Electronic Control Unit MARS 2 - USER GUIDE

Pavel Píša (pisa@cmp.felk.cvut.cz)

28th June 2000

Contents

1 Multiaxes Control System Description

Electronic control unit **MARS** is designed for position control of DC motors (up to the amount of three) with incremental sensors of position increment and one index mark per one motor revolve. Working area of the controlled equipment can be marked off by electromechanical terminal switches connected to power parts of the motors or by limit position sensors with logic outputs. To precise the initial position, it is possible to use terminal switches and sensors in a selectable combination with an index output.

Commands for the control unit can be inserted with the help of a selectable local keyboard or via I^2C interface of the connected keyboard. From a superior computer, the commands can be transmitted via I^2C interface, *RS-232* or via *RS-485*. The control unit can be equipped with an input for a three-axes analog joystick.

In addition to the motor control, 5 digital outputs, 4 power outputs and 2 outputs galvanicaly separated from the unit electronics can be used. Index marks of motors and up to nine digital inputs can be used as digital inputs. Five of them are also usable as analog inputs 0...2.5 V with resolution of 10 bits. When using joystick, three of the inputs are used for joystick.

When a change of the input state is defined and after selection of one of four digital inputs, each of two setable event triggers can send information about the position of the controlled axes and about the state of digital inputs. During the trigger activation it is possible to stop motion of the selected axes and set a state of digital outputs.

Unit MARS - Technical Data :

Variable	Value	Units
Supply voltage	18 - 36	VDC
Max. safe current for one motor	5	A
- current protection	electronic for each axes	
Position sensor input (IRC)	2 signals out of phase + TTL level index or RS-422	
Max. frequency of IRC signals	3	MHz
Inputs of joystick sensors	0 - 5	V
- supposed central position	2.5	V
- sensitivity of joystick sensors	range can be adjusted for a particular joystick with the help of resistors	
Fine tuning of sensitivity and zero position of joystick	by software	
Desired position generator	trapezoidal course with settable acceleration and maximal velocity	
Motors position control	sliding PID controller with nonlinear damping	
Setting of P, I, and D constants	by software	
Correction of power exciter non-sensitivity	by software	
Range of motor position	±8000.000	step 0.001
- range in periods of the IRC signal (4 phases)	4000000	
Absolute position calibration	automatic search of a selectable combination of the index and terminal switches	

For small and middle size position applications with high precision demands it is suitable to combine the **MARS** unit with **MAXON** RE 70 W / 42 VDC motors with **HP** HEDS 5540 incremental sensors with TTL outputs or **HP** HEDL 5540 with RS-422. These IRC sensors send 500 periods of out of phase signals per one motor revolve and are complemented by an index mark. The unit **MARS** can execute measurements with the accuracy of 1/2000 of one revolve, and realize position control and movement to the destination position with the accuracy of ±1/2000 of one revolve.

To mark off a working area it is possible to connect terminal (limit) switches with diodes to the power outputs for motors. It ensures return from the positions behind the limit switches and quick stop when getting out of the working area.

As a power supply a feeder with a double-insulated transformer or a pulse power supply can be used. The unit **MARS** , all peripheries and motors are not dangerous to touch when supplied.

Maintenance.

The electronic control unit MARS requires NO MAINTENANCE.

2 PC Control Via RS232 Port

The electronic control unit communicates with computers via serial asynchronous interface. The following configuration of the communication port is demanded: 9600 baud, 8 bits, no parity, and 2 stop-bits. Data flow control is executed with the help of hardware by signals **CST** and **RTS**. The communication speed is settable within the range from 2400 baud up to 19200 baud with the help of a local keyboard. The chosen value is loaded into the EEPROM memory along with other parameters.

2.1 Form of Commands and Requests Sent Via RS232

Each command consists of a name, operation symbol, and parameters. Individual parts of a command can be separated by spaces.

Name is created by an arbitrary combination of letters and numerals and always starts with a letter. Names of commands related to individual motors are supplemented with a symbol (A, B and C) specifying the motor 'm'.

