables channel x
STAT	SLx CHx		Returns channel status
*IDN?			Returns program ID
*RST			Resets the multiplexer. All the slaves are opened (signal relays turned OFF and GND relays turned ON). Guard relays turned OFF. Local mode. Internal trigger. Status byte to default value.
*CLS			Clears the multiplexer. All the slaves are opened (signal relays turned OFF and GND relays turned ON). Errors bits on status byte are cleared.
GTL			Go To Local operation
REM			Remote operation. Disables the front panel push buttons
TRG	INT | EXT		Trigger internal or external
TRGPOL	POS | NEG		Sets TRIGGER polarity
TIMER	< ms >		Sets the internal TIMER
TIMER?			Returns the internal TIMER in ms
START			ARM the multiplexer. Thus, it starts running the programmed sequence for each TRIGGER event.
STOP			Stops running the MUX
PAUSE			Pauses running the MUX
RESUME			Resumes the MUX after pausing it from the last sequence

COMMAND	ARGUMENT #1	ARGUMENT #2	DESCRIPTION
DELAY	<delay in ms>		An enable delay time is required to prevent short circuit when the signal relays are opened and the ground relays are closed or viceversa. This time depends on the relays and load.
DELAY?			Returns the current enable delay.
*STB?			Reports for the status byte
GRD	SLx CHx		ENABLE/DISABLE Guard at SLx CHx
ADDSEQ	SLx CHx W <no. trigger>		Adds a switching sequence
EDTSEQ	<seq> SLx CHx W <no. trigger>		Edits sequence number <seq>
LDSEQ	<no. of sequences to load>		Loads a sequence matrix into memory from PC (binary format)
STSEQ(¥)			Saves the current sequence into the EEPROM
RLSEQ(¥)			Recalls the last stored sequence from EEPROM to internal memory
GTSEQ(¥)			Reports the current sequence from Memory to PC (binary format)
DELSEQ			Deletes last entered sequence
SEQ?	< no. >		Returns the sequence no. configuration
NSEQ?			Returns the total number of sequences in memory
NSLAVES?			Returns the number of slaves connected
WSLAVES?			Returns the slaves position in a byte as The Bx is set if the slave is connected at the corresponding position.

		SL6	SL5	SL4	SL3	SL2	SL1
B7	B6	B5	B4	B3	B2	B1	B0
X	X	0/1	0/1	0/1	0/1	0/1	0/1

(¥) NOT YET IMPLEMENTED!

Important: To start the multiplexer a valid sequence must be in memory. Then the trigger must be set to EXT.

Example 1: Status byte

To read the status byte use *STB?

Status byte structure:

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LASTERR			RDY	MX_ENA	TRG POL	TRG INT /EXT	REM /LOC
			/RDY=0 RDY=1	ENA=0 DISA=1	POS=0 NEG=1	INT=0 EXT=1	REM=0 LOC=1

Bit description:

BIT 0	Is set to 0 when the MUX in on REMOTE operation. Is set to 1 when the MUX is on LOCAL operation.
BIT 1	Is set to 0 when the trigger event is internal. Is set to 1 when the trigger event is external.
BIT 2	Is set to 0 when the trigger polarity is positive. Is set to 1 when the trigger polarity is negative.
BIT 3	Not yet implemented MX_ENA is always equal to 0, the multiplexer is on Enable state.
BIT 4	RDY, is set to 1 (RDY) when the MUX is on IDLE. Otherwise is set to 0 (/RDY), indicatining that a sequence is running.
BIT 5 **BIT 6** **BIT 7**	See error codes table below

LASTERR: error codes table

Code number	Description	Condition
0 (000)	No error	There is no error
1 (001)	Command error	Indicates a wrong command
2 (010)	Sequence error	Indicates 0 sequence in memory
3 (011)	Sequence error	Indicates that the maximum number of sequences was reached
4 (110)	ENA command error	Indicates that an error in ENA command
5 (101)	SLV error	Indicates an error in ENA command when setting a SLAVE

6 (110)	GRD error	Indicates an error in GRD command
7 (111)	CH error	Indicates a channel error in the commands.

