Table of contents

1. Introduction

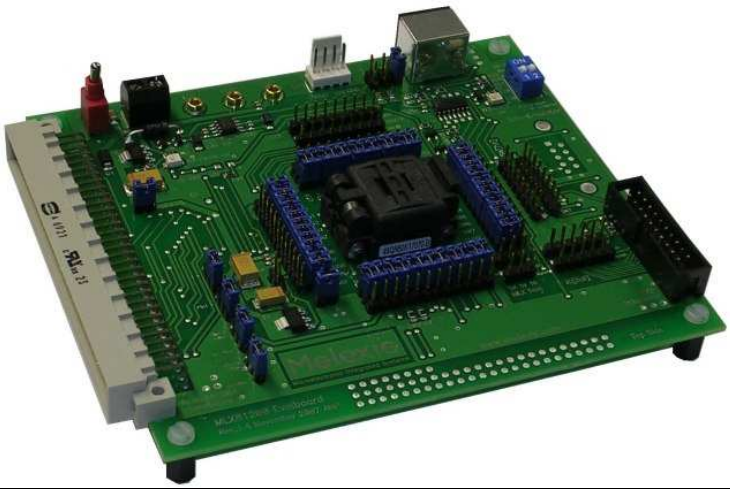
This document is intended to give a brief introduction of the different parts of the SW Development kit for the ASSP MLX81200. The aim of this document is to support a fast start up with this kit. Besides this document, several other important documents are necessary for a deeper understanding of more detailed development issues.

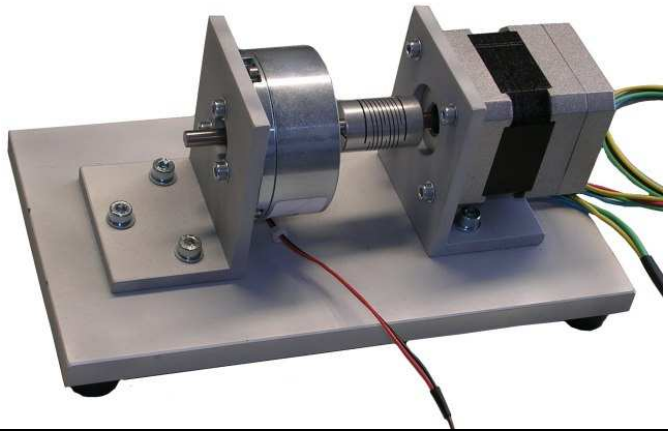
The most important documents related to the MLX81200 are:

MLX81200 Datasheet	-detailed description of the chip with all functions and features
MLX82001_Product_Specification	-MelexCM Datasheet and underlying documentation of different blocks included in the MLX82001
Melexis development system	-MLX Assembler, Linker, Obgen, Tabgen description
GNU C-Compiler GCC	-GCC User's Manual, incl. AS, LD and Getting Started Doc
Melexis emulator (flash programming SW)	-description of the Emulator
Melexis emulator SW	
Mlx16 C-Debugger	-Debugger User's Manual
Mlx16 CPU Simulator	-description of the Simulator
Mlx16X8 Data book	-explanation of the Mlx16X8 microcontroller core and the instruction set
Melexis Lin Master documentation	-description of the Melexis USB LIN Master
MelexCM LIN API (including required SW files)	-detailed description of the MelexCM LIN API
MLX81200 BLDC Demo Kit firmware	-MLX81200_SWDesignDescription_x.pdf
AppNote_MLX81100_MLX81200_Reflashing_on_module_x.pdf	
ApplNote_Flash_over_pin_LIN_x.pdf	

2. Contents of the MLX81200 SW Development kit

2.1. Hardware

Part	Picture
MLX81200	
Evaluation board	
Power board	

USB LIN Master	
Mini E-MLX Emulator	
BLDC Motor and hysteresis brake	
Hysteresis brake control board	

Tbl. 1 Contents Hardware

2.2. Software

Part	Function	Installation file
Mlx16 Simulator/Emulator	Simulator/Emulator PC Software	Mlx16Setup_x_xx_x.exe
Mlx81200x.mmf	Configuration file for Simulator/Emulator and Flash Programmer	Mlx81200Conf_x_x_x.msi
E-mlx MM/Mini E-MLX Emulator	Emulator Software and hardware driver	EmlxSetup_x_x_x.msi
Mlx16 Interactive C Debugger	Melexis C Debugger	MDC_Setup_x_x_x.exe
Flash Programmer Software for E-mlx MM and Mini E-MLX	Programming software for Flash memory	EMlxMMProgSetup_x_x_xx.msi
Gnu Tool chain	Mlx16 C Compiler and associated tools	Mlx16_GCC_setup_x_xx_x.exe
MLX81200 SW Platform	Includes general chip operation especially the LIN feature	Available on Softdist
Melexis USB LIN Master Software	PC software for LIN Master	Comes with MLX USB LIN Master on CD
MLX81200 BLDC Firmware	BLDC Firmware for the Demo kit setup	BLDC_DemoKit_Firmware_x.zip
MLX81200 PC program	PC program to control the BLDC firmware via LIN	MLX81200_Demokit_PCProgram_x.exe
MLX81200 SCT	PC program to configure, build und flash the BLDC-Demokit firmware	MLX81200_SCT_x.exe
MLX81200 Software Source code	BLDC Firmware sources for the BLDC Demokit	BLDC_x_Demokit.zip

Tbl. 2 Contents Software

3. How to use the MLX81200 SW Development kit

The following table introduces the philosophy of the MLX SW Development kit and gives an overview about the different options for development with the kit.

Task	Hardware	81200 specific hardware	Required software
Software development in Assembler			- Editor - Mlx16 Melexis Assembler
Software development in C			- Editor - Mlx16 GCC C Compiler
Software Simulation in Assembler	- PC		- Mlx16 Simulator
Software Simulation in C	- PC		- Mlx16 C Debugger
In-Circuit Emulation in Assembler	- PC - Mini E-MLX Emulator	- Evaluation board	- Mlx16 Simulator - Mlx16 Emulator - 81200x.mmf configuration file
In-Circuit Emulation in C	- PC - Mini E-MLX Emulator	- Evaluation board	- Mlx16 Simulator - Mlx16 Emulator - 81200x.mmf configuration file
Quick HW Test	- PC - Mini E-MLX Emulator	- Evaluation board	- Mlx16 Simulator - Mlx16 Emulator - 81200x.mmf configuration file - Demo Software for the Evaluation board
Configure, build and program via pin LIN	-PC -Mini E-Mlx Emulator or -USB-LinMaster 2.0	-Evaluation board -MLX81200 IC with loader support	-Mlx16 GCC Compiler -MLX81200 SCT

Tbl. 3 How to use the SW Development kit

4. The Software Kit

4.1. Software Evaluation flow

The documentation for the development tools listed above is included with those tools and is outside the scope of this document. Despite that, a brief flow of the software development for the MLX81200 is given below in Fig. 1. Software can generally be written in Assembler or in C.

4.1.1. Software Development in Assembler language

Assembler source file(s) can be written with all common non-formatting editors. The code is then compiled with a Melexis supplied assembler, which is customized for the Mlx16X8 instruction set and supports all standard features of other commercial compilers.
The linker will merge the object files generated by the assembler and create a final file (*.hex, *.cod). This final *.hex file can be used in two different ways:

- for simulation purposes with the Mlx16 Simulator
- for in-circuit debugging/testing (programmed to Flash)

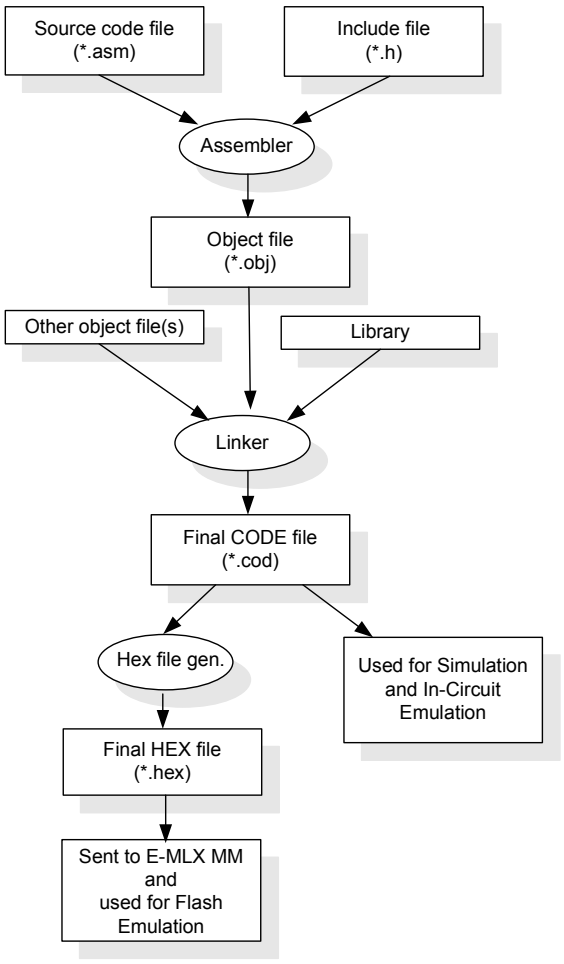


Fig. 1 Flow of the Software Development using Assembler and Melexis tools

4.1.2. Software Development in C language

Gnu has also developed a C compiler, Assembler and Linker. This tool chain supports all known features of other common compilers. The creation of C source file(s) can be achieved with any common non-formatting editor. The code is then compiled and linked with the above mentioned Gnu tools (see Tbl. 2)
With this methodology:

- All C language based features can be used
- Assembler parts (previously compiled to object file level) can be included
- ROM and In-circuit emulation is possible on C instruction level

NOTE: The object files of Gnu and Melexis supplied Assembler are not compatible.
Therefore the Gnu provided Assembler must be used, in case Assembler sources should be included with a Gnu C source.

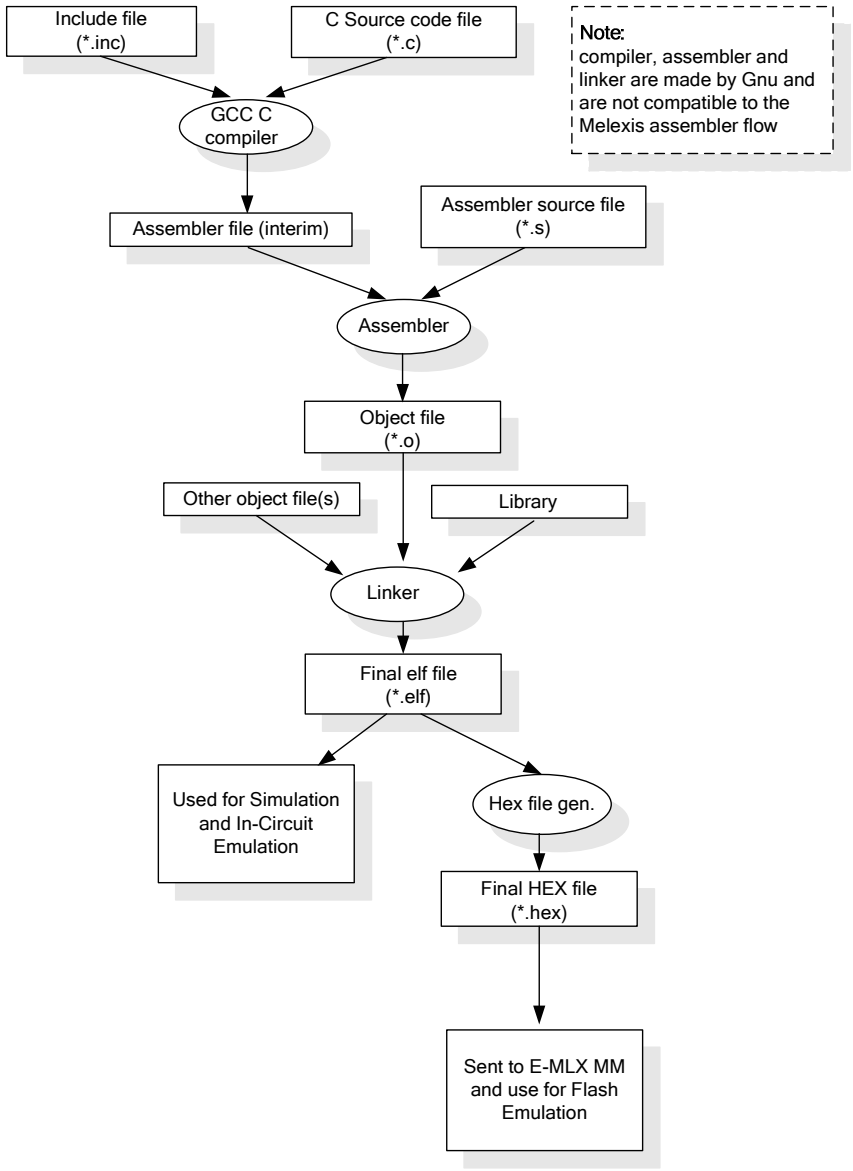


Fig. 2 Flow of the Software Development using the Gnu C Compiler tools

4.2. The installation of the Software Kit

4.2.1. Installation of the tools

All tools run under WinXP®. Win9x, WinNT® and Win2000® are not supported. All programs use the standard windows installer of WinXP®.

The following tools have to be installed:

• EmlxSetup_x_x_x.msi	Emulator Software and hardware driver
• Mlx16Setup_x_xx_x.exe	Mlx16 Simulator Software
• MDC_Setup_x_x_x.exe	Mlx16 Interactive Debugger
• EMlxMMProgSetup_x_x_xx.msi	Flash Programmer Software
• Mlx81200Conf_x_x_x.msi	Configuration file, consists of chip specific settings
• Mlx16_GCC_setup_x_xx_x.exe	Mlx16 C Compiler
• MLXLinMaster_Setup.exe	USB LIN Master Software
• MlxLinDebug.exe	Melexis USB LIN Master
• MLX81200_Demokit_PCProgram_x.exe	PC program to control the BLDC firmware via LIN
• MLX81200_SCT_x.exe	PC program to configure, build und flash the BLDC-Demokit firmware

x – Revision number

Demo Software can be copied at any top level directory:

• Software_Platform_MelexCM_x_x_x.zip	Software platform including the Demo Software
• BLDC_DemoKit_Firmware_x_x.zip	firmware for the Evaluation setup
• BLDC_x_Demokit.zip	the source code of the BLDC DemoKit firmware

4.2.2. Directory structure of the Software Tools

It is recommended to use the “standard” installation option. After installing with this installation option, the following path settings and directory structure appears:

C:\Programs\Melexis\

E-Mlx MM\Programmer	FLASH Programmer Software
Bin	MLX Assembler Package
Bin\Doc	MLX Assembler Development System Documentation
Emulator	Emulator Software and help file
Simulator	MLX16 Simulator/Emulator (Assembler)
MDC	MLX16 Interactive C Debugger and manual
MLX81200_DemoKit_PCprogram	MLX81200 Control Software

The release of Mx16 GCC suite installs in the following directory tree:
C:\mlx16-gcc\

bin	contains executable files.
Lib	contains libgcc library and include files and compiler specs.
mlx16	contains libmlx16 library and include files; start-up module; linker command files and memory map.
libexec	additional executable files.
config	configuration file and script for Mlx16 Simulator.
docs	documentation.
examples	sample programs.

C:\Programs\Common Files\Melexis Shared\Config\81200

81200.mmf	Configuration file for Simulator, Emulator and Flash Programmer

C:\Programme\LIN Commander\LinCommander.exe

LinCommander.exe	a sample interface program that can be used to debug via the LIN bus when developing an application.

4.2.3. Directory structure of the Demo Software

The setup works with relative paths, so it can be copied into any top level directory.

NOTE: Please make sure that directory path does not contain any spaces

It contains the following folders:

\bin	Utilities
\config	Makefile configurations
\doc	Documentation related to general MelexCM platform software
\include	Common libraries include directory
\lib	Common libraries (.a), library object files and linker files
\mmf	Test controller files and tools
\libsrc	Root of the common libraries sources
\libsrc\lib	Source code of the common libraries
\libsrc\LIN	LIN library source code
\libsrc\math	Math library source code
\projects	Root of the projects sources
\projects\Examples	Projects which are independent from analogue chip periphery
\projects\81200	Root of the MLX81200 project
\projects\81200\include	The MLX81200 specific declarations (analogue port names etc.)
\projects\81200\src	The MLX81200 common versions of the product specific functions (analog_trimming, power_down and vectors.S) and projects
\projects\LINLoader	LIN pin loader source code (LIN mode or standalone)

Tbl. 4 Folder description

5. Hardware Kit

5.1. Configuration

- Evaluation board in conjunction with the Mini E-MLX Emulator
- Used for In-Circuit debugging
- Flash programming possible
- LIN Communication possible

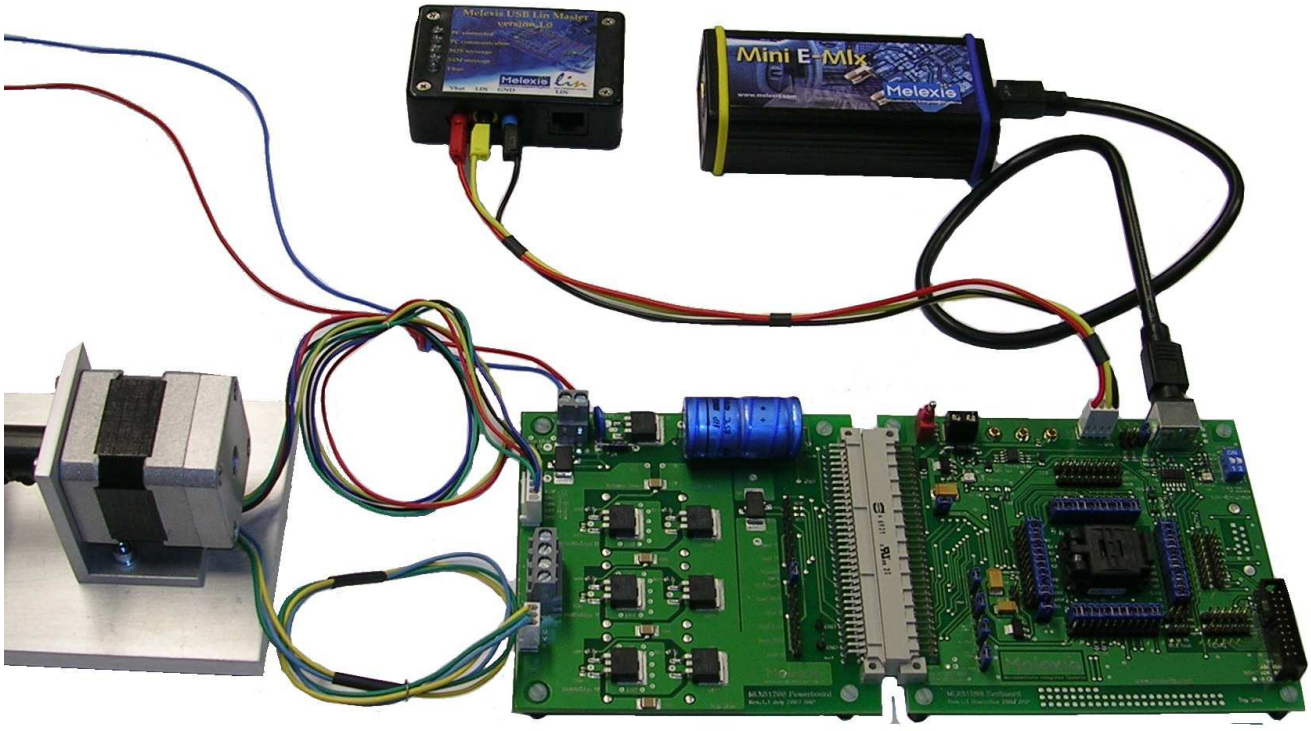


Fig. 3 Arrangement of Evaluation board in conjunction with Power board, Mini E-MLX Emulator , BLDC motor and Melexis USB LIN Master

5.2. General

The purpose of the HW kit is the development of software for the MLX81200 BLDC Motor Controller. Using this HW kit, the device can be evaluated in a detailed fashion.

