DVK90109

125kHz RFID Development Kit
User Manual

Features and Benefits

- ❑ Standard ISO communications
- ❑ User interface software
- ❑ RS232 serial communication
- ❑ Dedicated instruction set
- ❑ 6 to 9Volts supply compliant

Ordering Information

Part No.
DVK90109

General Description

The development kit DVK90109 has been developed by Melexis to create specific applications using 125 kHz protocols and to evaluate the advantages of the MLX90109 transceiver IC.

The DVK90109 results of the combination of the development board DVKRFID and the evaluation board EVB90109. The EVB90109 is plugged on the DVKRFID through a 10 pins connector (including Coil, VCC and Ground) and controlled by the microcontroller ATMEGA128.

A set of commands has been elaborated by Melexis to define the communication between the Computer and the evaluation board EVB90109.

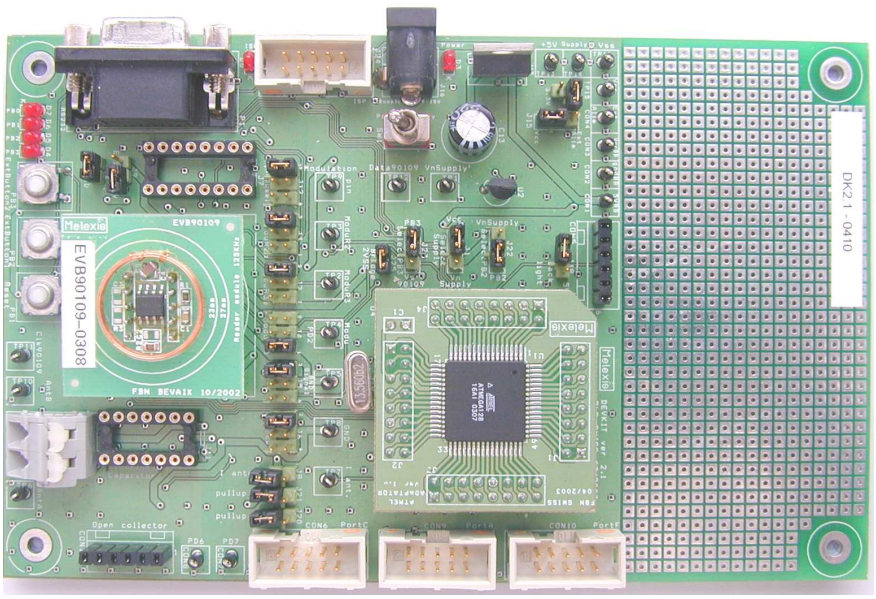


Table of Contents

1 UNPACKING THE SYSTEM 3

2 CONNECTING THE SYSTEM 3

3 DIGITAL COMMUNICATION EVB90109 - DVKRFID 4

4 DIGITAL COMMUNICATION DVKRFID – PC 5

4.1 Protocol assignment 5
4.2 List of instructions 5
4.3 Error Instruction 6
4.4 Instructions set 6
4.4.1 Version (VER) 6
4.4.2 Set Field (STF) 6
4.4.3 Configuration of the MLX90109 (CFG) 7
4.4.4 HEA : HEAder 7
4.4.5 REA : REAd 7
4.4.6 WRI : Write Q5 tag 8
4.4.7 SEL : Select tag Hitag1 9
4.4.8 RDH : Read Hitag1 memory 9
4.4.9 WRH : Write Hitag1 memory 9

5 MELEXIS RS232 INTERFACE PROGRAM 11

5.1 Start RS232 interface: 11
5.2 Select serial communication port COM 11
5.3 Open serial communication port 11
5.4 Generates commands file 11
5.5 Select commands file 12
5.6 Send a command 12

6 DISCLAIMER 13

1 Unpacking the system

The DVK90109 board is delivered with all components to build a complete RFID setup.

Contents of the box:

1) Development kit 125 kHz
   - o Development kit RFID (DVKRFID)
   - o Evaluation board MLX90109 (EVB90109)
   - o Alimentation block 3 to 12V
   - o 125 KHz Transponder in credit card format
2) Compact Disc (Software and documentation)
   - o EVB90109 documentation
   - o DVK90109 User Guide
   - o RS232 PC interface software

2 Connecting the system

1) The EVB90109 is plugged through a 10 pins connector (DIL10) on a 14 pins connector (DIL14) on the DVKRFID (EVB90109 AREA). The following pictures show how to connect the EVB90109 on the DVKRFID board.