Example 2: to load a sequence matrix

Use the command LDSEQ <number of states to be load>

The sequence will be loaded using three bytes for each state:

BYTE 1:

BIT 7	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**
SL4-CH2	SL4-CH1	SL3-CH2	SL3-CH1	SL2-CH2	SL2-CH1	SL1-CH2	SL1-CH1

BYTE 2:

BIT 7	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**
X	X	X	X	SL6-CH2	SL6-CH1	SL5-CH2	SL5-CH1

BYTE 3: a number between 1 to 255 indicating the number of trigger pulses to remain on the current sequence state.

Following you can see an example text file which can be uploaded using the VI or python script. The file can be generated by any text editor or by the VI module. The fields should be separated by TABs. Note that the byte 1 and 2 represents the states of the channels in binary format.

```
CH Byte 1   CH Byte 2   Delay Byte 3
      1           4           10
      10          0           10
      4           0           1
      8           0           1
      16          0           1
      32          0           1
      64          0           1
      128         0           1
      0           1           1
      0           2           2
      0           4           2
      0           8           2
      0           4           2
      0           2           4
      0           1           5
      1           0           4
      64          0           1
      10          0           1
```

Examples 3: configuring the scanner without sequence matrix

ADDSEQ SL1 CH2 SL2 CH1 SL3 CH1 SL4 CH2 SL5 CH1 SL6 CH2 W 3→ Add a sequence to the multiplexer memory. The sequence is: close channel 2 (slave 1), channel 2 (slave 2), channel 1 (slave 3), channel 2 (slave 4), channel 1 (slave 5) and channel 2 (slave 6) and wait 3 trigger pulses to change to the next sequence.

ADDSEQ SL1 CH1 SL2 CH2 SL4 CH1 W 1 → Sequence: close channel 1 (slave 1), channel 2 (slave 2) and channel 1 (slave 4), then wait 1 pulse to the next sequence.

Examples 4: useful commands

To enable a channel:

ENA SL1 CH1 ON → close channel #1 of Slave #1

ENA SL2 CH2 OFF → open channel #2 of Slave #1

To enable the guard relay:

GRD SL1 CH2 ON → close the GRD of Slave 1 channel 2

GRD SL3 CH1 OFF → open GRD of slave 3 channel 1

To set the internal timer:

TIMER 160 → set the internal TIMER to 160 ms (the internal timer is for testing proposes)

To set internal or external trigger

TRG INT → Set the trigger mode to internal triggered

TRG EXT → Set the trigger to external

To start the multiplexer:

START → start running the MUX according to the sequence in memory.

To ask for the number of slaves:

NSLAVES? → Return msg: “TOTAL SLAVES: 2”

WSLAVES? → Return msg: XX000101

Power-On State

When the multiplexer is powered-on it starts on the following state:

All relays open

Trigger INT

LOCAL mode

No valid sequence in memory

Internal TIMER 2 s

Push Buttons Description

The multiplexer has 4 push buttons:

1) RESET: Reset the multiplexer as the *RST command does.

2) uC RESET: This button reset the microcontroller, restarting it from power-on.

3) ENABLE: Runs the sequence in memory step-by-step.

4) REM/LOC and SEQUENCE Display. This has two functions. Pressing it for more than 5 seconds change the state from REM to LOCAL operation or viceversa. Pressing it for less than 5 seconds shows on the last line in the display the sequence in memory in the form:

SEQ no.: byte 1 in hex | byte two in hex | byte three in hex.