The evaluation system is composed of an evaluation board and the power board. The idea of the evaluation system is to have two stand-alone boards:

- A standard board (evaluation board) is used for all possible applications. It consists of a socket for the chip, several pin headers for all signals from the chip and interface connectors to the Emulator, LIN-Bus and the application board (power board).
- A customized application board (power board) has to be designed specifically for a certain application by the user. It is possible to connect this power board on 32-pin connector. The sample power board consists of 3 half Bridges with N-FET-transistors, 2 shunts for possible current supervision and interface connectors to Vbat and the BLDC-Motor.

5.3. Evaluation board

5.3.1. Evaluation board - jumper settings

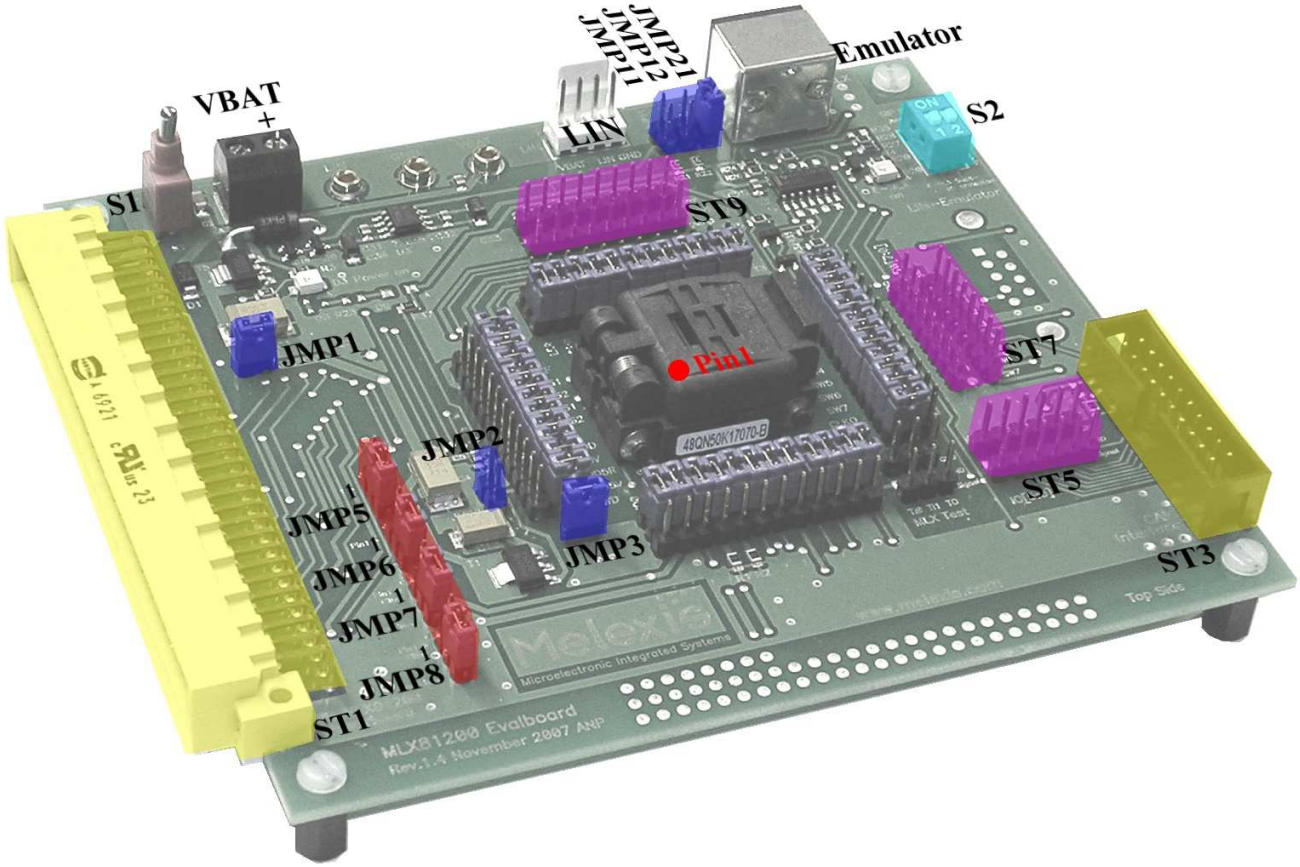


Fig. 4 Location of Connectors on Evaluation board

JMP1 short : connects Vbat to the chip.
open : disconnects Vbat from the chip

JMP2 short : connects CWD-pin to GND
open : CWD-pin of the chip is active.
NOTE: the external capacitor for analog Watchdog is active.

JMP3 short : disables the external regulator NPN transistor for higher VCC loads.
open : enables the external regulator NPN transistor for higher VCC loads

JMP5 1-2 : enable low pass filter R55/C15 and voltage divider R5/R50 for pin T
2-3 : enable only the voltage divider R5/R50 for pin T

JMP6 1-2 : enable low pass filter R66/C16 and voltage divider R6/R60 for pin SW5
2-3 : enable only the voltage divider R6/R60 for pin SW5

JMP7 1-2 : enable low pass filter R77/C17 and voltage divider R7/R70 for pin SW6
2-3 : enable only the voltage divider R7/R70 for pin SW6

JMP8 1-2 : enable low pass filter R88/C18 and voltage divider R8/R80 for pin SW7
2-3 : enable only the voltage divider R8/R80 for pin SW7

S2 consists of the jumper JMP9 and JMP10:
JMP9 short : the connection between the chip and the emulator is split
open : the connection between the chip and the emulator is established, the chip can be accessed by the emulator, if the emulator is not plugged the CPU is executing the flash program

JMP10 short : CPU does not execute the flash program
open : CPU is executing the flash program, if no emulator is connected and JMP9 is shorted

JMP11 short : the pull up resistors R21 and R22 are connected to the test interface inputs
+JMP12 (necessary if the E-mlx MM programmer is used)
open : the pull up resistors are disconnected (default for the Mini E-Mlx programmer)

JMP21 short : supply the additional voltage regulators inside the Mini E-Mlx programmer (required)
open :disconnect VBAT from the test interface (required for the E-mlx MM programmer)
NOTE: If the E-Mlx MM programmer is connected and the JMP21 is shorted the diode D4 can be destroyed.

5.3.2. Functionality of the Evaluation Board

Main Power Switch S1 selects between supply connector of the evaluation board or the supply connector of the power board.

IMPORTANT: If the evaluation board is supplied via the power board, the power supply connector of the Evaluation board should **NOT** be connected! Otherwise the board or the power supply will be damaged!

The PCB only requires a DC voltage of +12…+18V which is applied via the supply connector of the evaluation board. This is valid if the evaluation board is used stand-alone without the power board. In case the board is used alongside the power board, the power board delivers the supply voltage for both PCBs.

Both the evaluation board and the power board are protected against reverse polarity.

5.3.3. Evaluation board – Phase signal filters

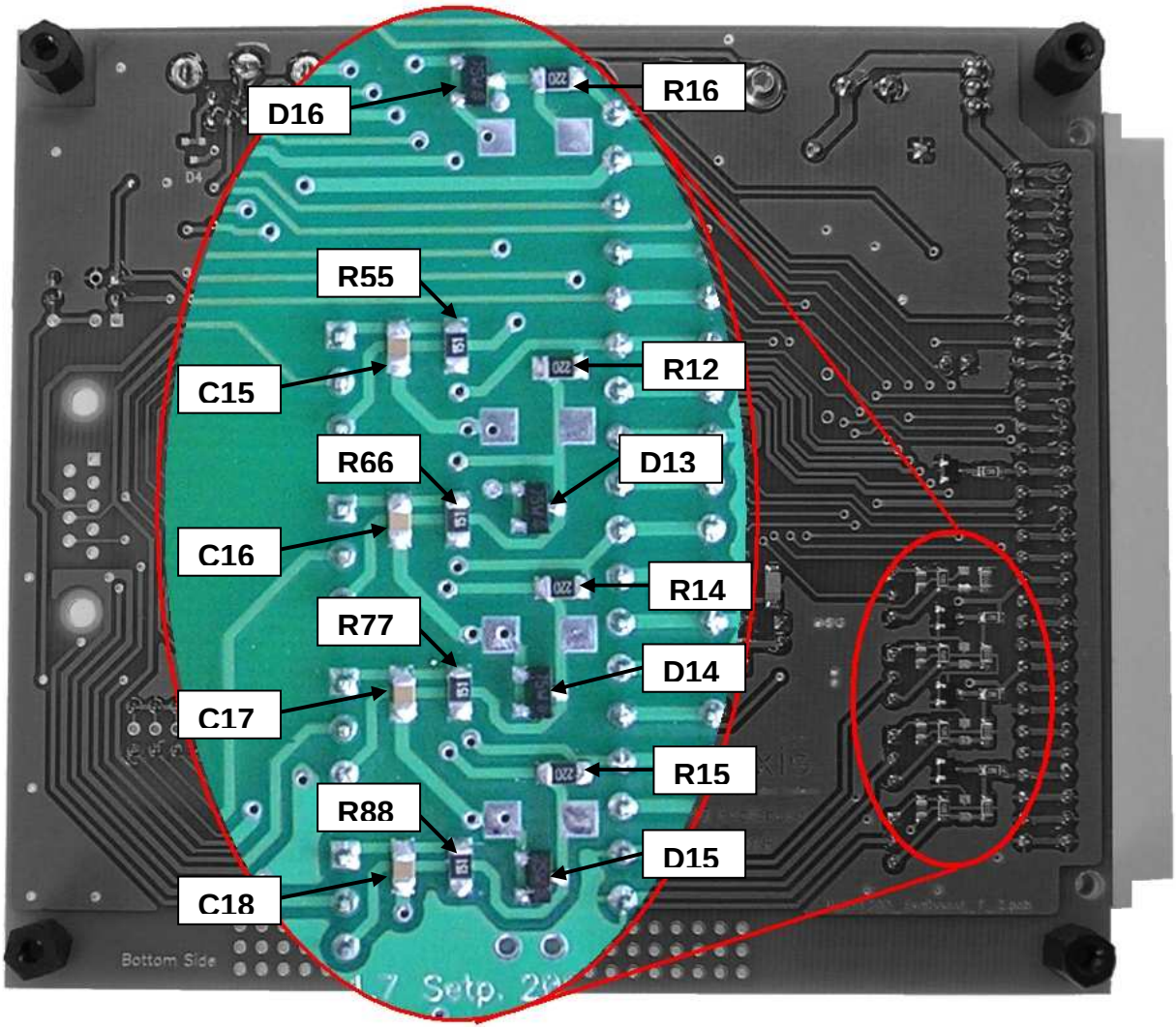


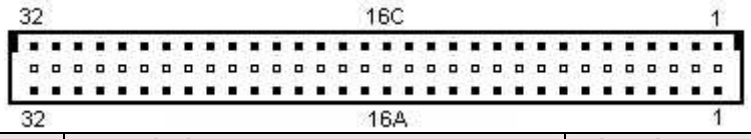
Fig. 5 Position of the filter components on the EVB

The resistors R13, R14, R15, R16 and the diodes D13, D14, D15, D16 protect the IC against negative current and voltage from the motor coils. The resistors increase the discharging time of the high side n-channel MOSFET transistor gates.
With the low pass filters R55/C15, R66/C16, R77/C17 and R88/C18 fast disturbances can be filtered out. The default value for the resistor is 150 ohm and for the capacitor is 10nF.

5.3.4. Evaluation board - Jumper ring

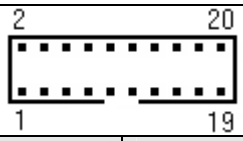
The jumper ring enables the user to either connect or disconnect all signal lines to the chip. By removing jumpers, the circuit from the chip to the application hardware can be disconnected e.g. for measuring current or applying external signals. All jumpers are described with the specific pin names on the board.

5.3.5. Connection between Power Board Port and Evaluation Board

Connector ST1 on Evaluation board			
Pin	**Name**	**Description**	**Direction**
1	GND	Ground	
2	GND	Ground	
3	GND	Ground	
4	W	IC-Pin 24	Input
5	HS2	IC-Pin 25	Output
6	SW0	IC-Pin 23	Input/Output
7	GND	Ground	
8	SW1	IC-Pin 22	Input/Output
9	V	IC-Pin 31	Input
10	HS1	IC-Pin 30	Output
11	GND	Ground	
12	SW2	IC-Pin 21	Input/Output
13	U	IC-Pin 33	Input
14	HS0	IC-Pin 34	Output
15	GND	Ground	
16A	VBAT_S2	IC-Pin 40	Input
16C	VBAT_S1	IC-Pin 39	Input
17	V5EXT	IC-Pin 2	Output
18	T	IC-Pin 37	Input
19	GND	Ground	
20	LS2	IC-Pin 27	Output
21	LS1	IC-Pin 28	Output
22	LS0	IC-Pin 36	Output
23	GND	Ground	
24	EN_VS	Bootstrap output from IC-Pin 15 -CLKO	Output
25	SW3	IC-Pin 20	Input/Output
26	GND	Ground	
27	GND	Ground	
28	GND_S2	IC-Pin 42	Input
29	GND_S1	IC-Pin 41	Input
30	GND	Ground	
31	VBAT	unprotected voltage from Power board	Input
32	VBAT	unprotected voltage from Power board	Input

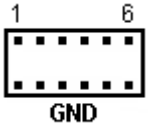
Tbl. 5 Connector ST1(male) on Evaluation board

5.3.6. Evaluation board – extension port ST3

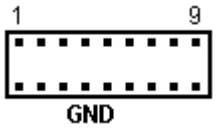
Connector ST3 on Evaluation board		
Pin	**Name**	**Description**
1	V5V or V5EXT	5V power supply or protected switchable V5ext
2	VS	Protected voltage
3	SW3	STB
4	SW4	Wake-up interrupt signal (INH)
5	GND	Ground
6	IO5	Interrupt from CAN module
7	GND	Ground
8	IO4	SPI clock
9	GND	Ground
10	IO3	SPI MOSI
11	GND	Ground
12	IO2	SPI MISO
13	GND	Ground
14	IO1	SPI chip select
15	GND	Ground
16	Not used	
17	Not used	
18	Not used	
19	Not used	
20	Not used	

Tbl. 6 Connector ST3 on Evaluation board (for example a CAN extension board can be connected)

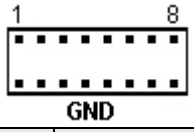
5.3.7. Evaluation board - Signal pin headers

Connector ST5 on Evaluation board		
Pin	**Name**	**Description**
1	IO0	IC Pin 1
2	IO1	IC Pin 7
3	IO2	IC Pin 10
4	IO3	IC Pin 12
5	IO4	IC Pin 9
6	IO5	IC Pin 3

Tbl. 7 Connector ST5 on Evaluation board

Connector ST7 on Evaluation board		
Pin	**Name**	**Description**
1	LS0	IC Pin 36
2	CP0	IC Pin 25
3	HS0	IC Pin 34
4	HS1	IC Pin 30
5	CP1	IC Pin 29
6	LS1	IC Pin 28
7	LS2	IC Pin 27
8	CP2	IC Pin 26
9	HS2	IC Pin 25

Tbl. 8 Connector ST7 on Evaluation board

Connector ST9 on Evaluation board		
Pin	**Name**	**Description**
1	SW0	IC Pin 23
2	SW1	IC Pin 22
3	SW2	IC Pin 21
4	SW3	IC Pin 20
5	SW4	IC Pin 19
6	SW5	IC Pin 18
7	SW6	IC Pin 17
8	SW7	IC Pin 16

Tbl. 9 Connector ST9 on Evaluation board

5.3.8. Evaluation board - Reset

Resets are always active low and can be achieved by the following options:

- Turning off the power switch on the Evaluation board resets all components of the Evaluation board as soon as the charge of the block capacitors falls below the reset threshold level

In case of working with the Mini E-Mlx Emulator, there are two more reset options (this is valid only for the digital part of the chip, MelexCM):

- Sending a new software (*.x16 or *.elf file) to the Emulator will reset the MelexCM chip automatically
- Reset can also be achieved by the *'Reset'* button of the PC Simulator/Debugger software.