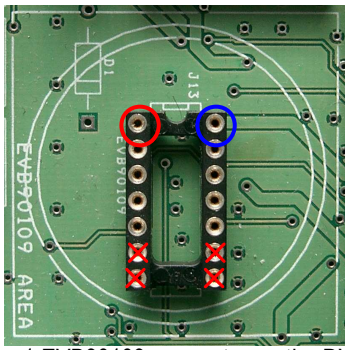


Figure 1:EVB90109 connector on the DVKRFID

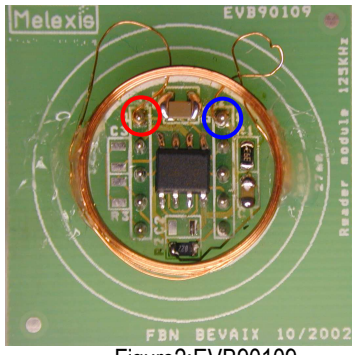


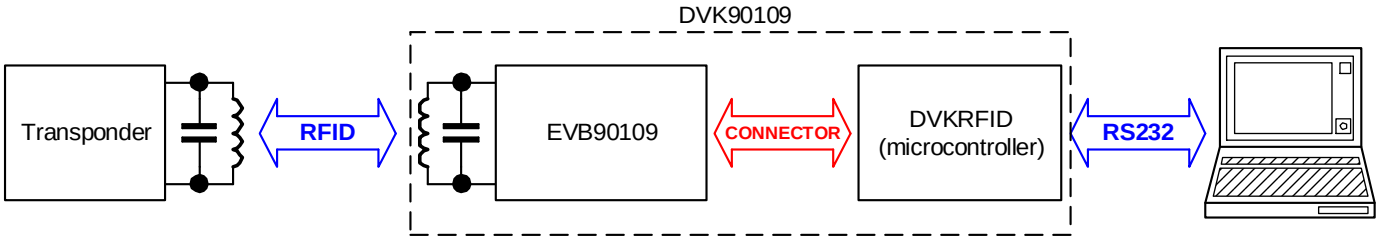
Figure2:EVB90109

2) Connect the board DVK90109 to the COM-port of your computer, with a RS232 cable. This allows communication between the DVK90109 and your computer.
3) Connect the power supply of the DVK90109.

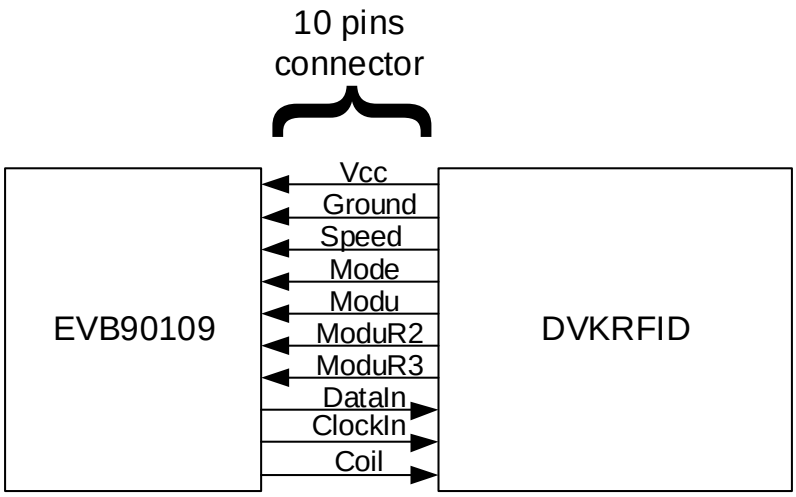
Important Note: The supply must be between **6 to 9 Volts DC**. Higher voltage may damage permanently the DVK90109 circuit.

3 Digital communication EVB90109 - DVKRFID

The microcontroller on the DVKRFID board communicates with the EVB90109 through a 10 pins connector.



