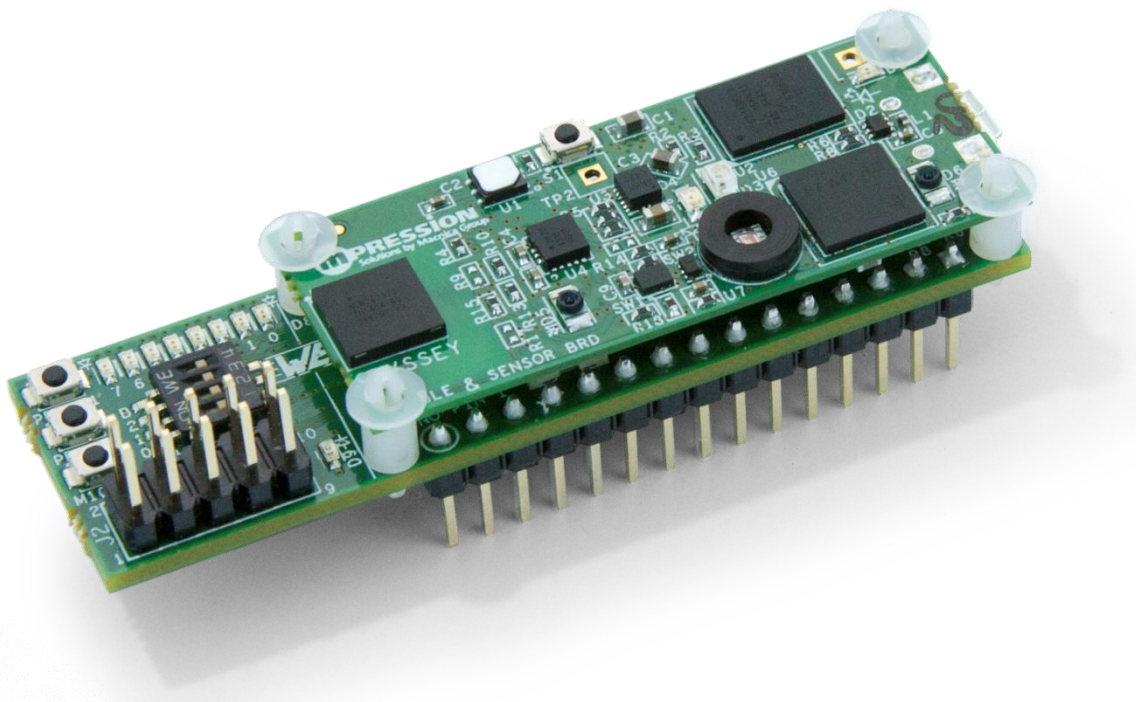




## User Guide



### Odyssey MAX 10 FPGA and BLE Sensor Kit

Model: ODYSSEY-MAX10-KIT

**Revision 2.1**

March 26, 2015

© Mpression – Solutions by Macnica Group

## Disclaimer

The information in this document (hereinafter called “Information”) is subject to change without notice.

Macnica Inc. (hereinafter called “Manufacturer”) makes no warranty of any kind regarding this document, or of any liability arising out of the application or use of information in this document, and assumes no responsibility for any errors that may appear in this document.

This document is distributed without any charge and reselling or copying without written authorization by Manufacturer is restricted.

IN NO EVENT WILL MANUFACTURER BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOST DATA AND LOST PROFITS, ARISING FROM OR RELATING TO YOUR USE OF THE INFORMATION, EVEN IF YOU HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE TOTAL CUMULATIVE LIABILITY OF MANUFACTURER IN CONNECTION WITH YOUR USE OF THE INFORMATION IN THIS DOCUMENT, WHETHER IN CONTRACT OR TORT OR OTHERWISE, WILL IN NO EVENT EXCEED THE AMOUNT OF FEES PAID BY YOU TO MANUFACTURER HEREUNDER FOR USE OF THE INFORMATION. YOU ACKNOWLEDGE THAT THE FEES, IF ANY, REFLECT THE ALLOCATION OF RISK SET FORTH IN THIS AGREEMENT AND THAT MANUFACTURER WOULD NOT MAKE AVAILABLE THE INFORMATION TO YOU WITHOUT THESE LIMITATIONS OF LIABILITY.

The Information is not intended for use in the development of on-line control equipment in hazardous environments requiring failsafe controls, such as in the operation of nuclear facilities, aircraft navigation or communications systems, air traffic control, life support, or weapons systems (“High-Risk Applications”). Manufacturer specifically disclaims any express or implied warranties of fitness for such High-Risk Applications. You represent that use of the Information in such High-Risk Applications is fully at your risk.

## Regulatory

### FCC:

Model: ODYSSEY-MAX10-KIT

FCC ID: 2AEDQODSY001

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

- (1) This device may not cause harmful interference, and
- (2) This device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

### IC:

Model: ODYSSEY-MAX10-KIT

IC: 20022-ODSY001

This device complies with Industry Canada licence-exempt RSS standard(s). Operation is subject to the following two conditions: (1) this device may not cause interference, and (2) this device must accept any interference, including interference that may cause undesired operation of the device.

Under Industry Canada regulations, this radio transmitter may only operate using an antenna of a type and maximum (or lesser) gain approved for the transmitter by Industry Canada. To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (e.i.r.p.) is not more than that necessary for successful communication.

### French:

Cet appareil est conforme avec Industrie Canada exempts de licence standard RSS (s) l'opération est soumise aux deux conditions suivantes: . ( 1) ce dispositif ne doit pas causer d'interférences et ( 2) cet appareil doit accepter toute interférence , y compris les interférences qui peuvent provoquer un mauvais fonctionnement de l'appareil.

Conformément à la réglementation d'Industrie Canada, cet émetteur radio ne peut fonctionner à l'aide d'une antenne d'un type et maximum ( ou moins ) Gain approuvé pour l'émetteur par Industrie Canada. Afin de réduire le risque d'interférence avec d'autres utilisateurs, le type d'antenne et son gain doivent être choisi que la puissance isotrope rayonnée équivalente (PIRE ) ne dépasse pas ce qui est nécessaire pour une communication réussie .

# Index

<b>1. Safe Usage</b>	<b>10</b>
1.1 Legend	10
1.2 Cautions	10
1.3 Developer Information	10
1.4 Inquiries	12
<b>2. Overview</b>	<b>13</b>
2.1 Introduction	13
2.2 Prerequisites	14
2.2.1 Mpression MAX 10 FPGA and BLE Sensor Kit	14
2.2.2 Power	14
2.2.3 Smartphone	15
2.2.4 Web Utility Account	15
2.2.5 Serial Communications Terminal	16
2.2.6 USB Virtual COM Port Driver	17
2.2.7 Development Software	17
<b>3. Components &amp; Interfaces</b>	<b>18</b>
3.1 The Odyssey IoT Solutions Boards	18
3.1.1 BLE sensor board	18
3.1.2 Battery board	21
3.1.3 FPGA board	22
3.2 Engineering Documents	25
3.2.1 BLE sensor board	25
3.2.2 Battery board	26
3.2.3 MAX 10 FPGA board	26
<b>4. Assembly/Disassembly</b>	<b>27</b>
4.1 Included HW	27
4.2 Tools	27
4.3 Disassembly	27
4.4 Reassembly	27
<b>5. Firmware Updates</b>	<b>28</b>
5.1 EFM32	28
5.1.1 Serial console	28
5.1.2 Simplicity Studio	30
5.2 BLE	30
5.2.1 Virtual COM Port	31
5.2.2 WICED SDK	35

<b>6. Run Preprogrammed Demonstrations (Personalities)</b>	<b>36</b>
6.1 Apply Power .....	36
6.2 Launch Smartphone App .....	36
6.3 Connect to Device .....	38
6.4 Select Personality .....	39
6.5 Exercise Personality Features.....	40
<b>7. The Mpression Odyssey IoT Solutions Framework</b>	<b>41</b>
7.1 The Smartphone .....	41
7.1.1 Communications link.....	41
7.1.2 The Application .....	41
7.1.3 Personality control .....	42
7.2 The Bluetooth Low Energy Device .....	42
7.2.1 Communications link.....	42
7.2.2 Odyssey Profile – The GATT Database.....	42
7.2.3 Command Interpreter .....	43
7.2.4 Sensor data processing.....	43
7.3 The EFM32 Microcontroller .....	44
7.3.1 Communications link.....	44
7.3.2 Personality loading and FPGA Programming.....	44
7.3.3 Flash Layout .....	44
7.3.4 Command Interpreter .....	46
7.3.5 Sensor data processing.....	46
7.4 The FPGA.....	46
7.4.1 Communications link .....	46
7.4.2 Custom logic.....	46
7.4.3 Programming.....	47
7.4.4 Sensor data processing.....	47
7.5 Personality Commands .....	47
7.5.1 I <sup>2</sup> C.....	48
7.5.2 Get HRM.....	49
7.5.3 CMD Stop .....	49
7.5.4 NOP Stop .....	49
<b>8. The Web Utility - Create A Custom Personality</b>	<b>50</b>
8.1 Sign in to Smartphone App Web Utility .....	50
8.2 Create New Personality.....	51
8.2.1 Text Fields .....	52
8.2.2 Autonomous Field .....	53
8.2.3 Comparison Indicator Fields (Not implemented) .....	53
8.2.4 FPGA Required Field.....	53
8.2.5 Personality Setup Commands Field.....	56
8.2.6 Background Commands Field.....	56