IMPORTANT: Reset of the whole chip including the analog part can ONLY be achieved by turning off the power switch.

5.4. Power board

5.4.1. Power board – Electrical continues parameter

The power board is designed for a maximum continues current up to 15 amps motor current. Motor currents up to 40 amps are allowed for maximum 1 minute. Higher currents up to 50 amps are only allowed for 5 seconds. The SMD power transistors can be replaced by TO220 types. This type of transistors can be cooled with additional heat sinks.

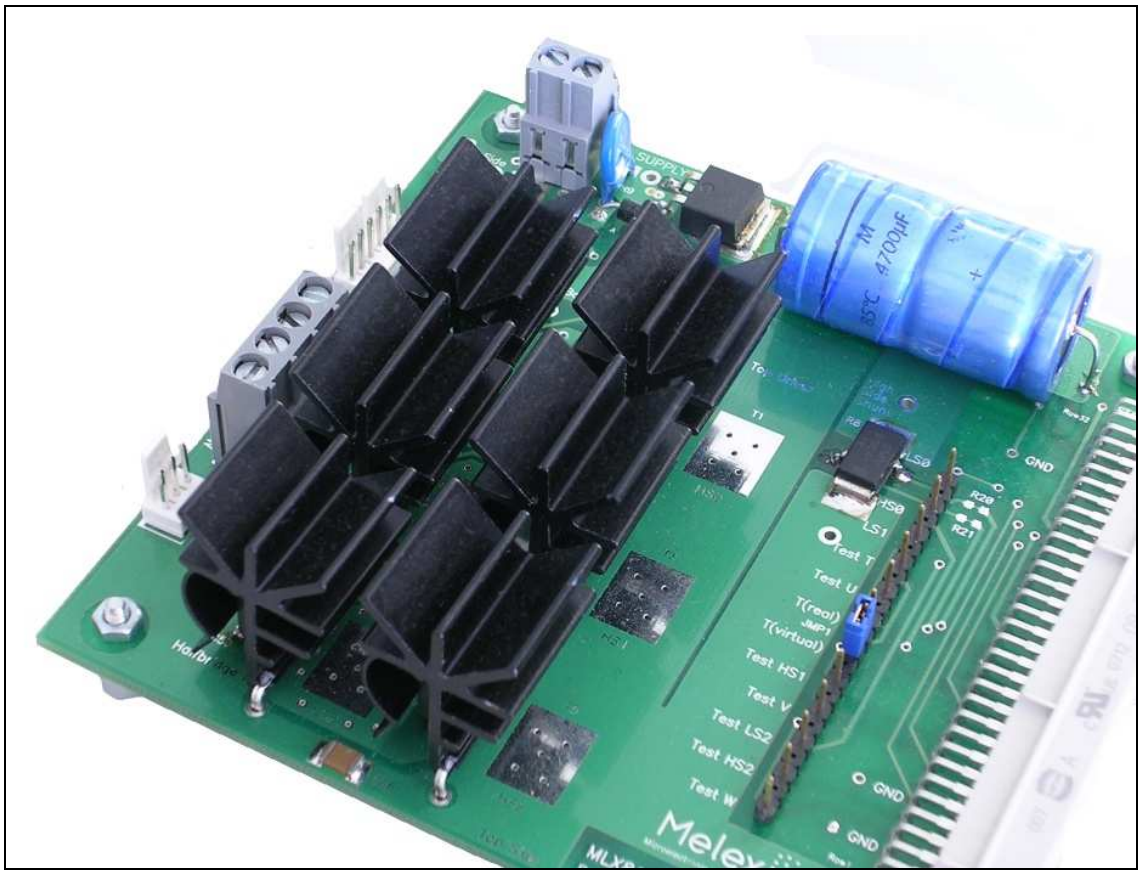


Fig. 6 Power board with TO220 transistors and heat sinks

The order number of the heat sinks can be found inside chapter 9.8 .

5.4.2. High and Low side current sensor

The high side shunt resistor is assembled with 1 mΩ. The low side shunt resistor is assembled with 5 mΩ. The maximum current can be calculated:

$$P_{tot} = 3W$$
$$P_{tot} = I^2 R$$
$$I_{\max} = \sqrt{\frac{P_{tot}}{R}}$$
$$I_{HS\max} = \sqrt{\frac{3W}{0.001R}}$$
$$I_{HS\max} \approx 54A$$
$$I_{LS\max} = \sqrt{\frac{3W}{0.005R}}$$
$$I_{LS\max} \approx 24A$$

If the shunt resistor is not used, it is possible to replace it with a copper wire.

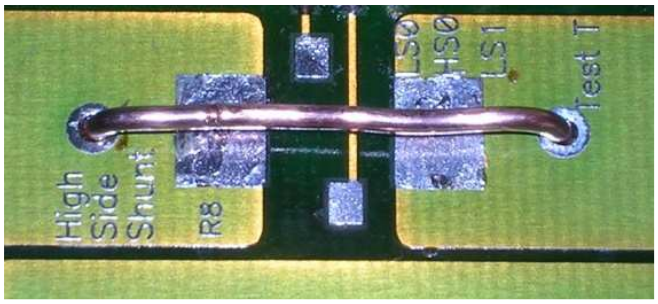


Fig. 7 Copper wire connected instead of shunt

5.4.3. Reverse polarity protection transistor T7

The transistor T7 can be substituted against a type with a smaller R_{DSon}. The bigger gate capacity doesn't matter because the transistor will be switched on only once. The temperature increasing can be calculated for PCB mounted transistors.

$$P_{tot} = I^2 R$$
$$P_{tot} = (20A)^2 * 0.008\Omega$$
$$P_{tot} = 3.2W$$
$$\Delta T = K_{th} * P_{tot} = 40\frac{K}{W} * 3.2W$$
$$\Delta T = 128K$$

5.4.4. Power board – Jumper settings

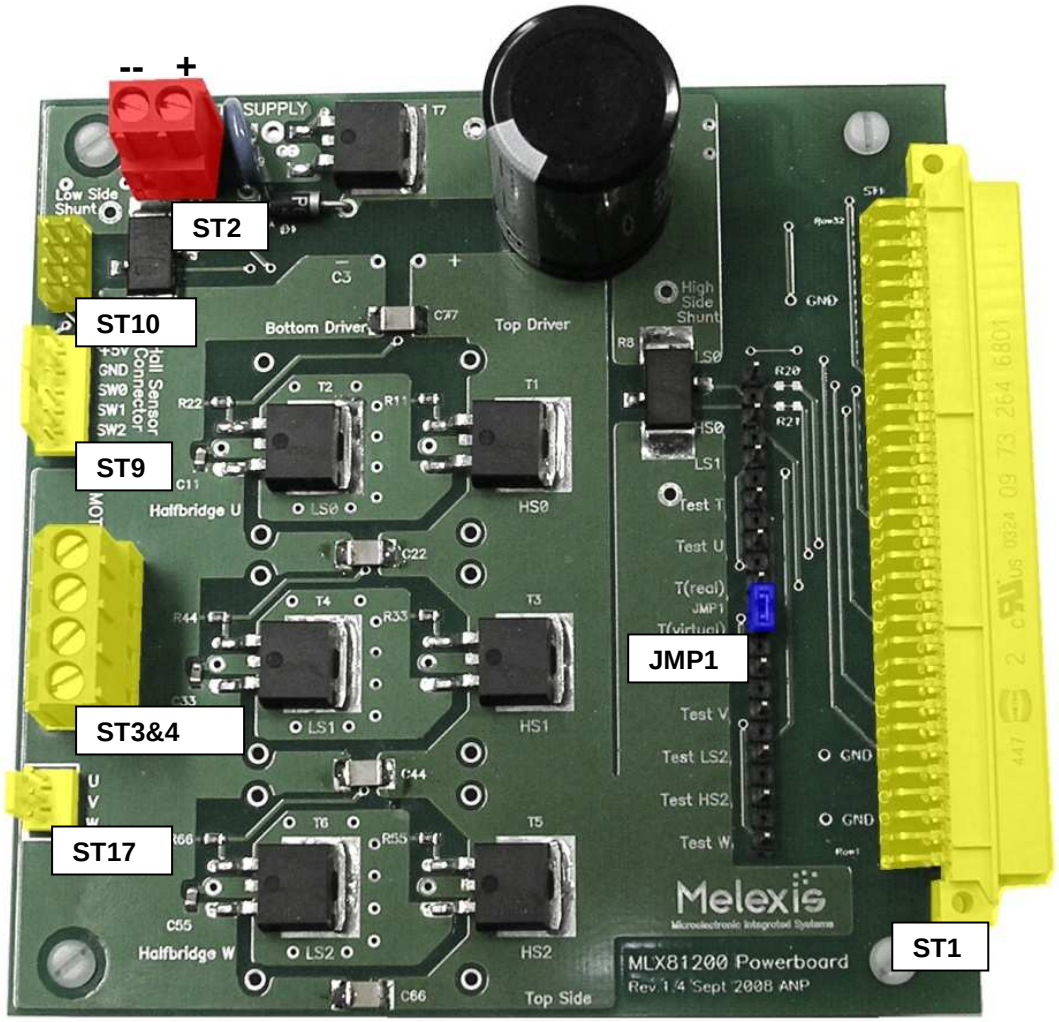


Fig. 8 Location of Connectors on Power board

JMP1 1-2: real star point connected to ST3&4
2-3: virtual star point used
R17, R18 and R19 are used to create the virtual star point

ST1: Evaluation board connector

ST2: Power terminal

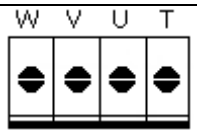
ST3&4: BLDC motor terminal

ST9: Hall sensor connector

ST10: SW0-3 input connector

ST17: BLDC demo kit motor connector

5.4.5. Power board – motor connectors ST3&4 and ST17

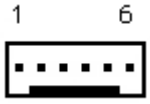
Connector ST3&4 on Power board		
Pin	**Name**	**Description**
U	Phase U	
V	Phase V	
W	Phase W	
T	Real star point	

Tbl. 10 Connector ST3&4 on Power board

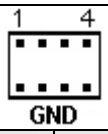
Connector ST17 on Power board		
Pin	**Name**	**Description**
1	Phase U	
2	Phase V	
3	Phase W	

Tbl. 11 Connector ST17 on Power board

5.4.6. Power board – hall sensor connectors ST9 and ST10

Connector ST9 on Power board		
Pin	**Name**	**Description**
1	V5EXT	IC Pin 2
2	GND	
3	SW0	IC Pin 22 – hall sensor input
4	SW1	IC Pin 21 – hall sensor input
5	SW2	IC Pin 20 – hall sensor input
6	Not used	

Tbl. 12 Connector ST9 on Power board

Connector ST10 on Power board		
Pin	**Name**	**Description**
1	SW0	IC Pin 22
2	SW1	IC Pin 21
3	SW2	IC Pin 20
4	SW3	IC Pin 19

Tbl. 13 Connector ST10 on Power board

6. Quick start up

6.1. Using the C flow

Assembling/Linking:

- change to the .\libsrc directory
- open a command prompt and type:
  '*gmake clean*'
  '*gmake install*'
- change to the directory where the source code is situated, e.g. .\projects\81200\BLDC\
- open a command prompt and type:
  '*gmake clean*'
  '*gmake all*'

-> executable *.elf file is created in the same directory
-> executable *.hex file is created in the same directory

Simulation:

- Start the MLX Interactive Debugger '…\Melexis\MDC\MLXDBGW.exe'
- Select target 'Mlx81200' and 'Simulator' from 'Tools' -> 'Target & Engine' menus as shown in Fig. 9
- Load the appropriate *.elf file – 'File' -> 'Open'
- Run the *.elf files by pressing the 'Go' or 'Step' icon

Emulation:

- Connect the Mini EMlx Emulator to the PC via an USB cable
- Insert a MLX81200 chip to the appropriate socket on the Evaluation board
- Power up the Evaluation board and connect the Mini EMlx Emulator to the EVB
- Start the MLX Interactive Debugger '…\Melexis\MDC\MLXDBGW.exe'
- Select target 'Mlx81200' and 'Mini EMlx' from '*Tools*' '*Target&Engine'* menus as shown in *Fig. 9*
- Load the appropriate *.elf file – 'File' -> 'Open'
- Run the *.elf files by pressing the '*Go'* or '*Step'* icon

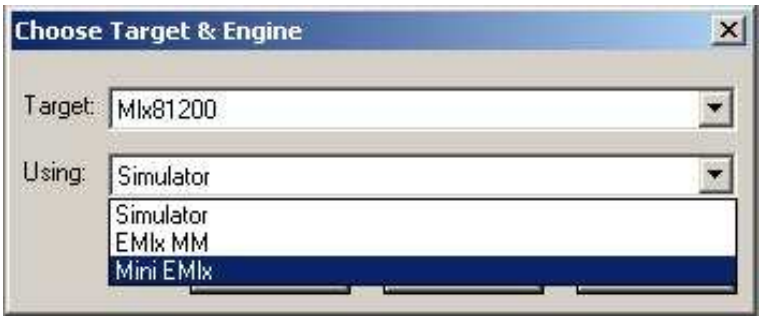


Fig. 9 Interactive Debugger: Selecting Simulator or Emulator mode

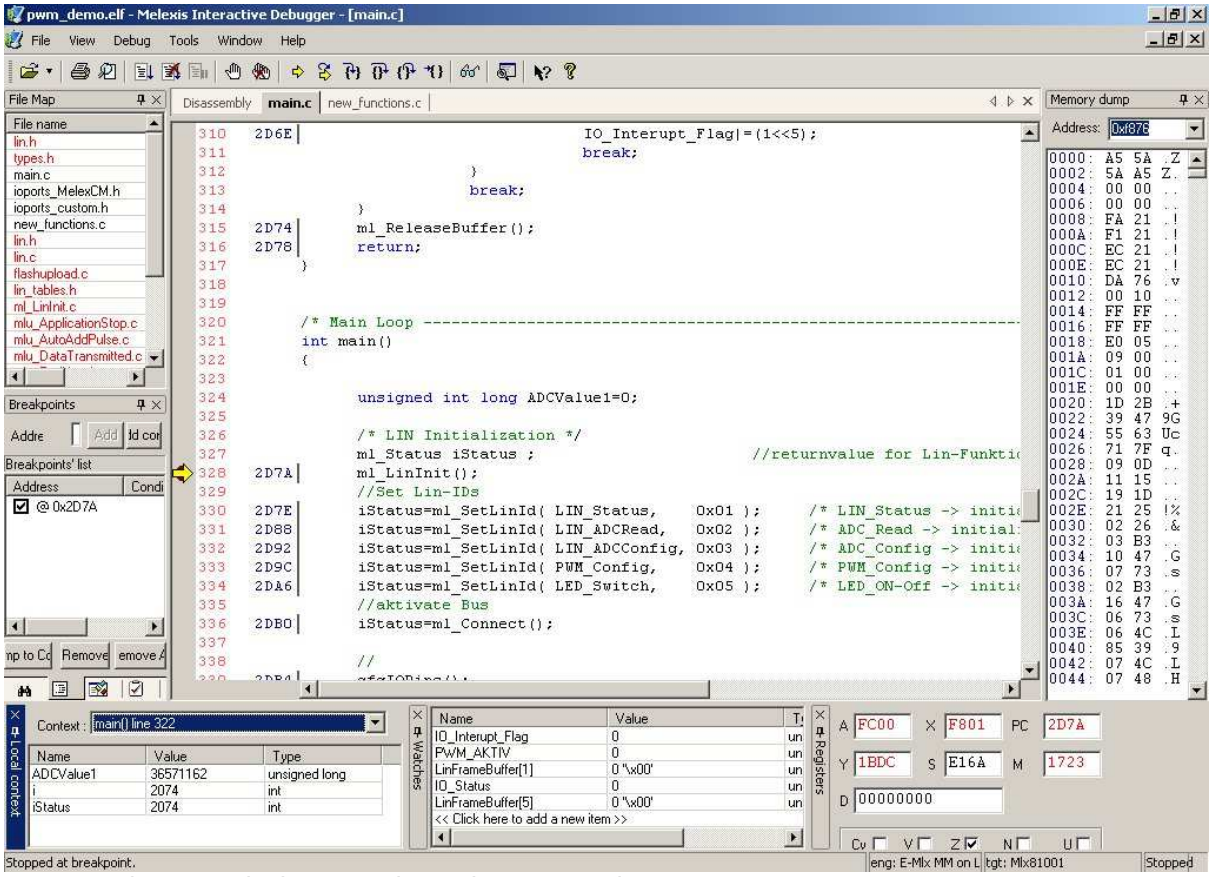


Fig. 10 Melexis Interactive Debugger session

6.2. Programming a HEX file to the FLASH

- Connect the Mini E-MLX Emulator to the PC via an USB cable
- Insert a MLX81200 chip into the socket and power up the EVB
- Connect the Mini E-MLX Emulator (programmer) to the EVB
- Start the programmer software '...\Programmer\EMlxMMProg.exe'
- Select File\Open… to open the Mlx81200 mmf file installed by Mlx81200Conf_x_x_x.msi

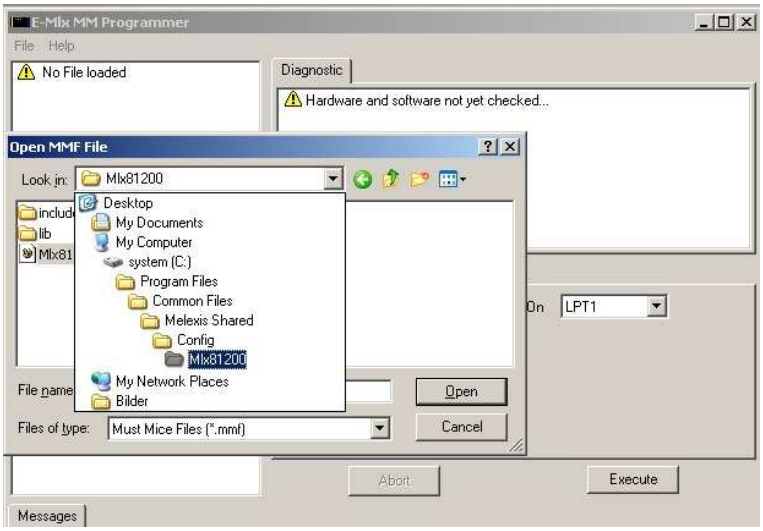


Fig. 11 E-Mlx MM Programmer: Path to the Mlx81200.mmf file

- Select *'Tools\Options\Programming'* tick checkbox *'Keep Supply between patterns'*