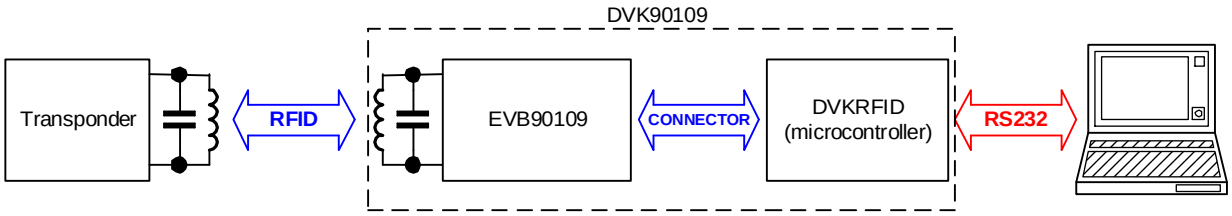
The picture and the table below describe all connections between the EVB90109 and the DVKRFID board.



Name	Function	Direction	Comment
Vcc	Supply voltage for the EVB90109	DVK -> EVB	
GND	Ground connection for the EVB90109	DVK -> EVB	
Speed	Data rate selection: GND = 4kBaud, Vcc = 2kBaud	DVK -> EVB	
Mode	Decoding selection: GND = Biphase, Vcc = Manchester	DVK -> EVB	
Modu	Power down mode	DVK -> EVB	NOT USED
ModuR2	Antenna voltage control: GND = Field ON, Vcc = Field OFF	DVK -> EVB	
ModuR3	-	DVK -> EVB	NOT USED
DataIn	Data Ouput	EVB -> DVK	DataIn and ClockIn will be internally pulled-up by the microcontroller.
ClockIn	Clock Ouput	EVB -> DVK	
Coil	Oscillator output	EVB -> DVK	

4 Digital communication DVKRFID – PC

The DVK90109 board communicates with a PC through a serial interface RS232.



This part specifies a dedicated set of commands useful for the operation with the **MLX90109** RFID transceiver IC.
This instruction set is defined for the development kit DVK90109.

The microcontroller ensures the following functions:

- o Control and configure the MLX90109 transceiver
- o Read the data information from the MLX90109 transceiver (on-chip decoding)
- o Communicate with a PC through the serial connection RS232.

4.1 Protocol assignment

The different commands are based on a Mnemonic assignment defined by three ASCII characters forming the Operation Code, followed by a specific number of ASCII or Hexadecimal parameters, depending on the type of the command.

OP CODE : 3 ASCII characters

PARAMETERS : ASCII characters or
Hexadecimal characters

The syntax of a command is typically: OpCode<Parameter1>,<Parameter2>,<Parameter3>,…
Each parameter should be separated by a comma.

4.2 List of instructions

Operation	Syntax	Function
VERsion	VER↵	Get the firmware version for the microcontroller
SeT Field	STF<On/Off>↵	Set the field ON or OFF
ConFiGuration	CFG<Encoding>,<Datarate>↵	Set the type of current transponder with the encoding method and the data rate used by the Transponder
HEAder	HEA<Header>,<Nbr Significant bit>↵	Define the header to read the Identification code (ID)
REAd Tag	REA<Op mode>↵	Perform a read operation and decode data with the encoding method of the transponder defined with the <Op mode> parameter
WRIte Tag	WRI,<data>,<address>↵	Perform a write operation with 100% modulation for Q5 tag
SELect Tag	SEL↵	Select a Hitag1 tag by sending an UID request then a select command. Return UID and configuration page
ReaD Hitag1	RDH,<command>,<address>↵	Perform a read operation and decode data with the encoding method of the transponder Hitag1

WRite Hitag1	WRH,<command>,<address>,<data>↵	Perform a write operation for Hiatg1 tag with the data defined in the <data> parameter

Note: Be careful that a value is necessary in all fields defined by the instruction. The default value defined in the following instruction set is only the value set by the program during the initialization phase. This means that if the CFG instruction is not sent, the <Encoding> and <Baudrate> fields will take default values being respectively biphase and 4kbit/s.

