Document Number: KT33912UG
Rev. 2.0, 10/2009

KIT33912EVME System Basis Chip with LIN Tranceiver Setup Instructions

Important Notice

Freescale provides the enclosed product(s) under the following conditions:

This evaluation kit is intended for use of ENGINEERING DEVELOPMENT OR EVALUATION PURPOSES ONLY. It is provided as a sample IC pre-soldered to a printed circuit board to make it easier to access inputs, outputs, and supply terminals. This EVM may be used with any development system or other source of I/O signals by simply connecting it to the host MCU or computer board via off-the-shelf cables. This EVM is not a Reference Design and is not intended to represent a final design recommendation for any particular application. Final device in an application will be heavily dependent on proper printed circuit board layout and heat sinking design as well as attention to supply filtering, transient suppression, and I/O signal quality.

The goods provided may not be complete in terms of required design, marketing, and or manufacturing related protective considerations, including product safety measures typically found in the end product incorporating the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge. In order to minimize risks associated with the customers applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards. For any safety concerns, contact Freescale sales and technical support services.

As a prototype, this product does not fall within the scope of the European Union directive on electromagnetic compatibility and therefore may not meet the technical requirements of the directive. Please be aware that the products received may not be regulatory compliant or agency certified (FCC, UL, CE, etc.).

Should this evaluation kit not meet the specifications indicated in the kit, it may be returned within 30 days from the date of delivery and will be replaced by a new kit.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. *Typical* parameters can and do vary in different applications and actual performance may vary over time. All operating parameters, including *Typical*, must be validated for each customer application by customer's technical experts.

Freescale does not convey any license under its patent rights nor the rights of others. Freescale products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale product could create a situation where personal injury or death may occur.

Should a Buyer purchase or use Freescale products for any such unintended or unauthorized application, the Buyer shall indemnify and hold Freescale and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale was negligent regarding the design or manufacture of the part.

Table of Contents

Paragraph Number | **Page Number**

Preface

Chapter 1 SBCLIN Evaluation Module Kit

Chapter 2 Setup Instructions

Chapter 3 FreeMASTER Control Page

Chapter 4 LIN Master Control Page

Chapter 5 User Helpful Control Page

Chapter 6 Embedded Slave Software

Chapter 7 FreeMASTER ActiveX Object

Chapter 8 ActiveX LIN Master Object

Chapter 9 Interconnect Board Description

Chapter 10 Jumper Connection

Appendices

List of Figures

List of Tables

Preface

This User's Guide provides the installation steps for the KIT33912EVME System Basis Chip with LIN Transceiver (SBCLIN)

Audience

This document is intended for application developers who are setting up Freescale's KIT33912EVME System Basis Chip with LIN Transceiver.

Suggested Reading

Additional documentation on the FreeMASTER software may be found at www.freescale.com.

Conventions

This document may use the following conventions:

Term or Value	Definition	Examples
Terminal Names	Terminal names are the physical connections and are shown in text as all upper case characters.	... the external supply voltage VBAT.
Terminal Values	Terminal values are the currents to/from a terminal and are shown as upper and subscripted text.	In Stop Mode the voltage regulator still supplies the MCU with V_{DD}.
Active High Signals (Logic One)	No special symbol attached to the signal name.	A0 SCLK
Active Low Signals (Logic Zero)	Noted with an overbar in text and in most figures.	$\overline{\text{RST}}$ $\overline{\text{CS}}$
Decimal Values	No special symbol attached to the number	1.0 34
Numbers	Considered positive unless specifically noted as a negative value	5.0 -10
Blue Text	Linkable on-line	...refer to
Bold	Reference sources, paths, emphasis	

References

The following sources were referenced to produce this manual:

[1] *MC33912 Data Sheet,* Freescale Semiconductor, Inc.

[2] *MC9S08DZ60 Data Sheet,* Freescale Semiconductor, Inc.

Chapter 1 Evaluation Module Kit

1.1 Introduction

The SBCLIN Evaluation Kit is a hardware and software solution to control System Basis Chip family devices from GUI installed on a PC. The kit is dedicated for laboratory use. The following sections provide the basic information:

- *1.2, KIT33912EVME System Basis Chip with LIN Transceiver*
- *1.3, System Requirements*
- *1.4, SBCLIN Evaluation Module Kit Quick Guide*

1.2 KIT33912EVME System Basis Chip with LIN Transceiver

DOCUMENTATION

• MC33912	Data Sheet
• MC9S08DZ60	Data Sheet
• FREEMASTERFS.pdf	Fact Sheet
• FreeMASTER for Embedded Applications	User Manual

SOFTWARE

• GUI	GUI FreeMASTER application
• FMASTERSW.exe	FreeMASTER Software
• Project.abs	HCS08X embedded software

HARDWARE

• KIT33912EVME	PCB
• Serial Cable	

1.3 System Requirements

To successfully create the demonstration system, each of the following systems are required:

- SBCLIN Evaluation Module Kit
- 12 V power supply with sufficient current capability
- Personal computer with:
  - — Windows operating system
  - — browser supporting ActiveX (Internet Explorer 6.0 recommended)
  - — monitor supporting a screen resolution of 1280x1024 recommended
  - — one or two serial ports (COM)
  - — CD-ROM drive
  - — $\geq$45 MB hard disk drive space

1.4 SBCLIN Evaluation Module Kit Quick Guide

This kit contains all the components to control SBCLIN devices:

- SBCLIN Printed Circuit Board
- FreeMASTER based GUI to control the devices

Figure 1-1 shows the SBCLIN Evaluation Module Kit quick guide.

NOTE

This Evaluation Module Kit isn’t appointed for EMC tests of MC33912 device.

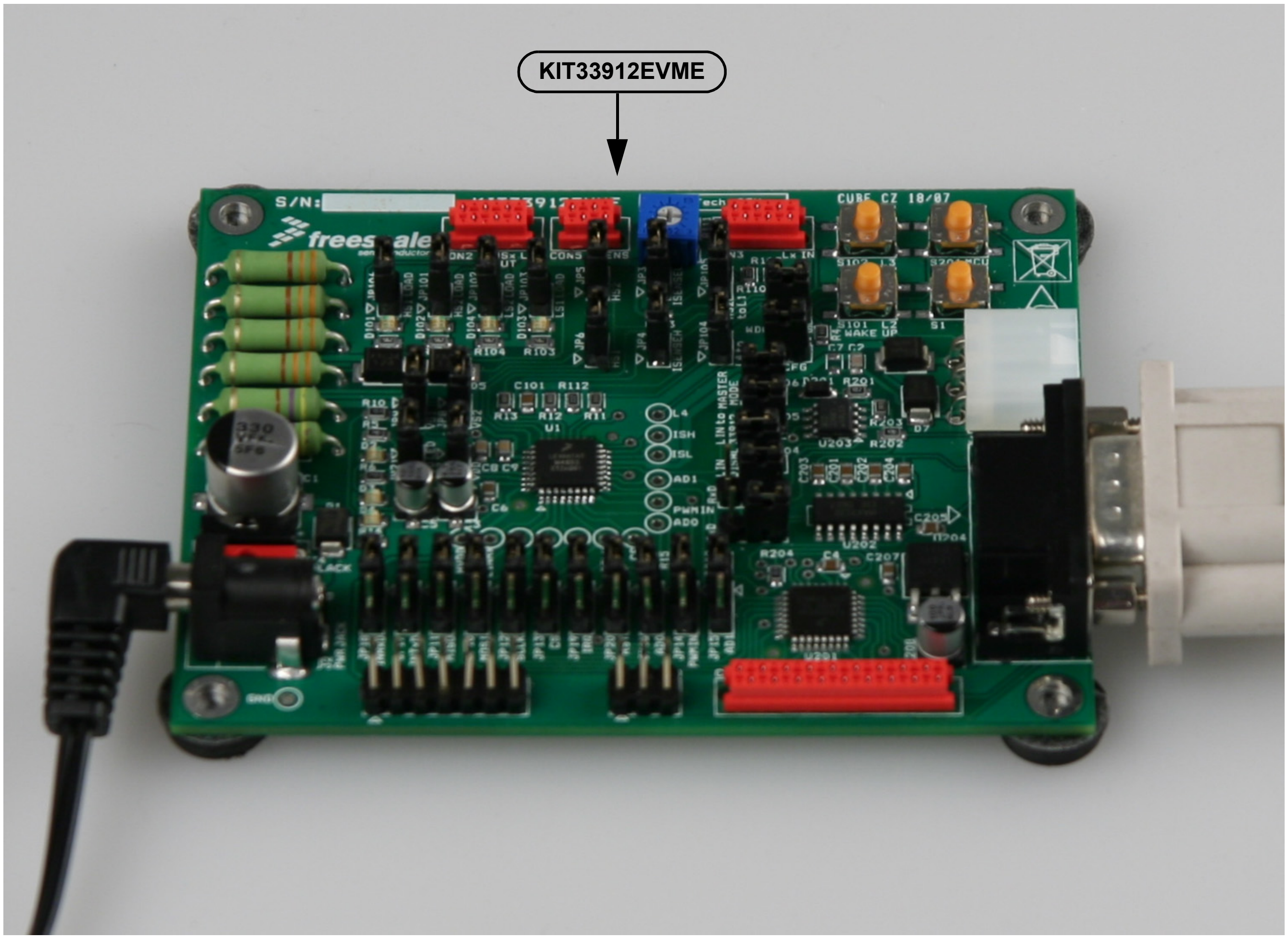


Figure 1-1. SBCLIN Evaluation Module Kit

Chapter 2 Setup Instructions

2.1 Overview

To successfully set up and run the evaluation environment the device control, the following sections provide the step-by-step instructions:

- *2.2, Install GUI on Host PC*
- *2.3, Quick System Setup to Run the SBCLIN Evaluation Module*
- *2.4, Start Graphic User Interface*

2.2 Install GUI on Host PC

To set up the GUI on your PC, you have to install the FreeMASTER software if not already installed. If it is installed, skip items 1 to 3.

1. Start the FMASTERSW.exe install shield wizard. The file can be downloaded from http://www.freescale.com.
   Results: The License Agreement box is displayed and you are prompted for further actions.

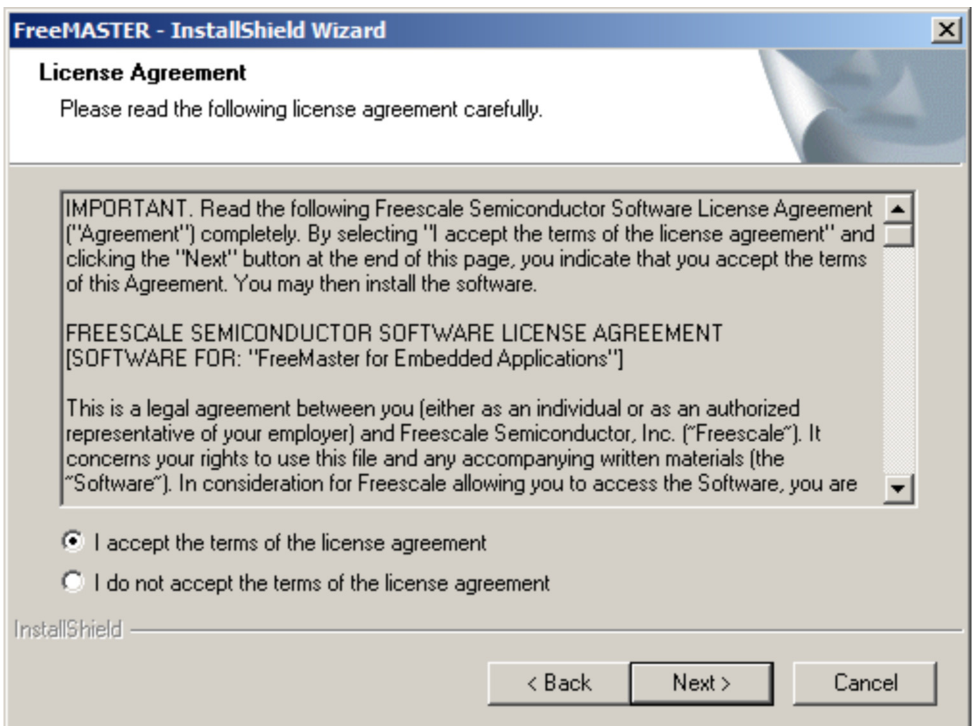


Figure 2-1. License Agreement Box

2. Clicking the **Next** button will start the installation program.
   Results: The Installation Wizard will prompt you for further actions.