- 8.2.7 Background Interrupt Commands Field (Not Implemented).....57
- 8.2.8 Personality Controls Fields .....57
- 8.3 Download Custom Personality ..... 60
- 8.4 Load Personality to Flash ..... 60
  - 8.4.1 Update FPGA design with Console Menu .....65
- 8.5 Run Custom Personality ..... 67
- 9. Tips and Tricks \_\_\_\_\_ 70**
  - 9.1 Consult FAQs ..... 70
  - 9.2 Cut standoffs ..... 70
  - 9.3 Stand-alone MAX 10 FPGA board ..... 70
    - 9.3.1 Parts list..... **Error! Bookmark not defined.**
  - 9.4 Getting leverage for the FPGA I<sup>2</sup>C interface ..... 70
  - 9.5 Startup delay ..... 71
- 10. Document Revision History \_\_\_\_\_ 72**

# List of Figures

Figure 2-1 Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit.....	13
Figure 2-2 Micro USB Cable .....	14
Figure 2-3 Mpression Odyssey App on Google Play Store.....	15
Figure 2-4 Mpression Smartphone App Configuration Web Utility .....	16
Figure 3-1 Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit.....	18
Figure 3-2 BLE Sensor Board Block Diagram .....	19
Figure 3-3 BLE Sensor Board .....	19
Figure 3-4 Battery Board.....	21
Figure 3-5 MAX 10 FPGA Board Block Diagram.....	23
Figure 3-6 MAX 10 FPGA Board.....	23
Figure 3-7 FPGA Board Power Tree .....	24
Figure 5-1 EFM32 Programming Mode Short .....	28
Figure 5-2 EFM32 Programming Serial Console Configuration .....	29
Figure 5-3 EFM32 Programming via XMODEM Protocol .....	29
Figure 5-4 EFM32 Programming File Selection .....	30
Figure 5-5 BLE Device Programming Jumper.....	31
Figure 5-6 Setting SDK_INSTALL Environment Variable .....	32
Figure 5-7 Auto-detect of Odyssey kit.....	32
Figure 5-8 File Selection for BLE Device Firmware Update .....	33
Figure 5-9 BLE Device Programming Status.....	33
Figure 5-10 Configuration File Selection for BLE MAC Address Update .....	34
Figure 5-11 Setting New MAC Address .....	34
Figure 5-12 BLE Device Programming Status.....	35
Figure 5-13 BLE Device Run Mode Jumper.....	35
Figure 6-1 Micro USB Cable .....	36
Figure 6-2 Login Screen.....	37
Figure 6-3 Device Discovery .....	38
Figure 6-4 Personality List.....	38
Figure 6-5 Activity LED.....	39
Figure 7-1 Bluetooth SMART Stack .....	41
Figure 8-1 Private and Public Personality List .....	50
Figure 8-2 Personality Editor.....	52
Figure 8-3 Smartphone App Fields .....	52
Figure 8-4 Create JAM File Selection .....	54
Figure 8-5 Create JBC .....	54
Figure 8-6 Successful JAM File Creation .....	55
Figure 8-7 FPGA Required Selection .....	55
Figure 8-8 FPGA JAM (JBC) File Selection .....	56

Figure 8-9 Upload JAM File.....	56
Figure 8-10 Personality Control Edit Fields .....	57
Figure 8-11 Available Commands .....	57
Figure 8-12 Command Specific Data Fields.....	58
Figure 8-13 Complete Custom Personality .....	59
Figure 8-14 Saved Custom Personality .....	60
Figure 8-15 COM Port Settings .....	61
Figure 8-16 MAX 10 Present/Console Menu Push-button .....	61
Figure 8-17 Console Menu.....	62
Figure 8-18 Tera Term New-line Setting.....	62
Figure 8-19 Flash Resident Personalities .....	63
Figure 8-20 Upload to Flash Personality #7 .....	63
Figure 8-21 Send File to Device .....	64
Figure 8-22 File Being Sent To Device.....	64
Figure 8-23 Custom Personality Loaded into Flash .....	65
Figure 8-24 Load JAM File.....	66
Figure 8-25 File Being Sent To Device.....	66
Figure 8-26 Select Personality to Associate JAM File.....	67
Figure 8-27 Custom Personality Available .....	68
Figure 8-28 Custom Personality Control Screen .....	68
Figure 8-29 Heart Rate Sensor Finger Position.....	69
Figure 8-30 Personality Data Fields Active .....	69
Figure 9-1 Clip Locking Standoffs .....	70
Figure 9-2 Stand-alone MAX 10 FPGA Board .....	70






# List of Tables

Table 1 Serial Terminal Emulator Settings.....	16
Table 2 BLE Sensor Board Push Buttons.....	21
Table 3 BLE Sensor Board LEDs/Diodes.....	21
Table 4 Battery Board J4 Jumper Settings.....	22
Table 5 BLE Sensor Board Documents.....	25
Table 6 Battery Board Documents.....	26
Table 7 MAX 10 FPGA Board Documents.....	26
Table 8 Assembly Parts.....	27
Table 9 Odyssey Profile.....	42
Table 10: Flash Memory Layout.....	45
Table 11: Personality File Format.....	45
Table 12: Personality Commands.....	47
Table 13 I <sup>2</sup> C Devices.....	49



# 1. Safe Usage


Be sure to follow the instructions given in this Manual which are intended to prevent harm to the user and others as well as material damage.

## 1.1 Legend

 <b>Danger</b>	Indicates an imminent hazardous situation which if not avoided will result in death or serious injury.
 <b>Warning</b>	Indicates a potentially hazardous situation which if not avoided could result in death or serious injury.
 <b>Caution</b>	Indicates a potentially hazardous situation which if not avoided may result in minor or moderate injury or in property damage.

## 1.2 Cautions

 <b>Danger</b>	<p>Coin cell batteries are not supplied with this kit. If a coin cell battery is used with the kit, keep this battery away from children. Children may swallow the battery causing severe injury or death from electrical or chemical reactions and burns. Children may also choke on the battery.</p> <p>Coin cell batteries are not supplied with this kit. If a coin cell battery is used with the kit, beware of shorting the battery terminals, as this can cause heating and fire.</p> <p>Coin cell batteries are not supplied with this kit. If a coin cell battery is used with the kit, please read and follow all warnings and directions of the coin cell manufacturer to avoid injury.</p> <p>This kit contains small parts that represent a choking hazard to children. Please keep the kit and all parts of the kit away from children.</p>
 <b>Warning</b>	<p>Do not apply strong impacts or blows to the kit. Doing so may cause the kit to emit heat, explode, or ignite, or the equipment in the kit to fail or malfunction. This may also cause fire.</p> <p>When disposing of the main unit, do not dispose of it along with general household waste. Throwing the main unit into fire may cause it to explode. Dispose of the main unit following the laws, regulations, and ordinances governing waste disposal. Do not use the kit in places subject to extremely high or low temperatures or severe temperature changes. Doing so may cause the kit to fail or to malfunction. Always be sure to use the kit in a temperatures ranging from 5°C to 35°C and a humidity range of 0% to 85%.</p> <p>Do not attempt to use or leave the kit in places subject to strong direct sunlight or other places subject to high temperatures such as in cars in hot weather. Doing so might cause the kit to emit heat, break, ignite, run out of control, warp, or malfunction. Also, some parts of the equipment might emit heat causing burn</p>