4.3 Error Instruction

No	Name	Meaning	Message
0	Instruction error	If the Opcode syntax is not correct or not defined.	err0↵
1	Definition error	If the definition syntax is not correct.	err1↵
2	No tag detected	If no transponder has been detected.	err2↵
3	Header not found	If the header cannot be found in the read code.	err3
5	Read operation failed	If the calculated CRC or parity check doesn't correspond to the read value.	err5↵
6	Others error		err6↵
8	Measure Frequency Antenna	If a problem occurs during antenna frequency measurement (value too high, no signal …)	err8↵

4.4 Instructions set

4.4.1 Version (VER)

Get the firmware version.
This command can be used to control the serial connection RS232.

Syntax	VER↵
Return	Melexis RFID <Development kit> V<*Version*>
Example	VER↵ Reply: Melexis RFID DVK90109 V2.0

4.4.2 Set Field (STF)

Set the magnetic field ON or OFF.

Syntax	STF<on/off>↵ <on/off> = Characters defining the current state of the field (<on> to put the field ON and <off> to put the field OFF). Default value: <on>.
Return	Ok
Example	STFon↵ Reply: Ok

4.4.3 Configuration of the MLX90109 (CFG)

Set the configuration of the transceiver to decode data information from a transponder (encoding mode and data rate).

Syntax	CFG<Encoding>,<Datarate>↵ <Encoding> = ASCII letter specifying the encoding method used by the Transponder (<M> for Manchester and <B> for Biphase). Default value: <B>. <Datarate> = ASCII number specifying the data rate used by the transponder (<2> for 2kbits/s, <4> for 4kbits/s). Default value: <4>.
Return	Ok
Example	CFGB,4↵ Reply: Ok

4.4.4 HEA : HEAder

Specify the header of the data sent by the transponder.

The program will search for the header in the data read from the transponder and display the data with the header first.

The header is a number of bits that cannot be reproduced in the rest of the ID. For example in the standard norm ISO11784, the first 11 bits of the ID code will be used as the header and fixed to “0000 0000 001”.

Syntax	HEA<Header>,<Nbr significant bit>↵ <Header> = Hexadecimal number defining the header. <Nbr unused bit> = Hexadecimal number defining the number of Header bits to be removed (mustn’t be higher than 7).
Return	Ok <Header in binary>
Example	HEA0020,5↵ Reply: Ok 00000000001 HEA002010,7↵ Reply: Ok 00000000001000000

Example: The header is calculated with two parameters. <*Header*> defining the header in hexadecimal (an entire number of byte is necessary) and <*Nbr Of Unused bit*> a decimal number defining the number of bits which are to be removed from the given Header to form the new Header. Thus, it is possible to define a length of header different of a power of 2.

HEA0020,5 = 0000 0000 001~~0 0000~~ = 0000 0000 001

4.4.5 REA : REAd

Read, decode and return data from a transponder.
If a problem occurs during operation (no header found, no transponder), an error message will be displayed as referenced in the table *1.3 Error Instruction*.

Syntax	REA<Opmode>,<Nbr of bit>↵ <Opmode> = Number defining the programmed decoding method used. o <1> Searches for the header, decodes ISO11784 mode, controls CRC value and displays the decoded data.

	o <2> Searches for the header, decodes Melexis 88bits mode, controls CRC value and displays decoded data. o <3> Searches for the header and displays the data read. o <4> Only displays the data read (no decoding, no search for the header). <Nbr of bit> = ASCII number specifying the number of bit read. <064>, <128>, <144> and <256> are implemented.
Return	<data> err2 (no tag detected) err3 (no header found) err5 (read failed)
Example	REA1,128↵ Read 128bits, searches for the header and decodes the ID read according to the standard norm ISO11784. Checks the CRC and returns corresponding message (data read or error). REA2,128↵ Read 128bits, searches for the header and decodes the ID read according to the Melexis 88 bits encoding method. Checks the CRC and returns corresponding message (data read or error). REA3,064↵ Read 64bits, searches for the header and returns corresponding message (data read or error). REA4,256↵ Reads and displays 256bits.

Note: The ISO11784/85 standard norm and the Melexis 88bits encoding method will be only performed on 128 bits read. So in these two mode (opmode <1> and <2>) the field <Nbr of bit> will be ignored and 128bits will be always read.