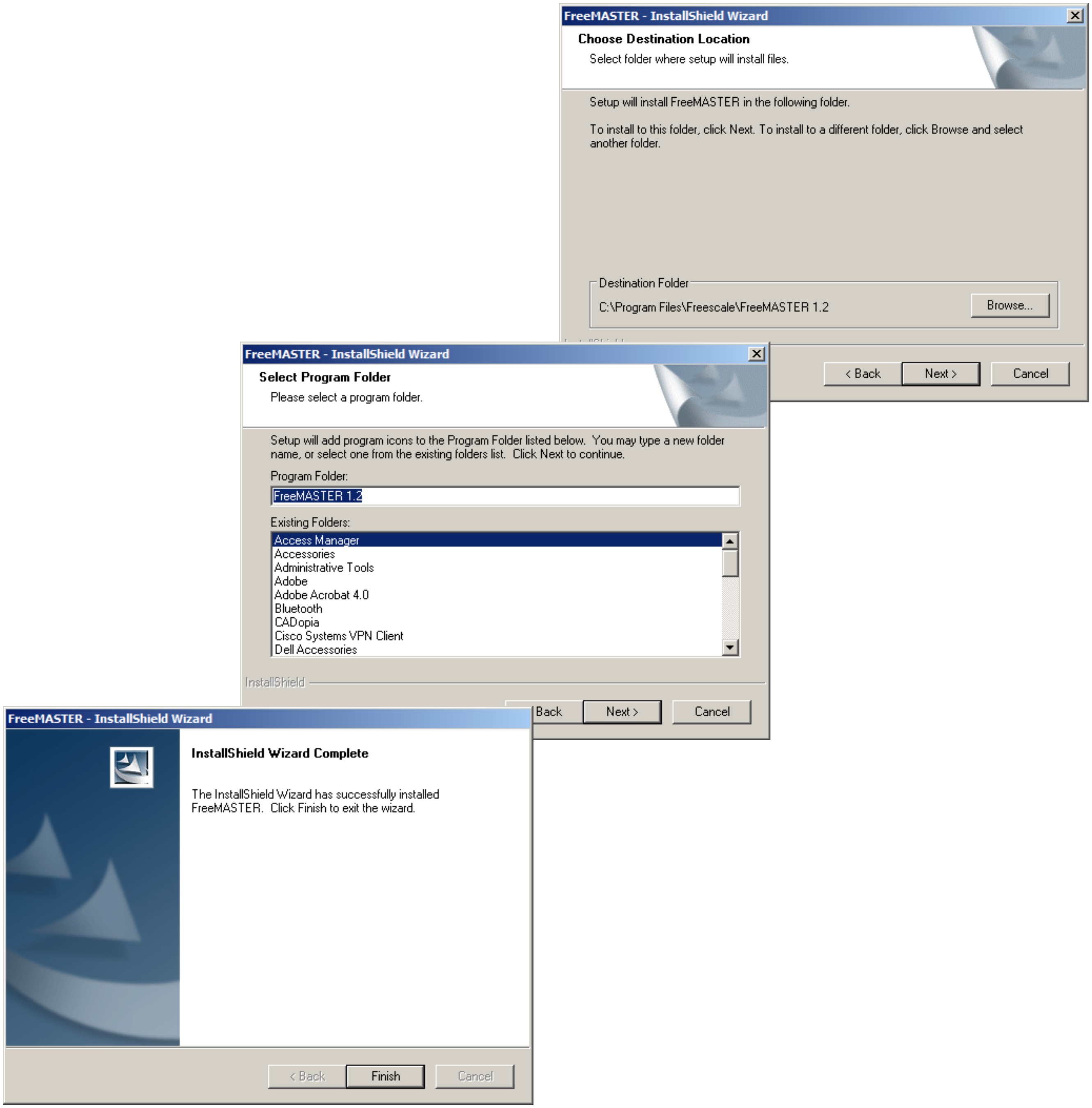


Figure 2-2. FreeMASTER Installation

3. Follow the instructions given by the Installation Wizard.
4. Install the GUI. Start the GUIxx.exe installation file.
   Results: The Installation Wizard will prompt you for further actions.

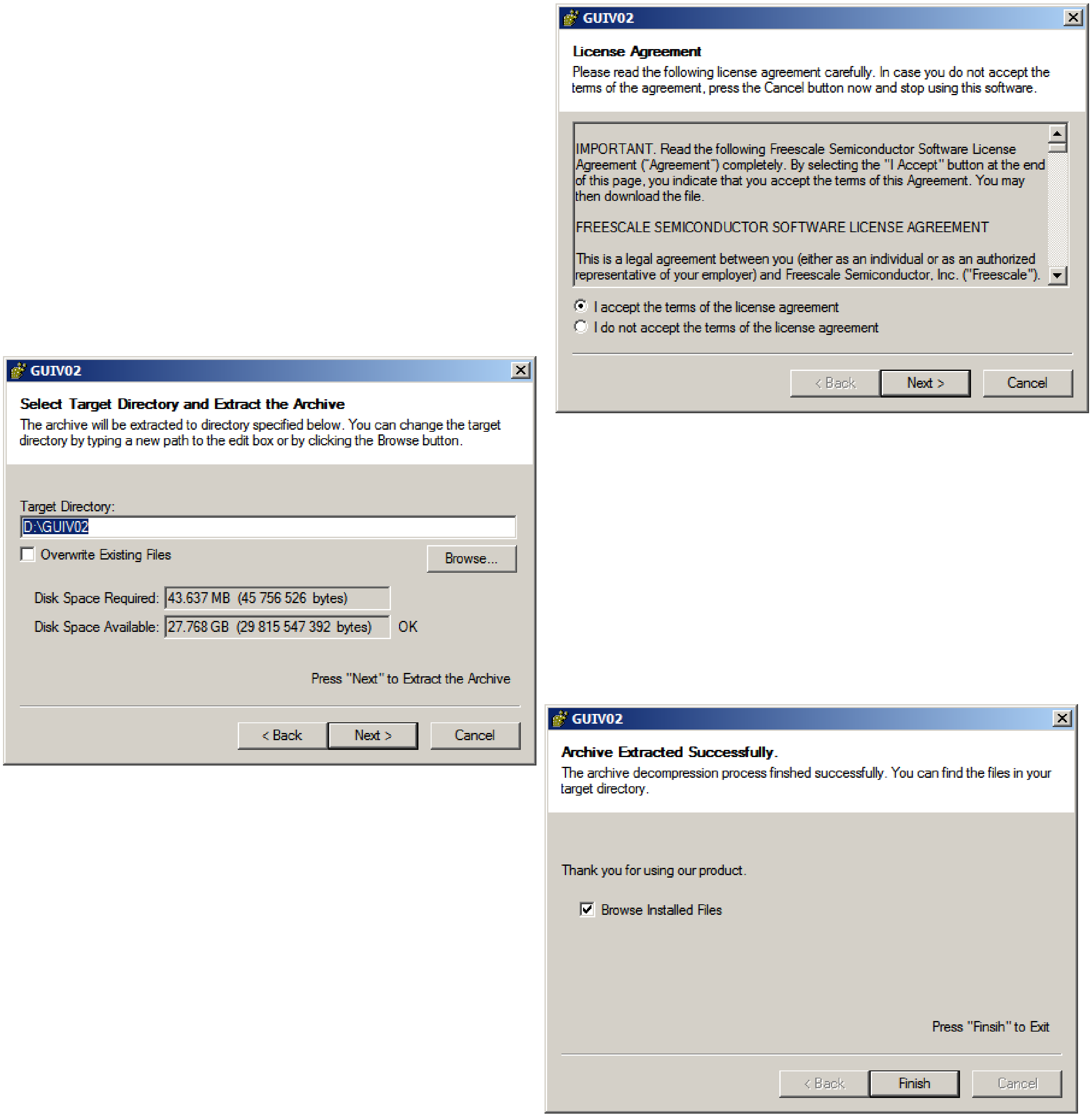


Figure 2-3. Graphical User Interface Installation Wizard

2.3 Quick System Setup to Run the SBCLIN Evaluation Module

To run an embedded application code, the KIT33912EVME must be configured by Jumpers. Follow the steps to configure the board.

1. Displace unnecessary connectors.
   a) Disconnect the programming header J201 (BKGD).
   b) Disconnect the LIN connector CON1 (LIN).
2. To supply the devices, place the following Jumpers.
   a) Place Jumper JP9 (VS1) to supply the MC33912 device.
   b) Place Jumper JP8 (VS2) to supply the High Side Switches in the MC33912 device.
   c) Place Jumper JP7 (VDD) to supply the MCU from an Internal Voltage Regulator.
3. For communication from the MCU to a personal computer, through FreeMASTER protocol, place Jumpers JP202 (TxD) and JP203 (RxD) in positions 1-2, which connect the MCU and the TTL / RS232 converter.
4. Configure the KIT33912EVME to an on-board control, by ensuring that Jumpers JP10 - JP16 and JP19 - JP20 are in position 1-2.
5. Connect the PC interface to the host PC by connecting the RS232 cable to connector J1 (RS-232) on the PC interface and to a serial port (COM) on the host PC.
6. Connect the 12 V power supply.

Results: The MC33912 device is in Normal Mode.

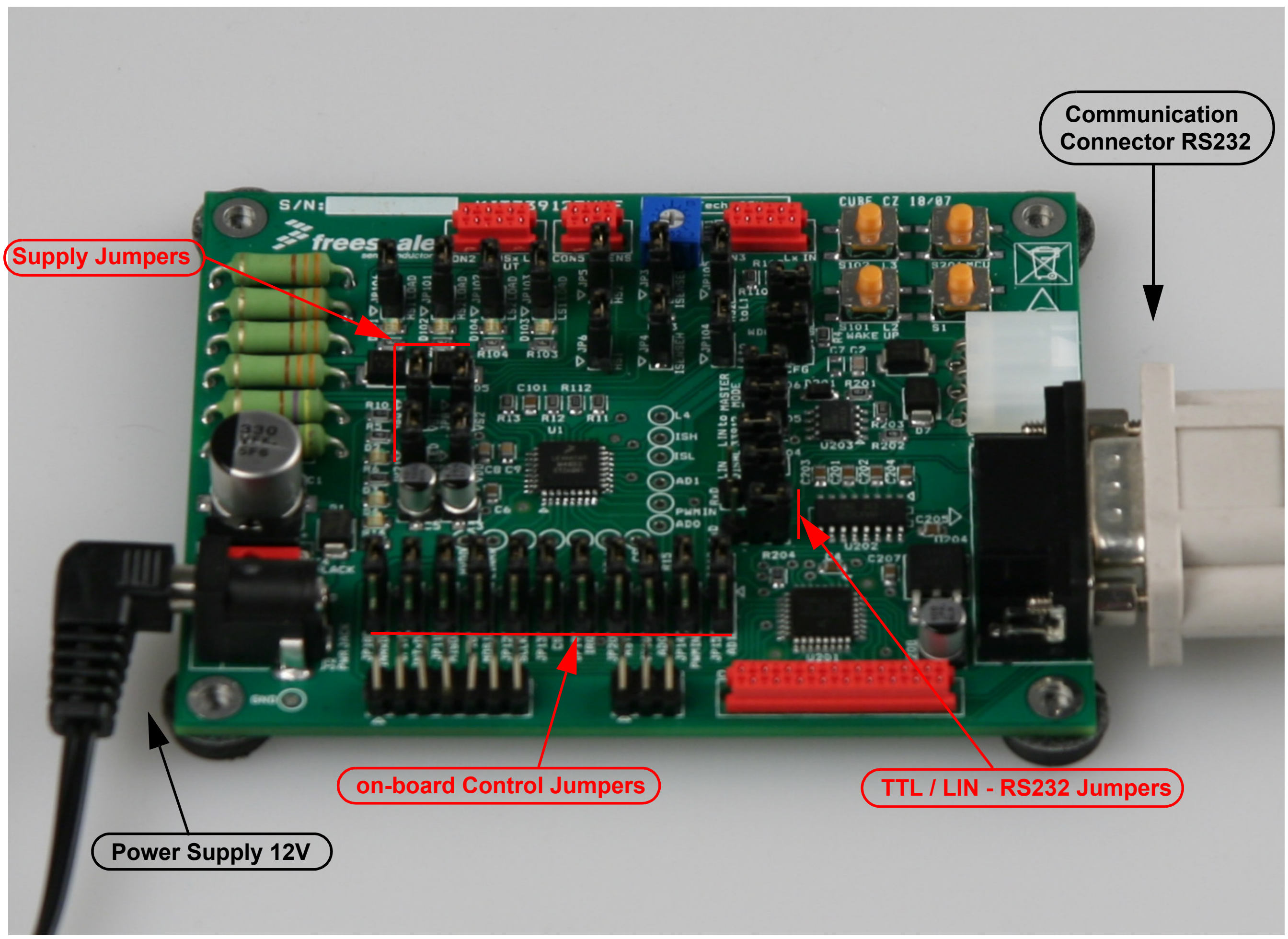


Figure 2-4. SBCLIN Power-Up Quick Guide

2.4 Start Graphic User Interface

Start the GUI:

1. From the folder where you have already installed the application, select and double-click on: **....\MC33912.pmp**.

   Results: On first-time use on the host PC, the GUI Initializing Required Components screen appears.

   On subsequent use on the host PC, the GUI Welcome screen appears.

   NOTE

   Screen resolution 1280x1024 is recommended.

   Starting GUI from a network folder is not recommended.

2. Click the **RETRY** button to initialize all the required components.

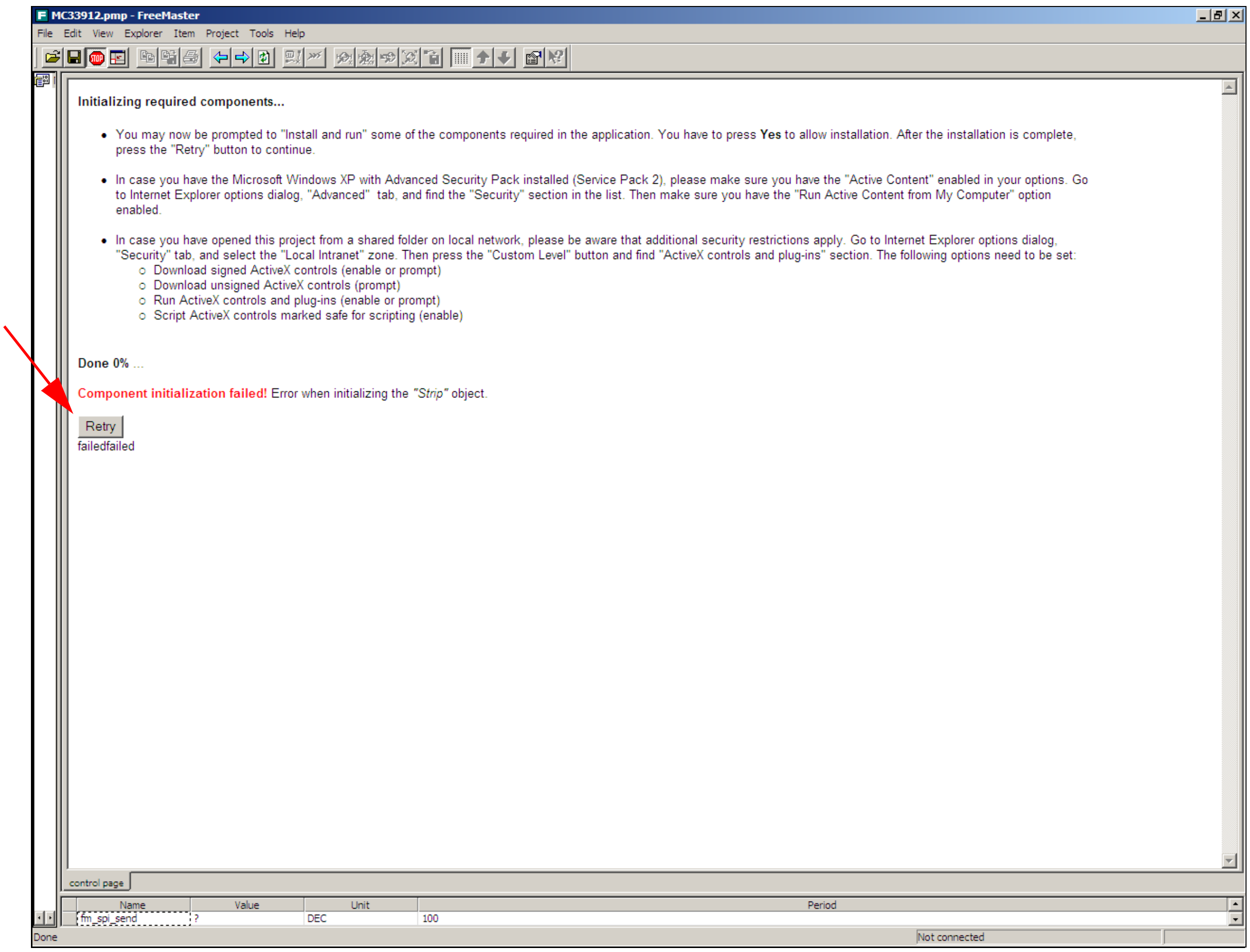


Figure 2-5. Graphical User Interface Initialization Screen

3. Default serial port setting is COM1/9600Bd. To choose a different setting, select from the main menu Project -> Options ->Comm.

