AWR Technology Microstep Protocol v1.17

1. Contents

2. Revision History

Date	Version	Firmware	Details
30 Jan 1998	v1.0	<= v00.03	• Conversion from extant documentation.
12 Feb 1998	v1.1	v00.04	• Some mods to improve consistency.
14 Aug 1998	v1.7	none yet	• Index Pulse (TBD) • Changed nonvol locations to be encapsulated for each possible movement.
9 Sept 1998	v1.8	none yet	• More consistent responses. • Re-optimised nonvol locations to reduce wasted space, while reflecting real microstep requirements.
24 Sept 1998	v1.9	none yet	• Reduced nonvol locations to save memory in ustep.
19 April 1999	v1.10	v00.05	• New command to send INDEX pulses to IH.
20 April 1999	v1.10a	v00.05	• Added to Serial Format section. • Added field automated using AUTHOR and TITLE.
21 Sept 1999	v1.11	v0.53	• Added X EVENT from MICROSTEP
4 Oct 1999	v1.12	v0.54	• Postscaler flags function change
15 Oct 1999	v1.13		• All messages are now packeted to improve link robustness and allow inter-packet debug information. • Removed adjust commands
6 Feb 2002	v1.14	v0.59	• Improvements in slow speed performance • Bigger command buffer to reduce error message flow. • Added BACKLASH corrections for both axes. • Added more EVENTS from Microstep.
22 Nov 2005	V1.15	V0.59	• Tidied up wording to match the actual transactions.
28 Apr 2007	V1.16	V0.72 on	• Added Command ‘R’ to retrieve status of relays and override setting. • All replies now postfixed with ‘#’
07 Jun 2013	V1.17	V0.77 on	• Added ‘L’ event for uncommitted lines in FRGEN version.

3. Serial Format

3.1 Hardware

Serial setup for the firmware is 9600,N,8,1 with no handshaking. RS232 is operated at 5V CMOS levels, with a transmit impedance of 220R. When category 5 UTP cable is used, the limit is likely to be potential drop in the ground return, giving a maximum length of about 10m.

3.2 Protocol

The protocol is essentially multi-master since both parties can originate a transaction. This can give rise to synchronism problems, so all messages coming in are packeted within START (**:**) and STOP (**#**) bracketing characters. All strings have a <CR> <LF> <#> appended.

Command-Response transactions are atomic, that is they will not be interrupted. However, due to buffering and soft timing considerations, it is expedient to expect the response packet to be received out of order, with perhaps a message in-between. The timeout for responses is 100ms. Packets whose contents are bad or meaningless will be discarded.

HEX digits must be UPPER CASE.

4. Commands to MICROSTEP

Unless otherwise stated, the response text is:

Y for success

N for failure

4.1 Button Presses

Command Text	Event	Details
1	**UP_PRESS**	Presses and holds the UP key.
2	**DN_PRESS**	Presses and holds the DOWN key.
3	**LT_PRESS**	Presses and holds the LEFT key.
4	**RT_PRESS**	Presses and holds the RIGHT key.
5	**RA_IDLE**	Releases the RIGHT and LEFT keys.
6	**DEC_IDLE**	Releases the UP and DOWN keys.

4.2 Speed Control

Command Text	Event	Details
`7`	`SPEED_30`	Sets movement rate to GUIDE (typically 30%).
`8`	`SPEED_2x`	Sets movement rate to CENTRE (typically 2x).
`9`	`SPEED_SLEW`	Sets movement rate to SLEW (as fast as possible).
`A`	`SPEED_32x`	Sets movement rate to MOVE (typically 32x).

4.3 User Outputs

These are THREE relays for custom switching tasks.

Command Text	Details
`Fn1`	Activates user relay n.
`Fn0`	De-activates user relay n. (Relay coil is OFF)

4.4 Soft Writes

These commands exist for committing and discarding soft-writes (see later section).

Command Text	Event	Details
`D`	`DISCARD`	Discards all soft writes, ie refreshes RAM from nonvol.
`E`	`COMMIT`	Commits all soft writes to nonvol.

5. Access to MICROSTEP Registers

Registers in the MICROSTEP contain all speed settings, and are non-volatile.