Note: REA command will not set the field of the EVB90109. Before performing a read operation, the field must be ON by using the STFon command.

4.4.6 WRI : Write Q5 tag

Write data in the Q5 tag memory.
This function will use the 100% modulation (please refer to the application note “100% modulation (On-Off Keying)”) to write data according to Q5 tag communication protocol.
An “Ok” will confirm the success of the operation.

Syntax	WRI,<data>,<address>↵ <data> = 4 bytes data (hexadecimal value) <MSB…LSB>. <address> = 1 bytes address of the Q5 memory (hexadecimal value).
Return	Ok Err2↵ (no tag detected)
Example	WRI,00AA00BB,07↵ Write data 0x00, 0xAA, 0x00 and 0xBB into the block 7 of the Q5 memory Ok>

Note: WRI command will not set the field of the EVB90109. Before performing a read operation, the field must be ON by using the STFon command.

4.4.7 SEL : Select tag Hitag1

Select the tag Hitag1 in the field by sending a Set_CC and a SELECT command.
This function will use the 100% modulation (please refer to the application note "100% modulation (On-Off Keying)") to select the tag according to Hitag1 communication protocol. The tag answer with its UID and its configuration page
The command SEL will return the UID and the configuration page in case of success of the operation.

Syntax	SEL↵
Return	UID: Unique tag identifier Configuration page: configuration of the tag (encoding, data rate, security features…) Err2↵ (no tag detected)
Example	SEL↵ UID = 7972C535 / Configuration page = FF370000>

Note: SEL command will not set the field of the EVB90109. Before performing a read operation, the field must be ON by using the STFon command.

4.4.8 RDH : Read Hitag1 memory

Read an area of the Hitag1 memory.
This function will use the 100% modulation (please refer to the application note "100% modulation (On-Off Keying)") to send the selected command and will decode the data sent by the tag according to the Hitag1 communication protocol.
The command RDH will return the data requested in case of success of the operation.

Syntax	RDH,<command>,<address>↵ <command> = 4 bits (binary value) corresponding to the command (refer to Hitag1 specification). <address> = 1 byte address (hexadecimal value)
Return	Data : size depends of the type of command and the address used Err2↵ (no tag detected)
Example	RDH,1101,10↵ Data received = AABBCCDD……..>

Note: SEL command has to be sent before performing a read operation.

4.4.9 WRH : Write Hitag1 memory

Write an area of the Hitag1 memory.
This function will use the 100% modulation (please refer to the application note "100% modulation (On-Off Keying)") to send the selected command and the selected data according to the Hitag1 communication protocol.
The command WRH will return a confirmation sentence in case of success of the operation.

Syntax	RDH,<command>,<address>,<data>↵ <command> = 4 bits (binary value) corresponding to the command (refer to Hitag1 specification). <address> = 1 byte address (hexadecimal value) <data> = 4 bytes data to write in the Hitag1 memory
Return	Confirmation sentence

	Err2↵ (no tag detected)
Example	WRH,1000,10,AAABBCCDD↵ 4 bytes data AABBCCDD written at address 10>