4. Choose the **FreeMASTER** button to open the FreeMASTER control page; see Chapter 3, FreeMASTER Control Page.
   Choose the **LIN Master** button to open the control page with LIN communication; see Chapter 4, LIN Master Control Page.
   Choose the **User Helpful** button to open the page with a user friendly environment and FreeMASTER communication; see Chapter 5, User Helpful Control Page

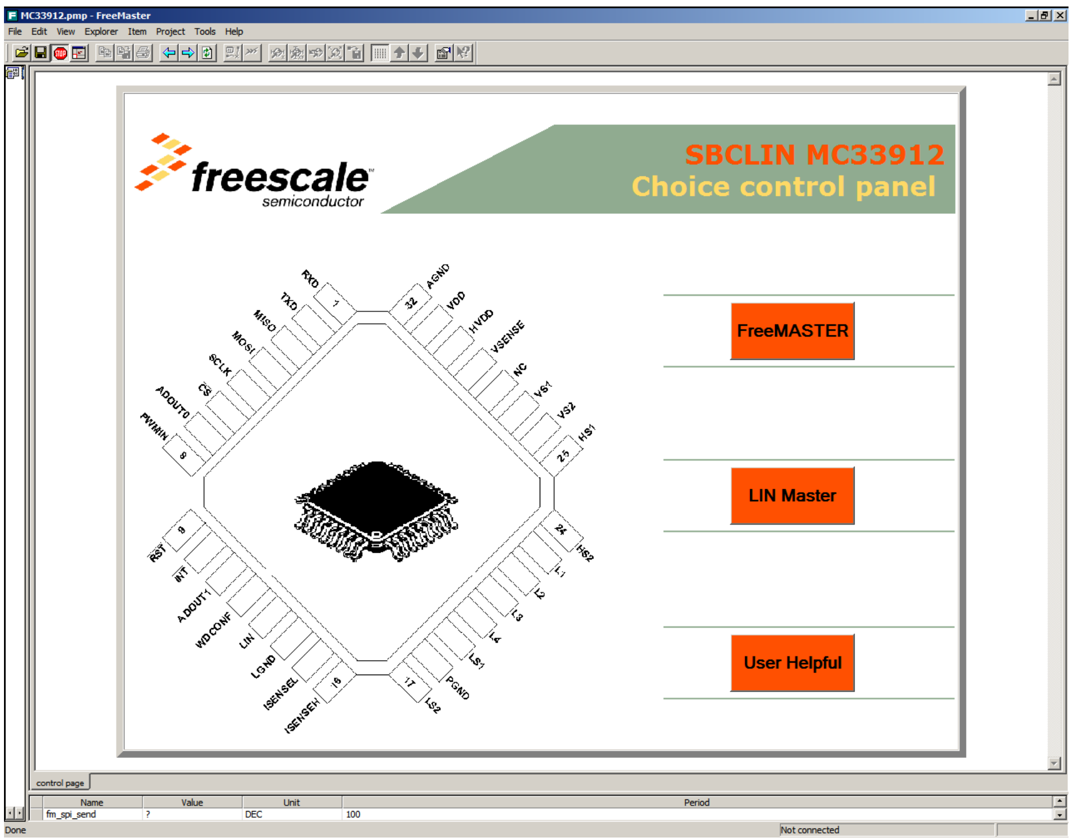


Figure 2-6. Graphical User Interface Welcome Screen

Chapter 3 FreeMASTER Control Page

3.1 Overview

The section describes basic guide to controlling the KIT33912EVME board by the FreeMASTER control page.

- *3.2, Setup Instruction*
- *3.3, Description of the Control GUI*
- *3.4, SPI Registers Control/Status Array*
- *3.5, Pulse Width Modulation*
- *3.6, Watchdog Service*
- *3.7, Wake-up Sources*

3.2 Setup Instruction

Before switching the power on, place the communication jumpers JP202 (TxD), JP203 (RxD) in position 1-2, short the supply jumpers JP7 (VDD), JP8 (VS2) and JP9 (VS1), and than place the control jumpers JP10 - JP16, JP19 and JP20 in position 1-2. To disable the LIN physical interface MCZ33661EF device, place jumper JP204.

3.3 Description of the Control GUI

The following paragraphs describe the indicators and the controls, on the GUI.

- An indicator is a graphical element displaying information (e.g., the status bit **VMS**).
- A control is a graphical element which displays information and allows altering the state of the element (e.g., the **SLEEP** button).

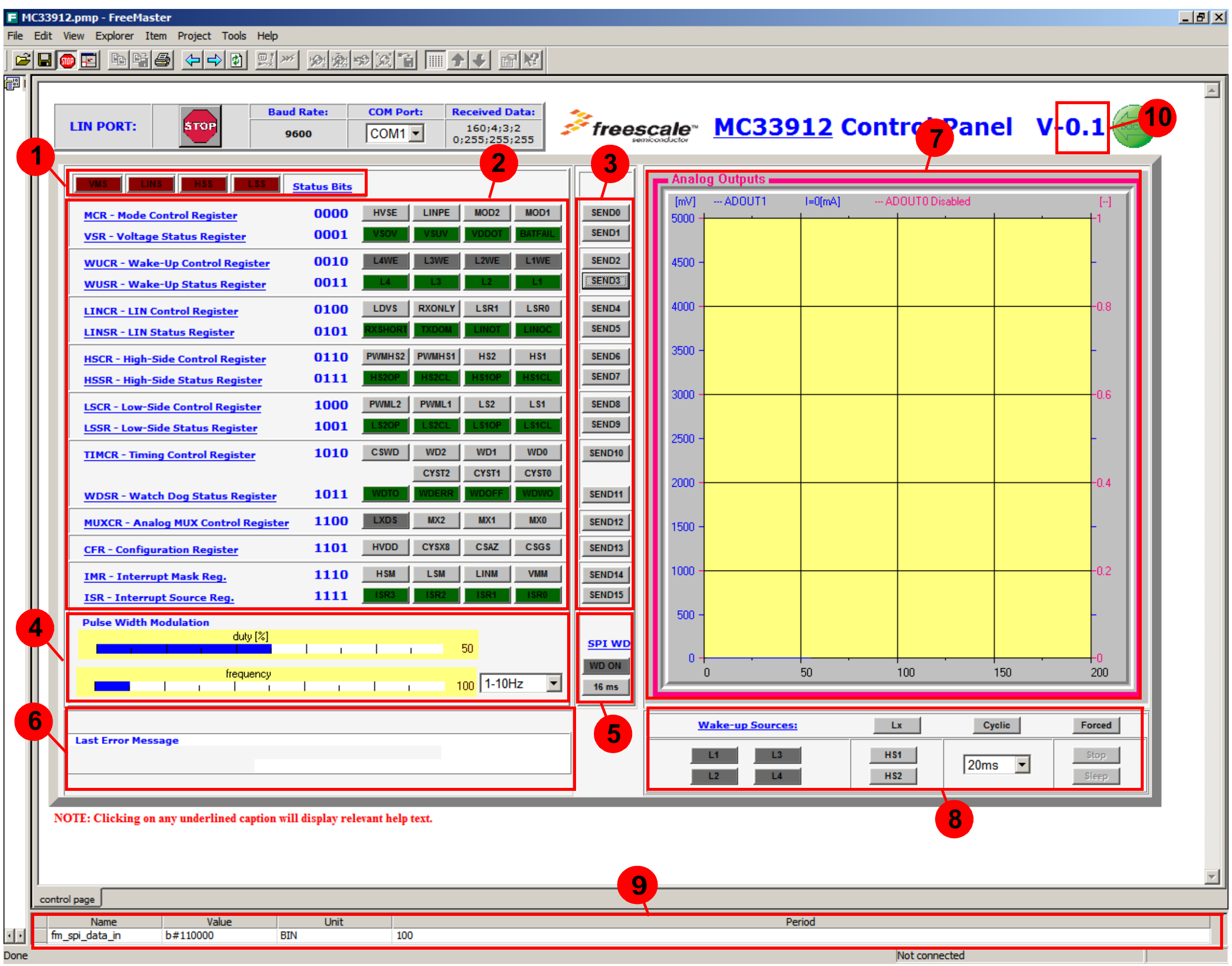


Figure 3-1. Description of Graphical User Interface - FreeMASTER Control

Table 3-1. Generic Control FreeMASTER Page Description

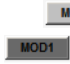
Item	Name	Description
1.	System Status Register	The System Status Register is always transferred with every SPI transmission and gives a quick system status overview. It summarizes the status of the Voltage Status Register VSR, LIN Status Register LINSR, High-Side Status Register HSSR and the Low-Side Status Register LSSR.
2.	SPI Registers	The SPI Registers Control/Status Array allows direct control of the individual bits of the SBCLIN SPI registers and indicates their status.
3.	Buttons SENDx	Clicking a **SENDx** button will generate the command send SPI data. The SPI data read back from the device is displayed in the Status Array.
4.	Pulse Width Modulation	The blue bars serve to change the frequency and duty of the PWM. Parameters are changed when a user drags and releases a mouse button. Frequency range of the Pulse Width Modulation is selectable by a drop-down menu.
5.	SPI Watchdog	These two buttons show the SPI Watchdog clearing time and offer a Watchdog switch off.
6.	Last Error Message	If an error occurs in communication, then the error message is shown. If the user chooses one of the wake-up capabilities and the wake-up event isn't configured correctly, then the user is given notice on what to change to configure the wake-up properly. If a reset condition is detected, then the user is warned and the control page reverts to the default state.
7.	Graph	The Graph indicator shows analogue voltage from the ADOUT0 and ADOUT1 outputs. The blue line indicates the voltage on the current sense input in mV, and the blue label calculates the current through the resistor connected to these inputs. A second line shows the variable in accordance with the Multiplexer Control Register MUXCR. The actual source of the voltage is displayed in the red caption of the Graph, and the appropriate axis is readjusted to the actual units.
8.	Wake-up Sources	This section provides three wake-up sources and helps the user to correctly set-up one of the wake-up events.
9.	Variable Watch Pane	The Variable Watch Pane displays the variables assigned to the watch. It displays the current variable values and allows the user to change them (if enabled in the variable definition). See FMASTXXX.pdf document available at http://www.freescale.com.
10.	Back Button	Clicking the Back button will open the Welcome screen.

3.4 SPI Registers Control/Status Array

The main control area is the Control/Status Array, which allows the control and read of bits in the SBCLIN SPI registers. There are two categories of components: controllers (write-only bits) and indicators (read-only bits). The bit categories and conditions are differentiate by color. See Figure 3-2 and text below.

Clicking on a control data change button will update the SPI register value if the appropriate SENDx button is pressed. Value of the bit is indicated by color.

— bit is not set 

— bit is set 

Data received from the device is represented in the indicator array. If a **SENDx** button is clicked, then the control register is configured. The status register is located at the same address, and updates the indicator array. Value of the bits are indicated by color, as follows.

After clicking the **SENDx** button in a row of indicators a command is processed to read the status bits only. Value of the bits are indicated by color.

— bit is not set 

— bit is set 

The SPI Registers Control/Status Array allows direct access individual bits. The Control/Status Array displays the names of the SBCLIN SPI registers. Clicking on any underlined register name opens a window with help text relating to that register.

If the MC33912 device generates an interrupt, then a read of the Interrupt Source Register is automatic and its value is displayed on the **ISR** register indicators row, so that the user recognizes the interrupt occurring.

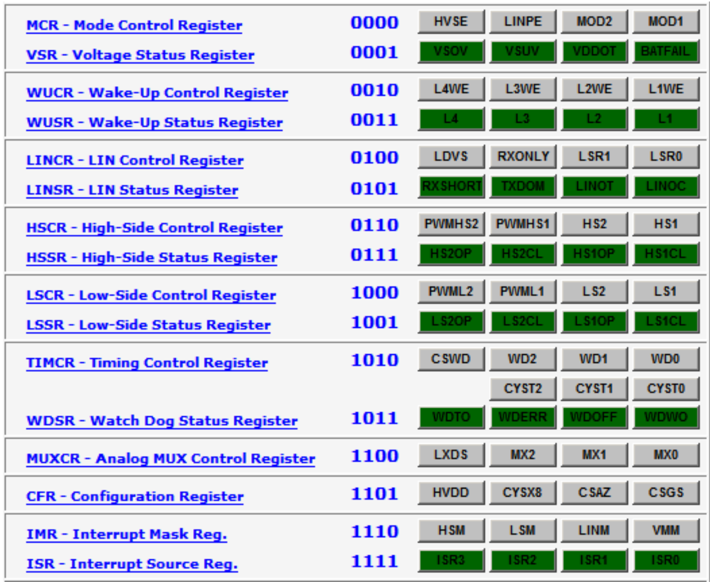


Figure 3-2. Control/Status Array

3.5 Pulse Width Modulation

The Pulse Width Modulation serves to directly control the High Side and the Low Side Switch Drivers. The operating pulses have an adjustable frequency from 1Hz to 1MHz. In the highest frequencies adjustment isn't precise. A frequency change is possible by dragging the blue frequency bar, with the range of the bar selectable by a drop down menu. Each following range has a scale one order higher. Width of the duty cycle is adjustable by percentage via the blue duty blue bar, with a minimum step five percent adjustment. For a zero percentage the Pulse Width Modulation terminal is permanently low, and for a hundred percent the terminal is high. See Figure 3-3.