5.1 Addresses

Flags structure :

Flags Field	Name	Note
BIT0	USM_ENABLE	axis enable
BIT1	USM_POSDIR	positive direction
BIT15..BIT8		counts postscaler

Address (hex)	Name	Notes
00.0	dir_sw_override	Override hardware direction senses
00.1	sw_ra_sense	RA software override detection
00.2	sw_dec_sense	DEC software override detection
01-02	drive_ra	Counts, Flags
03-04	drive_dec	Counts, Flags
05-06	guide_up	Counts, Flags
07-08	guide_right	Counts, Flags
09-0A	guide_left	Counts, Flags
0B-0C	centre_up	Counts, Flags
0D-0E	centre_right	Counts, Flags
0F-10	centre_left	Counts, Flags
11	move_up	Frequency
12	move_right	Frequency
13	move_left	Frequency
14	slew_up	Frequency
15	slew_right	Frequency
16	max_accel_ra	Acceleration
17	max_accel_dec	Acceleration
18		Reserved
19	ra_backlash	RA backlash steps
1A	dec_backlash	DEC backlash steps
3F	CRC	Use with care !
FF	VERSION (DD.DD)	Version (read only)

5.2 Read One Word

The command text for reading is

AA?

The response text for success is

AA?DDDD

The response text for failure is

N

Where

AA = Address in hex

DDDD = Data in hex

5.3 Read All Words

The command for reading all words is

?? **(**for example **:??#)**

The response is the same as if commands for all the addresses had been issued individually and sequentially.

5.4 Write One Word

The command text for writing is

AADDDD

The response text for success is

AAY

The response for failure is

AAN

Where

AA = Address in hex

DDDD = Data in hex

5.5 Soft Writes

A soft-write is a write to RAM only. The protocol is the same as nonvol write except the address has BIT7 set.

6. Messages from MICROSTEP (EVENTS)

None of these have responses.

These can however happen at any time but are not mixed up with other message/response transactions.

6.1 Errors

Error reporting occurs whenever an error is found. After the error is reported the system carries on - no errors are considered fatal.

Error Text	Cause
e1	watchdog timeout reset
e2	EEPROM CRC error
e3	event buffer overflow
e4	EEPROM verify error
e5	EEPROM other error
e6	MCLR reset
e7	brownout reset
e8	divide by zero
e9	divide overflow
eA	receive buffer overflow
eB	serial receive error
eC	protocol syntax error

6.2 Index Pulse

With INDEX PULSE hardware around the slow motion axis (RA) there is a command to send the index pulse data to the IH. This is necessary to synchronise periodic error correction.

Event Text	Event	Details
P	INDEX_PULSE	RA index pulse received.

6.3 Override Input

There is also an OVERRIDE input to stop the telescope from moving. This would be operated by an external limit switch (or several wired in series) so if one breaks the circuit the signal will be recognised. An override push button external to the drive box must be pressed to allow further movement of the telescope.

Event Text	Event	Details
`S1`	`FULL_STOP`	Override stop received
`S0`	`GO_FROM_ES`	Override stop removed

6.4 Axis Movement Status

This command gets sent whenever an axis starts moving from idle rate or returns to idle rate. Its use is to tell when a slew has finished.

Event Text	Event	Details
`Xab`	`MOVE_STATUS`	a = ra axis, b = dec axis 0 = idle, 1 = moving

6.5 Backlash Corrections

These commands get sent when a backlash correction is initiated and when it finishes. In use the RA axis will send a backlash command when the motor moves more than 2x sidereal in the Easterly direction and again when it reverses direction to normal tracking rate. The DEC axis backlash issues a command at the start of a movement if the movement is in a different direction to previous. The backlash correction is performed as a high speed blip and so takes very little time.

Event Text	Event	Details
`V1`	`RA_BACKLASH`	RA motor reversed
`V0`	`RA_IDLE`	RA normal direction
`W1`	`DEC_BACKLASH`	DEC motor direction towards Pole
`W0`	`DEC_BACKLASH`	DEC motor direction away from Pole

If the value in stores 19 and 1A is set to zero (0000) then there is no backlash performed. The value is the number of steps calculated to give the arc second backlash requirement.

6.6 Poll Status

Send the full command **`<:R?#>`** and it will reply with the status that has to be decoded bit wise to give the status of the three relays.

Cmd Text	Status reply	Details
R?	:Rab#	A is relay status B is Override status

6.7 FRGEN Input lines

The Frequency Generator version product (FRGEN) has 6 uncommitted lines which can be configured in the factory to operate other equipment. As INPUTS the 6 lines will give an event when the logic level on one of the lines changes. The values x and y correspond to individual lines and are logic OR together to make the word.

Event Text	Event	Details
L00xy	On port change (x) 0010 - D 0020 - E 0040 - F on port change (y) 0001 - A 0002 - B 0004 - C	Input lines A,B,C Input lines D,E,F

EXAMPLE - ‘L0035’ has lines A,C,D,E at logic HIGH and lines B,F at logic LOW.

NOTE: In AWR COMMS mode on the Intelligent Handset the whole word is echoed out to the host computer eg : <:L0035#>