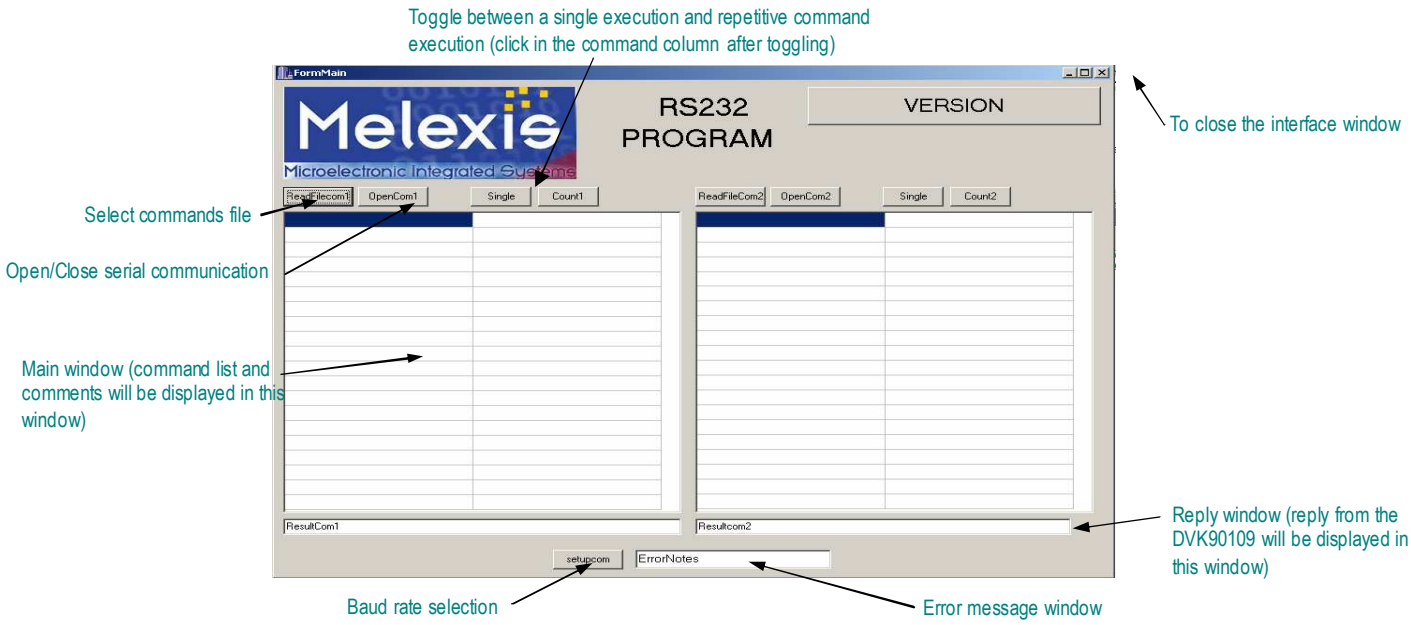
Note: SEL command has to be sent before performing a read operation.

5 Melexis RS232 interface program

The RS232 interface program is a easy-to-use windows software developed by Melexis to send commands through the RS232 interface of a Pc.
The RS232 interface program uses text files to generate the set of commands and is able to control the COM port of the Pc (COM 1 or 2 can be chosen) and to send the corresponding commands through it.

5.1 Start RS232 interface:

Double click on RS232.exe file. The following interface window will be displayed.



5.2 Select serial communication port COM

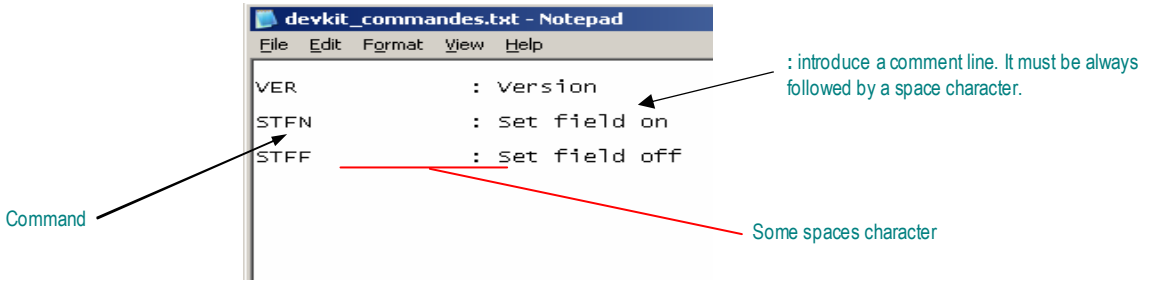
Click on “setupcom” button. On the new window displayed, choose “br9600” on COM1-2. Close the new window by clicking on the top right corner. The speed communication is now validated at 9600 bit/s.

5.3 Open serial communication port

Click on “OpenCom1” button. This will open the corresponding communication port (COM1-2).