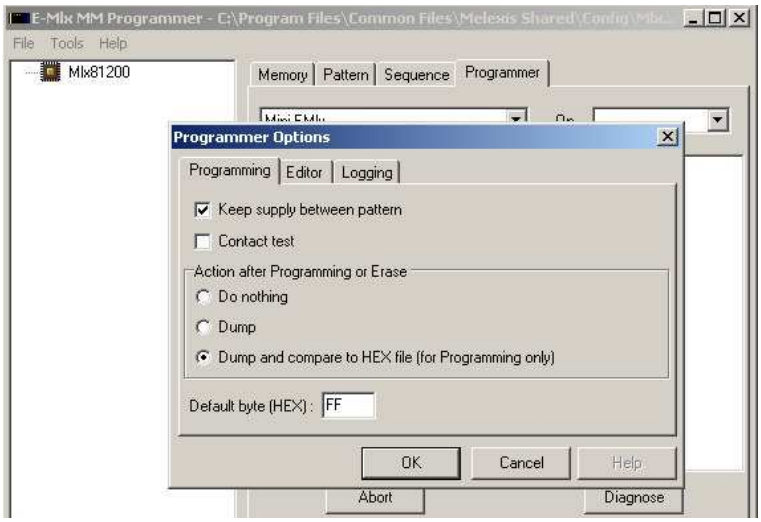


Fig. 12 E-Mlx MM Programmer: Option "keep supply between pattern"

- Select tab ‘Programmer’, choose from pull-down menu ‘Mini E-mlx’ on ‘Autodetect’

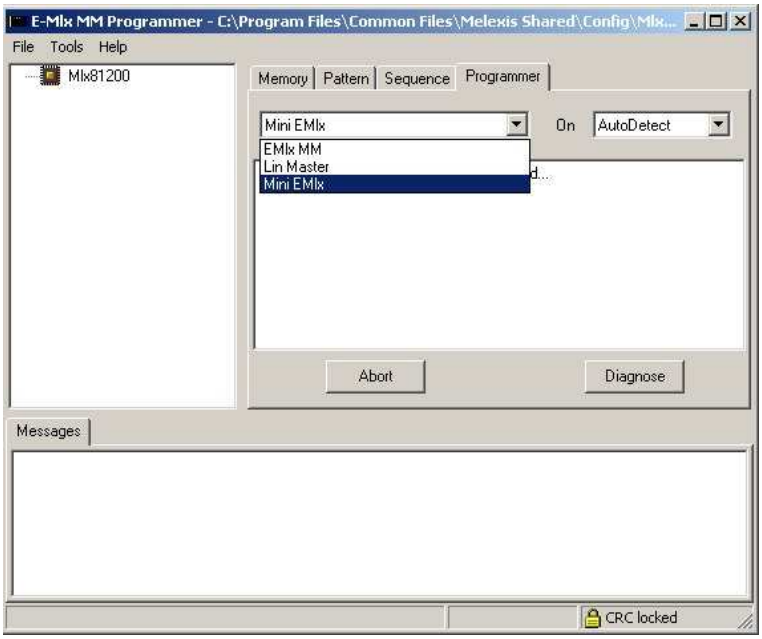


Fig. 13 E-Mlx MM Programmer: Chose the Mini EMlx

- Push the ‘*Diagnose’* button
  -> This message should appear : `>>> Info : Hardware successfully checked!`

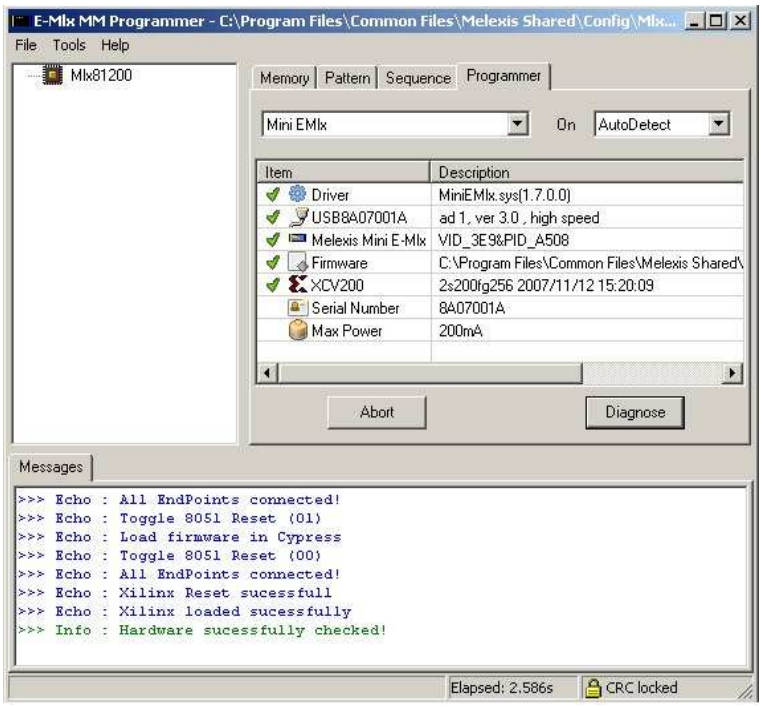


Fig. 14 E-Mlx MM Programmer: checking connected emulator

- Select tab ‘Pattern’ and go to pattern ‘POWER_ON’
- Push the ‘Execute’ button -> in addition to D3 “Power on” D20 “E-MLX on” should light up

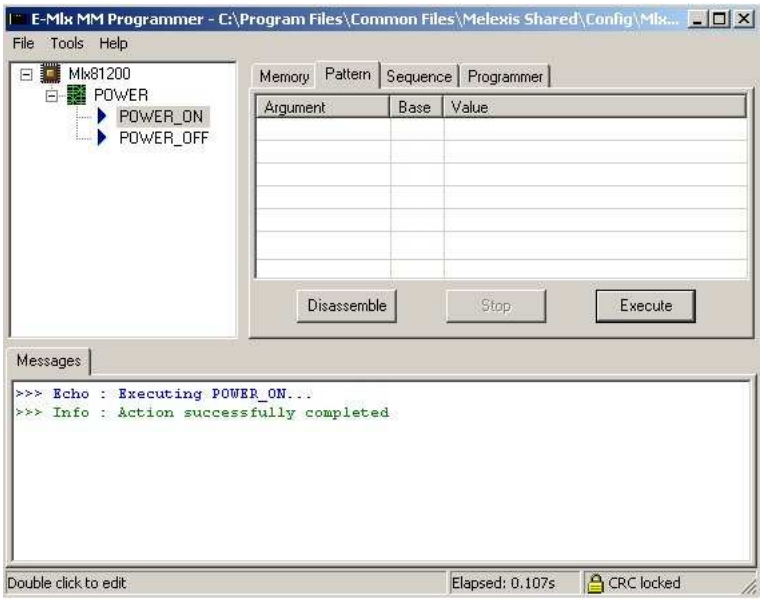


Fig. 15 E-Mlx MM Programmer: enable programmer interface

- Select ‘Flash’ from the tab ‘Memories’ sections
- Select the ‘Program using HEX file’ radio button
- Load a *.hex file and push the ‘Execute’ button
- Verify the programmed file by pushing the ‘Compare Dump to HEX file’ radio button and press ‘Execute’ again

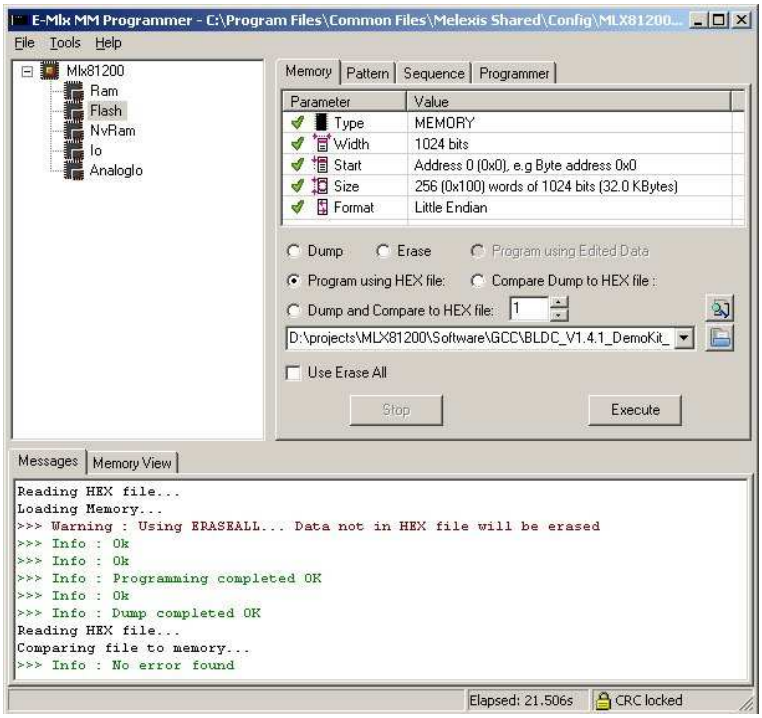


Fig. 16 E-Mlx MM Programmer: programming the flash

- Before disconnecting the Mini EMLX select tab *'Pattern' -> 'POWER_OFF'*
- Push the '*Execute*' button
- D20 "E-MLX on" should now be turned of

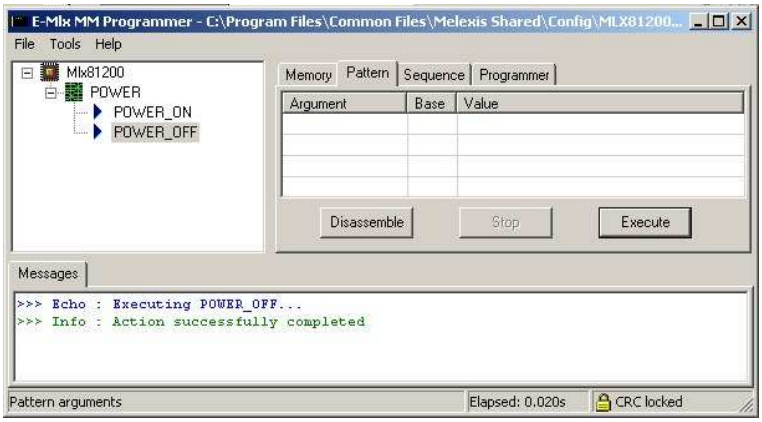


Fig. 17 E-Mlx MM Programmer: disable programmer interface

- The evaluation board can now be switched off
- Mini E-MLX Emulator can now be safely removed

For generating *.hex code please refer to 6.1

7. MLX81200 Software configuration tool (SCT)

7.1. General

The Software configuration tool can be used to configure, compile and flash the BLDC-Demokit firmware version.

The Software configuration tool is not suitable to write a complete application. If the file structure is changed or the source files are modified, the tool will not work correctly.

Note: The save function will overwrite the files motorctrl.h, sysdef.h, ramppar.c, .\config\chip.mk and the application Makefile with a defined content. That means any modification will be lost.

The Software configuration tool does NOT include any source code for the MLX81200. The required firmware source files package BLDC_Vx.x_DemoKit can be found on the Melexis Softdist server. The firmware needs to be unpacked to a local folder on the PC. Please make sure that directory path does not contain any spaces.

The SCT is divided into three major parts.

Part 1: locate the demokit firmware sources
Part 2: configure the motor control parameters
Part 3: compile the firmware and flash the firmware into the IC via the pin LIN

A working loader inside the IC is necessary for a successful flash process. Melexis will deliver preprogrammed devices, which include an initial working loader code.
If the preprogrammed loader code was destroyed, the first step must be always the programming of the loader software via test interface. This step can only be executed via the programmer software or debugger using E-mlx Emulator.

For programming a HEX file to the flash via the test interface please refer to chapter 6.2

For further information please see:

- the description for the MLX81200 BLDC Demo Kit firmware
- Application Note how to program on module
- Application Note how to flash over pin LIN

7.2. Locate the demokit firmware sources and load the configuration

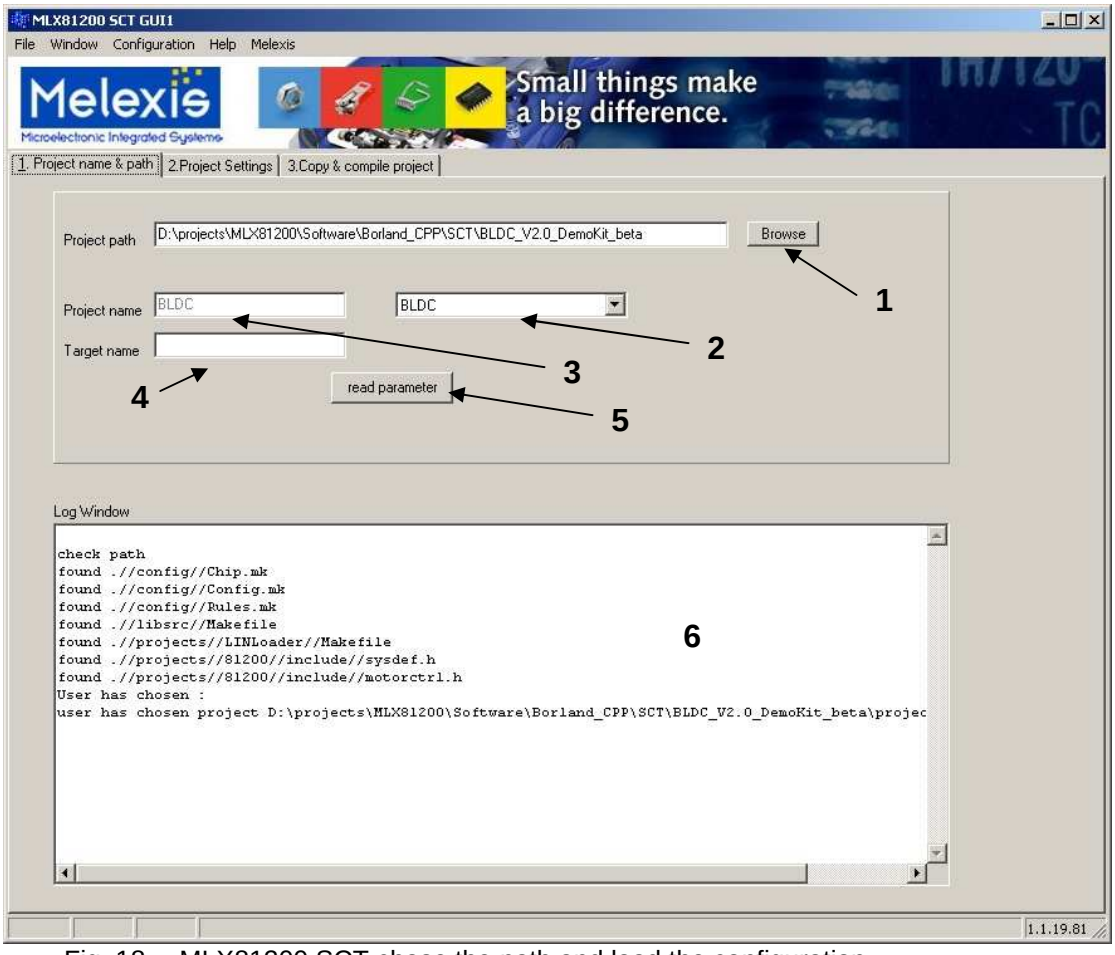


Fig. 18 MLX81200 SCT chose the path and load the configuration

1	Button “Browse”	the path to BLDC firmware source
2	Project selector	the current project name must be selected
3	Project name	shows the current project name
4	Edit field “Target name”	the name of the firmware
5	Button “read parameter”	the configuration will be read from the chosen directory
6	Log Window	each operation is shown in this window