	<p>injuries.</p> <p>USB cables are not provided with this kit, but can be used with the kit. Please note that cables can present a strangulation hazard, especially in small children. Please be aware of this hazard. Please keep cables away from children and always follow the manufacturer’s directions when using USB cables.</p>
 <p><b>Caution</b> (Continued from previous page)</p>	<p>Do not place this Board in locations where excessive force is applied to the Board. Failure to do so may cause the PC board to warp, leading to breakage of the PC board, missing parts or malfunctioning parts.</p> <p>Do not clean this Board by using a rag containing chemicals such as benzene or thinner. Failure to do so will likely to cause this Board to deteriorate. When using a chemical cloth be sure to comply with any directions or warnings.</p> <p>Do not immediately turn on the power if you find that water or moisture had condensed onto the main unit after removing the board from the package. Condensation might occur on this Board when taking it out of the box, if the board is cool yet the room temperature is warm. Do not apply power to the Board while water or moisture has condensed on it because the moisture may cause the Board to break or may shorten the service life of the parts.</p> <p>When you first take this Board out of the box be sure to leave it at room temperature for a while before using it. If condensation or moisture has occurred on this Board, first wait for the moisture to fully evaporate before installing or connecting the Board to other devices.</p> <p>Unplug any USB cable and remove any battery when carrying out maintenance or updates to the devices in the kit. Remove power this way when connecting or disconnecting any boards, cables, or connections to the board.</p> <p>Do not disassemble, dismantle, modify, alter, or recycle parts unless they are clearly described as customizable in this Manual. Although this kit is customizable, if parts not specified in this Manual as customizable are modified in any way, then the overall product operation cannot be guaranteed. Please consult with Manufacturer beforehand if you wish to customize or modify any parts that are not described in this Manual as customizable.</p>

### 1.3 Developer Information

Macnica Americas  
 380 Stevens Ave. Suite 206  
 Solana Beach, CA 92075  
<http://www.macnica-na.com/>

## 1.4 Inquiries

In case of any inquiries about the use this product, please contact the local Macnica office or make inquiries through the contact form in the following web site:

<http://www.m-pression.com/contact>

Macnica companies:

- China & HK: Cytech Technology <http://www.cytech.com/>
- ASEAN & India: Cytech Global <http://www.cytechglobal.com/>
- Taiwan: Galaxy Far East Corp. <http://www.gfec.com.tw/>
- North America: Macnica Americas <http://www.macnica-na.com/>
- Brazil: Macnica DHW <http://www.macnicadhw.com.br/en/>
- Japan: Altima <http://www.altima.co.jp>  
Elsena <http://www.elsena.co.jp>

## 2. Overview

### 2.1 Introduction

The Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit is an evaluation board ideal for doing Internet of Things (IoT) proof-of-concept development. The board is designed to allow users to evaluate the use of and write software for the BCM20737S BLE module, Max10 FPGA, Energy Micro Giant Gecko microcontroller, and the sensors included in the kit for use in their own end products.

This kit encompasses everything users need to connect wirelessly using Bluetooth® Low Energy (BLE), collect information from the environment and even add custom designs using programmable logic.



Figure 2-1 Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit

The Odyssey MAX 10 FPGA and BLE Sensor Kit contains a BLE sensor board, an FPGA expansion board and a battery board. The BLE sensor board has a Bluetooth SMART device, a microcontroller and various sensors. The sensors include UV and ambient light, pulse rate and blood oximetry, temperature and humidity as well as acceleration. It also features a physical interface to either the battery board or the FPGA expansion board. The FPGA expansion board includes an FPGA for programmable logic, a microphone, LEDs, switches, push-buttons and expansion capabilities to an Arduino Nano. The battery board allows the user to power the BLE sensor board off a coin cell battery for true wireless applications. An I<sup>2</sup>C expansion port and a J-Link interface to program the microcontroller are also integrated on the battery board.

The kit is designed to interact with a Smartphone via the BLE link. This communications link is used to exercise the sensors and other interfaces by means of a complex firmware infrastructure. Depending on individual needs, users can easily use the existing framework to implement a design or users can modify the firmware to achieve unique design goals.

It is the object of this document to both introduce the first time user to the details of connecting to and using the Odyssey kit as well as expose some details of the underlying framework for more experienced engineers that need to build upon the existing infrastructure.

## 2.2 Prerequisites

In order to have a successful experience using the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit it is highly recommended users follow the guidelines as presented here. This User Guide is written in such a fashion that the kit will provide a sufficient platform for understanding the sensor integration and firmware framework.

### 2.2.1 Mpression MAX 10 FPGA and BLE Sensor Kit

At a minimum a user will need to have the BLE sensor board. This board provides the minimum hardware and software necessary to enable the Odyssey system. If a user wishes to explore the capability of using and/or creating custom logic in a programmable device the FPGA board must be installed. Likewise, to enable a wireless solution using the BLE sensor board, the battery board must be installed. See section 4 below for assembly details.

#### 2.2.2 Power

##### A. USB Cable

The primary means to power to the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit is through the micro USB connector on the BLE sensor board. If a user is using the USB cable solely to supply power, it is not necessary to connect the cable to a PC, a wall adapter instead. However, in order to upload any configuration files or use the Console Menu, the USB cable will need to be connected to a PC. This cable is not supplied with the kits. Note, there are no LEDs or other indicators providing feedback that the kit is powered.



Figure 2-2 Micro USB Cable

- To power-up the kit, simply insert a micro USB cable into the BLE sensor board receptacle and the other end to a USB port supplying a standard 5 volts.

##### B. FPGA board DIP pins

An alternate method to power the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit is by means of the DIP pins of the FPGA board. A solderless bread board can be used to mount the FPGA board and supply power to the system. See section 3.1.3 and the FAQ below for more details.

##### C. Battery

When using the battery board to supply power to the Odyssey kit, install a coin cell battery, part number CR2032, into the battery retainer. When the FPGA board is being used, the battery

cannot be utilized and the kit must be powered via the USB cable or via the DIP interface of the FPGA board.

## 2.2.3 Smartphone

### A. Device

In order to interact with the Odyssey kit users will need to have a device running either the Android operating system version 4.4 (KitKat) or higher or an Apple device running iOS7 or higher. The device must have Bluetooth SMART or be Bluetooth SMART Ready. The device must also have access to the Internet to enable the cloud based infrastructure of the Odyssey system. The Internet connection can be provided either via cellular data or WiFi. Throughout the remainder of this document, this device will be referred to as a “Smartphone.”

### B. Application

The Mpression software team has created a custom application that will be used on the Smartphone to interact with the Odyssey kit over the Bluetooth Low Energy communications link. This application is available on Google Play and the Apple App Store. It is titled “Mpression Odyssey.” (Note: At the time this User Guide was released, the iOS app is not available on the Apple App Store. It will be released in the near future.)

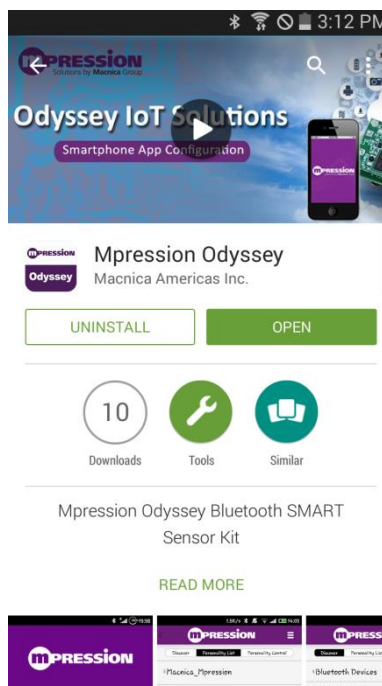


Figure 2-3 Mpression Odyssey App on Google Play Store

## 2.2.4 Web Utility Account

As users progress into understanding the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit, users will most likely want to create a custom personality to interact with the various sensors and create a custom Smartphone control page. In order to create a custom personality the user must create an account to access the Mpression Odyssey IoT Solutions Smartphone App Configuration Web Utility. The Web Utility is also the location where new and updated reference Smartphone app personalities can be downloaded from or used as templates for custom personalities.

The Mpression Odyssey IoT Solutions Smartphone App Configuration cloud based utility is located at <https://odyssey.m-pression.com>

Users can also create an account via the Mpression Odyssey Smartphone app directly.