Operation symbol defines a command (symbol ':') or a request (symbol '?'). The symbol '\' can be used to specify acknowledgements.

Parameter The meaning of the parameter is determined by the command. In the next, 'xxx.xxx' represents a decimal number, 'xxx' an integer number, and 'x' is a numeral. Negative numbers begin with the symbol '-'.

Names of Individual Commands.

- Movement to destination position and movement to reference mark commands, for more information see Section 2.2.1

  Gm, GRm, APm, HHm, HH

- Controller parameters setting, for more information see Section 2.2.2

  REGPm, REGIm, REGDm, REGS1m, REGS2m, REGMSm, REGACCm, REGMEm, REGCFGm, REGDBGm, REGTYPEm, REGSFRQ

- Controller response tuning support, for more information see Section 2.2.3

  REGDBGm, REGDBGHIS, REGDBGPRE, REGDBGGNS, REGDBGGNR

- Clearing, stopping and controller release, for more information see Section 2.2.4

  STOPm, STOP, PURGE, CLEARm, CLEAR, RELEASEm, RELEASE

- Acknowledgement of reception and completing commands, for more information see Section2.2.5

  READY, R, Rm, REPLY

- Joystick and keyboard, for more information see Section 2.2.6

  KEYLOCK, JOYSTICKm, JOYSTICK, JOYOFFSm, JOYRESm, JOYHYSm, JOYCAL

- System status, for more information see Section 2.2.7

  STm, ST

- Direct velocity control, for more information see Section 2.2.8

  SPDm, SPDTm

- Direct control of inputs and outputs, for more information see Section 2.2.9

  PWMm, DIGO, DIGI, ADCx, TRIGt

- System commands, for more information see Section 2.2.10

  VER, HEXLD, TEST, CFGNVSAVE, CFGDEFAULT

2.2 Commands Description

The detailed description of the individual commands is given in this section. The commands indicated by '*opt' are available only for several system configurations.

2.2.1 Movement to Destination Position And Movement to Reference Mark Commands

Name	Op	Parameters	Function
Gm	:	xxx.xxx	Movement to absolute position
		<-8000,8000.000>	

Moves the specified motor 'm' to the absolute position 'xxx.xxx'.

Name	Op	Parameters	Function
GRm	:	xxx.xxx	Movement to relative position
		<-8000,8000.000>	

Relative distance 'xxx.xxx' of the motor 'm' movement.

Name	Op	Parameters	Function
APm	?		Actual position

Returns the actual position of the specified motor 'm' (format 'xxx.xxx').

Name	Op	Parameters	Function
HH	:		Reference position

Finds the reference mark and resets the position data for all motors.

Name	Op	Parameters	Function
HHm	:		Reference position of 'm'

Finds the reference mark and resets the position data for the motor 'm'.

2.2.2 Controller Parameters Setting

Name	Op	Parameters	Function
REGPm	: ?	xxx	Proportional gain for 'm'
		<0,255>	
REGIm	: ?	xxx	Integral constant for 'm'
		<0,255>	
REGDm	: ?	xxx	Derivative constant for 'm'
		<0,255>	
REGS1m	: ?	xxx	1. auxiliary constant for 'm'
		<0,255>	
REGS2m	: ?	xxx	2. auxiliary constant for 'm'
		<0,255>	

These commands enable to set and read controller constants for the motor 'm'. The precise meaning of the parameters and their range depend on the type of the chosen controller for the given motor. For **PID** controllers , the parameters **S1** and **S2** serve to suppress the output exciter non-sensitivity.

Name	Op	Parameters	Function
REGMSm	: ?	xxx	Maximal velocity for 'm'
		<0,30000>	

This command enables to set the maximal velocity of the motor 'm' (format 'xxx'). The velocity parameter is used in the command of Destination Position Movement and also as a velocity limit when control is executed by joystick. The value is given directly in the IRC increment multiplied by 256 in one sampling period.