7.3. Configure the demokit firmware

7.3.1. Test pulse configuration

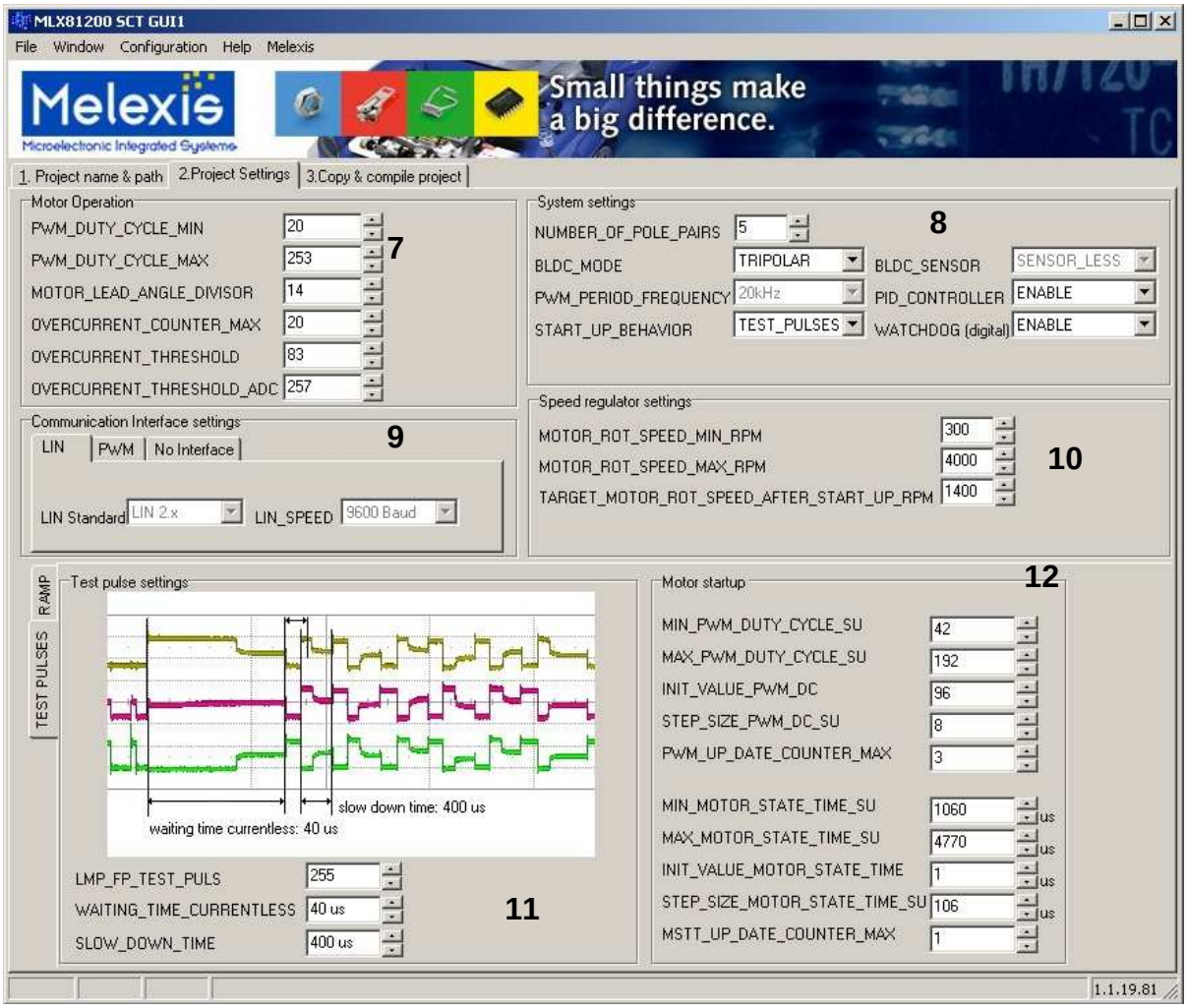
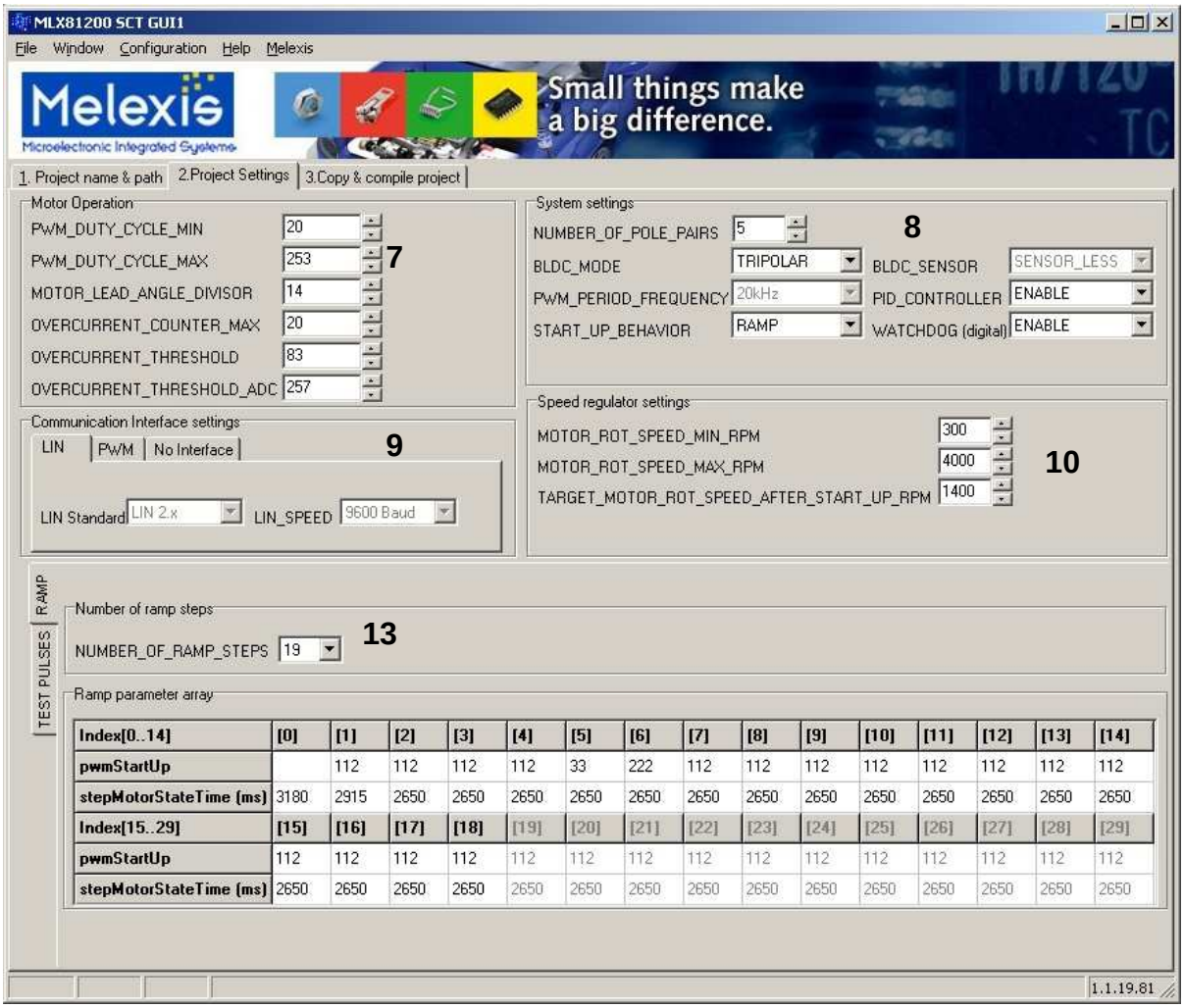


Fig. 19 MLX81200 SCT – modify the parameter of the BLDC-demo kit

7 Parameter set “motor operation”
8 Parameter set “System setting”
9 Parameter set “Communication Interface settings”
10 Parameter set “Speed regulator settings”
11 Parameter set “Test pulse settings”
12 Parameter set “Motor start up test pulses”

7.3.2. Ramp configuration



Index[0..14]	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]
pwmStartUp		112	112	112	112	33	222	112	112	112	112	112	112	112	112
stepMotorStateTime (ms)	3180	2915	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650
Index[15..29]	[15]	[16]	[17]	[18]	[19]	[20]	[21]	[22]	[23]	[24]	[25]	[26]	[27]	[28]	[29]
pwmStartUp	112	112	112	112	112	112	112	112	112	112	112	112	112	112	112
stepMotorStateTime (ms)	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650	2650

Fig. 20 MLX81200 SCT – modify the parameter of the BLDC-demo kit

7 Parameter set “motor operation”
8 Parameter set “System setting”
9 Parameter set “Communication Interface settings”
10 Parameter set “Speed regulator settings”
13 Parameter set “Motor start up Ramp”

7.4. Save, compile and program the firmware

7.4.1. Configure the hardware in use

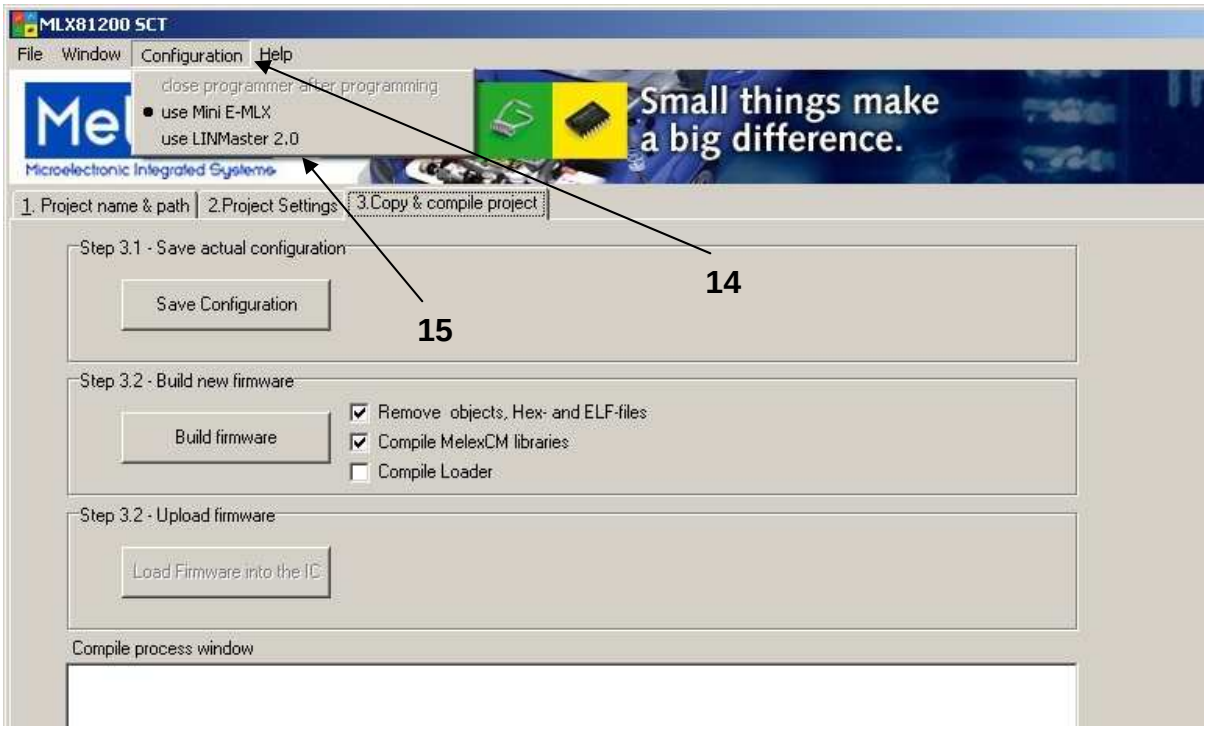


Fig. 21 MX81200 SCT – configure the hardware in use

14	Main menu item “Configuration”	the SCT can be configured in this menu
15	Menu item “use Mini E-Mlx” and “use LINMas..”	the USB LINMaster or the Mini E-MLX can be chosen

7.4.2. Build process

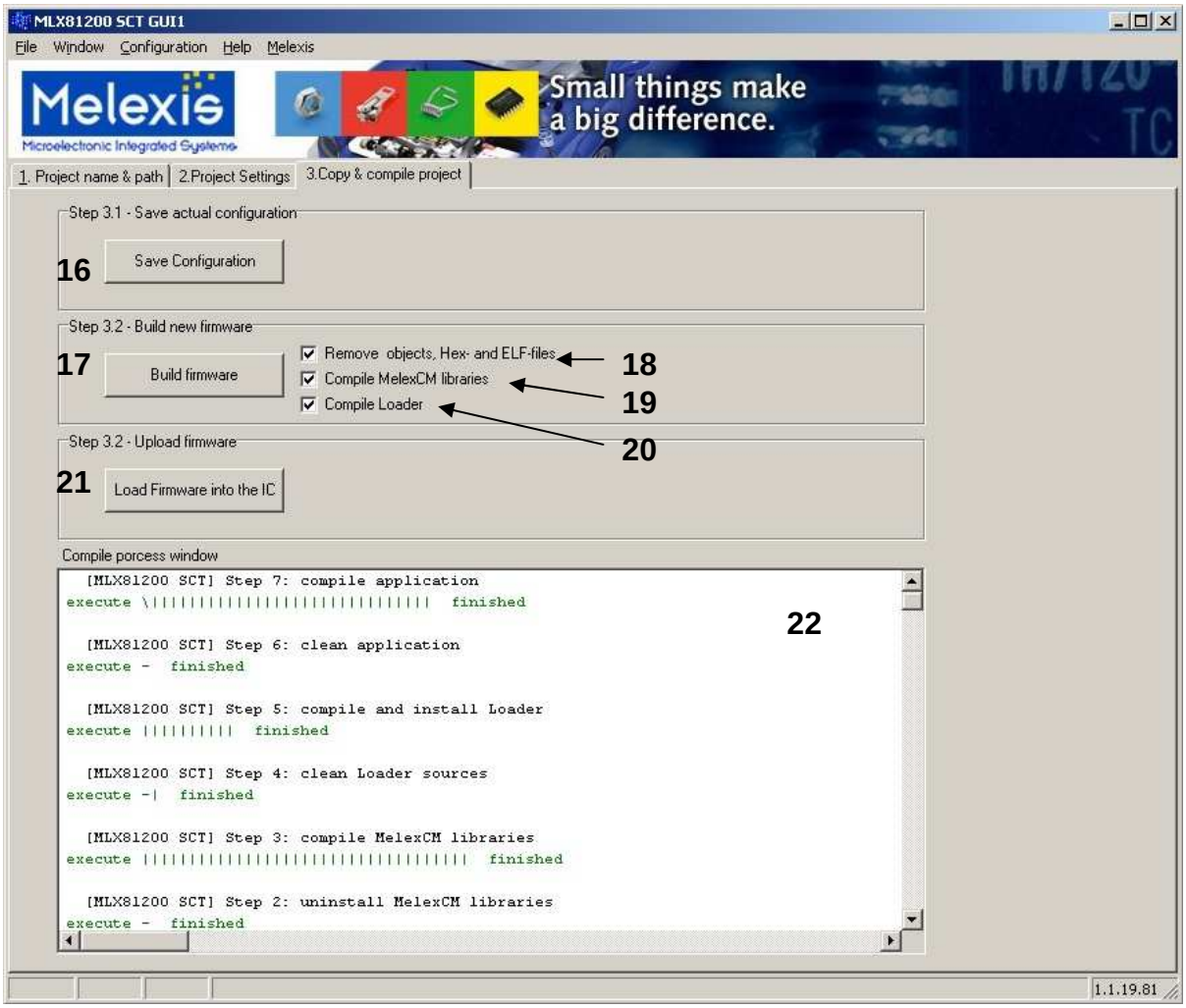


Fig. 22 MX81200 SCT – screenshot running compile process of the BLDC-demo kit firmware

16	Button “Save configuration”	the configuration will be saved into the files
17	Button “Build firmware”	start the compile process
18	Checkbox “Remove objects, Hex- and Elf-files”	remove the old object and firmware files
19	Checkbox “Compile MelexCM libraries”	compile the complete MelexCM libraries
20	Checkbox “Compile Loader”	build the loader firmware from current configuration
21	Button “Load Firmware into the IC”	start the programmer tool and load the actual firmware into the MLX81200 via pin LIN
22	Log window “Compile process window”	The compile log messages are shown inside

7.4.3. Upload the firmware

- Disconnect all other devices from the LIN bus
- Connect the USB-LIN master version 2.0 or the Mini E-Mlx emulator to the pin LIN and to the PC
- Press the Button “Load Firmware into the IC” - The programmer tool is started and will be configured for the used loader protocol
- Do a POR of the MLX81200 when the message “`>>> Echo : Entering programming mode...`” is shown. Uploading the firmware is only possible immediately after POR and for a certain time (500ms). Afterwards uploading is not possible anymore.
- Wait until the E-Mlx MM Programmer tool has finished with the message “`Info : Programming completed OK`”
- Close the E-Mlx MM Programmer tool.

A working loader inside the IC is necessary for this process.

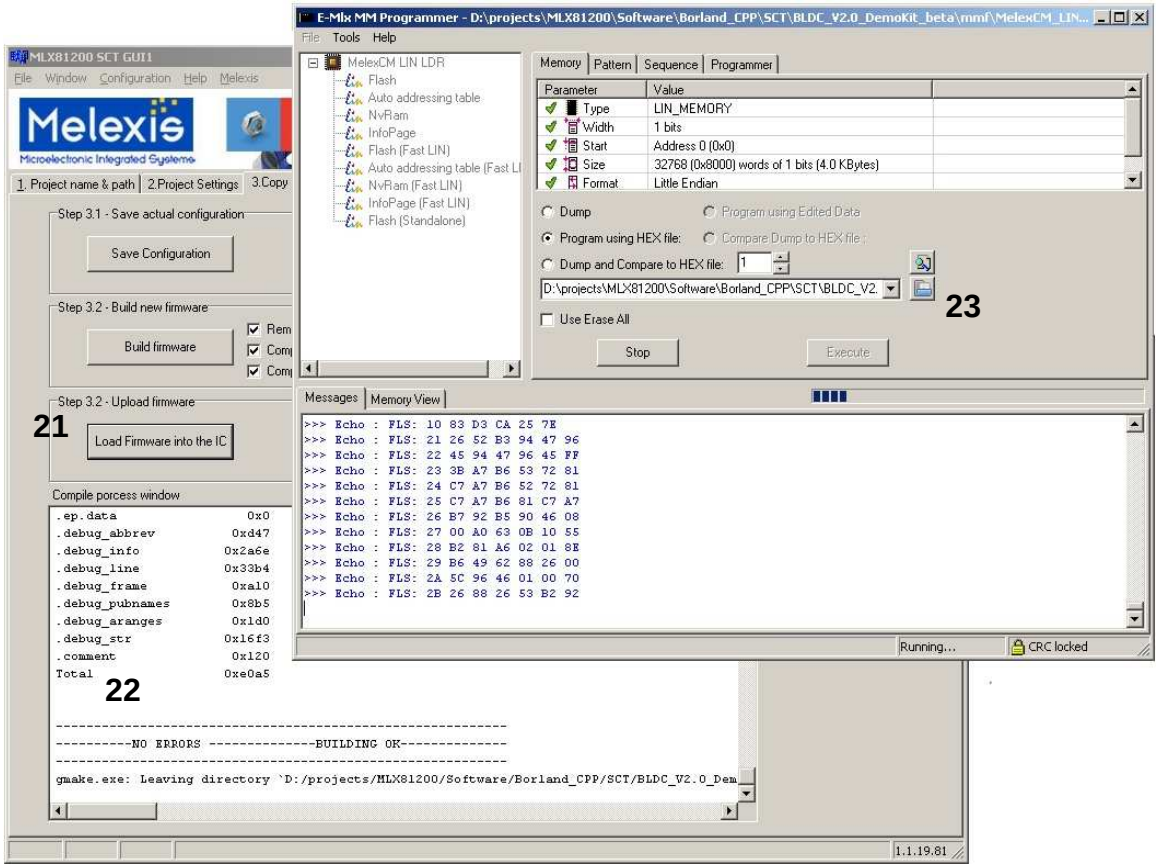


Fig. 23 MLX81200 SCT – uploading procedure with the programmer tool

21	Button “Load Firmware into the IC”	Start the programmer tool and load the actual firmware into the MLX81200 via pin LIN
22	Log window “Compile process window”	The compile log messages are shown inside
23	E-Mlx MM Programmer tool	The Melexis programmer tool

8. BLDC DemoKit PC program

8.1. General

The Motor control Software does NOT include any source code for the MLX81200. The required firmware file BLDC_V20_DemoKit_XY.hex MUST be loaded first into the MLX81200. The program MUST be running either by starting the program in the Melexis interactive debugger or let the CPU run free without the emulator out of power on reset.