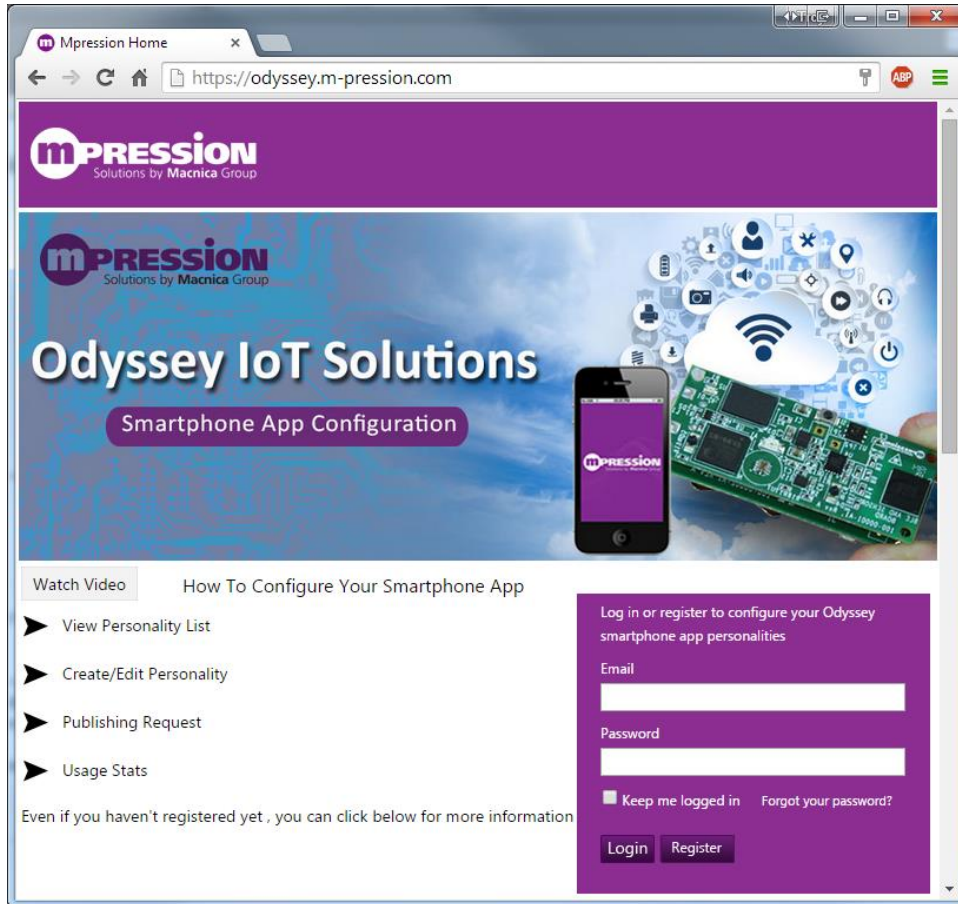


Figure 2-4 Mpression Smartphone App Configuration Web Utility

## 2.2.5 Serial Communications Terminal

In order to interact with the Console Menu on the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit, a serial communications terminal emulator is required. All development and testing was done using the open-source Tera Term emulator. All serial communications use the following settings.

Table 1 Serial Terminal Emulator Settings

Baud rate	115200
Data bits	8
Parity	None
Stop bits	1
Flow control	None
New-line	Auto or LF

The installation package is available here:

<http://en.sourceforge.jp/projects/ttssh2/downloads/62353/teraterm-4.85.exe/>



## 2.2.6 USB Virtual COM Port Driver

In order for the user's PC to recognize the Odyssey MAX 10 FPGA and BLE Sensor Kit and communicate with it using the above serial terminal emulator, a Silicon Labs USB virtual COM port driver needs to be installed. It can be obtained here:

<http://www.m-pression.com/solutions/boards/odyssey/odyssey-downloads>

## 2.2.7 Development Software

Currently, it is beyond the scope of this document to detail the specifics of doing stand-alone microcontroller, BLE device or FPGA development. However, the tools used are as follows.

### A. Silicon Labs Simplicity Studio

To develop software for the EFM32 microcontroller Simplicity Studio is required.

<http://www.silabs.com/products/mcu/Pages/simplicity-studio.aspx>

### B. Broadcom WICED Smart SDK

To program the Broadcom BLE device, a batch file as described in section 5.2.1 below relies on several files included as part of the WICED Smart SDK. Users will need to register with the Broadcom Community to gain access to and install the development tools.

To develop software for the Broadcom BLE device, the WICED Smart SDK 2.1.1 or higher is required. <http://community.broadcom.com/community/wiced-smart>

### C. Altera Quartus-II

Quartus-II version 14.1 or later is required to develop MAX 10 FPGA designs.

<https://newwww.altera.com/products/design-software/fpga-design/quartus-ii/overview.html>

## 3. Components & Interfaces

### 3.1 The Odyssey IoT Solutions Boards

As discussed above, the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit is comprised of several boards. Below, each board is independently discussed and features highlighted.



Figure 3-1 Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit

#### 3.1.1 BLE sensor board

The BLE sensor board is the heart and soul of the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit. Fundamentally, it includes most of the sensors, the Bluetooth Low Energy communication device and a microcontroller.

The BLE device, EFM32 microcontroller and sensors communicate via a network of I<sup>2</sup>C and SPI busses while configuration and interactions with the outside world are handled via USB and JTAG ports as seen below.

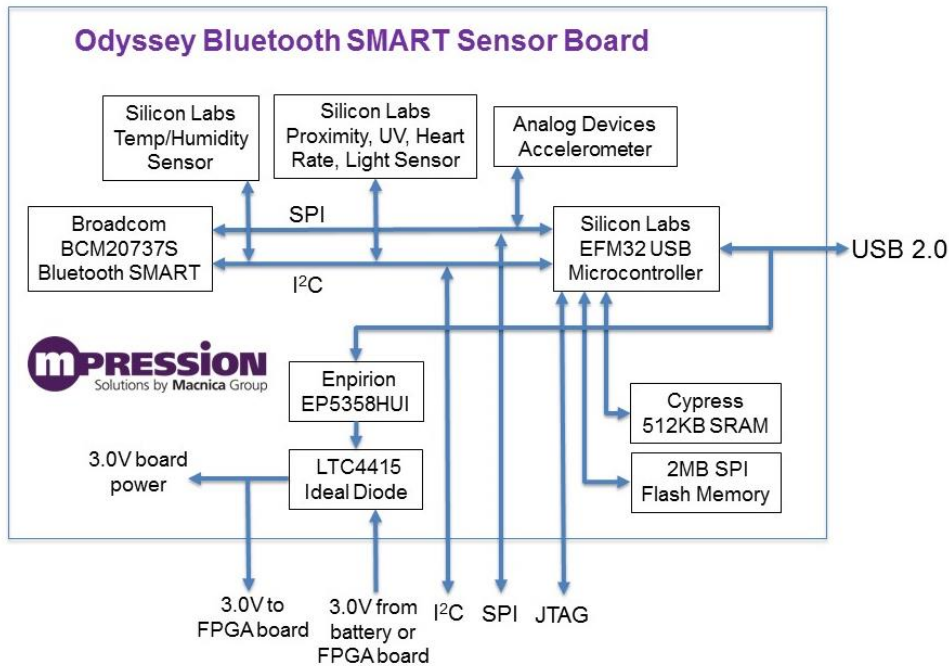


Figure 3-2 BLE Sensor Board Block Diagram

Details of each component are outlined here.

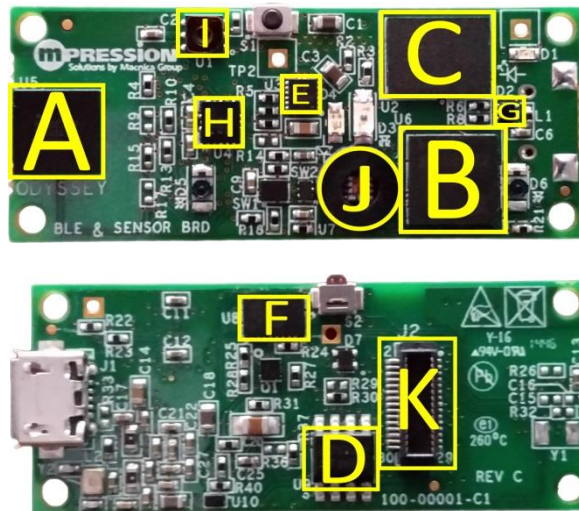


Figure 3-3 BLE Sensor Board

**A. Broadcom BCM20737S Bluetooth SMART (BLE) module (U5)**

The BCM20737S from Broadcom gives the board its wireless connectivity and plays the master role in executing and interpreting commands for the Odyssey system. The module is a Bluetooth Low Energy (BLE/Bluetooth SMART) module that includes the BLE radio, an onboard ARM Corex-M3 processor, memory, and peripherals making it an all-in-one solution for BLE connected devices and wearables. It supports an easy-to-use embedded stack for host and controller functions and includes full capability for developing and implementing new profiles and applications. In addition to an onboard stack and profiles, this 6.5 x 6.5 SIP Module includes BT SMART transceiver, crystal, EEPROM, antenna, passives, and regulatory approvals as well. The module uses a development environment called the WICED Smart SDK which can be downloaded from Broadcom's community portal. See section 2.2.7B.

#### B. Silicon Labs EFM32GG395F512 low-power Giant Gecko 32-bit microcontroller (U6)

The EFM32 low-power Giant Gecko microcontroller provides the USB interface and acts as the programmer/debug port for the above Broadcom Bluetooth SMART (BCM20737S) module. It also manages the loading of the MAX 10 FPGA. In addition, it enables loading of new "personalities" into the flash memory on the Odyssey board so users can leverage the Mpression smartphone app in their custom designs. Certain sensor functions are also enabled by this processor.