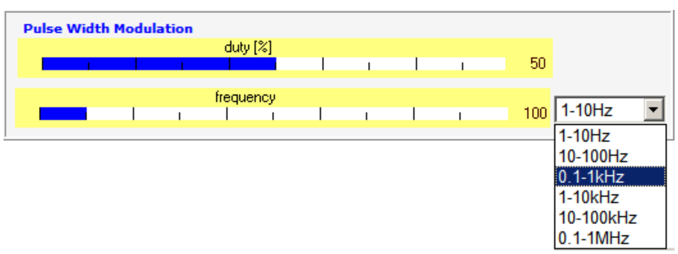


Figure 3-3. Pulse Width Modulation Control Panel

3.6 Watchdog Service

In the SBCLIN device a watchdog pin is included, which is a configuration terminal for the internal watchdog. When the pin is directly connected to the ground (place JP1, JP2), this terminal disables the watchdog function. When this terminal is left open (move JP2), the watchdog period is fixed to its default value of 150ms. If the on-board resistor is connected (place JP2, move JP1) to this terminal, then it configures the watchdog period to 16ms.

If the on-board watchdog resistor is connected and the GUI is started or a reset condition passes watchdog servicing is activated automatically; see Figure 3-4.

The user can extend the watchdog clear process by selecting the **WDx** bits in the Timing Control Register, then the button labelled **16 ms** redisplays with the relevant clearing time. This configuration is valid only if the windowing watchdog is active.

The user can stop the watchdog servicing by clicking the button labelled **16 ms**, which means the watchdog counter is not properly cleared while it's window is open for clears, and the MC33912 device will reset the microcontroller. The same sequence of events will occur, if the user disables the watchdog service by clicking the button labelled **WD ON**.

When there is a direct connection of the watchdog terminal to the ground, the watchdog service is disabled automatically.

Figure 3-4. Watchdog Control Buttons

3.7 Wake-up Sources

The MC33912 device offers two low power modes, Stop and Sleep, with various wake-up capabilities. There are three wake-up events, which require setup before entrance to the low power modes; see Figure 3-5. To set a wake-up from the wake-up inputs L1, L2, L3, L4 without cyclic sensing, select button **Lx**. To set a cyclic sense timer and wake-up inputs L1, L2, L3, L4, press button **Cyclic**. To set a forced wake-up, choose **Forced**.

After clicking one of the wake buttons, all unnecessary control units are disabled and the low power buttons **Stop** and **Sleep** are enabled. If the user enters the wrong wake-up settings and places the mouse over the low power buttons, the invalid cell will be red and a help hint will be displayed in the Last Error Message panel.

A forced wake-up requires the High Side Switch outputs disabled and a predetermined low power period. This period is selectable in a drop down menu. On first use, the period value must be changed to a setting corresponding to bits in the TIMCR register.

Before entering low power modes with cyclic sensing, the following setup has to be performed. Select Lx inputs, enable HSx outputs and predetermine the cyclic sense period.

In order to select and activate direct wake-up from the Lx inputs, those inputs must be enabled.

When all the required wake-up set-up has been done and the user presses the **Stop** or **Sleep** buttons, the status of the important registers (WUCR, HSCR, TIMCR, CFR, MCR) will be send to the device. The device will then enter one of the low power mode.

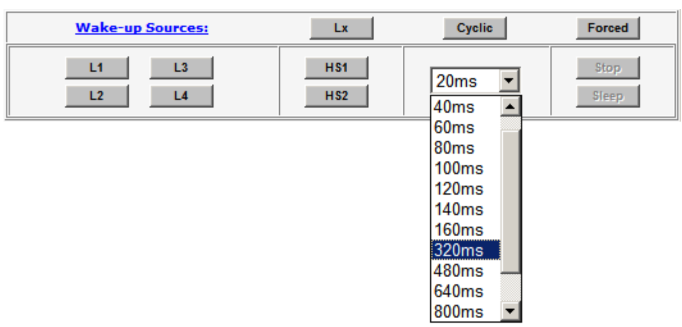


Figure 3-5. Wake-up Configuration Panel

Chapter 4 LIN Master Control Page

4.1 Overview

This section shows control of the KIT33912EVME board from the control page through LIN communication protocol.

- 4.2, *Setup Instruction*
- 4.3, *Description of the Control GUI*
- 4.4, *LIN Port Control*

4.2 Setup Instruction

Before switching the power on, place the supply jumpers JP7 (VDD), JP8 (VS2) and JP9 (VS1), then place the control jumpers JP10 - JP16, JP19 and JP20 in position 1-2. For LIN communication, place jumpers JP202 (TxD), JP203 (RxD) in position 2-3. Route the LIN signal to the MC33912 device by placing jumper JP205 and connect the MC33912 to the MCU through the SCI jumpers JP17, JP18 in position 1-2. To enable the LIN physical interface MCZ33661EF device, move jumper JP204.

4.3 Description of the Control GUI

The LIN control page has the same functions and controls as the FreeMASTER control page (see 3, *FreeMASTER Control Page*), except for the LIN port control section; see Figure 4-1.

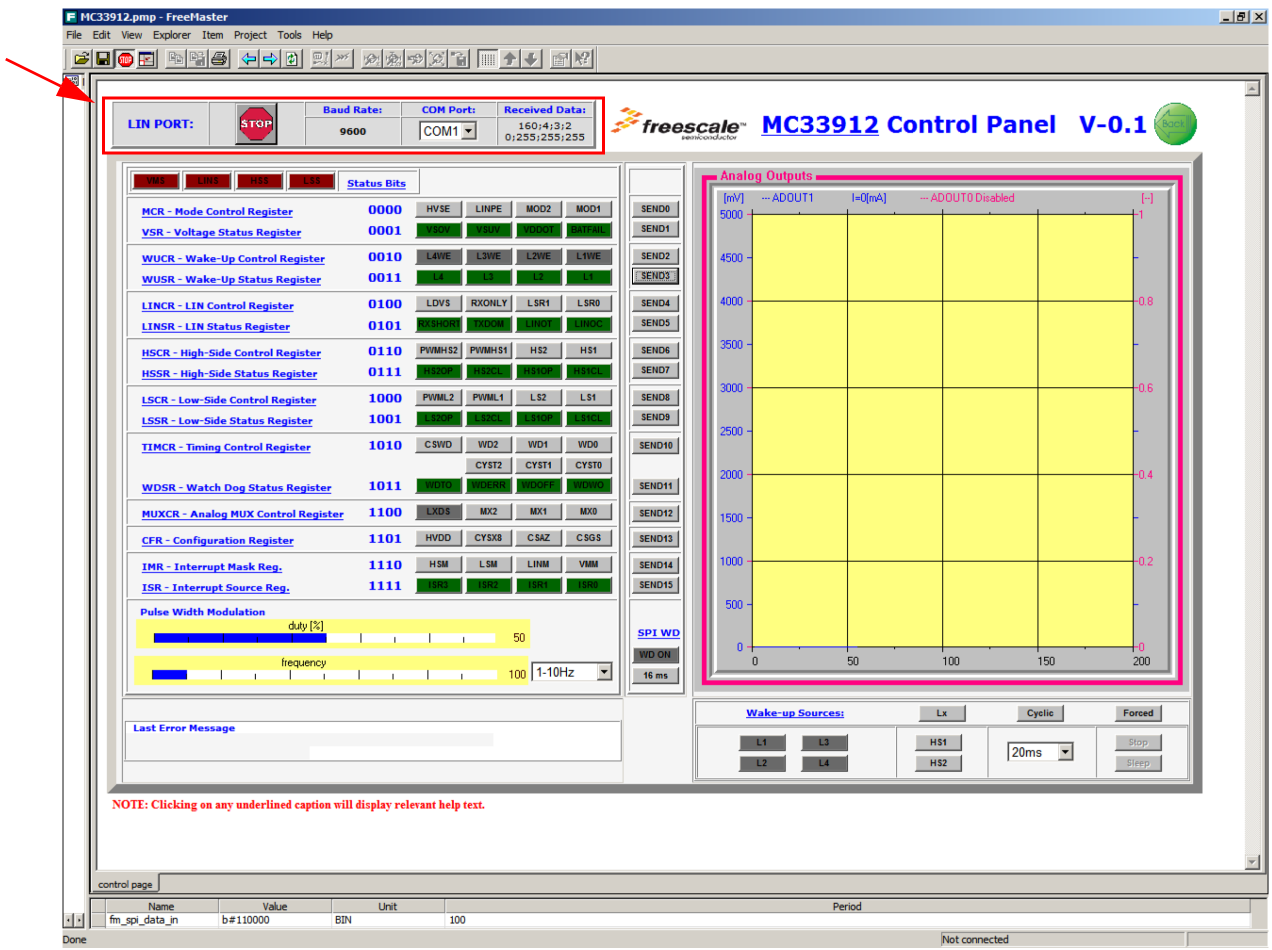


Figure 4-1. Description of Graphical User Interface - LIN Control

4.4 LIN Port Control

KIT33912EVME board includes a LIN communication channel. For communication through LIN, stop the FreeMASTER communication port. Select COMx port in section LIN PORT:, then start communication by clicking the **STOP** button, see; Figure 4-2. Transmission speed is set by default to 9600 Baud.

Figure 4-2. LIN Port Control Panel

If an order is carried out to enter one of the low power modes, LIN Master communication will be suspended. This is due to one simple reason, which is the possibility of a wake-up from a low power mode by a LIN dominant wake-up event. If this wake-up event occurs, the slave node sends to the master node the wake-up message frame, and the communication is restored. LIN Master communication is restored again by clicking any **SENDx** button.

Chapter 5 User Helpful Control Page

5.1 Overview

This section provides an overview on how to control the KIT33912EVME board from the User Helpful control page. The aim of this control environment is simplify control of the evaluation module.

- *5.2, Setup Instruction*
- *5.3, Description of User Helpful Page*
- *5.4, SPI Status Control Registers Tab Panels*
- *5.5, System Status Register and SPI Status Registers*
- *5.6, Sequence Pane*

5.2 Setup Instruction

Before switching the power on, place the communication jumpers JP202 (TxD), JP203 (RxD) in position 1-2, short the supply jumpers JP7 (VDD), JP8 (VS2) and JP9 (VS1), and then place the control jumpers JP10 - JP16, JP19 and JP20 in position 1-2. To disable the LIN physical interface MCZ33661EF device, place jumper JP204.

5.3 Description of User Helpful Page

The User Helpful control page has some of the same panels as the FreeMASTER or LIN Master control pages, such as Graph, Pulse Width Modulation, Wake-up Sources, SPI Watchdog, Last Error Message and System Status Register. The array of SPI Status/Control Registers is rearranged between tab panels, where each tab shows the control or presents possibilities for the relevant register. The status registers are included in the tab panels too, but for a convenient overview of the device state, they are presented again and together in one different block. The control panel contains one additional function, which is the ability to create a sequence of control commands.