The BLDC Demokit Firmware can be controlled with the PC program via the LIN bus. The hysteresis brake board can be controlled via the LIN bus too.

For programming a HEX file to the flash please refer to chapter 6.2

8.2. Command line parameter

It is possible to configure the minimum and maximum speed slider values of the graphical interface. The default limits are 500 rpm for the minimum and 5000 rpm for the maximum.
Following parameters are supported:

- -maxspeed:x set the maximum limit
- -minspeed:x set the minimum limit

Example: The command:

“MLX81200_Demokit_PC_program.exe –maxspeed:4000 –minspeed:2000”

starts the BLDC DemoKit PC program with a speed range from 2000 to 4000 rpm.

8.3. Main window

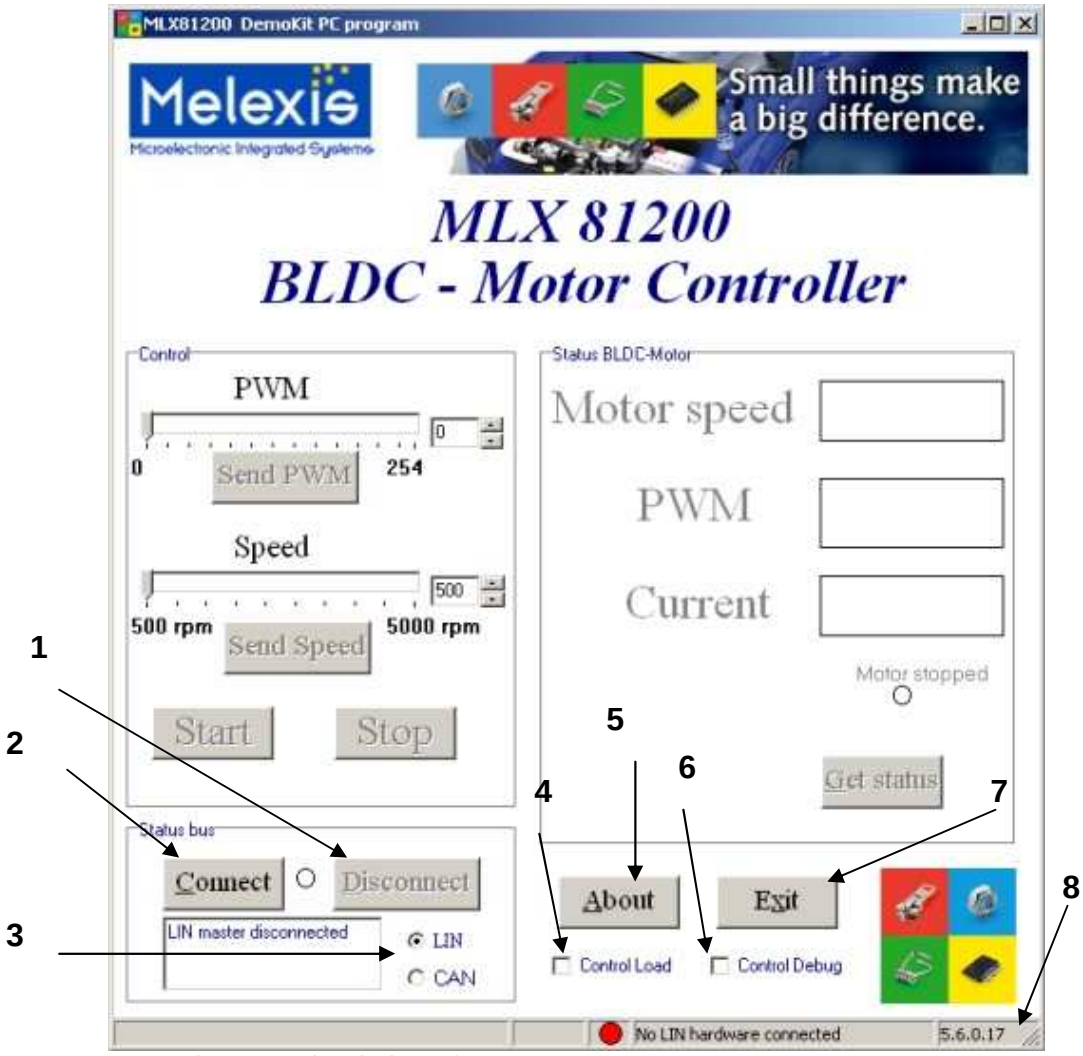


Fig. 24 Main window after start

1	Button “Disconnect”	disconnect the LIN or CAN master. The application does not close. The software can be reactivated by clicking the ‘connect’ button.
2	Button “Connect”	connect the LIN master or the CAN master
3	Interface radio buttons	The interface can be selected with this two radio buttons
4	Checkbox “Control Load“	open a window to control the hysteresis brake board via the LIN bus
5	Button “About”	
6	Checkbox “Control Debug”	open the debug window to control the wave form
7	Button “Exit”	close all windows and exit the application
8	Status bar	show the status of the LIN bus and the revision number of the firmware

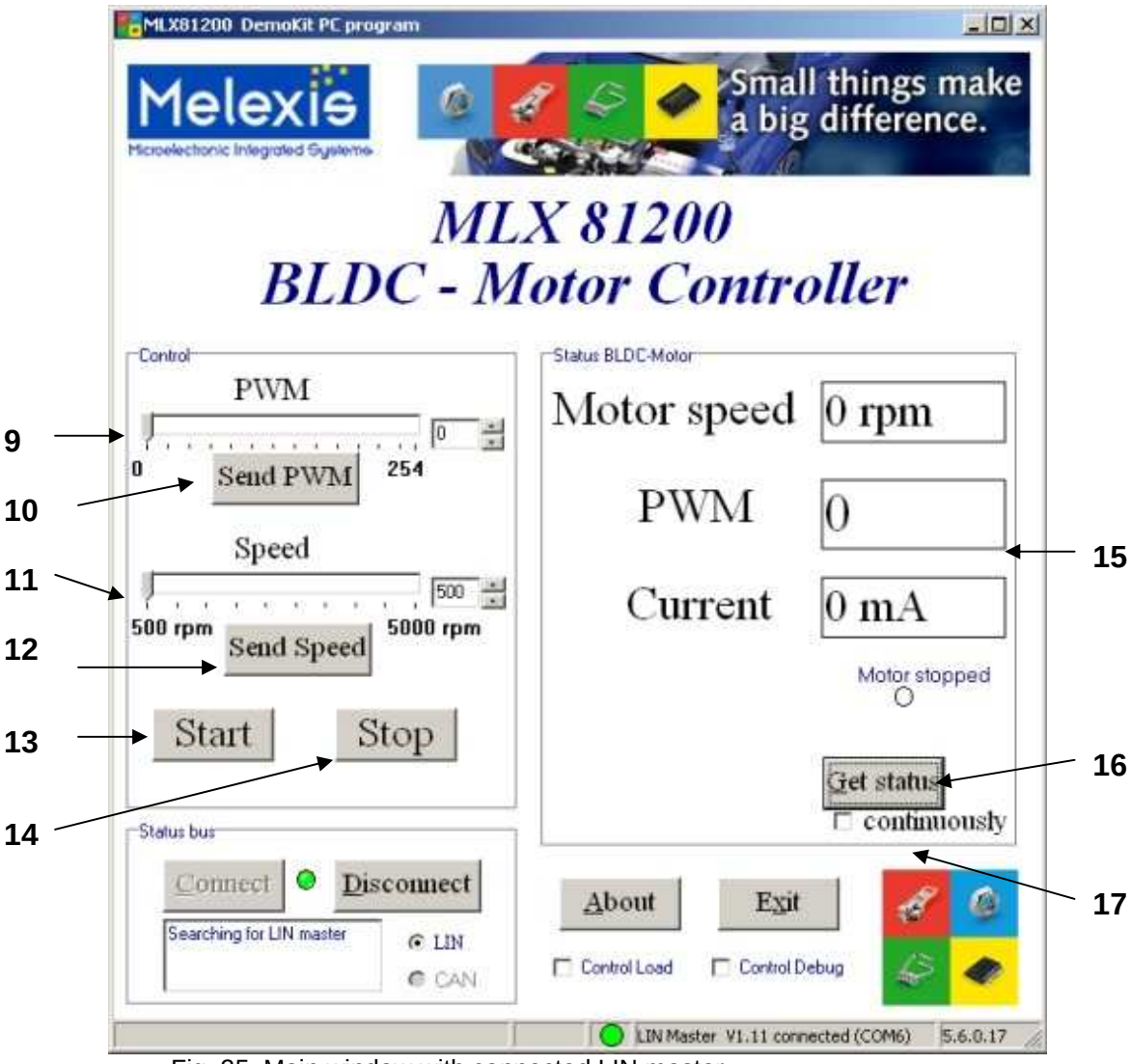


Fig. 25 Main window with connected LIN master

9	Control bar “PWM”	adjust the PWM value (send to MLX81200 with button 10)
10	Button ”Send PWM”	send the adjusted PWM value to the MLX81200
11	Control bar “Speed”	adjust motor speed (send to MLX81200 with button 12)
12	Button “Send Speed”	send the adjusted speed value to the MLX81200
13	Button “Start”	start the motor Note: The default speed target after start command is 1200U/min
14	Button “Stop”	stop the motor immediately
15	Status window	shows the current motor speed , PWM value , motor status and the current Note: If the CAN interface is used, the value of the current will not be transferred
16	Button “Get Status”	Click this button to get the actual status of the motor (shown in 15) Note: the received values of the PWM and speed will be transferred automatically to the control bars (9 and 11)
17	Checkbox “continuously”	get the motor status and send the target speed value every 150 ms.

8.4. Control Load window

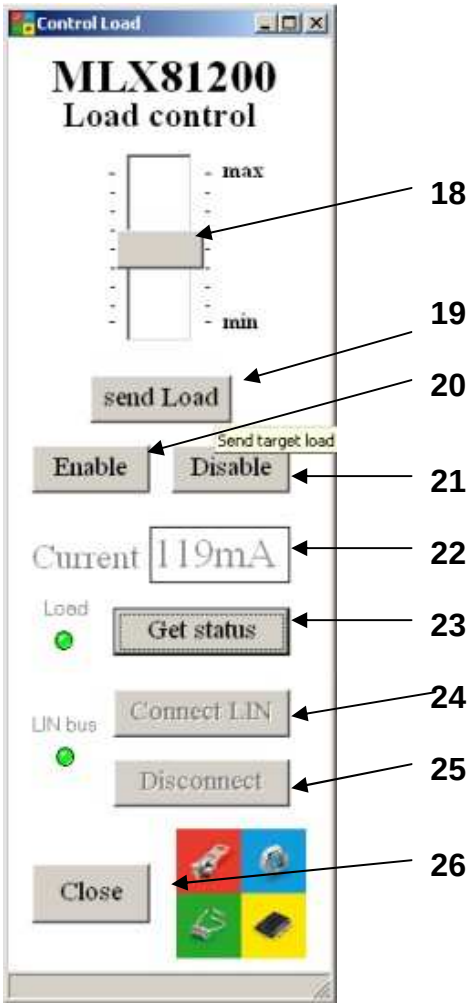


Fig. 26 Control Load window

18	Control bar “Load “	adjust the load value (send to brake board with button 19)
19	Button “send Load”	send the adjusted load value to the brake board
20	Button “Enable”	switch the load on
21	Button “Disable”	switch the load off
22	Status window “Current”	shows the actual current of the load
23	Button “Get status”	click this button to get the actual current of the load (shown in 22)
24	Button “Connect LIN”	connect the LIN master if the CAN interface is used in main window
25	Button “Disconnect”	disconnect the LIN master
26	Button “Close”	close the window “Control load” and return to the main window