#### C. Cypress CY62148 4MB SRAM (U2)

The CY62148EV30 is a 4-Mbit MoBL® SRAM that serves as an expansion memory for the EFM32 microcontroller to support FPGA JTAG loading (JAM player, which is a very memory-intensive function). The CY62148EV30 is designed for extending battery life in portable devices, with ultra-low active current and an automatic power-down feature that consumes only 1uA typical standby current. Cypress is well known for its broad array of reliable, high-performance SRAMs and nonvolatile RAMs used in mission-critical systems.

#### D. Micron M25PX16 Flash (U9)

The M25PX16 is a 16Mbit NOR SPI flash from Micron used to store "personality" information on the Odyssey board. The device is controlled from the EFM32 and features an operating voltage down to 2.3V and deep power-down current of 10uA max.

#### E. Enpirion EP5358 power regulator (U3)

The EP5358HUI PowerSoC from Altera is a tiny DC-DC switching regulator with built-in inductor. It provides up to 600mA of power to the Odyssey board from the USB port at up to 93% efficiency which helps to avoid heating the nearby humidity/temp sensor.

#### F. Linear Technology LTC4415 dual ideal diode / power path controller (U8)

The LTC4415 is a dual ideal diode allowing power to be switched seamlessly between the coin cell battery and the USB source without the voltage drop associated with diode-or'ing.

#### G. Wurth WE-TVS diode (D2)

It's always important to protect high speed data lines such as the cabled USB 2.0 interface from over-voltage caused by electrostatic discharge (ESD), electrical fast transients (EFT) and lightning strikes (surge). The Odyssey board utilizes the Wurth Elektronik WE-TVS Diode array to provide protection for the USB port lines with very low (<2pf) capacitance.

#### H. Analog Devices ADXL362 accelerometer (U4)

The ADXL362 accelerometer allows the board to detect and report motion to the smartphone app based on a threshold set by the user. The ADXL362 3-axis accelerometer is one of the lowest power in the industry at less than 2µA @ 100Hz sampling, and also includes a unique mode allowing motion-triggered wake up which operates at an amazing 270 nano Amps. This allows a system to be design such that it is optionally powered down completely until motion is detected, saving battery power.

#### I. Silicon Labs Si7020 temperature/humidity sensor (U1)

The Si7020 allows the Odyssey to measure temperature and relative humidity and report the values via the I<sup>2</sup>C interface. It's factory calibrated and available with an optional factory-installed protective cover.

J. Silicon Labs Si1147 proximity/UV/ambient light/heart rate monitor sensor (U7)

The Si1147 provides a variety of sensor functions including measuring UV levels, ambient light, proximity and gestures, heart rate, and blood oximetry. The data is communicated over the I<sup>2</sup>C interface.

K. Molex 501920-3001 mating connector (J2)

The BLE sensor board mates to either the battery board or FPGA board via this 30 position (2x15) plug.

L. Push buttons

Table 2 BLE Sensor Board Push Buttons

S1	BLE Reset
S2	EFM32 Reset

M. LEDs/Diodes

Table 3 BLE Sensor Board LEDs/Diodes

D1	Green LED	EFM32 GPIO
D2	TVS Diode	USB Protection
D3	IR LED	Pulse Rate/Ox Detector
D4	Red LED	Pulse Rate/Ox Detector
D5	IR Emitter	Proximity Detector
D6	IR Emitter	Proximity Detector

### 3.1.2 Battery board

The battery board serves many purposes. As to be expected it can be used to supply power to the BLE sensor kit for true wireless operation. This board also includes connectivity for direct EFM32 microcontroller programming and debug as well as I<sup>2</sup>C expansion.

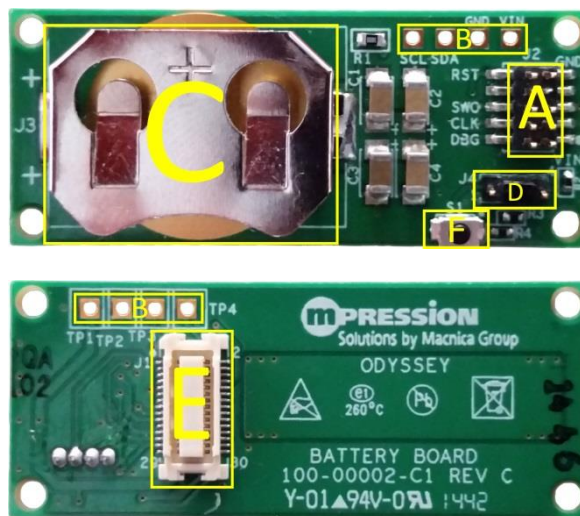


Figure 3-4 Battery Board

Details of each component are outlined here.

#### A. Samtec 2x5 programming and debug header (J2)

This 50MIL spaced 10 position header is used to connect a Segger J-Link ARM programming and debug module.

#### B. I<sup>2</sup>C expansion port (SCL, SDA, GNC, VIN/TP1-4)

The battery board allows a user to connect any I<sup>2</sup>C capable device via these 4 unpopulated through-holes. The signaling for this port comes via the J1 mating connector.

#### C. CR2032 battery retainer (J3)

Battery power to the BLE sensor kit is achieved by installing a CR2032 coin cell battery in this retainer. Operational power is only used from the battery when the USB or FPGA DIP interface supplied power is unavailable as determined by the LTC4415 dual ideal diode on the BLE sensor board.

#### D. Samtec 1x4 header (J4)

This 50MIL spaced 4 position header is used to control the BLE device mode of operation on the BLE sensor board.

Table 4 Battery Board J4 Jumper Settings

3-4 Shorted	BLE Normal Operating Mode
2-3 Shorted	BLE Programming Mode

#### E. Molex 52991-0308 mating connector (J1)

The battery board will mate to the BLE sensor board via this 30 position (2x15) receptacle.

#### F. EFM32/BLE GPIO (S1)

This push-button is used as a GPIO into the BLE and the EFM32. Its primary function is to enable the Console Menu.

### 3.1.3 FPGA board

The FPGA board is ideal for doing proof-of-concept experiments that require custom programmable logic. The board's primary feature is the Altera MAX 10 FPGA. Along with the enormous processing power of the FPGA, this board includes additional interfaces such as a MEMS microphone, LEDs, dip switches and push-buttons for user interaction. This FPGA board can also be expanded by means of the DIP connectors that are Arduino Nano compatible. It is also possible to use the FPGA board in a stand-alone configuration. See Tips & Tricks section below.

The FPGA board communicates with the BLE sensor board via I<sup>2</sup>C and the FPGA can be programmed by either the EFM32 on the BLE sensor board or directly through JTAG as shown here.

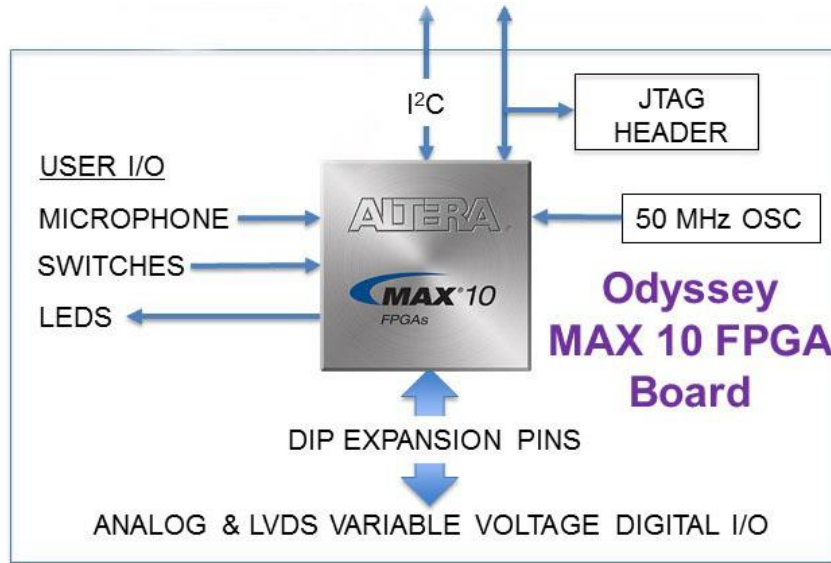


Figure 3-5 MAX 10 FPGA Board Block Diagram

Details of each component are outlined here.