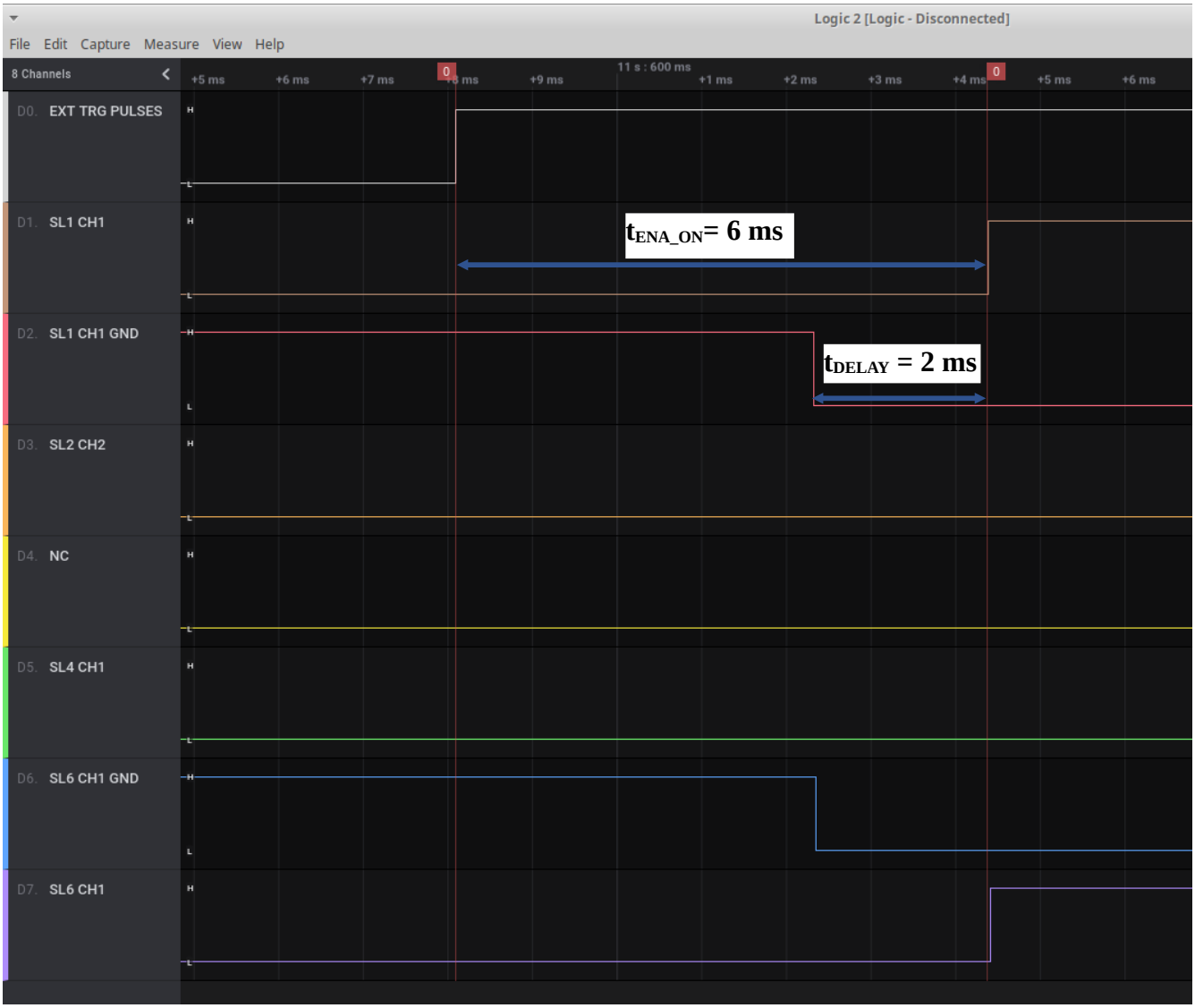
For example:

1: 1 4 A => which corresponds to a sequence number 1 as: byte 1 = 1, byte 2 = 4, byte 3 = 10 in decimal