8.5. Control Debug window

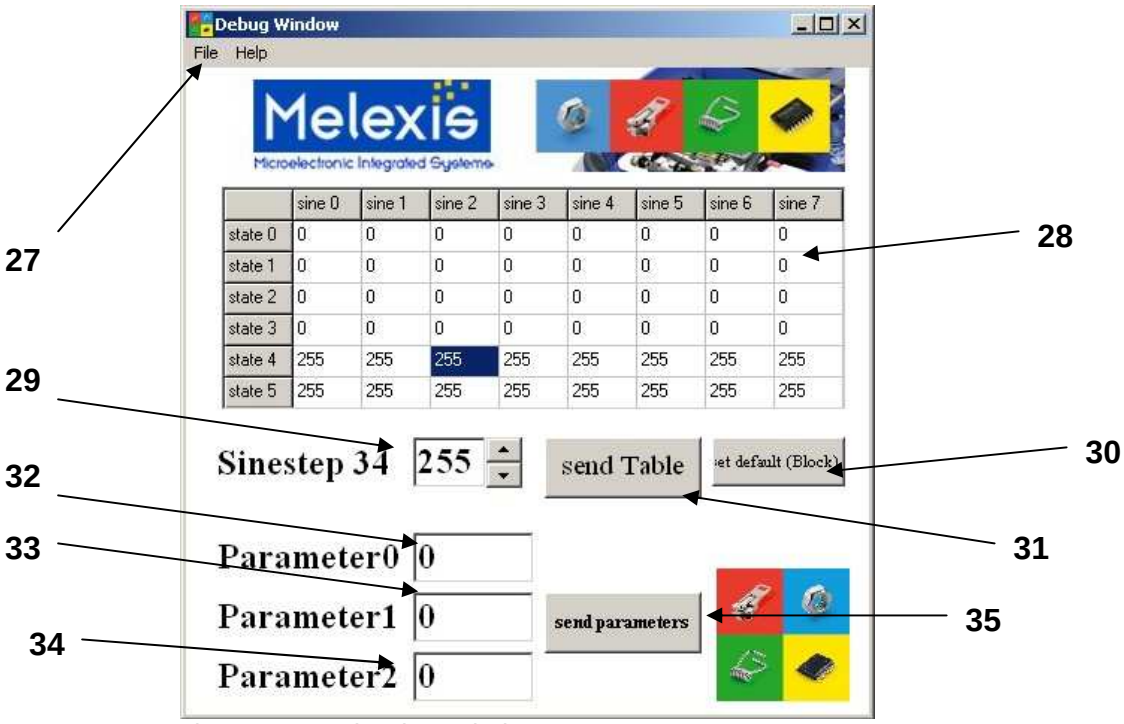


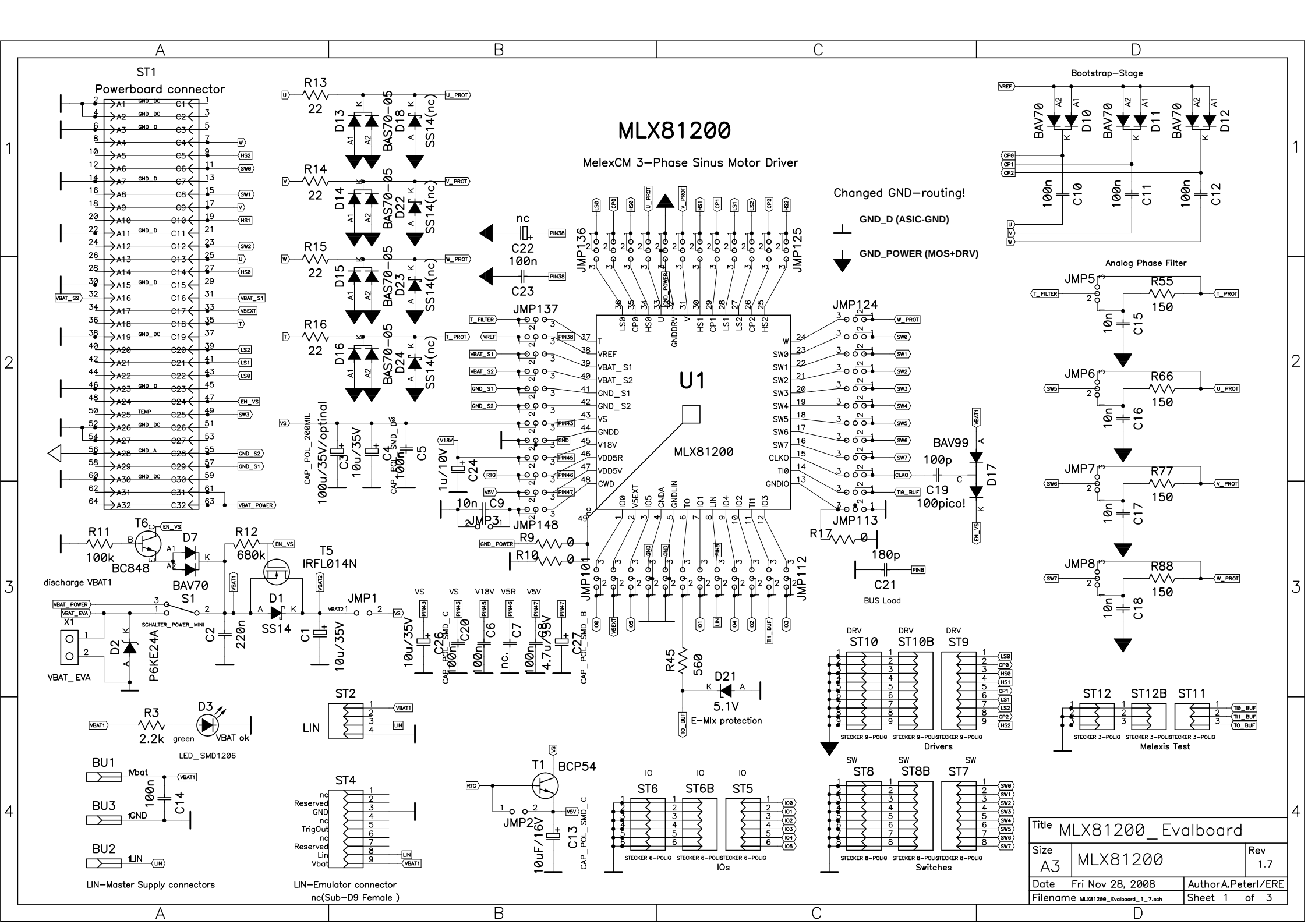
Fig. 27 Control Debug window

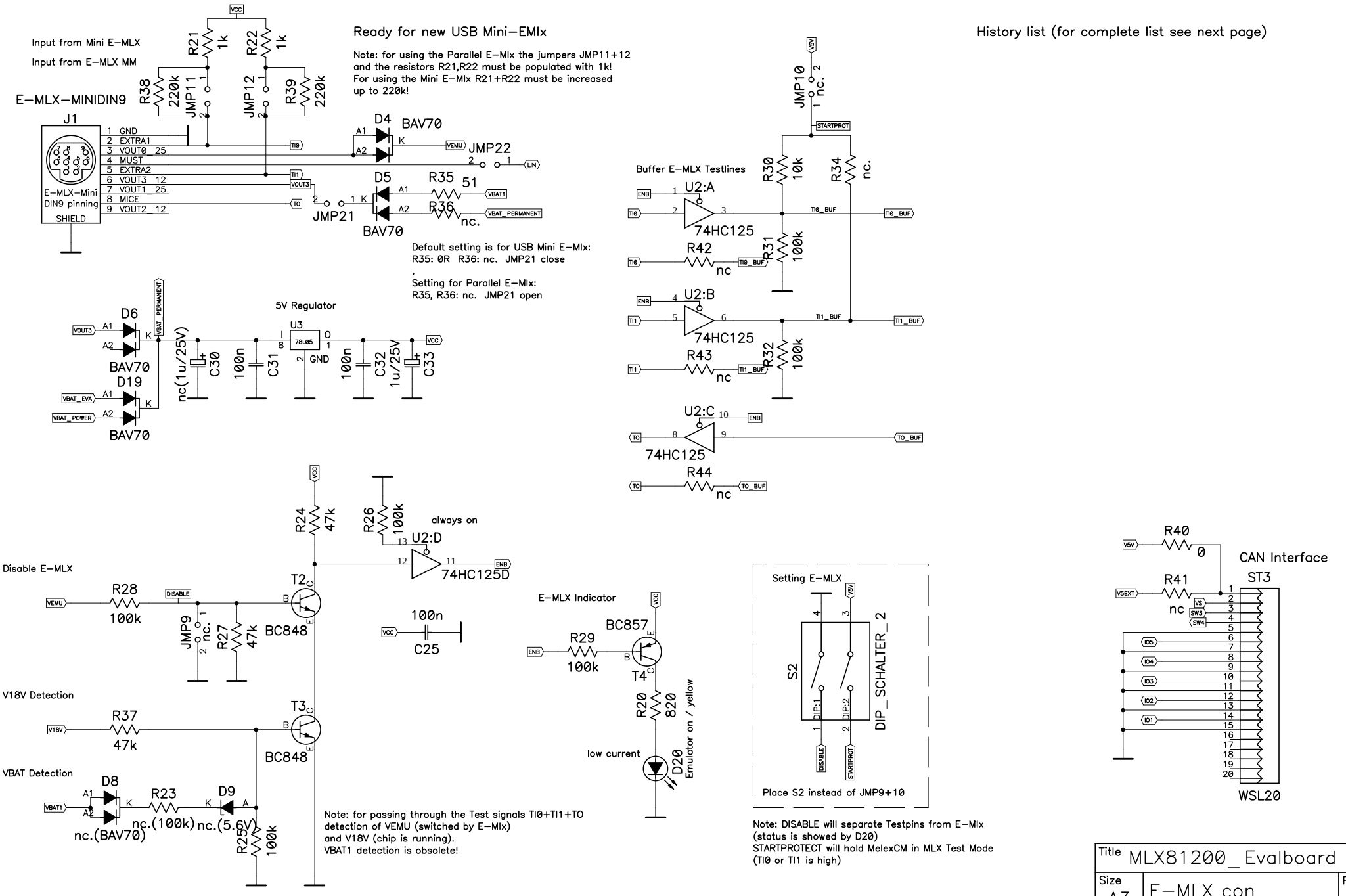
27	Pull down menu:	-"open table" load a saved sine table -"save table" save the sine table -"save table as…" save the sine table to a selected path -"close window" close the debug window
28	Sine table	each element can be chosen by mouse and modified with field 29
29	Edit field "Sinestep"	the value for each sine step can be modified direct or with up/down buttons
30	Button "set default(Block)"	set the sine table to a default block
31	Button "send Table"	send the table to the MLX81200 via LIN
32	Edit field "Parameter0"	not used
33	Edit field "Parameter1"	not used
34	Edit field "Parameter2"	not used
35	Button "send parameter"	not used

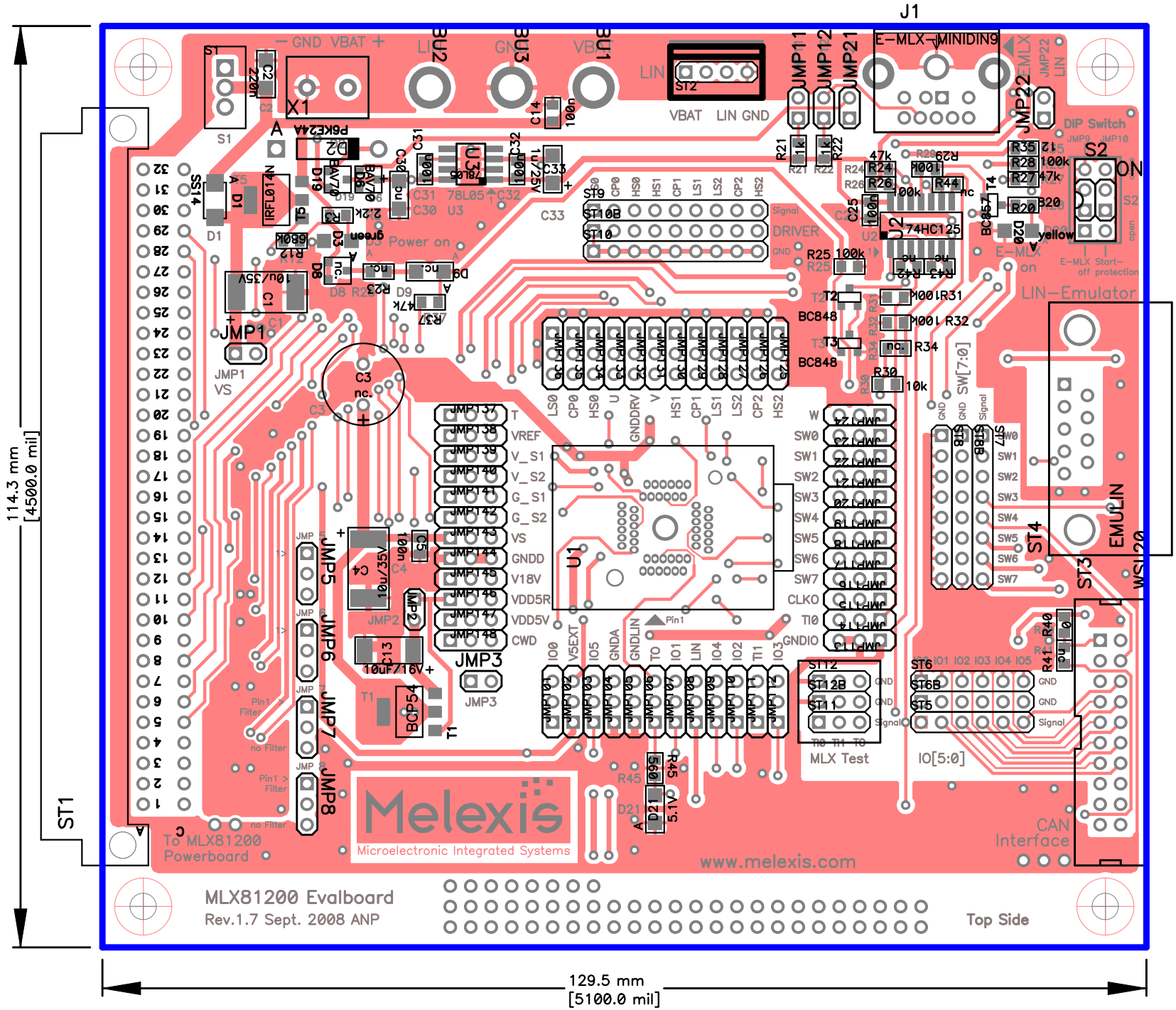
8.6. PID controller and PC program

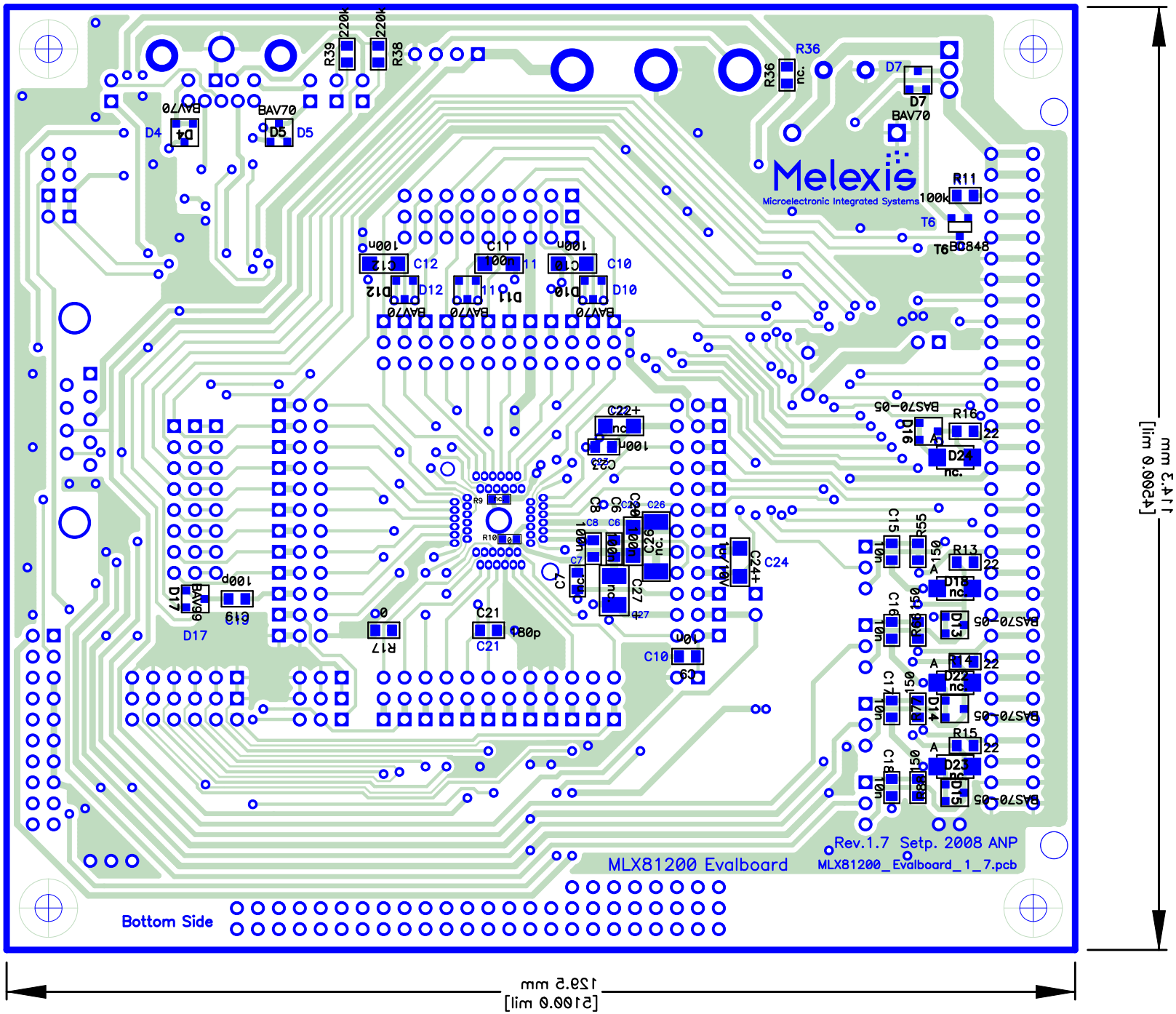
The firmware revision 1.6 for the MLX81200 BLDC DemoKit includes a PID controller.

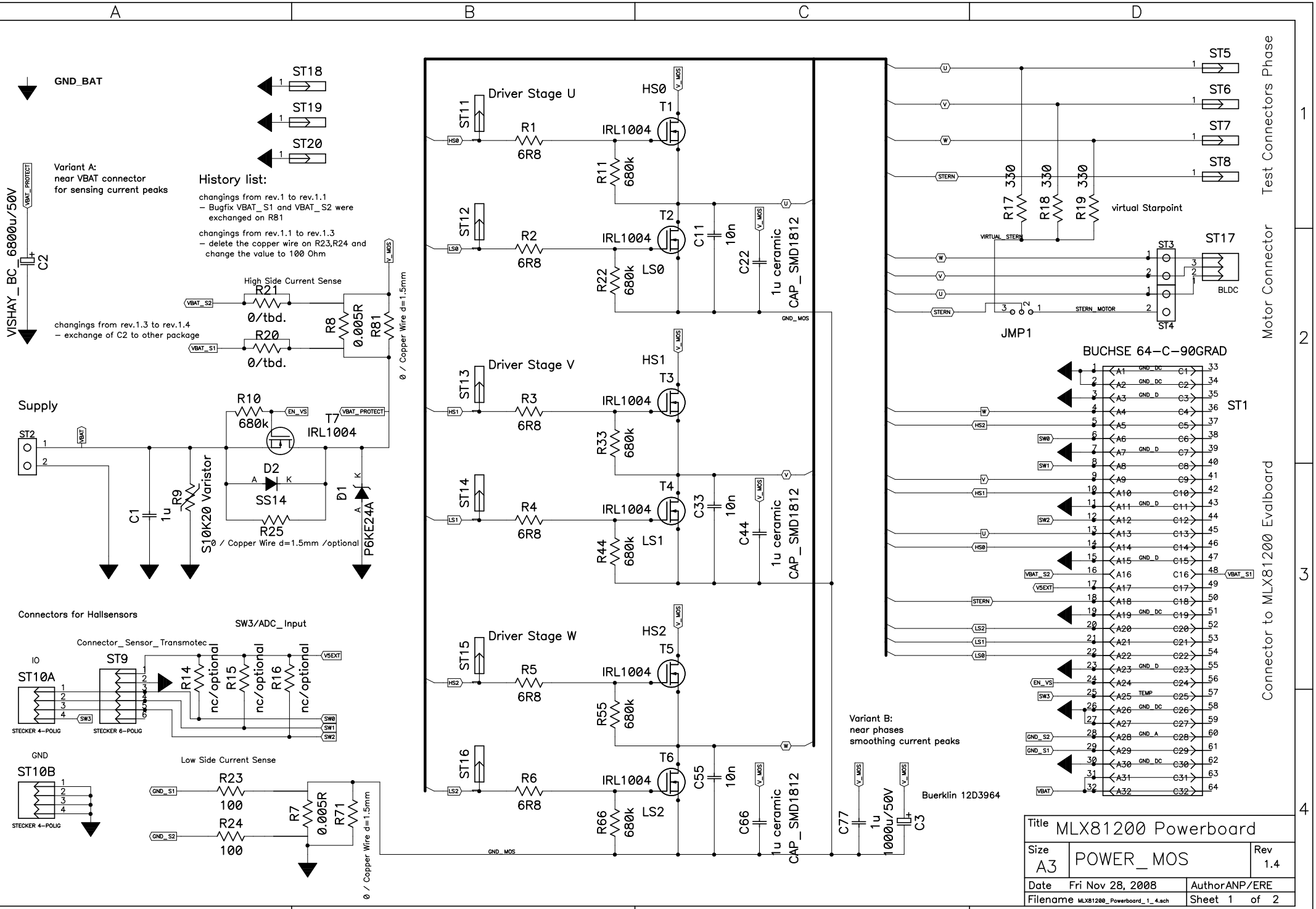
After the button "Start" is pressed , the start algorithm begins to rotate the motor. After a defined number of motor states (ECM must be active), the PID controller is activated and the predefined target rational speed is set up by the PID controller. Every time the button "Send PWM" is used to change the PWM value of the motor, the PID controller is disabled and only the adjusted PWM is used by the firmware. The PID controller can be reactivated by sending a new speed target to the MLX81200.

