Commands

Motor command:	M	1 Byte
Select motor:	0 to N-1	1 Byte
Select type:	0 = FUNCTION	1 Byte
	1 = FREQUENCY	
	2 = MULTIPLIER	
	3 = PHASE	
	4 = REQUEST STATUS	(changed in the future to 255)

Case 1 = FREQUENCY

Value:	0 to 511	2 Bytes

Command back from FPGA:

END COMMAND	/R/N	2 Bytes

Case 2 = MULTIPLIER

Value:	0 to 255	1 Byte

Command back from FPGA:

END COMMAND	/R/N	2 Bytes

Case 0 = FUNCTION

Value:	0 to 4	1 Byte *(additional function: DC Generator will be set to type 4, type 5 to N will be custom RAMs)*

Hint: 0 = Sine; 1 = Triangle; 2 = Sawtooth; 3 = Rectangle; 4 = DC generator; 5 to N+4 = Custom

Command back from FPGA:

END COMMAND	/R/N	2 Bytes

Case 3 = PHASE

Motor:	0 to N-1	1 Byte
Phase value:	0 to 360	2 Bytes
Motor reference:	0 to N-1	1 Byte

Command back from FPGA:

END COMMAND	/R/N	2 Bytes

Case 4 = REQUEST STATUS

Type:	0 to 3 (0= FREQUENCY, 1=MULTIPLIER,2=FUNCTION, 3=PHASE)	1 Byte

Command back from FPGA:

Value	0 to 65535	2 Bytes
END COMMAND	/R/N	2 Bytes

ANSWERE from FPGA

(0= FREQUENCY, 1=MULTIPLIER,2=FUNCTION, 3=PHASE)

ANSWERE Case 1 = FREQUENCY

Frequency value:	0 to 511	2 Bytes
END COMMAND	/R/N	2 Bytes

ANSWERE Case 2 = MULTIPLIER

Multiplier value:	0 to 255	1 Byte
END COMMAND	/R/N	2 Bytes

ANSWERE Case 0 = FUNCTION

Multiplier value:	0 to 4	1 Byte
END COMMAND	/R/N	2 Bytes

ANSWERE Case 3 = PHASE

Phase value:	0 to Num Steps	2 Bytes (NUM_STEPS=511 or 1023)
END COMMAND	/R/N	2Bytes

Extra Command -> Write to CUSTOM ROM

RAM command:	C	1 Byte
Custom function:	0 to N-1	1 Byte
RAM Address:	0 to Num Steps	2 Bytes (NUM_STEPS=511 or 1023)
Value	0 to Num Steps	2 Bytes (NUM_STEPS=511 or 1023)

Command back from FPGA:

END COMMAND	/R/N	2 Bytes

Case Error

- *If command does not exist*
- *If time exceeds 5 seconds without finishing command*

Command back from FPGA:

Message	ERROR\R\N	7 Bytes

Future:

More CUSTOM function RAM (approx. Additional 5 RAMS) will be supported (-> paper)

Simple DC generator will be supported

Case 4 REQUEST STATUS will be changed to 255

ESP Implementation for easy use

Web GUI for values

Python interface for RS232 and UDP

Usage

Hint: All commands must be sent binary without exceeding 5 seconds!

Set Motor 1 to rectangle

M	1	0	3
01001101	00000001	00000000	00000011

Set multiplier of Motor 60 to 50%

M	60	2	50
01001101	00111100	00000010	00110010

Set Motor 8 to 512 Hz

M	1	1	511
01001101	00001000	00000001	11111111

Set Phase of Motor 4 to 180° in relation to motor 12

M	4	3	180°		12
01001101	00000100	00000011	00000000	10110100	00001100

Request frequency of Motor 7

M	1	255	1
01001101	00000111	11111111	00000001

➔ ***Answere in case of frequency was set to 65Hz***

65		/R	/N
00000000	01000001	00001101	00001010

Write to custom RAM 2 at address 100 a pwm value of 500

C	2	100		500	
01000011	00000010	00000000	01100100	00000001	11110100