Timming diagrams

t_{ENA_ON}: time to close a signal channel after an external trigger pulse

t_{DELAY} = delay time

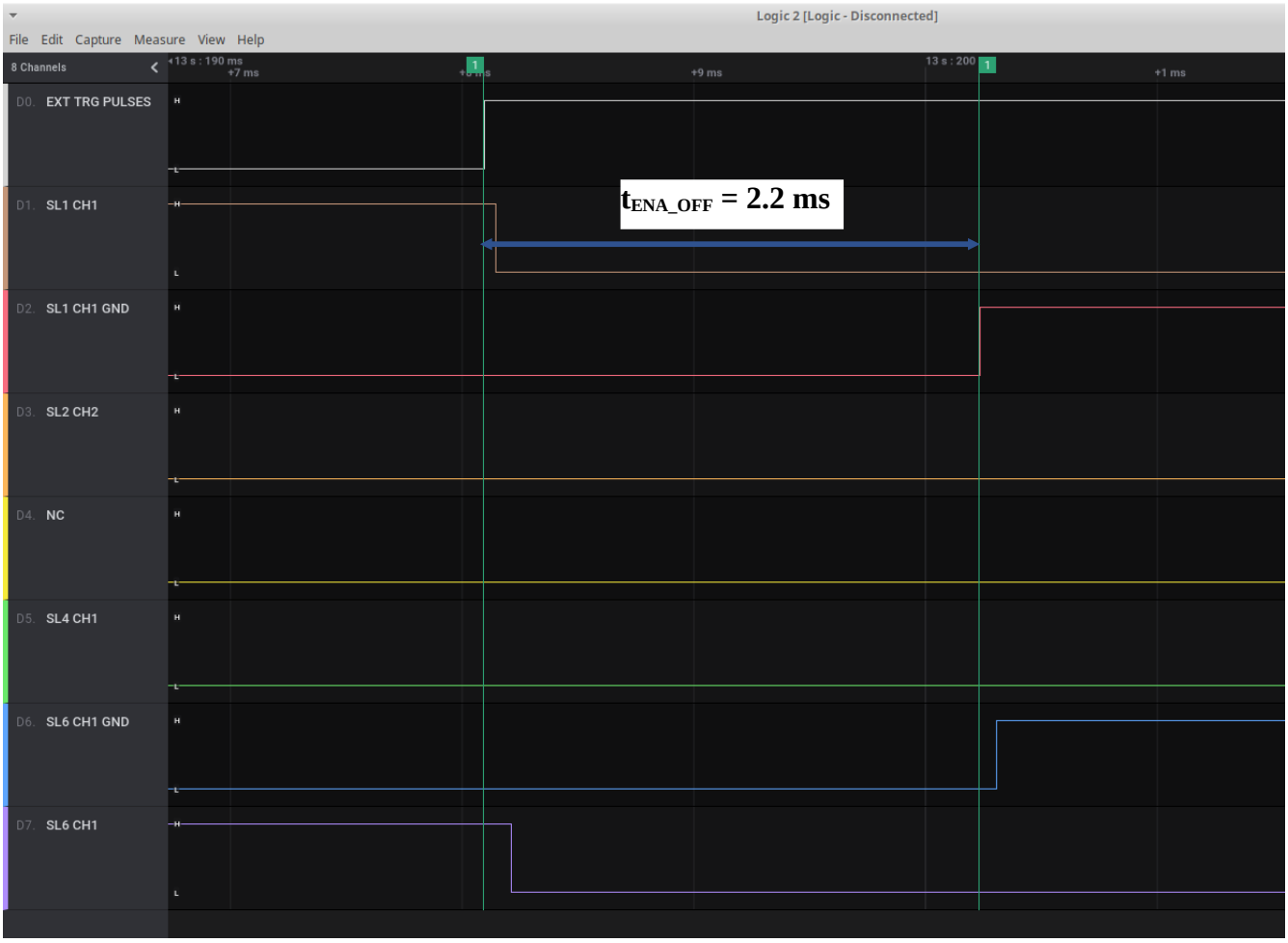


To estimate t_{ENA_ON} time use the following formula:

$$t_{ENA-ON} = (3 * DELAY + 0.225) ms$$

DELAY is the current DELAY time setting

t_{ENA_OFF}: Time to wait until the channel is open



To estimate t_{ENA_OFF} time use the following formula:

$$t_{ENA-OFF} = (DELAY + 0.125)ms$$

Important: ENABLE ON/OFF times also depends on the source and load impedance