Name	Op	Parameters	Function
REGACCm	?	xxx	Acceleration for motor 'm'
		<0,30000>	

This command enables to set the acceleration of the motor 'm' (format 'xxx'). The acceleration parameter is used in the command Destination Position Movement with a trapezoidal course of velocity. In the configuration with a trapezoidal course of velocity, the parameter is used to slow down the velocity when the **STOP** command is applied or when the movement is interrupted by next **Gm**, **GRm**. The acceleration command can be also used to control velocity with the help of commands **SPDm** and **SPDTm**. The value determines the velocity increment per one sampling period.

Name	Op	Parameters	Function
REGMEm	: ?	xxx	Maximal PWM for 'm'
		<0,30000>	

The value of this parameter specifies the PWM level for motor 'm'. Reduction of the maximal PWM level reduces the maximum motor voltage represented in the percentage related to the supply voltage. In practice, this parameter is used to protect motors with the nominal voltage less than the supply voltage. The value of 32000 corresponds to the maximal voltage.

Name	Op	Parameters	Function
REGCFGm	: ?	xxx	Configuration word for 'm'
		<0,30000>	

For the motor 'm', the configuration word defines a velocity course when moving from one position to another one, a way of finding of the zero position, and conversions of logical and physical coordinates. The received decadic number is interpreted as a bit field xxxxxNxTxLCRDSSS. A complete description is given in Section 2.3.

Name	Op	Parameters	Function
REGTYPEm	:	x	Controller change for 'm'
*opt		<0,5>	

This command changes the type of the controller for the motor 'm'. The following table shows the possible types of controllers that are implemented in the contemporary software version. According to the software modification the default type is linear PID or PID with a nonlinearity.

Value	Controller
0	Default type
1	PID with nonlinearity
2	Linear PID
3	Discrete filter
4	Relay characteristic
5	Output without IRC coupling

Important warning: The contemporary software version does not enable to store the controller type into the setting in the EEPROM memory. If it is necessary to use another type of the controller for the motor than is the default controller, it is always necessary to execute a new setting after the unit switching on. According to the user demands the manufacturer is able to prepare a modified version with various default parameters and types of controllers for individual axes.

Name	Op	Parameters	Function
REGSFRQ	: ?	x	Sampling frequency of controllers
*opt		<0,4>	

This command sets a sampling frequency of the control loop of all motors. The meaning of the individual values is as follows: 0 .. default, 1 .. 600 Hz, 2 .. 800 Hz 3 .. 1000 Hz, 4 .. 1200 Hz. This value is stored along with other parameters into the EEPROM memory.

2.2.3 Controller Response Tuning Support

Name	Op	Parameters	Function
REGDBGm	:	x	Debugging of 'm' enabled
		<0,1>	

This command enables storage of the history of the motor 'm' motion and declares the motors which all other **REGDBGxxx** commands will relate to. For the motors with a set flag the actual values of velocity and values sent to the PWM generators are always stored after starting the motion.

Name	Op	Parameters	Function
REGDBGHIS	:	xxxx	Motion history reading
		<0,3000>	

This command reads the motion history. The parameter indicates a number of the sent numbers. Each number is sent in a separated line.

Name	Op	Parameters	Function
REGDBGPRE	:	xxxx	Data preprocessing for response debugging
		<0,3000>	

This command stores data for motor response debugging commands into the memory. The parameter indicates the number of the stored values. The values are sent in the separated lines.

Name	Op	Parameters	Function
REGDBGGNS	:		Motor response

It controls PWM outputs of the motors with **REGDBGm** set to 1 according to the values stored with the help of the command **REGDBGPRE**. After executing the command, the command **REGDBGHIS** enables to read the actual course of the motor motion.

Name	Op	Parameters	Function
REGDBGGNR	:		Response of the motor control

This command sets the required change of position for motor controllers with **REGDBGm** set to one according to the values stored with the help of command **REGDBGPRE**. After executing the command, the command **REGDBGHIS** enables to read the actual course of the motor motion.

2.2.4 Clearing, Stopping and Controllers Release

Name	Op	Parameters	Function
CLEARm	:		Control switching off and clearing of 'm'
CLEAR	:		Control switching off and clearing of all motors

Switching off of the motor 'm' control and clearing the position reading.