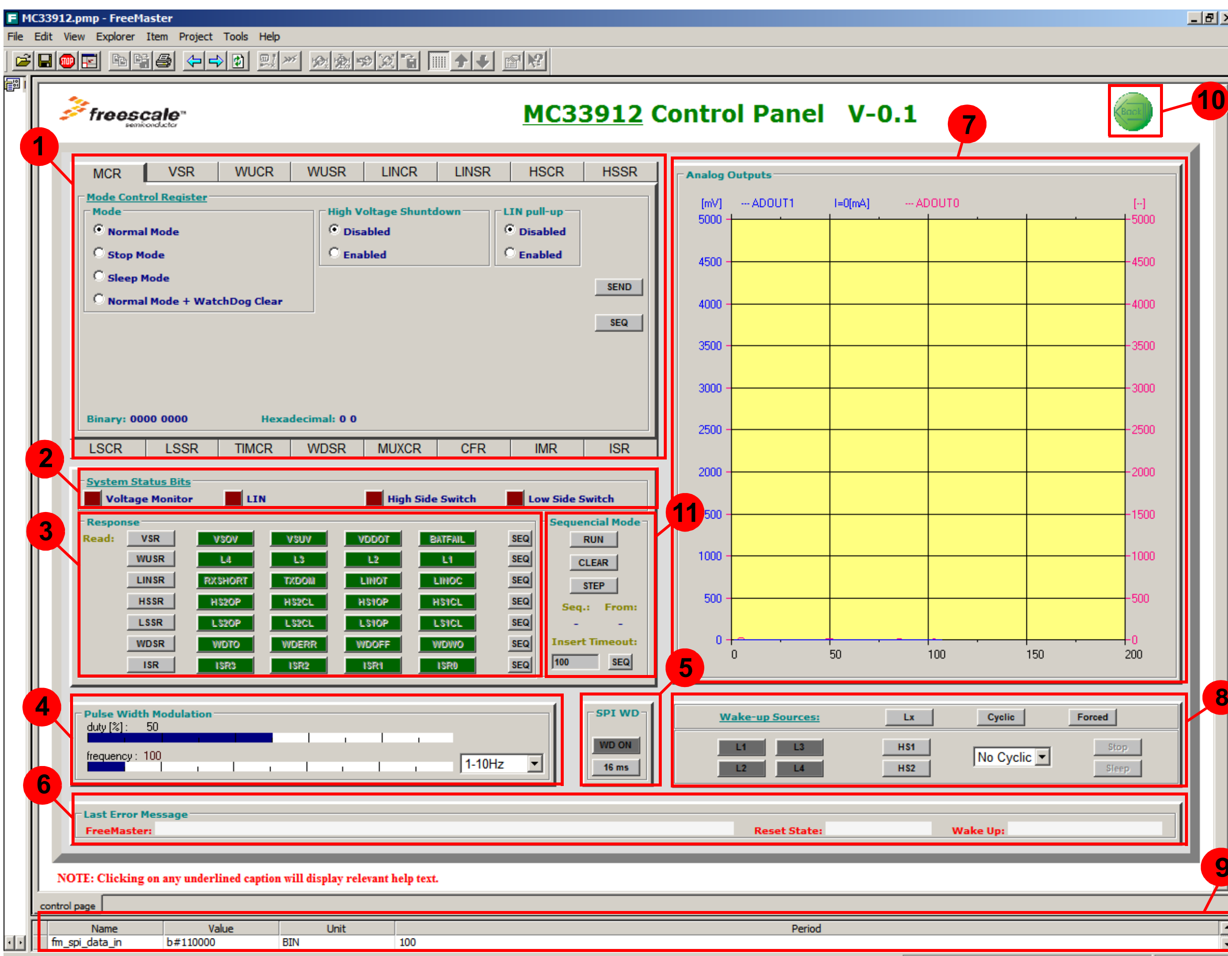


Figure 5-1. GUI for SBCLIN - User Helpful Control

Table 5-1. Generic Control User Helpful Page Description

Item	Name	Description
1.	SPI Registers Tab Panel	The Control SPI Registers tabs give a basic view on the functionality of the MC33912 device and allows the use of these functions. The Status SPI registers tabs show possible states of the MC33912 device and indicate their status.
2.	System Status Register	The System Status Register is always transferred with every SPI transmission and gives a quick system status overview. It summarizes the status of the Voltage Status Register VSR, LIN Status Register LINSR, High-Side Status Register HSSR and the Low-Side Status Register LSSR.
3.	SPI Status Registers	System status registers indicate the state of MC33912 device.
4.	Pulse Width Modulation	The blue bars serve to change the frequency and duty of the PWM. Parameters are changed when a user drags and releases a mouse button. Frequency range of the Pulse Width Modulation is selectable by a drop-down menu.
5.	SPI Watchdog	These two buttons show the SPI Watchdog clear time and offer a switch off for the Watchdog clear routine.
6.	Last Error Message	If in communication an error occurs, then the error message is shown in the first white area. If the user chooses one of the wake-up capabilities and the wake-up event isn't configured correctly, then user is given a notice on what to change to configure the wake-up properly. This message is shown in second white area. If a reset condition is detected, then the user is warned and the control page reverts to the default state and the reset flag is displayed in the third white area.
7.	Graph	The Graph indicator shows analogue voltage from the ADOUT0 and ADOUT1 outputs. The blue line indicates the voltage on the current sense input in mV, and the blue label calculates the current through the resistor connected to these inputs. A second line shows the variable in accordance with the Multiplexer Control Register MUXCR. The actual source of the voltage is displayed in the red caption of the Graph, and the appropriate axis is readjusted to the actual units.
8.	Wake-up Sources	The section shows three wake-up sources and helps the user to correctly set-up one of the wake-up events.
9.	Variable Watch Pane	The Variable Watch Pane displays the variables assigned to the watch. It displays the current variable values and allows user to change them (if enabled in the variable definition). See FMASTXXX.pdf document available at http://www.freescale.com.
10.	Back Button	Clicking the Back button will open the Welcome screen.
11.	Sequence Pane	The Sequence Pane allows control in executing the command sequence.

5.4 SPI Status Control Registers Tab Panels

Each SPI Status or Control Register is arranged in its own panel. Each of the sixteen panels display the possibilities for the appropriate register. Access to the other tab panels is possible by clicking on the abbreviated register names, located above and below the panel.

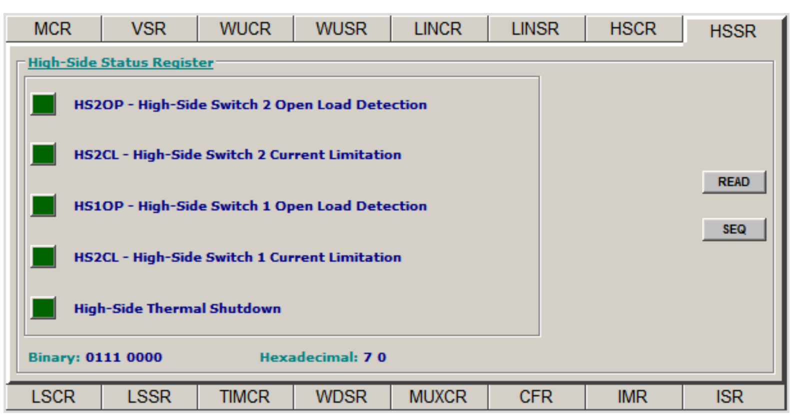


Figure 5-2. Status Register Tab Panel

Each Status Register panel displays underlined name of the relevant register. By clicking on that label, a new window with help text opens up. In the body place of the status panel is description of the states of the MC33912 device. On each state description row is an indicator element showing if the state occurred or not. Indication is given by color;

— state didn't occur

— state did occur

Address and actual value of status register is displayed at bottom of the panel. There are two representations of the register value, in binary and hexadecimal. The final elements to be found on the panel are the **READ** and **SEQ** buttons; see Figure 5-2. After clicking **READ** button, a command is executed to read the appropriate register and this value is shown on the panel. The **SEQ** button saves the read command to a sequence variable, more in Section 5.6, *Sequence Pane*.

Control panels include underlined names with help hints too. Each panel is divided into logical fields with titles. Each titles indicates the possible control elements located in that field. On the control panels there are three types of control elements. The most frequent components are radio buttons, which allow only one selection in the given field. Control bits, which have only one function and have no relation to other control bits, are handled by check boxes. The final control component is the drop down menu, to select the appropriate register bit combinations; see the following pictures and Figure 5-3.

— elements aren't selected

— elements are chosen

— drop down menu 20ms

After each change in the control elements, the binary and hexadecimal values of the potential command are updated and shown at the bottom of the panel. This value is transferred to the device by clicking the **SEND** button. The **SEQ** button saves the actual panel configuration (an exact byte representation of the register) to a sequence variable; more in Section 5.6, *Sequence Pane*.

The Timing Control Register allows the configuration of the cyclic sensing periods, together with the CYSX8 bit in the Configuration Register. If this bit has changed after clicking **SEND** button in the Timing Control Register panel, the actual value of the Configuration Register is transmitted to the MC33912 device too. Therefore, the CYSX8 bit isn't in the Configuration Register panel, because it is handled from the Timing Control Register panel.

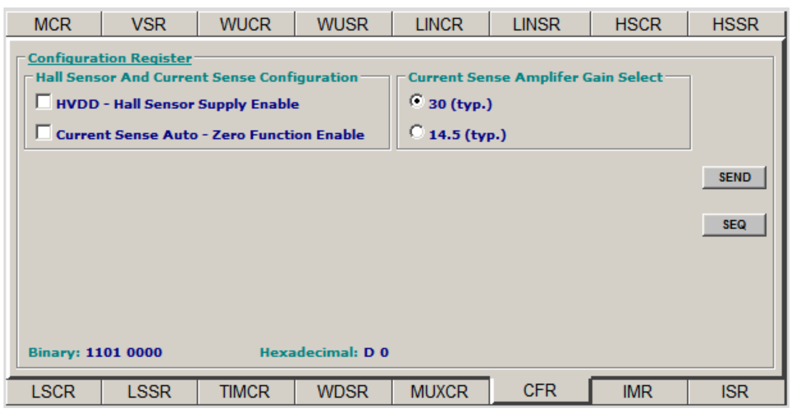


Figure 5-3. Control Register Tab Panel

When communication starts, all the SPI Control Registers are set to their default values. This also applies after a reset.

5.5 System Status Register and SPI Status Registers

The System Status Register gives basic information on the device state, described in Table 5-1. In the System Status Bits panel are indicators with a description. They have two states:

— bit isn't set

— bit is set

The SPI Status Registers field groups together all the device status registers (except the System Status Register). Clicking on active buttons displaying the abbreviated name of an appropriate register, allows a read of that register. The returned register value is displayed on inactive green buttons in line with the selected button. The value is also represented on the relevant tab panel. **SEQ** buttons save the read command to a sequence variable; more in Section 5.6, *Sequence Pane*.

— indicator bit isn't set VSUV

— indicator bit is set VSUV

All Status Register are updated after a reset or power up. If the MC33912 device generates an interrupt, then a read of the Interrupt Source Register is automatic and its value is displayed on the Status Register array, and also on the **ISR** tab panel, so that the user recognizes the interrupt occurring.

Figure 5-4. SPI System Status Registers Field

5.6 Sequence Pane

This additional function of graphical user interface allows the user to create a sequence of control commands and in a future step to execute these commands; see Figure 5-5 and text below.

It's possible to create each step of the sequence by clicking the **SEQ** button of the appropriate register. After clicking the SPI command is saved to a variable and on the Sequence pane a command quantity counter is incremented. Behind each sequence, an automatically included 100 ms timeout gives a proper reading of the response from the MC33912 device. The timeout can be extended by entering the value of that timeout in milliseconds and clicking the continuous **SEQ** button, that increments the quantity counter too.

Execution of the sequence is done by clicking the **RUN** button. Throughout the flow of control, the number of the actual command is displayed in the pane and the state of the control and status elements are changed or updated. The **RUN** button changes itself to a **STOP** button, with which the execution of the sequence can be halted. A further click will continue interrupted sequence.

For checking the created sequence there is a **STEP** button, with which every click will execute only one command, the counter of the actual sequence being incremented. The included timeouts are skipped and the counter of the actual sequence jumps to the next control command.

When the sequence reaches the end, the user can repeat it by clicking the **RUN** or **STEP** buttons. The last button on the Sequence pane is **CLEAR**, and clicking it will erase the sequence variable and the user can create a new sequence of commands.

Figure 5-5. Sequence Pane

Other blocks such as Pulse Width Modulation, SPI Watchdog, Last Error Message, Graph, Wake-up Sources and Variable Watch Pane are described in Chapter 3, FreeMASTER Control Page, and their functionality is similar.

Chapter 6 Embedded Slave Software

6.1 Overview

This chapter describes the embedded generic Slave software. It gives basic information on how to use the embedded software features successfully. The following sections provide the basic information:

- *6.2, Generic Control Slave Software Description*
- *6.2.1, Generic Control PDU Structure*
- *6.2.2, CMD data interpretation*
- *6.3, MCU Reprogram*

6.2 Generic Control Slave Software Description

Generic Control of the MC33912 device is through the SPI channel. Control data is transported to the MCU by FreeMASTER protocol or LIN protocol, as well as data from the A/D converter and MCU memory.

The units that are transported in a LIN diagnostic frame are called PDU's (Packet Data Unit). A PDU used for Generic Control of the device is a Single Frame message based on the User Defined Diagnostics specification. The diagnostic format sends a fixed master request frame with eight data bytes provided. This is used to issue specific Generic Control fixed frames.

Messages issued by the master are called requests, and messages issued by the slave are called responses. Requests are always sent in master request frames and responses are always sent in slave response frames. The Master request frame has a fixed identifier, 60 (0x3C) and a fixed size (8 bytes). The Slave response frame has a fixed identifier, 61 (0x3D) and a fixed size (8 bytes).

The meaning of each byte in a PDU is defined in the following sections.

6.2.1 Generic Control PDU Structure

The Generic Control PDU structure is described in this section; see Figure 6-1.

Request	**NAD**	**CMD**	**M0**	**M1**	**M2**	**M3**	**M4**	**M5**
Response	**NAD**	**CMD**	**S0**	**S1**	**S2**	**S3**	**S4**	**S5**

The left byte (NAD) is sent first

Figure 6-1. Generic Control PDU Structure

NAD

NAD is the address of the slave node being addressed by a request. NAD is also used to indicate the source of the response.

Generic Control is uses Free NAD values, meaning that the NAD value must be selected from the range 128 (0x80) - 255 (0xFF), and not to be interpreted as a diagnostic. For the KIT33912EVME NAD - 160 (0xA0) is used.

CMD

CMD is a command being sent by the master in a request to be interpreted by the slave. CMD is also used to interpret the status data in a response.

CMD code:

0 = SPI - read and write SPI data registers through the SPI channel
1 = PWM - write a value to the MCU pulse width modulation registers
2 = ADC - read data from the A/D converter the MCU registers
3 = TOOLS_TO_MCU - special commands to change the MCU state and activity
4 = TOOLS_TO_PC - special commands to read the MCU state and activity

M0 - M5

Data being sent by the master in a request. The meaning of the data is related to the CMD. It will be described in the next section.

S0 - S5

Status data sent in a response by an addressed slave. The meaning of the data is related to the CMD. It will be described in the next section.

If a PDU is not completely filled, the unused bytes are filled with ones, meaning that their value will be 255 (0xFF). This is necessary since a user defined diagnostic frame is always eight bytes long.

6.2.2 CMD data interpretation

SPI

The SPI command will perform the "*ReadWriteSPI8bit(addr + data)*" function of the embedded software, returning the SPI status data structure. This function waits for the LIN response frame.

This command only uses the M0 and S0 bytes. The message structure is described as follows:

Request
- M0 - SPI request
- M1 - 0xFF
- M2 - 0xFF
- M3 - 0xFF
- M4 - 0xFF
- M5 - 0xFF

Response
- S0 - SPI response
- S1 - 0xFF
- S2 - 0xFF
- S3 - 0xFF
- S4 - 0xFF
- S5 - 0xFF

PWM

PWM commands transmit data to the pulse width modulation registers in the MCU. To adjust the pulse width modulation requires the frequency, duty cycle and range of the MCU clock prescaler. Values of the frequency and duty cycle are separated into two bytes.

This command uses bytes M0, M1, M2, M3, M4. The message structure is described as follows:

Request
- M0 - lower part of value appointed to the frequency register
- M1 - upper part of value appointed to the frequency register
- M2 - lower part of value appointed to the duty register
- M3 - upper part of value appointed to the duty register
- M4 - value of the range appointed to change the MCU clock prescaler
- M5 - 0xFF

Response - No response.

ADC

ADC commands announce a data read from the A/D converter. After receiving a frame with the CMD value equal to ADC, the response frame waits to transmit the value from the A/D converter register, which is separated into two bytes.

This command uses bytes S0, S1, S2 and S3. The message structure is described as follows:

Request
- M0 - 0xFF
- M1 - 0xFF
- M2 - 0xFF
- M3 - 0xFF
- M4 - 0xFF
- M5 - 0xFF

Response
- S0 - lower part of register value from the first converter
- S1 - upper part of register value from the first converter
- S2 - lower part of register value from the second converter
- S3 - upper part of register value from the second converter
- S4 - 0xFF
- S5 - 0xFF

TOOLS_TO_MCU

TOOLS_TO_MCU is a request LIN frame for watchdog, interrupt and reset information.

For these commands bytes M0, M1, M2 are used. The message structure is described as follows:

Request
- M0 - stop watchdog window clear
- M1 - accept reset
- M2 - accept interrupt
- M3 - 0xFF
- M4 - 0xFF
- M5 - 0xFF

Response - No response

TOOLS_TO_PC

TOOLS_TO_PC is a response LIN frame with the watchdog, interrupt and reset information.

