Telescope Control Protocol (WIP)

This defines the trame protocol used for my final project of the FabAcademy 2016, witch consists of the motorization of a telescope mount.

I2C will be used to communicate between the main board and the motor board.

The trame will be sent complete with all the fields, even the unused ones (they must be ignored by the slave)

Available commands

Code	Byte	Description
STOP	0x00	Stop (with decceleration) the actual moves
SPEED	0x05	Sets the motors speed
ACCEL	0x08	Sets acceleration
GOFAST	0x10	Move motor(s) with desired number of steps in the desired direction with the maximum speed. Reset microsteps value and set them back after the move is done
GOSLOW	0x15	Moves motor(s) with desired number of steps in the desired direction using the actual speed.
FOLLOW	0x20	Move a motor indefinatly with a given speed
MS	0x30	Set microstepping mode

Available options (values) (WIP)

Value	Type	Description
Motor num	Char	Define witch motor the command is addressed to 0x01 = motor 1 0x02 = motor 2 0xFF = all motors
Speed	u_int16	Define the speed in steps/s Must be ignored by the CMDs : ACCEL, SETMSTEP if zero, default speed is used, else sets the new default speed
Acceleration	u_int16	Defines the acceleration in steps/s^2 Must be ignored by the CMDs : SPEED, SETMSTEP, FOLLOW if zero, default acceleration is used, else sets the new default acceleration
Distance	Int32 (long)	Number of steps to turn the motor by. Must be ignored by CMDs :

		SPEED, ACCEL, SETMSTEP
Microsteps	Char	From 1 to 255 theorical. Each value will point to an array in witch driver's microstepping configuration will be stored. Need to be configured at compile time for each driver model
Direction	Char	1 : Clockwise, -1 = Counter-clockwise

Example :
Set acceleration of 512 steps/s² for motor 2

CMD_CODE	MOTOR_NUM	Speed	Acceleration	Distance	Microsteps	Dir
ACCEL	Motor 2	N/A	512 steps/s²	N/A	N/A	N/A
0x08	0x02	0x0000	0x0200	0x00000000	0x00	0x00

Sets motor’s 1 speed to 2000 steps/s and turn it clockwise for a lot of steps (microstepping will be set to 1 and then back to the initial value)

CMD_CODE	MOTOR_NUM	Speed	Acceleration	Distance	Microsteps	Dir
GOFAST	Motor 1	2000 steps/s	N/A	159875 steps	N/A	Clockwise
0x08	0x01	0x07D0	0x0000	0x00027083	0x00	0x01

Commands options and values

SPEED :
- Motor number : CHAR (0x01 = motor 1, 0x02 = motor 2, etc.) ; 0xFF = ALL motors
- Speed : U_INT16 (in Steps/s)

ACCEL :
- Motor number : CHAR (0x01 = motor 1, 0x02 = motor 2, etc.) ; 0xFF = ALL motors
- Acceleration : U_INT16 (in Steps/s²)

GOFAST :
- Motor number : CHAR (0x01 = motor 1, 0x02 = motor 2, etc.) ; 0xFF = ALL motors
- Distance : [long] INT32 (in steps)
- Direction : char. 1 = clockwise, -1= counter-clockwise
- Speed : U_INT16 in steps/s

FOLLOW :
- Motor number :CHAR (0x01 = motor 1, 0x02 = motor 2, etc.) ; 0xFF = ALL motors
- Distance :long int32, set a long distance (will need to be stopped on the next GOFAST/SLOW command)
- Speed : U_INT16 in steps/s
- Direction : char. 1 = clockwise, -1= counter-clockwise

MS :
- Microsteps :Value from 1 to 255, did not correspond to actual microstep value but point to a preconfigured array where port pin is mapped on driver's microstepping pin. Need to be configured at compile time for each stepper driver