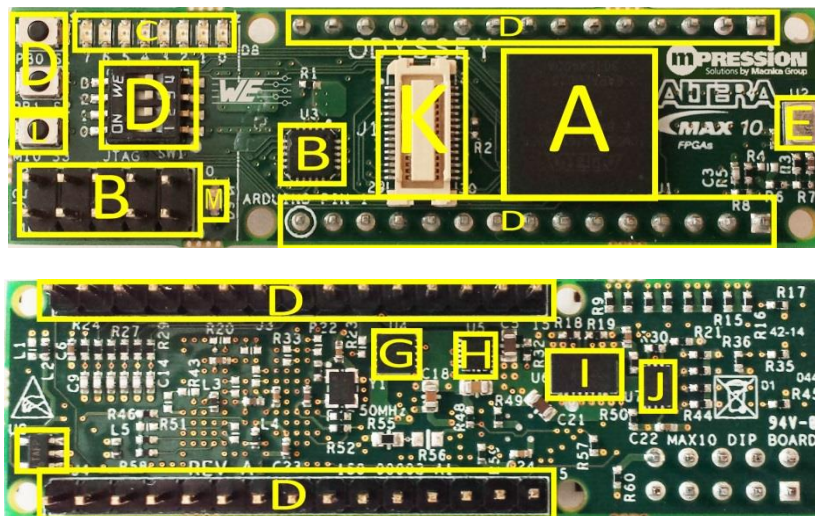


Figure 3-6 MAX 10 FPGA Board

**A. Altera® MAX 10 FPGA - 10M08SAU169C8GES (U1)**

An Altera 10M08 FPGA is the featured device on the MAX® 10 FPGA Evaluation board. On this board, a single core voltage supply version of the MAX 10 FPGA is used in a space-efficient 169-pin UPGA that measures 11x11mm. Integrated features include analog-to-digital converters (ADCs) and dual configuration flash allowing a user to store and dynamically switch between two images on a single chip. Unlike CPLDs, MAX 10 FPGAs also include full featured FPGA capabilities such as digital signal processing (DSP) blocks, PLLs, block memory, and soft DDR3 memory controllers (on certain devices). The MAX 10 FPGA also enables the use of the Nios® II soft core embedded processor.

### B. Analog Devices ADG788 IC switch (U3, J2)

An Analog Devices ADG788 quad SPDT analog IC switch is used to switch the JTAG connection between the JTAG header (2x5 pin header from Wurth) and the connector from the BLE sensor board. When a USB Blaster is plugged into the MAX 10 FPGA board, it automatically switches the JTAG lines so that MAX 10 FPGA loading can be performed from a USB Blaster rather than from the EFM32 Giant Gecko microcontroller on the BLE sensor board.

### C. Wurth Elektronik WE-SMCW LEDs (D1-D8)

On the Odyssey MAX 10 FPGA board, mono-color bright green top view style LEDs from Wurth Elektronik is used for signaling.

### D. Switches, pushbuttons, and DIP pins (S1, S2, SW1, J3, J4)

Other user IO is provided by switches (Wurth), pushbutton switches (Omron), and two 15-pin gold plated headers (Wurth) that give the FPGA board the appearance similar to the dual in-line packages (DIP) common in the '80s and '90s. This DIP design is good for easy prototyping, debug and supplying power. See section 3.2 below to access the schematics for pinout details. Additionally, a close look at the pinout shows that it is compatible with the Arduino Nano boards (Note: Pin 1 on J4 does NOT align with the pin 1 silk screen label for Arduino on the MAX 10 FPGA board.)

### E. Audio input circuit (U2, U8)

An interesting input option for Odyssey is the MEMS microphone from Invensense. Invensense makes versions of this tiny microphone with digital outputs, but this one drives an analog signal into a Linear Technology LT6220CS5 op amp used as a non-inverting gain stage (gain of about 72). This is then fed into one of the analog inputs on the MAX 10 FPGA.

### F. Power tree

The FPGA board is designed to operate on a desk, powered only by the USB via the BLE sensor board, or bread-boarded with power supplied from a bench supply (with or without the BLE sensor board attached). Also, the MAX 10 FPGA has multiple IO banks that can run at different voltages. The Odyssey system is designed to allow two of these banks to be powered from a variable power supply through one of the DIP/header pins. If power is not supplied to the variable rail, the variable banks default to 2.5V. A power diagram is shown below, and the components broken out separately after that.

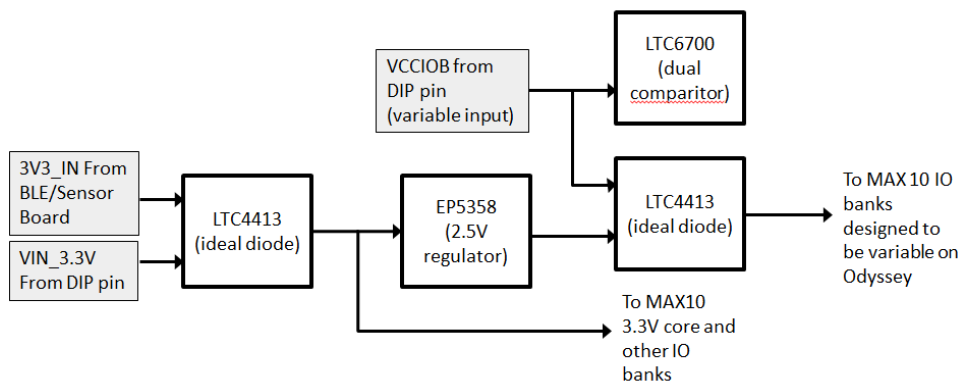


Figure 3-7 FPGA Board Power Tree



### G. Linear Technology LTC4413 dual ideal diode / power path controller (U4)

The Linear Technology LTC4413 dual ideal diode selects where to source the 3.3V board power from either the 3.3V\_in DIP pin of the FPGA board or the power supplied from the BLE sensor board. The dual ideal diode does this seamlessly without the voltage drop associated with diode or'ing. This allows the FPGA board to be completely powered by the USB port on the BLE sensor board or from a bench supply when prototyping with a breadboard.

### H. Enpirion EP5358 power regulator (U5)

The EP5358LUI PowerSoC is a tiny DC-DC switching regulator with built-in inductor that creates a steady 2.5V rail as an option for two of the MAX 10 IO banks. 2.5V is useful when driving LVDS or a 2.5V IO standard (2.5V LVCMOS for example). It provides up to 600mA of power at up to 93% efficiency.

### I. Linear Technology LTC4415 dual ideal diode / power path controller (U6)

The Linear Technology LTC4415 dual ideal diode seamlessly selects the IO voltage for the variable IO voltage rail on the MAX 10 FPGA without the voltage drop associated with diode or'ing. IO power can be supplied via a DIP pin or from the on-board 2.5V supply created by the Enpirion power regulator device.

### J. Linear Technology LTC6700 dual comparator (U7)

A Linear Technology LTC6700-1 comparator is used in conjunction with the LTC4415 dual ideal diode to determine which MAX 10 FPGA IO rail supply to use. If the power supplied via the DIP pin falls into the range of 1.7-3.4V, the DIP pin power rail will be used to power the MAX 10 FPGA's variable IO banks. If the DIP supply falls outside of that range, the static 2.5V rail will be selected.

### K. Molex 52991-0308 mating connector (J1)

The FPGA board will mate to the BLE sensor board via this 30 position (2x15) receptacle.

### L. EFM32/BLE GPIO (S3)

This push-button is used as a GPIO into the BLE and the EFM32. Its primary function is to enable the Console Menu.

### M. IR LED (D9)

Infrared LED that can be used as an IR link to IR receive capable devices. This LED is connected to the MAX 10 FPGA on a GPIO pin.

## 3.2 Engineering Documents

Macnica Americas has made all the engineering details for each board readily available on the web.

### 3.2.1 BLE sensor board

Table 5 BLE Sensor Board Documents

Schematic	<a href="http://www.m-pression.com/odyssey-ble-sch">www.m-pression.com/odyssey-ble-sch</a>
Bill of materials	<a href="http://www.m-pression.com/odyssey-ble-bom">www.m-pression.com/odyssey-ble-bom</a>
Board drawings	<a href="http://www.m-pression.com/odyssey-ble-assy">www.m-pression.com/odyssey-ble-assy</a>

## 3.2.2 Battery board

Table 6 Battery Board Documents

Schematic	<a href="http://www.m-pression.com/odyssey-battery-sch">www.m-pression.com/odyssey-battery-sch</a>
Bill of materials	<a href="http://www.m-pression.com/odyssey-battery-bom">www.m-pression.com/odyssey-battery-bom</a>
Board drawings	<a href="http://www.m-pression.com/odyssey-battery-assy">www.m-pression.com/odyssey-battery-assy</a>

## 3.2.3 MAX 10 FPGA board

Table 7 MAX 10 FPGA Board Documents

Schematic	<a href="http://www.m-pression.com/odyssey-max10-sch">www.m-pression.com/odyssey-max10-sch</a>
Bill of materials	<a href="http://www.m-pression.com/odyssey-max10-bom">www.m-pression.com/odyssey-max10-bom</a>
Board drawings	<a href="http://www.m-pression.com/odyssey-max10-assy">www.m-pression.com/odyssey-max10-assy</a>

## 4. Assembly/Disassembly

### 4.1 Included HW

The primary BLE sensor board will come installed with the FPGA board by means of 4 nylon stand-off connectors and washers. The kit can be used immediately out-of-the box without any modifications. However, the need may arise to disassemble/reassemble the kit. Please follow the instructions below to avoid damage.

