Interfacing the MindSet with Arduino

Features

- Interfaces with the Mindset with the BlueSMiRF module for Bluetooth communication
- Acquiring the Attention and Meditation eSense values from the data stream.

Introduction

This step-by-step guide is intended to quickly get a typical Arduino board user up and running with the MindSet. This guide will show how to setup the BlueSMiRF module quickly for communication with the MindSet and gives example code to parse the output stream. The parsed output stream can then be used to perform various tasks such as to control the speed of a motor, to light up LEDs, and to control the position of a servo motor, etc.

Interfacing to the MindSet

Interfacing to the MindSet requires the end application to appear as a Bluetooth host. The following describes easiest way to quickly get up and running using an Arduino board paired with a BlueSMiRF module. Both the Arduino and BlueSMiRF are sold by a vendor listed in the sources section. The MindSet is available at the NeuroSky Store.

Required Materials

This application requires the following materials:

- MindSet
- Arduino Diecimila or Duemilanove
- BlueSMiRF Bluetooth Module
- Arduino development environment

System Overview

The system uses the Aduino's hardware UART for both USB communication and Bluetooth communication. After the system is up and running, the Arduino may be detached from the computer and run independently.

The basic programming flow is as follows:

1. Find the MAC address of your MindSet.
2. Connect the BlueSMiRF to the Arduino board.

3. Assuming the BlueSMiRF Bluetooth module is in a factory default state, program using the ProgramBlueSMiRF sketch.
4. Download the provided example code into the Arduino board using the Arduino development environment.
5. Power up the whole system and observe the Attention eSense value in a terminal software window.

Detailed Instructions

Finding the MindSet MAC Address

The MindSet's MAC address can be found on MindSet itself. The MAC address is on a sticker on the inside of the MindSet's headband. The MAC address is 12 digits and usually starts with 00:13:EF:xx:xx:xx. Alternatively, the MAC address can be found after pairing the MindSet with your computer and reading the MAC address from the Bluetooth stack.

Setting up the BlueSMiRF

To convert the BlueSMiRF from RS232 logic to 3.3V logic, short the RTS and CTS-1 pins with a wire.

Powering the Arduino

The Arduino may be powered by USB, by battery, or by an external power supply. Please see the Arduino hardware page here http://arduino.cc/en/Main/Hardware for more information.

Getting started with the Arduino development environment

Please see Arduino Getting Started guide here http://arduino.cc/en/Guide/HomePage

Connecting and programming the BlueSMiRF module

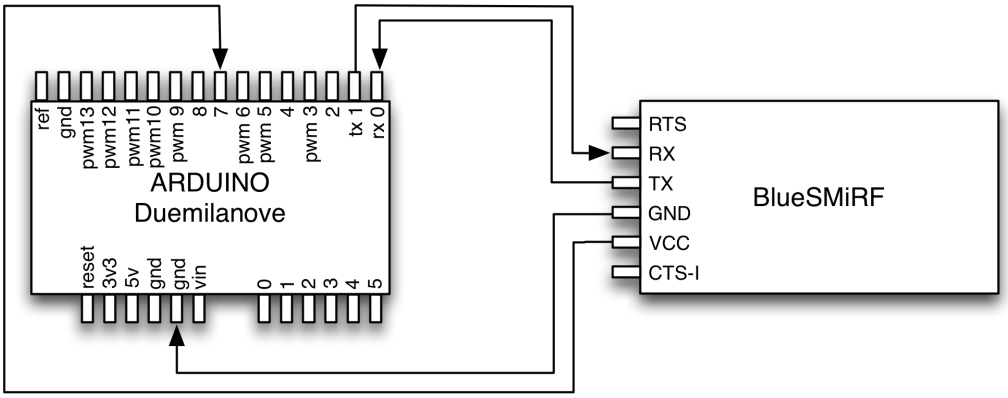

Figure 1: Programming layout

1. Connect the BlueSMiRF module to the Arduino according to the diagram above.
2. Load the ProgramBlueSMiRF.pde sketch in the Arduino development environment.

March 16, 2011 | © 2009 NeuroSky, Inc. All Rights Reserved.

3. On line 23, replace the MAC address with the MAC address of your MindSet without colons.

   (a) `char mac[13] = "0013EF00077A";` » `char mac[13] = "Your MindSet MAC address";`