For these commands bytes S0, S1, S2 are used. The message structure is described as follows:

Request
- M0 - 0xFF
- M1 - 0xFF
- M2 - 0xFF
- M3 - 0xFF
- M4 - 0xFF
- M5 - 0xFF

Response
- S0 - hand over the MC33912 watchdog pin configuration
- S1 - hand over the reset occur
- S2 - hand over the interrupt state
- S3 - 0xFF
- S4 - 0xFF
- S5 - 0xFF

6.3 MCU Reprogram

In case of loose the MCU embedded code, it is possibility to reprogram MCU flash memory from file on CD**...\GraphicalUserInterface\Project.abs.s19**.

Chapter 7 FreeMASTER ActiveX Object

7.1 Overview

All HTML pages used in FreeMASTER are rendered using the standard Microsoft Internet Explorer component. The advantage of using the HTML and Internet Explorer component is that it fully supports scripting languages and enables scripts to embed and access third-party ActiveX controls. The FreeMASTER application itself implements a (non-visual) ActiveX component to let script-based code access and control the target board application.

The chapter gives basic description of the ActiveX FreeMASTER object.

- *7.2, Quick Reference*
- *7.2.1, WriteVariable Function Example*
- *7.2.2, ReadVariable Function Example*

7.2 Quick Reference

The FreeMASTER object is registered in the system registry during each start of the FreeMASTER application. Its class ID (CLSID) is

{48A185F1-FFDB-11D3-80E3-00C04F176153}

The registry name is "MCB.PCM.1"; version independent name is "MCB.PCM".

FreeMASTER functions can be called from any HTML code via the FreeMASTER ActiveX control. Insert the FreeMASTER ActiveX control into your HTML code by the Class ID number (see the example below) and set the dimensions (height and width) to zero to make the object invisible.

<object name="PCMaster" width="0" height="0" classid="clsid:48A185F1-FFDB-11D3-80E3-00C04F176153">

The SBCLIN GUI uses only three functions of FreeMASTER ActiveX control:

- WriteVariable writes a value to a FreeMASTER-defined variable
  - This function writes a value to a FreeMASTER variable.
- ReadVariable reads a value from a FreeMASTER-defined variable
  - This function reads a value from a FreeMASTER-defined variable.

Short examples on how the functions are used in the GUI are in the following sections. For more details on the FreeMASTER functions, refer to the “FreeMASTER for Embedded application” user manual.

7.2.1 WriteVariable Function Example

```
// write to FreeMASTER variable, display error box in case of error

function write_fmaster_variable(varname, value){

         var succ = pcm.WriteVariable(varname, value);

         if (!succ) set_fmaster_err();

}
```

7.2.2 ReadVariable Function Example

```
// read FreeMASTER variable, display error box in case of error on success, value is returned
function read_fmaster_variable(varname){
var value
    var succ = pcm.ReadVariable(varname);
        if(succ){
                  value = pcm.LastVariable_vValue;
         }else{
                  set_fmaster_err();
         }
    return value;
}
```

Chapter 8 ActiveX LIN Master Object

8.1 Overview

The chapter gives a basic description of the ActiveX LIN Master Object. This object is able to perform LIN Master tasks on a PC and can be used for LIN network control from an HTML page. The following section provides the basic information:

- *7.2, Quick Reference*

8.2 Quick Reference

An ActiveX control is an object that supports the LIN Master functionality. It is delivered as the cabinet (.cab) file "flinbox.cab". An ActiveX control is identified by object in an HTML file. If the control has been stored in a ".cab" file, the object must include a CODEBASE attribute that specifies the URL for this ".cab" file. The following excerpt from a sample HTML file demonstrates how the object and the CODEBASE attribute are used.

```
<OBJECT id="lin" name="lin" height="1" width="1"

classid="clsid:A6D06BA3-A530-45FF-A02F-48A9B82BB25B"

codebase="flinbox.cab#Version=1,2,0,4">

<PARAM name="BaudRate" value="9600">

<PARAM name="PortName" value="COM1">

</OBJECT>
```

The FLINBOX.LinMaster object has 6 methods and 3 properties:

- **bool OpenPort([vPortName, vBaudRate]);**
  - — Open RS232 port. It must be called before any other function. Returns true when successful.
    - – **vPortName** (string) - RS232 port to open. Optional, default COM1.
    - – **vBaudRate** (numeric) - Communication speed. Optional, default 9600.

- **void ClosePort()**;
  - — Close RS232 port

- **bool SendMessage(vIdent, vData [, vDataLen])**;
  - — Send LIN message and return true if succeeded.
    - – **vIdent** (numeric) - Message ID byte.
    - – **vData** (string/array) - Data to send, passed either as a VBScript array or a semicolon-delimited string.
    - – **vDataLen** (numeric) - Length of data to be sent. Optional, default 0 which forces calculation of the length from the vData parameter (do not use 0 for VBScript arrays, as they always contain one dummy item at the end).

- **string RecvMessage(vIdent, vExpectedDataLen [, vData])**;
  - — Requests and receives a LIN message. Function returns the number of bytes received. It returns 0 for checksum errors or timeouts.
    - – **vIdent** (numeric) - Message ID byte.
    - – **vExpectedDataLen** (numeric) - Length of data to be received. When set to 0, up to 255 bytes are received using the maximum timeout value for that size.
    - – **vData** (output array) - VBScript array of the returned data. Optional, use the **LastRecvData** property to get the data after this call.

- **bool Sleep();**
  - — Put the node the Sleep mode. In this mode, the port remains open and the master waits for the wakeup character (0x80, 0xC0 or 0x00). After this is received, the OnWakeup event is triggered. You can handle the OnWakeup event using the VBScript handler. The Sleep mode is aborted by a call to the Wakeup, SendMessage or RecvMessage functions. Even in the OnWakeup event, you remain in the Sleep mode until one of these three functions is called.

- **bool Wakeup();**
  — Abort the Sleep mode. The Sleep mode is aborted by a call to the Wakeup, SendMessage or RecvMessage functions.

- **bool SetRTS(vSet)**;
  — Sets RTS to a given state.
    – **vSet** (numeric) - if vSet is nonzero (true) then set RTS to +, otherwise -;

- **bool SetDTR(vSet)**;
  — Sets DTR to a given state.
    – **vSet** (numeric) - if vSet is nonzero (true) then set DTR to +, otherwise -;

- **property: array LastRecvData**;
  — This property retrieves the VBString array of the last data received by the last RecvMessage() call

- **property: numeric LastRecvDataLength**;
  — This property retrieves the number of bytes received by the last RecvMessage() call

- **property: string LastErrorMsg**;
  — This property retrieves the last error message.

Chapter 9 Interconnect Board Description

9.1 Overview

This chapter gives a basic description of the Interconnect Board. The following sections provide the basic information:

- *9.2, J1 RS232*
- *9.3, J201 Background Debug Mode (BDM)*
- *9.4, J3 External Control*
- *9.5, CON1 LIN Bus Connector*
- *9.6, CON2 HSx LSx Outputs*
- *9.7, CON3 Lx Inputs*
- *9.8, CON4 The MCU Input Output Pins*
- *9.9, CON5 ISENSE Inputs*

9.2 J1 RS232

The RS232 connector J1 allows connection to a PC. Table 9-1 shows the terminal definitions for the J1 connector.

Table 9-1. J1 Serial Peripheral Interface Terminal Definitions

Terminal	Terminal Name	Definition
1	Not used	Carrier detect.
2	TXD	Transmit data from MCU to PC.
3	RXD	Received data from PC to MCU.
4	Not used	Data terminal ready.
5	GND	Ground.
6	Not used	Data set ready, Carrier detect.
7	NC	No connect.
8	NC	No connect.
9	NC	No connect.

9.3 J201 Background Debug Mode (BDM)

The MCU contains a single wire background debug interface that supports in-circuit programming of on-chip nonvolatile memory and sophisticated non-intrusive debug capabilities. Table 9-2 shows the standard definition for a 6 pin connector.

Table 9-2. J201 Program Debug Terminal Definition

Terminal	Terminal Name	Definition
1	BKGD	Background debug.
2	GND	Ground.
3	NC	No connect.
4	$\overline{\text{RESET}}$	Reset.
5	NC	No connect.
6	VDD	+5V voltage supply.

9.4 J3 External Control

The following connector, see Table 9-3, is appointed as the external control of the MC33912 device. The choice between on-board control through the MCU, or external control, is made through Jumpers JP10 - JP20.

Table 9-3. External Control Pin Description

Terminal	Terminal Name	Definition
1	LINRXD	This terminal is the receiver output of the LIN interface which reports the state of the bus voltage to the MCU interface.
2	LINTXD	This terminal is the transmitter input of the LIN interface which controls the state.
3	MISO	SPI (Serial Peripheral Interface) data sent to the controller by the MC33912.
4	MOSI	SPI (Serial Peripheral Interface) data received by the MC33912.
5	SCLK	SPI (Serial Peripheral Interface) clock input.
6	$\overline{CS}$	SPI (Serial Peripheral Interface) control chip select input terminal.
7	PGND	Power ground.
8	PWMIN	High Sides and Low Sides Pulse Width Modulation Input.
9	$\overline{RST}$	Reset I/O terminal. This is bidirectional allowing a reset of the entire system. It is driven low when any internal reset source is asserted.
10	$\overline{IRQ}$	Interrupt output terminal, indicating wake-up events from Stop Mode or events from Normal and Normal Request Modes.
11	AGND	Analogue ground.
12	AGND	Analogue ground.
13	ADOUT1	Current sense analogue output.
14	ADOUT0	Analogue multiplexer output.

9.5 CON1 LIN Bus Connector

The LIN bus terminal provides a physical layer for single wire communication in automotive applications. The LIN terminal is optionally attached to the LIN physical interface device (place Jumper JP205). Table 9-4 shows the three pin connector standard definition.

Table 9-4. LIN Connector Pin Out

Terminal	Terminal Name	Definition
1	LGND	LIN ground.
2	VBAT	Supply voltage
3	LIN	LIN interconnect terminal

9.6 CON2 HSx LSx Outputs

The MC33912 device, including two 60mA High Side Switches and two 150mA Low Side Switches with output protection, are available for driving resistive and inductive loads. The interconnection of the MC33912 device with an output connector is shown in Table 9-5.

Table 9-5. CON2 Output Connector Pin Definitions

Terminal	Terminal Name	Definition
1	HS1	High Side Switch output HS1.
2	HS2	High Side Switch output HS2.
3	PGND	Power ground.
4	LS1	Low Side Switch output LS1.
5	LS2	Low Side Switch output LS1.
6	PGND	Power ground.

9.7 CON3 Lx Inputs

The MC33912 device has four high voltage inputs available for use in contact monitoring or as external wake-up inputs. The terminals can be used as high voltage analogue inputs. For the L2, L3, L4 inputs, 33k ohm series resistors are used between the connector and the device pins. L1 input is directly linked to the connector. Refer to Table 9-6.

Table 9-6. CON3 Output Input Terminal Definitions

Terminal	Terminal Name	Definition
1	PGND	Power ground.
2	L1	L1 terminal is a wake-up capable digital input. In addition, can be sensed as analogue via the analogue multiplexer.
3	L2	L2 terminal is a wake-up capable digital input. In addition, can be sensed as analogue via the analogue multiplexer.
4	L3	L3 terminal is a wake-up capable digital input. In addition, can be sensed analogue via the analogue multiplexer.
5	L4	L4 terminal is a wake-up capable digital input. In addition, can be sensed analogue via the analogue multiplexer.
6	PGND	Power ground.

9.8 CON4 The MCU Input Output Pins

Connector CON4 is linked to the MCU pins, not used in the application. The connector includes a Hall sensor supply terminal, ground and a +5V output supply terminal. See Table 9-7.

Table 9-7. MCU Pins not used in the Application

Terminal	Terminal Name	Definition
1	HVDD	Hall sensor switchable supply terminal.
2	PTD0	Not used in application.

Table 9-7. MCU Pins not used in the Application

Terminal	Terminal Name	Definition
3	PTD1	Not used in application.
4	PTD3	Not used in application.
5	PTD4	Not used in application.
6	PTD5	Not used in application.
7	NC	No connected.
8	PTB0	Not used in application.
9	PTB1	Not used in application.
10	NC	Not connected.
11	PTA0	Not used in application.
12	PTA3	Not used in application.
13	PTA4	Not used in application.
14	PTA5	Not used in application.
15	PTA6	Not used in application.
16	NC	Not connected.
17	VDD	+5V main regulator output terminal of the MC33912 device.
18	PGND	Power ground.

9.9 CON5 ISENSE Inputs

The ISENSEH and ISENSEL terminals are the input terminals to a ground compatible differential amplifier designed to be used to sense the voltage drop over a shunt resistor. ISENSEx pins are terminated in connector CON5; see Table 9-8.

Table 9-8. ISENSE Connector Pin Out

Terminal	Terminal Name	Definition
1	PGND	Power ground.
2	ISENSEH	Current sense differential (+) input.
3	PGND	Power ground.
4	ISENSEL	Current sense differential (-) input.

Chapter 10 Jumper Connection

10.1 Jumper Connection

The chapter gives a quick guide about the possible board configuration accomplish by the jumper selection.