Name	Op	Parameters	Function
STOPm	:		Stopping of the motor 'm' motion
STOP	:		Stopping of all motors motion

Stops the motor 'm' motion, but control goes on. When a trapezoidal course of velocity is enabled, the motion fluently slows down before it stops.

Name	Op	Parameters	Function
PURGE	:		Stops control with an error

This command stops motor control with errors and resets the errors. Other motors continue their activities.

Name	Op	Parameters	Function
RELEASEm	:		Releases control and stops 'm'
RELEASE	:		Releases control and stops all motors

This command releases controllers and enables a manual control or stopping motors.

2.2.5 Acknowledgement of Commands Receiving and Completing

Name	Op	Parameters	Function
READY	:	x	Operation completed response
		<0,1>	

This command turns ON/OFF a message about finishing all operations. After finishing activities of all motors it sends the line 'R!' or 'FAIL!' (in case of error flag for at least one motor).

Name	Op	Parameters	Function
R	:		Only one operation completed response

This command sends only one 'R!' or 'FAIL!' after finishing just running operations or immediately when none of the motors is busy.

Name	Op	Parameters	Function
Rm	:		Response for a specific motor

This command sends 'Rm!' or 'FAILm!' after finishing activities of the motor 'm'.

Name	Op	Parameters	Function
REPLY	:	x	Command acknowledgement
		<0,1>	

This command turns ON/OFF confirmation of the lines by their copy. The copy starts with a backslash '\'.

2.2.6 Joystick and Keyboard

Name	Op	Parameters	Function
KEYLOCK	:	x	Keyboard locking
		<0,1>	

This command enables and disables control via the unit keyboard.

Name	Op	Parameters	Function
JOYSTICKm	:		Joystick control of the 'm' axis
JOYSTICK	:		Joystick control of all axes

Turns ON joystick control of the motor position.

Name	Op	Parameters	Function
JOYRESm	: ?	xxx	Joystick axis resolution
		<-32000,32000>	
JOYOFFSm	: ?	xxx	Central position offset of joystick axis
		<0,65000>	
JOYHYSm	: ?	xxx	Hysteresis angle of joystick axis
		<0,65000>	

For the individual joystick axes corresponding to the individual motors these commands set resolution, central position offset and hysteresis around the central position.

Name	Op	Parameters	Function
JOYCAL	:		Central position calibration

Calibration of the joystick central position (JOYOFFS) for all axes.

2.2.7 System Status

Name	Op	Parameters	Function
STm	?		Motor 'm' status

This command returns motor 'm' status. Decadic numbers need to be interpreted as bit fields.

Bit	Meaning
0	IRC reading enabled
1	Controller enabled
2	Generator enabled
3	Error
4	Last command execution
5	Control course storage ON

Name	Op	Parameters	Function
ST	?		Status of all motors

This command returns logical addition of states of all motors.

2.2.8 Direct Velocity Control

Name	Op	Parameters	Function
SPDm	:	xxx	Movement with given velocity
*opt		<-32000,32000>	

With this command the given velocity of the 'm' motor can be demanded. A velocity change can be executed by the acceleration command **REGACCm**.

Name	Op	Parameters	Function
SPDTm	:	xxx,yyy	Movement with the velocity of 'xxx' continuing within the max. interval 'yyy'
*opt		<-32000,32000>	
		<0,32000>	

With this command the given velocity of the 'm' motor can be demanded. A velocity change can be executed by the acceleration command **REGACCm**. The movement continues within 'yyy' sampling periods. If a next command for the given motor has not come until this time, the movement is smoothly stopped.

2.2.9 Direct Control of Inputs and Outputs

Name	Op	Parameters	Function
PWMm	:	xxx	Direct PWM setting
*opt		<-32000,32000>	

Direct PWM output setting of the 'm' motor to the 'xxx' value.