The kit comes with the boards mated and no additional parts are required to connect the boards. If additional or replacement parts are needed, the part numbers are as follows.

Table 8 Assembly Parts

Part	Description	Manufacturer	Part Number
Stand-offs	Polyamide spacer	Wurth	709651000
Washers*	Polyamide spacer	Seatrom Manufacturing	5610-424-20
Heart Rate O-ring	ID 1/8", OD 1/4"	McMaster-Carr	4061T111
Jumper shunts	1.27 mm shunt	Digi-Key	S9345-ND
Alternate shunts	1.27 mm shunt	Harwin, Inc	M50-1900005

\*Not required for board mating – creates tighter fit.

### 4.2 Tools

Due to the small form factor and delicate nature of the mating connector, no large-scale tools are to be used for disassemble and reassembly. The only tool necessary is a pair of needle nose pliers.

### 4.3 Disassembly

As shipped, the method by which the boards are connected are the locking stand-offs. In order to pull the two boards apart, the locking portion of the stand-offs need to be compressed.

Gently use needle nose pliers to squeeze just the locking head of the stand-off while gently pulling the boards apart, allowing the head to pull out of the mounting hole. Do this for each of the four corners of either the battery board or the FPGA board, depending on the kit setup. Optionally, leave the stand-offs and washers attached to the BLE sensor board as they can be used for reassembly. Once the stand-offs have been released use only hands to separate the two boards.

### 4.4 Reassembly

Reassembly of the kit requires no tools other than hands. Simply line-up the mating connector between the two boards along with the stand-offs at each corner and gently press the boards together at each corner until the stand-offs are all seated and then ensure the mating connector is seated as well. If a tighter fit is desired, install the washers over the locking head of the stand-offs.

## 5. Firmware Updates

As the development team continues to improve and enhance the functionality of the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit, updates will be made available that a user can download to the board. Both the microcontroller and Bluetooth LE device can be updated. All updates can be accessed via the Mpression Odyssey IoT Solutions downloads page.

<http://www.m-pression.com/solutions/boards/odyssey/odyssey-downloads>

### 5.1 EFM32

As discussed earlier, the Silicon Labs EFM32 microcontroller is an integral part of the Odyssey system. There are two methods by which a user can update the EFM32. For users that are not familiar with Simplicity Studio and the EA Commander, the Silicon Labs EFM32 development platform, a serial console method can be used.

#### 5.1.1 Serial console

The EFM32 can directly receive a firmware update via the USB serial connection.

- Be sure the Odyssey kit is unpowered first then connect the battery board to the BLE sensor board according to section 4 above.
- In order to place the EFM32 microcontroller into the appropriate maintenance mode, create a short between pins 1 and 4 on J2 (VIN and CLK) on the battery board using a jumper with micro-clips. **Be sure not to short the adjacent pins.**

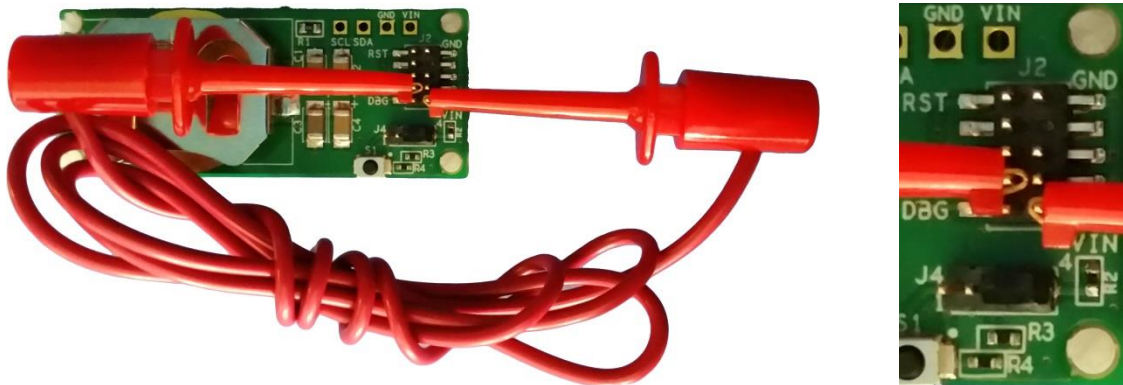


Figure 5-1 EFM32 Programming Mode Short

- Power-up the kit by plugging the USB cable into a PC with the serial terminal application as described in section 2.2.5 above. Note: The short needs to be in place when power is first applied.
- Using the PC's Device Manager, determine the COM port number assigned to the Odyssey kit and open a Tera Term session with the following settings.

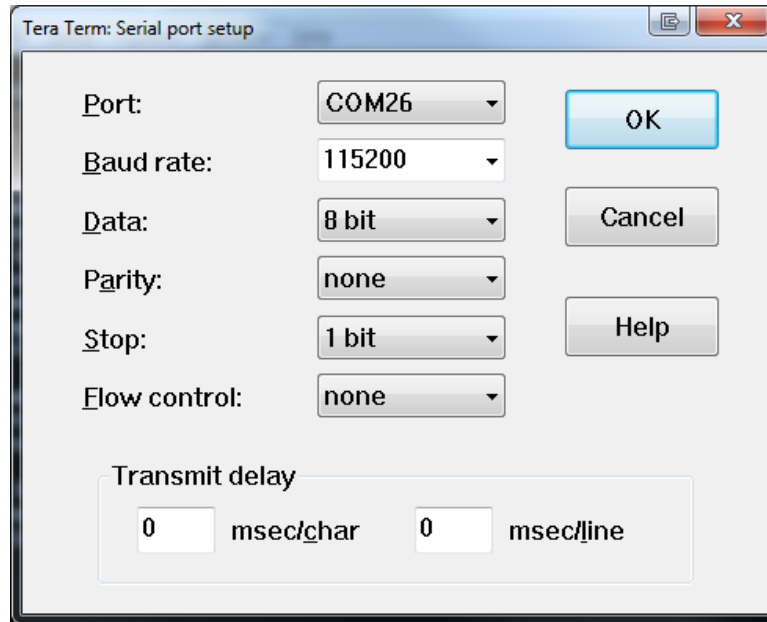


Figure 5-2 EFM32 Programming Serial Console Configuration

- First type “i” to verify a good console session is established. The boot-loader information will be displayed.
- Type “u” in the Tera Term session window – this starts the programming procedure. The console will show the word “Ready” followed by “C”. The “C” will repeat about every 5 seconds if the file transfer is not immediately started.
- Transfer the EFM32 update file using the XMODEM CRC protocol.
  1. In Tera Term, go to the **File** menu
  2. Select **Transfer** menu option
  3. Select **XMODEM** menu option
  4. Select **Send...** menu option

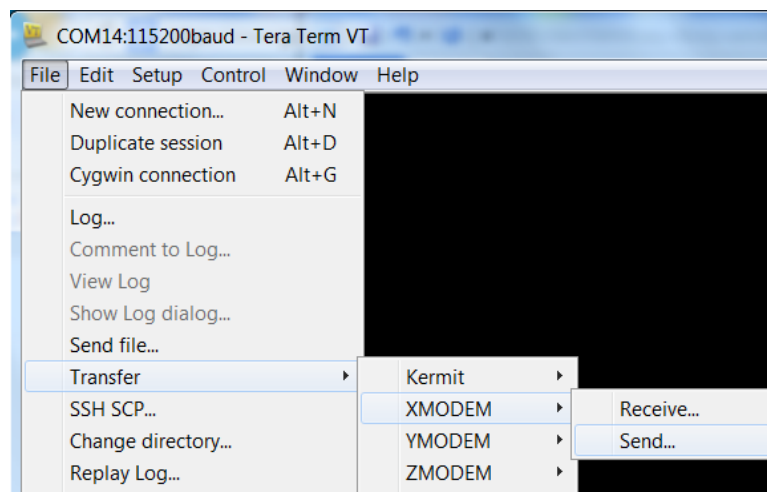


Figure 5-3 EFM32 Programming via XMODEM Protocol

5. This opens a file selection window. Browse to the EFM update image file, select the **CRC** radio button, and click **Open**.