Table 10-1. Jumper Connection

Name	Description
JP1	1-2 position: Disabled watchdog only if jumper JP2 is closed
	Floating: Watchdog enabled only if selected jumper JP2
JP2	1-2 position: Watchdog time out is 16ms, only if jumper JP1 is open
	Floating: Watchdog time out is 150ms
JP3	1-2 position: (ISENSEL) input is connected to ground
	Floating: (ISENSEL) is not connected to ground
JP4	1-2 position: (ISENSEH) input is connected to current sensing resistor to measure current from (HS1) and (HS2)
	Floating: (ISENSEH) is not connected to current sensing resistor
JP5	1-2 position: (HS2) is connected to current sensing resistor, and with shorted JP4 to (ISENSEH) input
	Floating: (HS2) is not connected
JP6	1-2 position: (HS1) is connected to current sensing resistor and with shorted JP4 to (ISENSEH) input
	Floating: (HS1) is not connected
JP7	1-2 position: Voltage Regulator output (VDD) is connected to supply MCU
	Floating: (VDD) output is not connected to supply MCU
JP8	1-2 position: Supply MC33912 High Side Switches Module from VBAT
	Floating: High Side Switches Module is not powered
JP9	1-2 position: Supply MC33912 from VBAT
	Floating: MC33912 is not powered
JP10	1-2 position: SPI signal Master Out Slave In (MOSI) is connected from MCU to MC33912
	2-3 position: SPI signal Master Out Slave In (MOSI) is connected from J3 header to MC33912
	Floating: (MOSI) is not connected
JP11	1-2 position: SPI signal Master In Slave Out (MISO) is connected from MCU to MC33912
	2-3 position: SPI signal Master In Slave Out (MISO) is connected from J3 header to MC33912
	Floating: (MISO) is not connected
JP12	1-2 position: SPI Clock (SCLK) is connected from MCU to MC33912
	2-3 position: SPI Clock (SCLK) is connected from J3 header to MC33912
	Floating: (SCLK) is not connected
JP13	1-2 position: SPI Chip Select (CS) is connected from MCU to MC33912
	2-3 position: SPI Chip Select (CS) is connected from J3 header to MC33912
	Floating: (CS) is not connected

Table 10-1. Jumper Connection

Name	Description
JP14	1-2 position: MC33912 Pulse Width Modulation Input (PWMIN) is connected to MCU
	2-3 position: MC33912 Pulse Width Modulation Input (PWMIN) is connected to J3 header
	Floating: (PWMIN) is not connected
JP15	1-2 position: MC33912 Analog Output from Current Sense Module (ADOUT1) is connected to MCU
	2-3 position: MC33912 Analog Output from Current Sense Module (ADOUT1) is connected to J3 header
	Floating: (ADOUT1) is not connected
JP16	1-2 position: MC33912 Analog Output from Analog Multiplexer (ADOUT0) is connected to MCU
	2-3 position: MC33912 Analog Output from Analog Multiplexer (ADOUT0) is connected to J3 header
	Floating: (ADOUT0) is not connected
JP17	1-2 position: MC33912 LIN (TxD) input is connected to MCU
	2-3 position: MC33912 LIN (TxD) input is connected to J3 header
	Floating: (TxD) is not connected
JP18	1-2 position: MC33912 LIN (RxD) output is connected to MCU
	2-3 position: MC33912 LIN (RxD) output is connected to J3 header
	Floating: (RxD) is not connected
JP19	1-2 position: Interrupt output (IRQ) is connected from MC33912 to MCU interrupt pin
	2-3 position: Interrupt output (IRQ) is connected from MC33912 to J3 header
	Floating: (IRQ) is not connected
JP20	1-2 position: Reset output (RST) is connected from MC33912 to MCU reset pin
	2-3 position: Reset output (RST) is connected from MC33912 to J3 header
	Floating: (RST) is not connected
JP21	1-2 position: LED diode D2 is connected to (VDD) supply voltage source
	Floating: LED diode D2 is not indicated presence of voltage on (VDD) pin
JP101	1-2 position: (HS2) on-board load is conneced
JP102	1-2 position: (LS2) on-board load is connected
JP103	1-2 position: (LS1) on-board load is connected
JP104	1-2 position: Analog input (L4) is connected to potentiometer R111
JP105	1-2 position: (HS2) is connected to (L1) input
JP106	1-2 position: (HS1) on-board load is connected
JP202	1-2 position: MCU input (RxD) is connected to MAX232 converter
	2-3 position: LIN physical interface MCZ33661EF input (RxD) is connected to MAX232 converter
	Floating: SCI input of MAX232 is disconnected
JP203	1-2 position: MCU output (TxD) is connected to MAX232 converter
	2-3 position: LIN physical interface MCZ33661EF output (TxD) is connected to MAX232 converter
	Floating: SCI output of MAX232 is disconnected

Table 10-1. Jumper Connection

Name	Description
JP204	1-2 position: LIN physical interface is disabled
	Floating: LIN physical interface is enabled
JP205	1-2 position: LIN signal from LIN physical interface MCZ33661EF is connected to device MC33912 and connector CON1
	Floating: LIN signal from LIN physical interface MCZ33661EF is not connected
JP206	1-2 position: LIN pull-up resistor and diode are connected {Master Mode}
	Floating: LIN pull-up resistor and diode are disconnected {Slave Mode}