Name	Op	Parameters	Function
DIGO	:	xxxxx	Digital outputs setting
*opt		<0,65535>	

This command sets the digital outputs 0 .. 15 according to the individual bits of the parameter binary form.

Name	Op	Parameters	Function
DIGI	?		Digital input state reading
*opt			

Reads the digital inputs 0 .. 15 and sends the read state as a decadic value.

Name	Op	Parameters	Function
ADCa	?		Analog input reading
*opt			

Returns a value of the analog input 'a'.

Name	Op	Parameters	Function
TRIGt	:	s,m,do	Trigger setting
*opt			

Event triggers - for more information see Section 2.4.

2.2.10 System Commands

Name	Op	Parameters	Function
VER	?		Returns software version

This command returns the unit firmware version.

Name	Op	Parameters	Function
IHEXLD	:		Loads new software

This command loads a program in the Intel-HEX format into the RAM memory of the **MARS** unit.

Name	**Op**	**Parameters**	**Function**
TEST	: ?		Connection test

Connection checking and input or output command sheet test.

Name	**Op**	**Parameters**	**Function**
REBOOT	:		Unit software reboot

Resets the unit software. All parameter values are loaded from EEPROM memory again.

Name	**Op**	**Parameters**	**Function**
CFGNVSAVE	:		Saves parameters into EEPROM

Saves parameters settings for individual motors and sampling periods into the EEPROM memory. It guarantees that the saved parameters values will be loaded after the next switch ON of the unit.

Name	**Op**	**Parameters**	**Function**
CFGDEFAULT	:		Default parameters setting

Loads standard parameters setting for all motors compiled in the software by the manufacturer.

2.3 Configuration

Configuration setting for individual axes executed by the command REGCFGm determinates a course of velocity when moving from one position to another one and a way of zero position finding. The inserted decadic number is interpreted as a bit field.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Flags	x	x	x	x	x	N	x	T	x	L	C	R	D	S	S	S

The meaning of the individual flags is given in the following table.

Flags	Meaning
SSS	Stop search speed during HH is REGMSm / 2^SSS
D	Initial direction for stop search for HH
R	To find HH the revolve mark from HP HEDS is used
C	Mark middle search
L	Limit switch usage
T	Trapezoidal velocity profile usage
N	Coordinates from RS232 are not converted
LCR=111	Mark search only
CR=11	Mark middle search only

To simplify the configuration computation, all combinations of LCR bits are given bellow. To the number it is necessary to add only 16 (D) to change the initial direction,(SSS) to reduce the stop search speed and 256 (T) for fluent starts.

Value	Meaning
0	Motor limit switch search only
16	Search the first revolve mark (first from the hardware terminal switch)
32	Search the middle of the first revolve mark (first from the hardware terminal switch)
48	Search the middle of the mark
64	Search the limit switch only
80	Search the first revolve mark (first from the limit switch)
96	Search the middle of the first revolve mark (first from the limit switch)
112	Mark search only

2.4 Event Triggers

Event trigger setting is executed by the following command:

TRIGt:s,m,do		
t	Trigger number (0 or 1)	
s	Trigger source 0 .. 3 (-1 trigger disconnection)	
	s.0 s.1	Source number
	s.4	Rising edge
	s.5	Trailing edge
	s.6	Sent input state
	s.7	Set digital outputs
m	Mask of the motors that should be stopped and transmission of their position	
	m.0	Stop A
	m.1	Stop B
	m.2	Stop C
	m.4	Send A
	m.5	Send B
	m.6	Send C
do	Value of digital outputs set at trigger event	

Triggers transmit:

TGt!di,ma,mb,mc	
t	Trigger number (0 or 1)
di	Digital inputs at trigger event or "N", if transition disabled
ma,mb,mc	According to the choice of m in the setting it sends the IRC position

3 Control Via IIC Interface

A message consists of an information about a type of the message and of data. Types of the I^2C messages used in the unit **MARS** are given in the following table.