4. Verify the sketch.

5. Upload the sketch the the Arduino and it will begin to program the BlueSMiRF. This process takes less than 3 seconds.

6. If successful, the LED on the Arduino will blink.

Using the MindSet with the Arduino

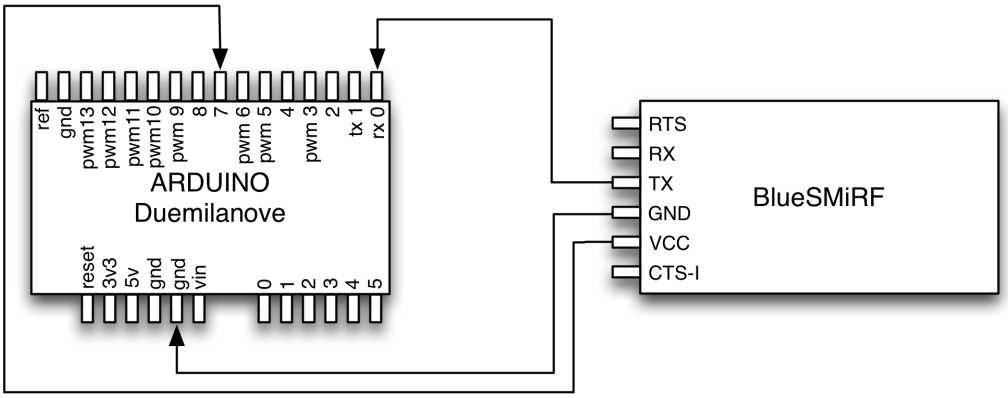

Figure 2: Default layout

1. Permanently short the RTS and CTS-1 pins of the BlueSMiRF. This changes the board from RS232 logic to 3.3V logic. (note that diagram does not indicate RTS and CTS-1 pins to be shorted)

2. Connect the BlueSMiRF module as shown in a diagram above. Note the BlueSMiRF RX is **not** connected.

3. Open the Tutorial.pde sketch in the Arduino development environment.

4. Verify the sketch and upload to the Arduino.

5. Turn on the MindSet. Once connected, the MindSet will have a solid blue light and the BlueSMiRF will have a solid green light.

6. Open the Serial Monitor in the Arduino development environment. Set the baud rate to 57600 baud.

7. If the Arduino is receiving data from the MindSet, you should see output like this:

```
PoorQuality: 200 Attention: 0 Time since last packet: 1006
```

Arduino Code Description

Arduino code is based on the C/C++ language and requires very little else to get started. All the details or regular microprocessor development such as processor clock speed, programming speed, clock sources, etc. are all initialized by the Arduino core. With this in mind, the Arduino system will

March 16, 2011 | © 2009 NeuroSky, Inc. All Rights Reserved.

still be able to perform much of the required tasks with no problem at all. The simplicity of using the microprocessor peripherals combined with the C language along with the sample code provided provides for very quick interface to the MindSet. The user is encouraged to extend the Arduino code by appending code after the indicated section where the packet has been checked for a valid checksum.

Thereafter, the user may use the eSense values to do one of many things possible. Some of which is to control the brightness of an LED, the height of an object, etc.

The bulk of the code is used for parsing the packet data. For further information, see the MindSet Communication Protocol document in the MindSet Development tools (MDT). The MDT is available at the NeuroSky Store for free. http://store.neurosky.com

Arduino Reset Code Description

It may be necessary to reset the BlueSMiRF module back to the factory settings to reprogram it for another MindSet. It could also be the case where the module could be incorrectly programmed and caused it to become out of sync with the microprocessor. To reset the BlueSMiRF module follow these instructions:

1. Connect the BlueSMiRF according to the Programming layout detailed earlier.
2. Open ProgramBlueSMiRF.pde sketch.
3. On line 14, change `#define FACTORYRESET 0` to `#define FACTORYRESET 1`
4. Verify and upload the code to the Arduino.
5. The BlueSMiRF will now be reset to the factory defaults.
6. Change line 23 to your desired MAC address as detailed earlier.
7. Change line 14 back to `#define FACTORYRESET 0` and upload the code to the Arduino.
8. The BlueSMiRF is now reset.

Material Sources

The BlueSMiRF and Arduino board can be obtained from:

Sparkfun Electronics
6175 Longbow Drive, Suite 200
Boulder, CO 80301
http://www.sparkfun.com

More information about the Arduino can be found here: http://www.arduino.cc

March 16, 2011 | © 2009 NeuroSky, Inc. All Rights Reserved.

Corporate Address
NeuroSky, Inc.
125 S. Market St., Ste. 900
San Jose, CA 95113
United States
(408) 600-0129

Questions/Support: http://support.neurosky.com
or email: support@neurosky.com

Community Forum: http://developer.neurosky.com/forum

Information in this document is subject to change without notice.

Reproduction in any manner whatsoever without the written permission of NeuroSky Inc. is strictly forbidden. Trademarks used in this text: eSense™, ThinkGear™, MindKit™, NeuroBoy™and NeuroSky®are trademarks of NeuroSky, Inc.

Disclaimer: The information in this document is provided in connection with NeuroSky products. No license, express or implied, by estoppels or otherwise, to any intellectual property rights is granted by this document or in connection with the sale of NeuroSky products. NeuroSky assumes no liability whatsoever and disclaims any express, implied or statutory warranty relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or non-infringement. In no even shall NeuroSky be liable for any direct, indirect, consequential, punitive, special or incidental damages (including, without limitation, damages for loss of profits, business interruption, or loss of information) arising out of the use of inability to use this document, even if NeuroSky has been advised of the possibility of such damages. NeuroSky makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. NeuroSky does not make any commitment to update the information contained herein. NeuroSky's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

NeuroSky
Brain-Computer Interface Technologies