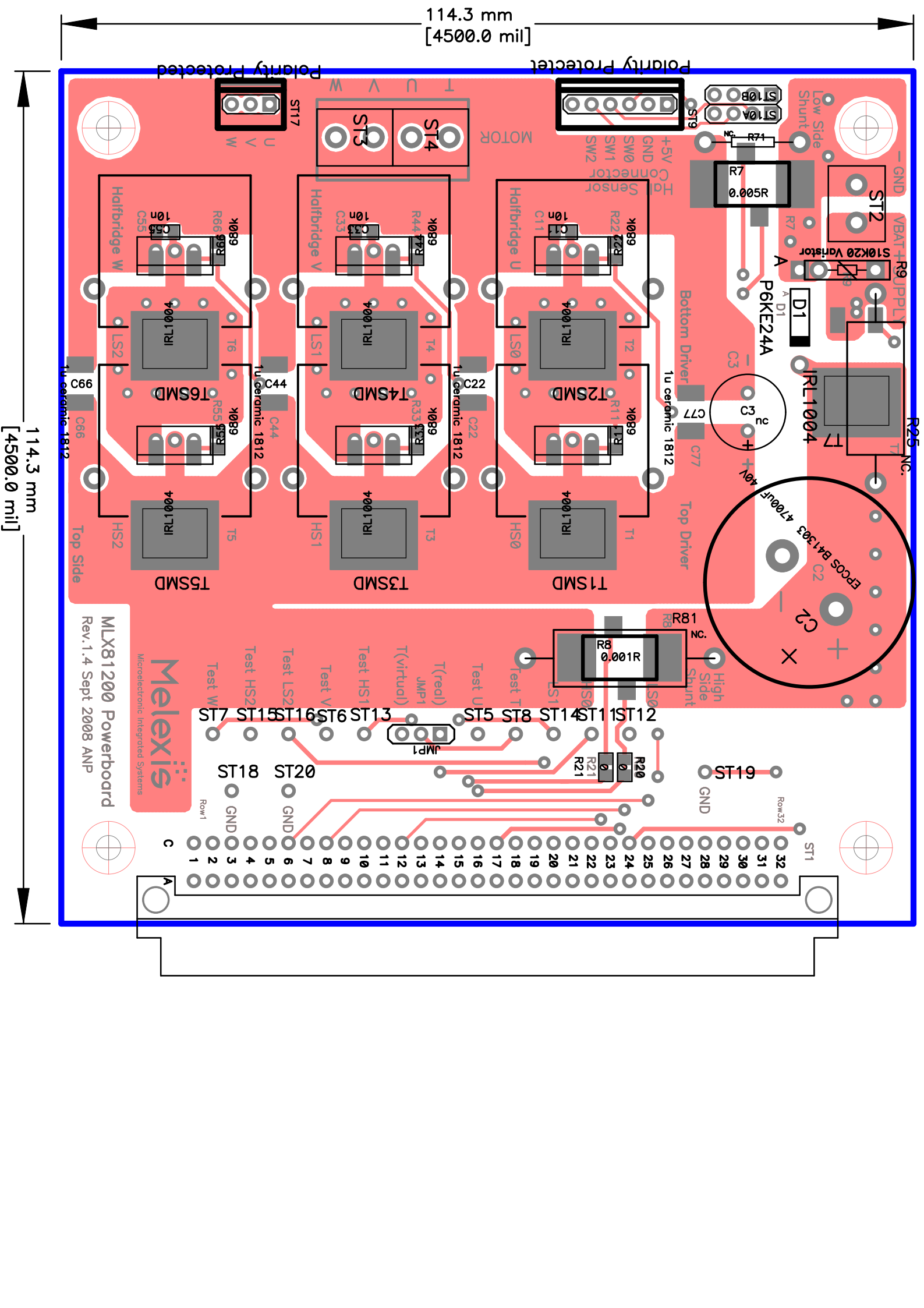

MLX81200
MelexCM 3-Phase Sinus Motor Driver
Changed GND-routing!
GND_D (ASIC-GND)
GND_POWER (MOS+DRV)
ST1
Powerboard connector
U1
MLX81200
Bootstrap-Stage
Analog Phase Filter
Drivers
Melexis Test
IOs
Switches
LIN-Master Supply connectors
LIN-Emulator connector
nc(Sub-D9 Female)
BUS Load
E-Mlx protection
discharge VBAT1
Title MLX81200_Evalboard
Size A3
MLX81200
Rev 1.7
Date Fri Nov 28, 2008
Author A.Peterl/ERE
Filename MLX81200_Evalboard_1_7.sch
Sheet 1 of 3

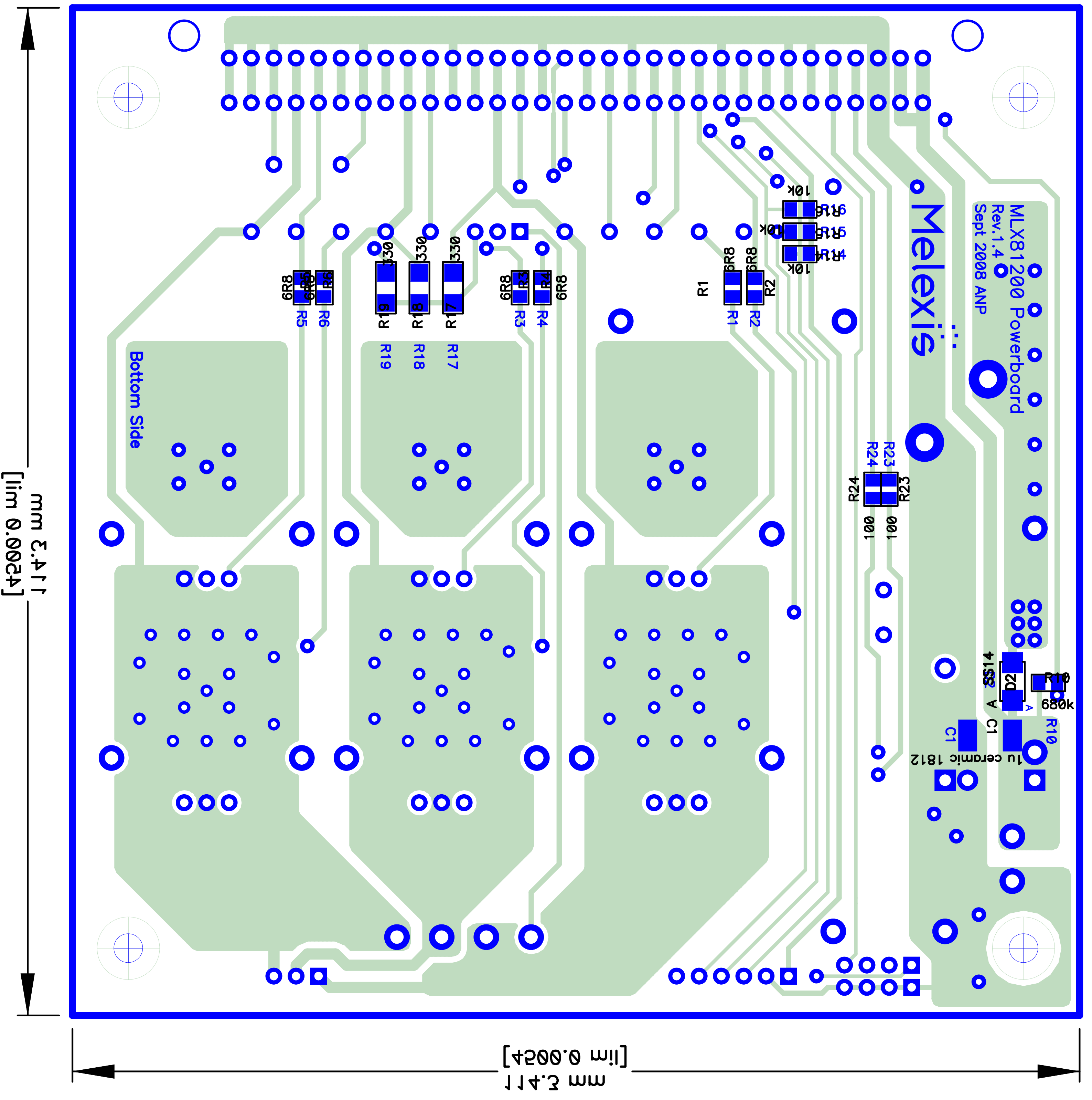

Input from Mini E-MLX
Input from E-MLX MM
E-MLX-MINIDIN9
J1
E-MLX-Mini DIN9 pinning
SHIELD
1 GND
2 EXTRA1
3 VOUT0_25
4 MUST
5 EXTRA2
6 VOUT3_12
7 VOUT1_25
8 MICE
9 VOUT2_12
R21 1k
R22 1k
R38 220k
R39 220k
JMP11
JMP12
Ready for new USB Mini-EMIx
Note: for using the Parallel E-MIx the jumpers JMP11+12 and the resistors R21,R22 must be populated with 1k!
For using the Mini E-MIx R21+R22 must be increased up to 220k!
D4 BAV70
D5 BAV70
JMP22
JMP21
R35 51
R36 nc.
Default setting is for USB Mini E-MIx:
R35: 0R R36: nc. JMP21 close
.
Setting for Parallel E-MIx:
R35, R36: nc. JMP21 open
5V Regulator
D6 BAV70
D19 BAV70
U3 78L05
C30 nc(1u/25V)
C31 100n
C32 100n
C33 1u/25V
Buffer E-MLX Testlines
U2:A 74HC125
U2:B 74HC125
U2:C 74HC125
R30 10k
R34 nc.
R31 100k
R32 100k
R42 nc
R43 nc
R44 nc
JMP10 nc.
STARTPROT
History list (for complete list see next page)
Disable E-MLX
R28 100k
JMP9 nc.
R27 47k
T2 BC848
R24 47k
R26 100k
always on
U2:D 74HC125D
C25 100n
V18V Detection
R37 47k
T3 BC848
VBAT Detection
D8
D9
R23 nc.(100k)
nc.(BAV70)
nc.(5.6V)
R25 100k
Note: for passing through the Test signals TI0+TI1+TO detection of VEMU (switched by E-MIx) and V18V (chip is running).
VBAT1 detection is obsolete!
E-MLX Indicator
R29 100k
BC857
T4
R20 820
low current
D20 Emulator on / yellow
Setting E-MLX
S2 DIP_SCHALTER_2
Place S2 instead of JMP9+10
Note: DISABLE will separate Testpins from E-MIx (status is showed by D20)
STARTPROTECT will hold MelexCM in MLX Test Mode (TI0 or TI1 is high)
R40 0
R41 nc
CAN Interface
ST3
WSL20
Title MLX81200_Evalboard
Size A3
E-MLX con
Rev 1.7
Date Fri Nov 28, 2008
Author A.Peterl/ERE
Filename MLX81200_Evalboard_1_7.sch
Sheet 2 of 3


114.3 mm
[4500.0 mil]
129.5 mm
[5100.0 mil]
ST1
J1
E-MLX-MINIDIN9
X1
GND VBAT +
LIN
VBAT LIN GND
JMP11
JMP12
JMP21
JMP22
DIP Switch
S2
LIN-Emulator
EMULIN
ST4
ST3
WSL20
E-MLX on
E-MLX Start-off protection
JMP1
JMP1 VS
JMP2
JMP3
JMP5
JMP6
JMP7
JMP8
U1
U2
U3
74HC125
78L05
BC848
BCP54
IRFL014N
MLX Test
IO[5:0]
CAN Interface
To MLX81200 Powerboard
Melexis
Microelectronic Integrated Systems
www.melexis.com
MLX81200 Evalboard
Rev.1.7 Sept. 2008 ANP
Top Side


Melexis
Microelectronic Integrated Systems
MLX81200 Evalboard
Rev.1.7 Setp. 2008 ANP
MLX81200_Evalboard_1_7.pcb
Bottom Side
R36
D7
BAV70
R11
100k
T6
BC848
R39
220k
R38
220k
D4
D5
C11
100n
C12
C10
D12
D11
D10
C22+
nc
100n
C23
BAS70-05
D16
R16
22
D24
nc.
C15
10n
R13
22
D18
nc
D13
C16
10n
R14
22
D22
nc
D14
C17
10n
R15
22
D23
nc
D15
C18
10n
C24
C24+
C21
680p
R17
0
C10
100n
D17
BAV99
C19
100p
mm 114.3
[inch 0.000+A]
mm 129.5
[inch 5100.0]


GND_BAT
ST18
ST19
ST20
Variant A:
near VBAT connector
for sensing current peaks
History list:
changings from rev.1 to rev.1.1
– Bugfix VBAT_S1 and VBAT_S2 were exchanged on R81
changings from rev.1.1 to rev.1.3
– delete the copper wire on R23,R24 and change the value to 100 Ohm
changings from rev.1.3 to rev.1.4
– exchange of C2 to other package
VISHAY_BC_6800u/50V
C2
VBAT_PROTECT
High Side Current Sense
R21
0/tbd.
R20
0/tbd.
R8
0.005R
R81
0 / Copper Wire d=1.5mm
V_MOS
Supply
ST2
VBAT
R10
680k
EN_VS
T7
IRL1004
VBAT_PROTECT
D2
SS14
R25
0 / Copper Wire d=1.5mm /optional
D1
P6KE24A
C1
1u
R9
S10K20 Varistor
Connectors for Hallsensors
SW3/ADC_Input
Connector_Sensor_Transmotec
IO
ST10A
ST9
R14
nc/optional
R15
nc/optional
R16
nc/optional
V5EXT
SW0
SW1
SW2
SW3
STECKER 4-POLIG
STECKER 6-POLIG
GND
ST10B
STECKER 4-POLIG
Low Side Current Sense
GND_S1
GND_S2
R23
100
R24
100
R7
0.005R
R71
0 / Copper Wire d=1.5mm
Driver Stage U
ST11
HS0
R1
6R8
IRL1004
T1
R11
680k
ST12
LS0
R2
6R8
T2
R22
680k
C11
10n
C22
1u ceramic
CAP_SMD1812
GND_MOS
Driver Stage V
ST13
HS1
R3
6R8
T3
R33
680k
ST14
LS1
R4
6R8
T4
R44
680k
C33
10n
C44
1u ceramic
CAP_SMD1812
Driver Stage W
ST15
HS2
R5
6R8
T5
R55
680k
ST16
LS2
R6
6R8
T6
R66
680k
C55
10n
C66
1u ceramic
CAP_SMD1812
Variant B:
near phases
smoothing current peaks
C77
1u
1000u/50V
C3
Buerklin 12D3964
ST5
ST6
ST7
ST8
R17 330
R18 330
R19 330
virtual Starpoint
VIRTUAL_STERN
ST3
ST17
BLDC
STERN_MOTOR
ST4
JMP1
BUCHSE 64-C-90GRAD
ST1
Test Connectors Phase
Motor Connector
Connector to MLX81200 Evalboard
Title MLX81200 Powerboard
Size A3
POWER_MOS
Rev 1.4
Date Fri Nov 28, 2008
Author ANP/ERE
Filename MLX81200_Powerboard_1_4.sch
Sheet 1 of 2


114.3 mm
[4500.0 mil]
114.3 mm
[4500.0 mil]
Polarity Protected
Polarity Protected
ST17
ST3
ST4
MOTOR
ST9
ST10B
ST10A
Low Side Shunt
R71
NC.
R7
0.005R
ST2
GND
VBAT
R9
SUPPLY
S10K20 Varistor
P6KE24A
D1
IRL1004
T7
R25
NC.
Hall Sensor Connector
SW2
SW1
SW0
GND
+5V
Halfbridge W
Halfbridge V
Halfbridge U
C55
10n
R66
680k
C33
10n
R44
680k
C11
10n
R22
680k
IRL1004
LS2
LS1
LS0
T6
T4
T2
T6SMD
T4SMD
T2SMD
C66
1u ceramic 1812
C44
1u ceramic 1812
C22
1u ceramic 1812
R55
680k
R33
680k
R11
680k
Bottom Driver
C77
1u ceramic 1812
C3
nc
40V
470uF
EPCOS B41303
C2
Top Side
HS2
HS1
HS0
T5
T3
T1
T5SMD
T3SMD
T1SMD
Top Driver
R81
NC.
R8
0.001R
High Side Shunt
MLX81200 Powerboard
Rev.1.4 Sept 2008 ANP
Melexis
Microelectronic Integrated Systems
Test W
Test HS2
Test LS2
Test V
Test HS1
T(virtual)
T(real)
JMP1
Test U
Test T
ST7
ST15
ST16
ST6
ST13
ST5
ST8
ST14
ST11
ST12
JMP1
R21
R21
R20
ST18
ST20
ST19
GND
GND
GND
Row1
Row32
ST1
C
A
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32


Melexis
MLX81200 Powerboard
Rev.1.4
Sept 2008 ANP
Bottom Side
114.3 mm
[4500.0 mil]

9.8. Component order codes

Component		Name	Manufacturer	Distributor	Order code
Power board	R8	Shunt	Isabellenhuette SMV0R0010	Buerklin	16 E 696
Power board	R7	Shunt	Isabellenhuette SMV0R0050	Buerklin	16 E 702
Power board	T1-T7	Power N-Fet	IR IRL1104S		
Power board		Heat sink TO220	AAVID THERMALLOY HF20	Farnell	121 3432
			AluTronic CK 960/20/SE	Reichelt	V CK960/20
EVB	J1	Connector Mini Din 9 pol	unbranded 4850.295	Farnell	152 392

10. History record

Rev.	No.	Change	Date
1.0	1	Creation of document	20/Dec/06
1.1	2	Changed description from EMLX-MM programmer to Mini E-MLX and adapted to hardware revision EVB rev1.4 and Powerboard rev1.1	13/Feb/08
1.2	3	Add chapter 7 BLDC DemoKit PC program	1/May/08
1.3	4	Add chapter 7 MLX81200 Software configuration tool	24/Okt/08
1.4	5	Add chapter “8.2 Command line parameter”, adapted new SW structure and add chapter “7.4.1 Configure the hardware in use”	05/Nov/08
1.5	6	-Adapted schematic and assembly drawings to EVB rev 1.7 and PWRB rev 1.4 -add component order codes for exotic parts -add chapter power board	12/Feb/09

11. Disclaimer

Devices sold by Melexis are covered by the warranty and patent indemnification provisions appearing in its Term of Sale. Melexis makes no warranty, express, statutory, implied, or by description regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. Melexis reserves the right to change specifications and prices at any time and without notice. Therefore, prior to designing this product into a system, it is necessary to check with Melexis for current information. This product is intended for use in normal commercial applications. Applications requiring extended temperature range, unusual environmental requirements, or high reliability applications, such as military, medical life-support or life-sustaining equipment are specifically not recommended without additional processing by Melexis for each application.
The information furnished by Melexis is believed to be correct and accurate. However, Melexis shall not be liable to recipient or any third party for any damages, including but not limited to personal injury, property damage, loss of profits, loss of use, interrupt of business or indirect, special incidental or consequential damages, of any kind, in connection with or arising out of the furnishing, performance or use of the technical data herein. No obligation or liability to recipient or any third party shall arise or flow out of Melexis' rendering of technical or other services.

For the latest version of this document, go to our website at:
www.melexis.com

Or for additional information contact Melexis Direct:

Europe and Japan:
Phone: +32 1367 0495
E-mail: sales_europe@melexis.com

All other locations:
Phone: +1 603 223 2362
E-mail: sales_usa@melexis.com