Symbol	Code	Type of message
IC_CMD	40H	Commands for motor control
IIC_CMM	41H	Reserved
IIC_STM	42H	Reserved
IIC_TEC	51H	Control of IIC keyboard and a display
IIC_TEK	52H	Information about pressed keys
IIC_TED	53H	Data display

Messages of IIC_CMD type are used to send commands for position controllers, to set controller parameters, and to read position and state of individual motors. A format of a full message is shown bellow.

Write to IIC addr 10H+W	
0	Type of the IIC_CMD message
1	Operation
2	Motor number
3,4,[5,6]	Parameter value for given motor
Read from IIC addr 10H+R	
0,1	System status
2,3,[4,5]	Parameter value of given motor

If the unit **MARS** receives a message containing only a type of the message no operation will be executed and during the next reading a system status will be sent. If the message contains only a type of the message and an operation the command will be run for all axes and a next reading will return a system status. When a motor number is added, the value of the parameter specified by the operation and the motor number are also read during the reading operation. If the message also contains a value, the value is saved into a specified parameter. The unit **MARS** enables to receive and read 16-bit and 32-bit parameters. Longer representations are advantageous especially for position data.

Codes for the individual operations are given in the table bellow. In the second column (Parameter) there are given the parameter values that are settable or readable for the given operation.

Code	Parameter	Description
0	Actual position	Actual position reading
1	Desired position	Desired position setting
2	Desired position	After inserting the position the motion starts up
3	Status	Runs hard home
4	REG P	Setting and reading of P constant
5	REG I	Setting and reading of I constant
6	REG D	Setting and reading of D constant
7	REG ME	Maximal duty of PWM
8	REG MS	Maximal speed of motion
9	REG ACC	Maximal acceleration
A	REG SCM	Multiplicative constant for coordinates conversion
B	REG SCD	Dividing constant for coordinates conversion
C	Min LS	Not implemented
D	Max LS	Not implemented
E	REG CFG	Configuration flags
F	Status	Motor status

Flags setting for hard home is the same as the setting via RS-232 (see 2.3).
State flags read via I^2C are given in the following table.

Bit	Symbol	Description
0	TRP_SGN	
1	FL_HH	
2	FL_ATIM	
3	FL_ENOV	Energy overflow for a motor start
4	ENE_ON	
5	ERR_FLG	Error during control
6	CMD_LCK	Next command can not be received
7	CMD_BSY	Last command execution

4 Manual Control

The MARS control unit can be handled directly by the local keyboard or a keyboard connected via the I^2C interface.

Button Function

A, B, C Switches the display over to control individual motors: A, B, and C. Writing a number and pressing the ENTER key runs a movement to the given position. Numbers are inserted by the keys 0 .. 9 and a decimal point. Signs can be changed by the key IMPL *.*

LIST Displays summary data for all motors. Individual motors are accessible via cursors ↑ and↓.

RUN Enables control of all axes by joystick.

HOLD Runs hard home (zero position search) for all axes

END Disconnects controllers and sets the actual position to zero

PURGE Stops control with an error - other operations go on.

MOTOR STOP Stops all motions. Control of all motors goes on.

5 Connectors and Wiring

Connector for connecting joystick

For the **MARS** unit, an analog joystick with three potentiometers is required. The unit connector has the following wiring:

Pin	**Signal**	**Alt. TTL input**	**DB**9
1	VCC		8,9
2	INPUT A	I13	3
3	INPUT B	I14	4
4	INPUT C	I15	5
5	GND		1,2

In case of encasing the electronic control unit, the input of the joystick is wired to the female DB9 Canon connector.