Appendices

Appendix A Quick Guide

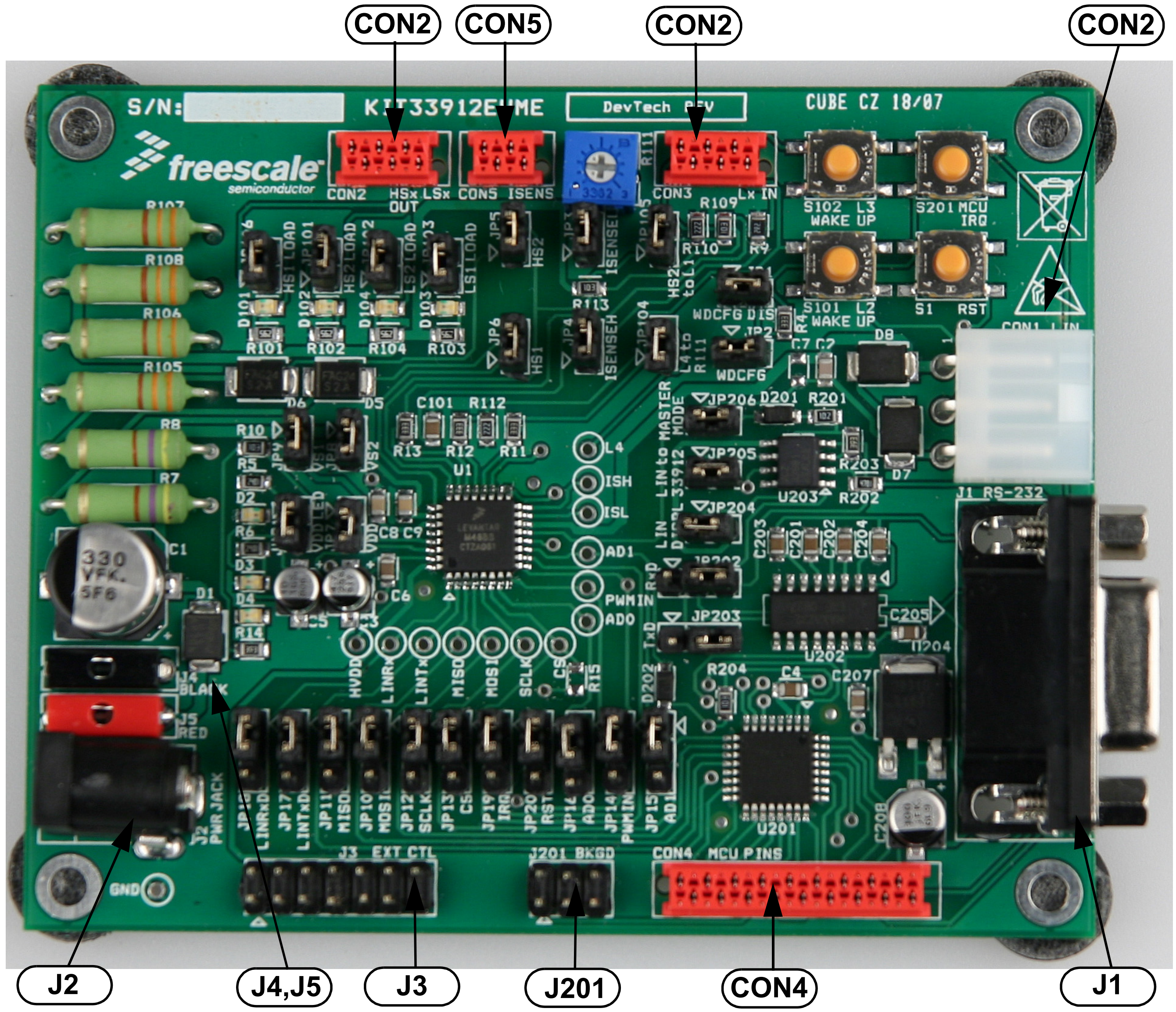


Figure A-1. Quick Guide

Appendix B Schematics

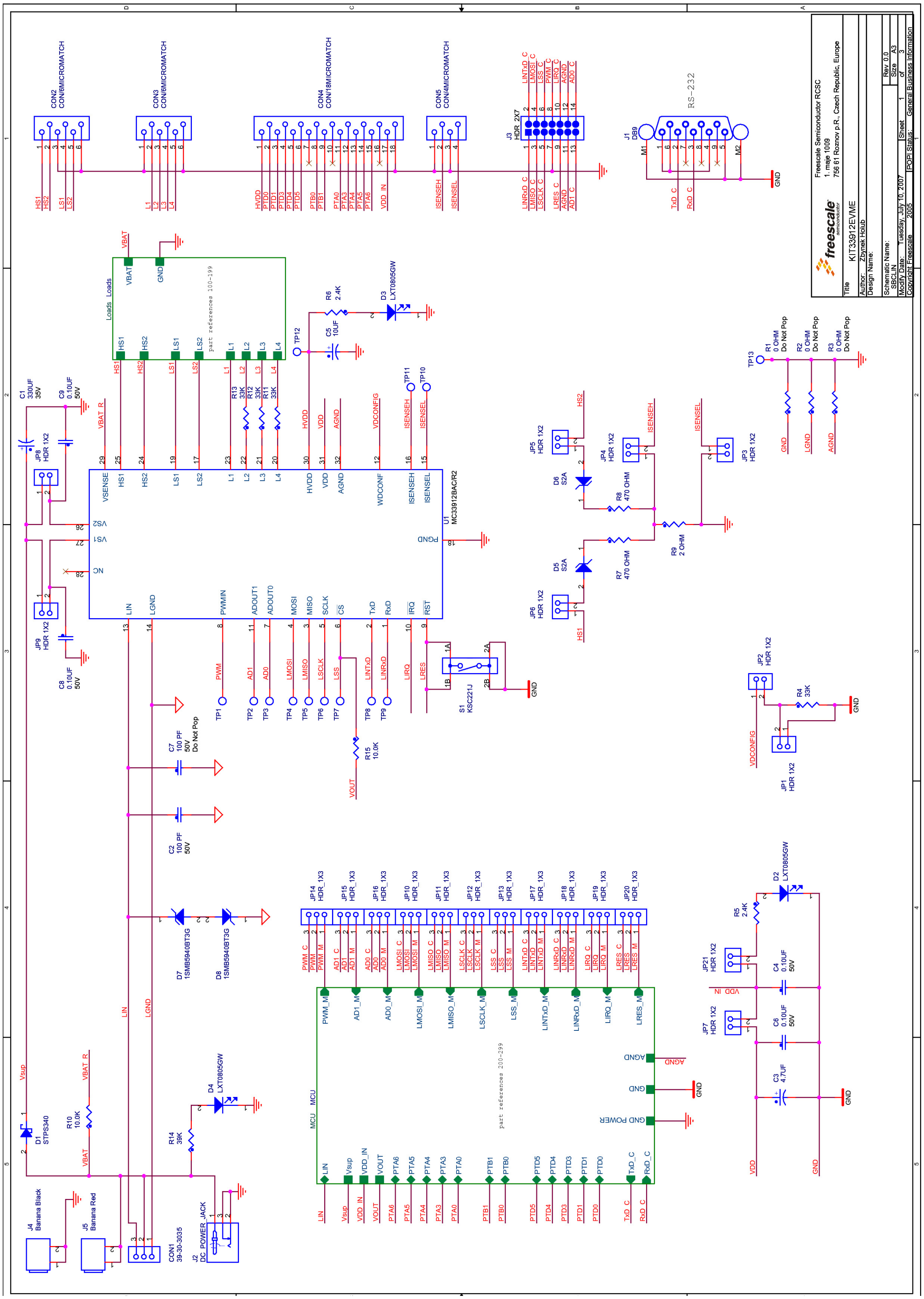


Figure B-1. Schematics - System Basis Chip

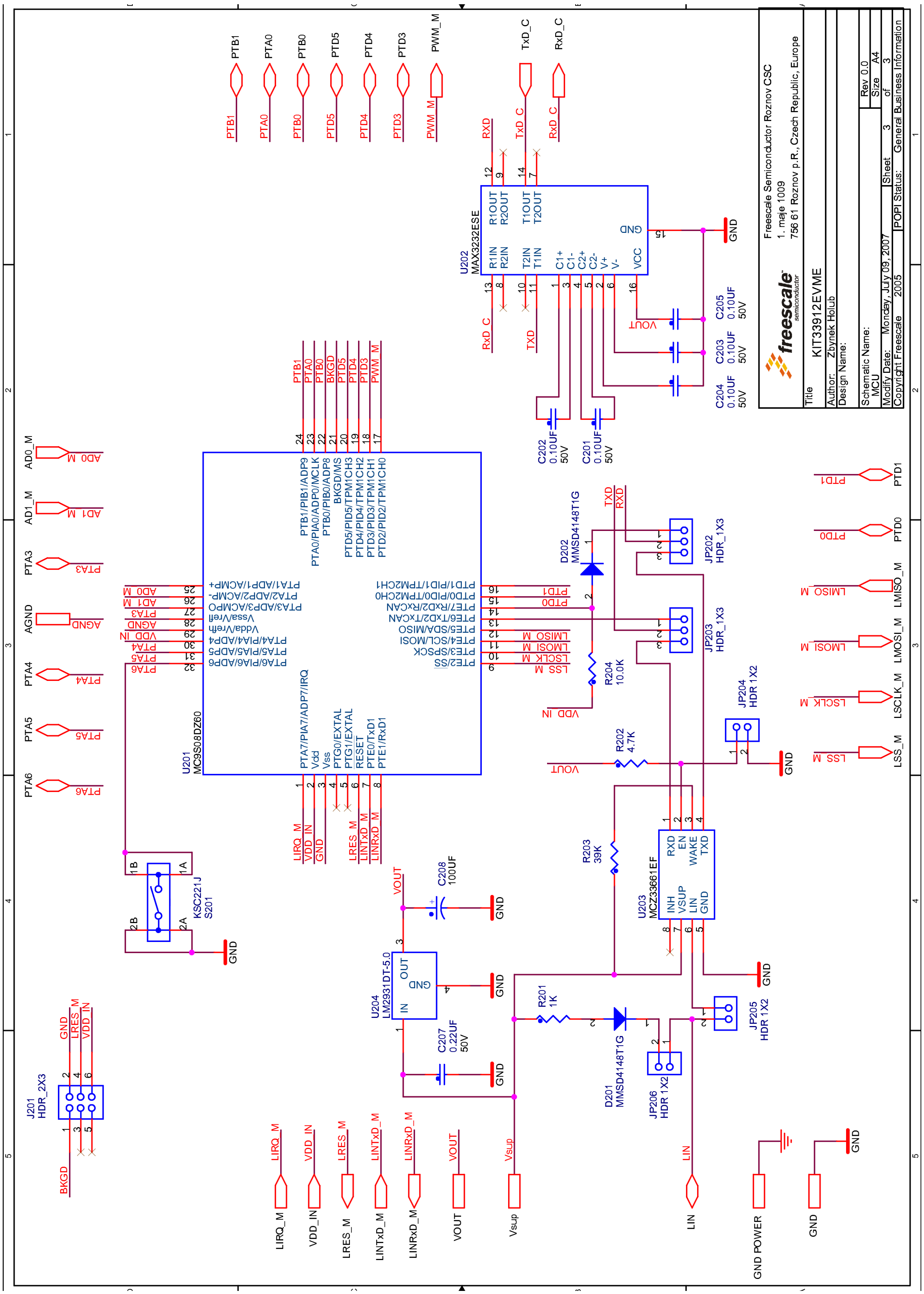


Figure B-2. Schematics - MCU

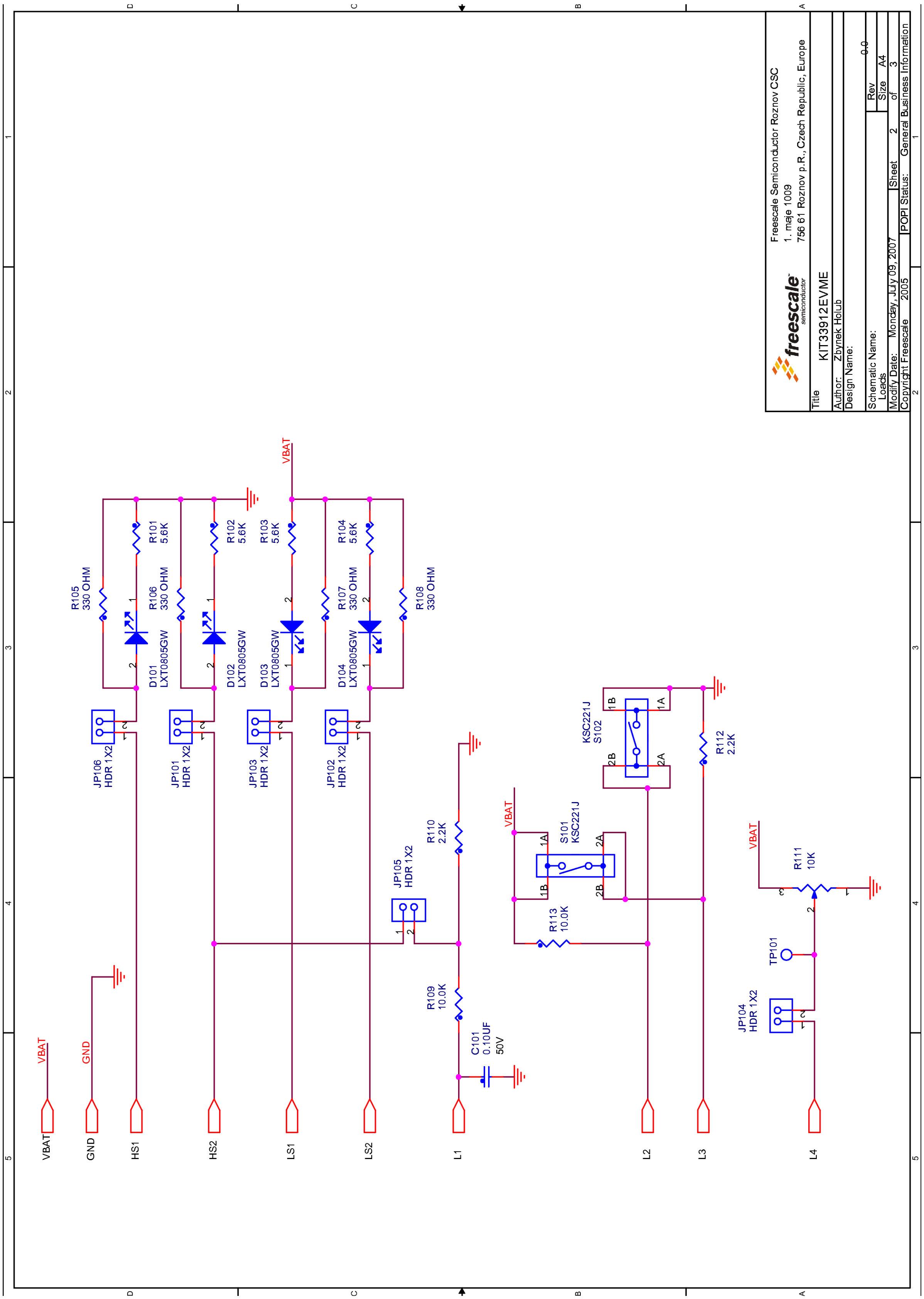


Figure B-3. Schematics - Loads

Appendix C Placement and Layout

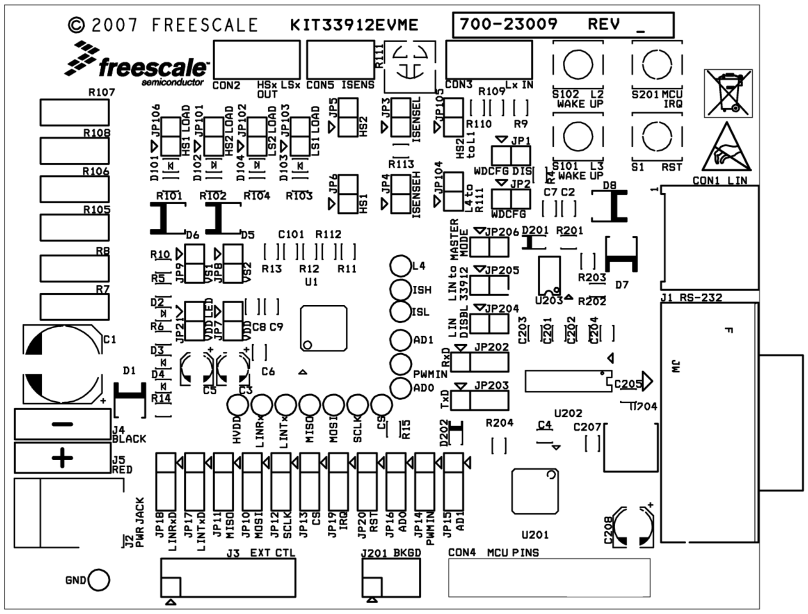


Figure C-1. Placement Top

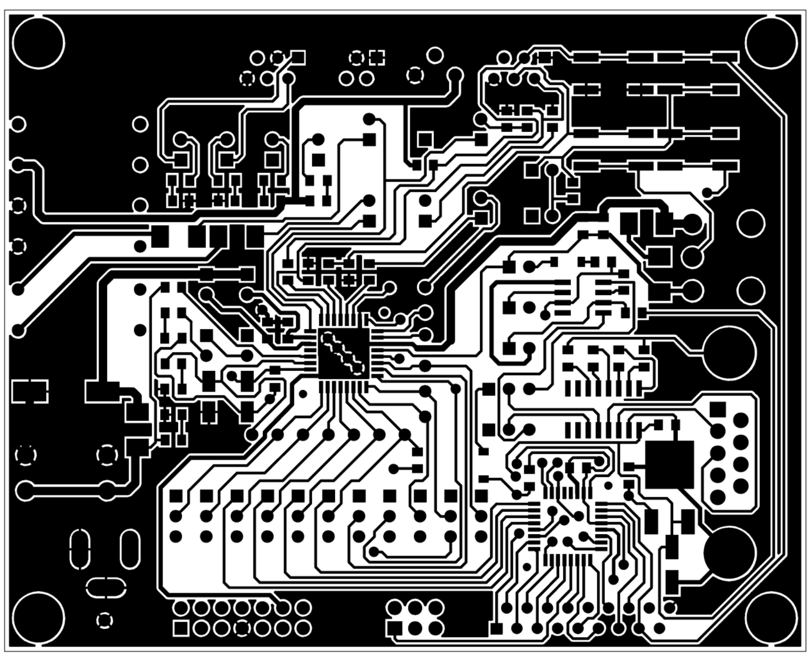


Figure C-2. Copper Top

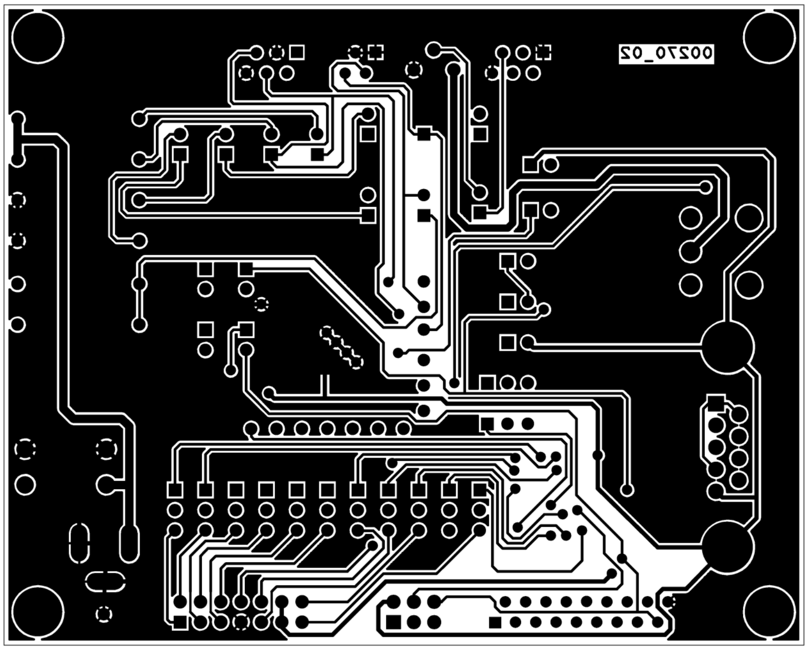


Figure C-3. Copper Bottom (mirrored)

Appendix D Bill of Materials

Table D-1. Bill of Materials

Number	Quantity	Reference	Part	Comments
1	1	CON1	CON/3MOLEX	3 Pin Connector, 39-30-3035
2	2	CON2, CON3	CON/6MICROMATCH	6 Pin Connector, 7-215079-6
3	1	CON4	CON/18MICROMATCH	18 Pin Connector, 8-215079-8
4	1	CON5	CON/4MICROMATCH	4 Pin Connector, 7-215079-4
5	1	C1	330UF	Aluminium Electrolytic Capacitor, 330µF, 35V, EEEFK1V331AP
6	1	C2	100PF	100pF Ceramic SMD 0805, 50V
7	1	C3	4.7UF	Aluminium Electrolytic Capacitor, 4.7µF, 25V, EEEHA1E4R7R
8	10	C4, C6, C8, C9, C101, C201, C202, C203, C204, C205	.1UF	0.1µF Ceramic SMD 0805, 50V
9	1	C5	10UF	Aluminium Electrolytic Capacitor, 10µF, 16V, EEE1CA100SR
10	1	C207	.22UF	0.22µF Ceramic SMD 0805, 50V
11	1	C208	100UF	Aluminium Electrolytic Capacitor, 100µF, 6.3V, EEEFKJ101UAR
12	1	C7	100PF	Not Populated.
13	1	D1	STPS340U	Schottky Diode, STPS340U
14	7	D2, D3, D4, D101, D102, D103, D104	LXT0805GW	LED, SMD - 0805, Green
15	2	D5, D6	S2A	Diode, DO-214AA, S2A
16	2	D7, D8	1SMB5940BT3G	Zener Diode, 43V, 3W, 1SMB5940BT3
17	2	D201, D202	MMSD4148T1G	Diode, 1N4148, SOD-123, MMSD4148T1G
18	19	JP1, JP2, JP3, JP4, JP5, JP6, JP7, JP8, JP9, JP21, JP101, JP102, JP103, JP104, JP105, JP106, JP204, JP205, JP206	HDR_1X2	Header 2X1, Diameter 2.54
19	13	JP10, JP11, JP12, JP13, JP14, JP15, JP16, JP17, JP18, JP19, JP20, JP202, JP203	HDR_1X3	Header 3X1, Diameter 2.54
20	1	J1	DB9	D Subminiature Connector, 5747844-6
21	1	J2	DC_POWER_JACK	Power Jack, Pin Dia 2.1mm, PJ-102AH

Table D-1. Bill of Materials (continued)

Number	Quantity	Reference	Part	Comments
22	1	J3	HDR_2X7	Header 7X2, Diameter 2.54
23	1	J4	BANANA BLACK	Test Jack, 105-0753-001
24	1	J5	BANANA RED	Test Jack, 105-0752-001
25	1	J201	HDR_2X3	Header, 3X2, Diameter 2.54
26	3	R1, R2, R3	0 OHM	Not Populated
27	4	R4, R11, R12, R13	33K	Resistor, SMD, 33k, 0805
28	2	R5, R6	2.4K	Resistor, SMD, 2.4k, 0805
29	2	R7, R8	470 OHM	Metal Oxide Resistor, 470R, 2W
30	1	R9	2 OHM	Resistor, SMD, 2R, 0805
31	5	R10, R15, R109, R113, R204	10K	Resistor, SMD, 10k, 0805
32	2	R14, R203	39K	Resistor, SMD, 39k, 0805
33	4	R101, R102, R103, R104	5.6K	Resistor, SMD, 5.6K, 0805
34	4	R105, R106, R107, R108	330 OHM	Metal Oxide Resistor, 330R, 2W
35	2	R110, R112	2.2K	Resistor, SMD, 2.2k, 0805
36	1	R201	1K	Resistor, SMD, 1k, 0805
37	1	R202	4.7K	Resistor, SMD, 4.7k, 0805
38	1	R111	10K	Cermet potentiometer, 3362P-1-103LF
39	4	S1, S101, S102, S201	KSC221J	Switch, KSC221JLFS
40	14	TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP101	Terminal	Not populated.
41	1	U1	MC33912BAC/R2	SBCLIN, MC33912BAC/R2
42	1	U201	MC9S08DZ60	8-bit Automotive microcontroller family, 32pin, MC9S08DZ60MLC
43	1	U202	MAX3232ESE	RS232/TTL converter, Maxim-Dallas, MAX3232ESE+
44	1	U203	MCZ33661EF	LIN Driver, MCZ33661EF
45	1	U204	LM2931DT-5.0G	Linear Voltage Regulator 5V, 100mA, LM2931DT-5.0G
46	32	Not refer	Jumper	Jumper Socket

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

KT33912UG
Rev. 2.0
10/2009