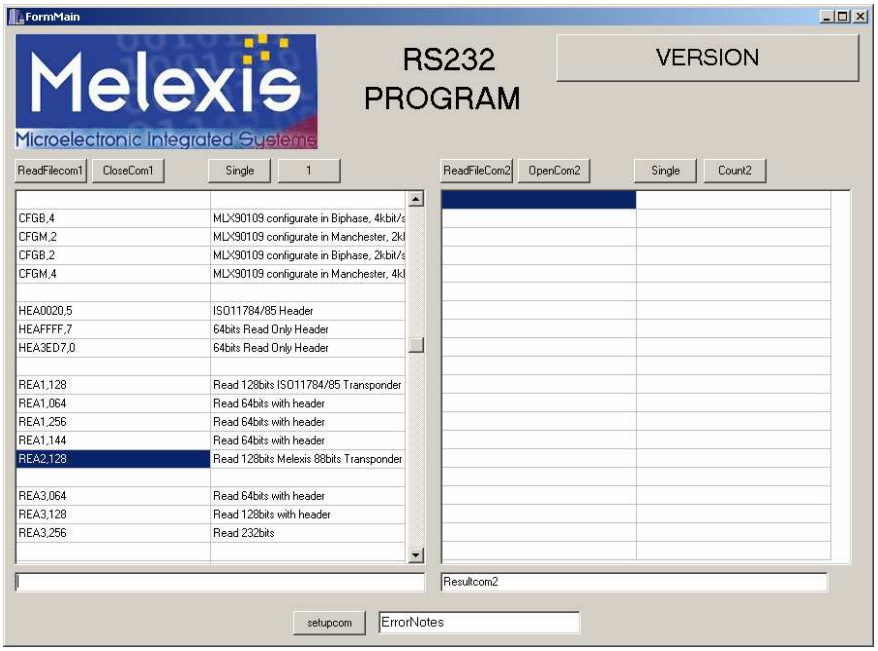
5.4 Generates commands file

The RS 232 program uses a text file which contains all commands. All commands must be defined like specified below.



5.5 Select commands file

Click on “ReadFilecom1” button. On the new displayed windows, find the text file which contains the command list (MLX-Programmer.txt is available together with RS232.exe) and double click on it.
This file will be displayed on the left table of the interface window. On the left column, several lines of commands are displayed with some comments on the right column like shown in the picture below.



5.6 Send a command

To send a command, double click on it and the response will be displayed in the “ResultCom” window.

6 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.

© 2004 Melexis NV. All rights reserved.

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

ISO/TS 16949 and ISO14001 Certified

OCEAN CHIPS
Океан Электроники
Поставка электронных компонентов

Компания «Океан Электроники» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Поставка оригинальных импортных электронных компонентов напрямую с производств Америки, Европы и Азии, а так же с крупнейших складов мира;
- Широкая линейка поставок активных и пассивных импортных электронных компонентов (более 30 млн. наименований);
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Помощь Конструкторского Отдела и консультации квалифицированных инженеров;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Поставка электронных компонентов под контролем ВП;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- При необходимости вся продукция военного и аэрокосмического назначения проходит испытания и сертификацию в лаборатории (по согласованию с заказчиком);
- Поставка специализированных компонентов военного и аэрокосмического уровня качества (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Actel, Aeroflex, Peregrine, VPT, Syfer, Eurofarad, Texas Instruments, MS Kennedy, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Компания «Океан Электроники» является официальным дистрибьютором и эксклюзивным представителем в России одного из крупнейших производителей разъемов военного и аэрокосмического назначения «**JONHON**», а так же официальным дистрибьютором и эксклюзивным представителем в России производителя высокотехнологичных и надежных решений для передачи СВЧ сигналов «**FORSTAR**».

AVIC JONHON

«JONHON» (основан в 1970 г.)

Разъемы специального, военного и аэрокосмического назначения:

(Применяются в военной, авиационной, аэрокосмической, морской, железнодорожной, горно- и нефтедобывающей отраслях промышленности)

«FORSTAR» (основан в 1998 г.)

ВЧ соединители, коаксиальные кабели, кабельные сборки и микроволновые компоненты:

(Применяются в телекоммуникациях гражданского и специального назначения, в средствах связи, РЛС, а так же военной, авиационной и аэрокосмической отраслях промышленности).

FORSTAR®

Телефон: 8 (812) 309-75-97 (многоканальный)
Факс: 8 (812) 320-03-32
Электронная почта: ocean@oceanchips.ru
Web: http://oceanchips.ru/
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, д. 2, корп. 4, лит. А