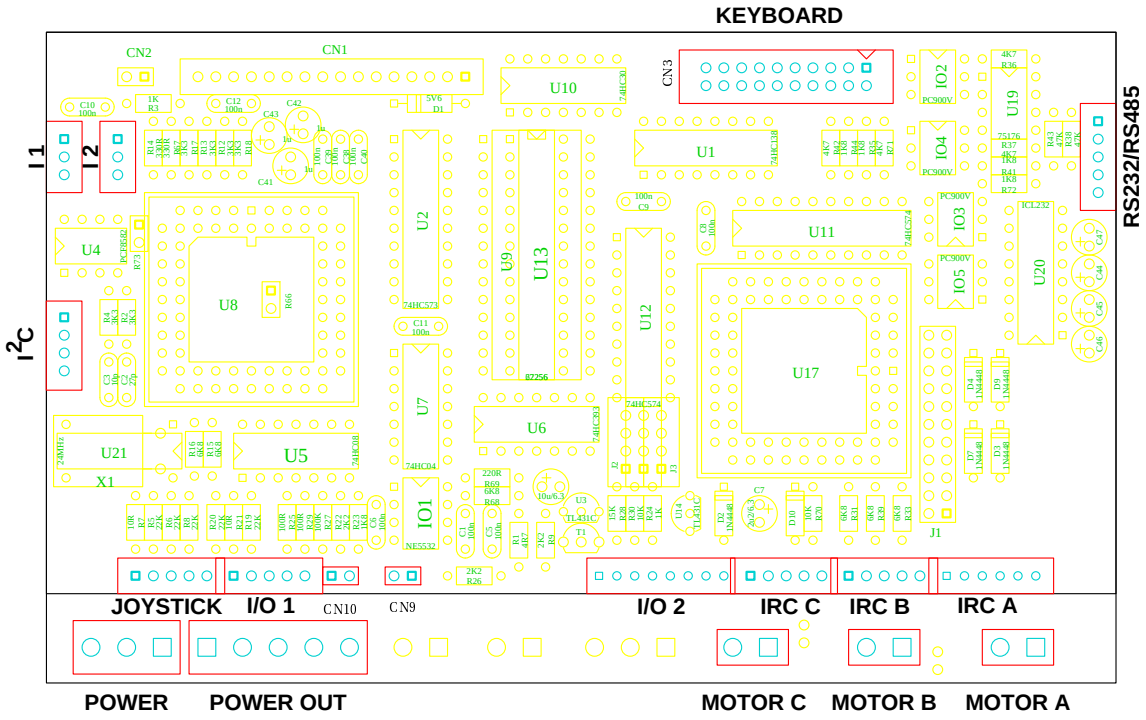

Figure 1: Connections layout on the unit **MARS**

Communication connector RS232

Cable is connected to the unit board via a crimped connector. To the control unit, a transfer connector is supplied to use standard connectors DB9 or DB25. To connect the unit to PC, the connection with DB 25V is given in the table bellow.

MARS			PC		
Board pin	Signal	DB9	Signal	DB25	DB9
1	TxD	2	RxD	3	2
2	RxD	3	TxD	2	3
3	RTS	8	CTS	5	8
4	CTS	7	RTS	4	7
5	GND	5	GND	7	5

Connectors for connecting motors and IRC sensors

Power outputs for individual motors are accessible via MOTOR A, MOTOR B and MOTOR C connectors. Incremental sensors of position need to be connected to the unit via IRC A,IRC B and IRC C connectors.

Pin	Signal
1	GND 0V
2	Phase A
3	VCC +5V
4	Phase B
5	INDEX

Index inputs can be also used and read as digital inputs

IRC A INDEX	I0
IRC B INDEX	I1
IRC C INDEX	I5

Digital inputs and outputs

Pin	Signal
Connector I1	
1	VCC
2	I11 (I2)
3	GND
Connector I2	
1	VCC
2	I12 (I3)
3	GND
Connector I/O1	
1	GND
2	I6
3	VCC
4	I7
5	O14
Connector I/O2	
1	I5
2	I4
3	O0
4	O1
5	O2
6	O3
7	VCC
8	GND
POWER OUT	
1	PWR 12 - 24V
2	O4
3	O5
4	O6
5	O7

Communication I^2C

Pin	Signal
1	GND
2	SCL
3	VCC
4	SDA

6 Manufactured by:

The electronic control unit is manufactured and supplied by:

PiKRON Ltd.
Kaňkovského 1235
18200 PRAGUE 8

Phone: Fax: (++420)(2) 688 46 76
Phone : (++420)(2) 409 76 71