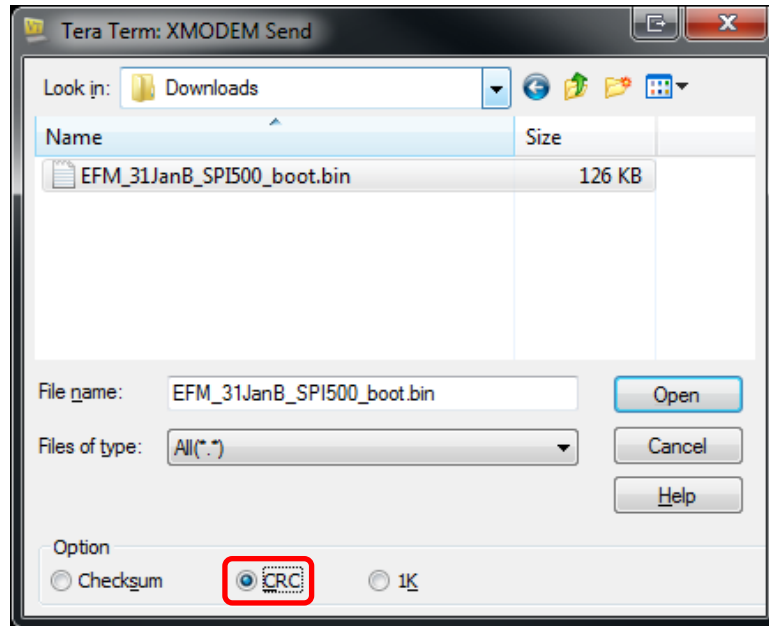


Figure 5-4 EFM32 Programming File Selection

6. The file will be transferred and programmed into the EFM32. This takes about 2-3 seconds.
7. Programming is now complete. Option: typing “c” in Tera Term will show the CRC as a check.
8. Disconnect the USB cable and remove the short.

### 5.1.2 Simplicity Studio

It is beyond the current scope of this document to detail programming the EFM32 via the Silicon Labs Simplicity Studio development tools.

Please see the Silicon Labs web site for additional details.

<http://www.silabs.com/products/mcu/Pages/32-bit-microcontroller-tools.aspx>

## 5.2 BLE

The Broadcom Bluetooth SMART device is an integral part of the Odyssey system. From a functionality point of view, the BLE device provides the wireless communications link between the Smartphone and the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit. It also handles command processing and masters the I<sup>2</sup>C and SPI busses. As part of this Bluetooth wireless protocol, a default Bluetooth (BT) MAC address\* is assigned to each Odyssey at the factory but it may be necessary to change this address to fit a user’s needs. The BLE device also encompasses a highly integrated firmware architecture that enables the Odyssey to take on different personalities and interact with the many different sensors. There are two methods by which a user can update the BLE firmware or just the BT MAC address. For users that are not familiar with the Broadcom Bluetooth SMART development platform, to do the firmware updates, a virtual communications port script can be used.

\* Note: All factory programmed kits will have the SAME MAC address. If more than one kit is to be used within the same vicinity, it will be necessary to change the MAC address to unique values.

## 5.2.1 Virtual COM Port

The BLE device can directly receive a firmware update, inclusive of a new BT MAC address, via the USB connection using a script file created by the Mpression development team. This batch file and the BLE device update firmware files can be accessed via the link in section 5 above. The batch file requires that the Broadcom WICED SMART SDK is accessible & installed on a PC. See section 2.2.7B above.

- First, the BLE Firmware Update Utility script and firmware files need to be downloaded to the PC. Place the script and firmware files on the PC in a location of choice. **Note: The firmware file name and directory path cannot exceed 260 characters in overall length.**
- Be sure the Odyssey kit is unpowered first and then connect the battery board to the BLE sensor board according to section 4 above.
- In order to place the BLE device into the appropriate programming mode, attach the jumper on the middle pins of J4 (pins 2 and 3) using the shunt provided.



Figure 5-5 BLE Device Programming Jumper

- Power-up the kit by plugging the USB cable into the board and to the PC with installed BLE Firmware Update Utility script and firmware files.
- Double click on the **BLE\_Firmware\_Update.bat** file. If the `SDK_INSTALL` environment variable is not set, the utility will prompt the user to select the Broadcom WICED Smart SDK installation folder. See section 2.2.7B above. Select the folder which contains the “WICED-Smart-SDK” folder. (NOT the WICED-Smart-SDK folder, but the folder that contains it. “~/WicedSmart 2.2.0/SDK” in the example below.) Note: The batch file will create a permanent environment variable so that this step will not be necessary the next time the batch file is run.

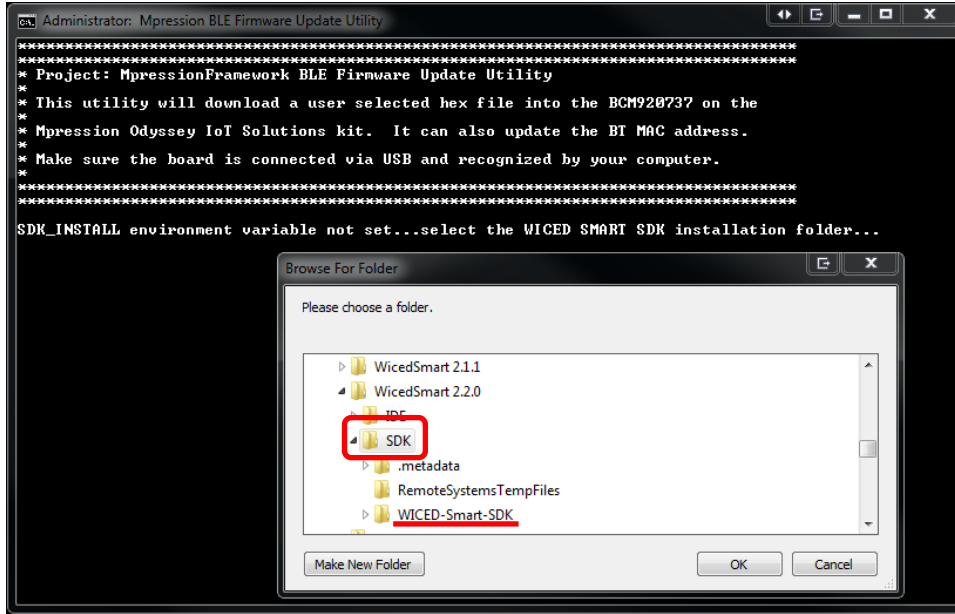


Figure 5-6 Setting SDK\_INSTALL Environment Variable

- The BLE Firmware Update utility will automatically scan all COM ports on the PC and determine on which one the kit is active. If the device is not detected, verify the J4 jumper is installed correctly and that there are no other serial console applications open and accessing the kit via the virtual COM port. i.e. Tera Term from either EFM32 updates or Console Menu.

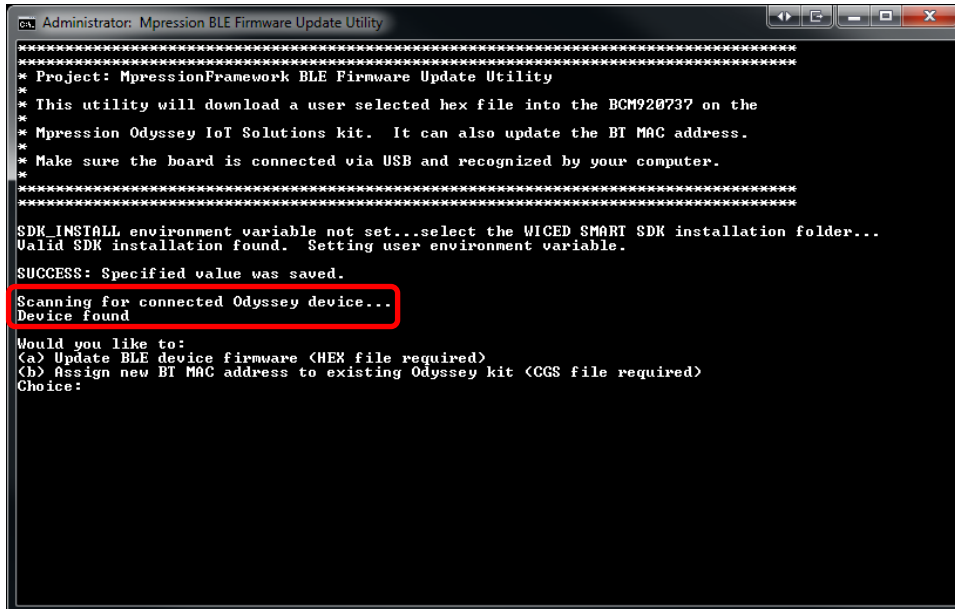


Figure 5-7 Auto-detect of Odyssey kit

- At this point the user needs to choose to either update the BLE firmware or change the BT MAC address.
- If the user chooses option “a” the script will prompt for the BLE device firmware update hex file. (Note: The hex file name and directory path cannot exceed 260 characters in overall length.)



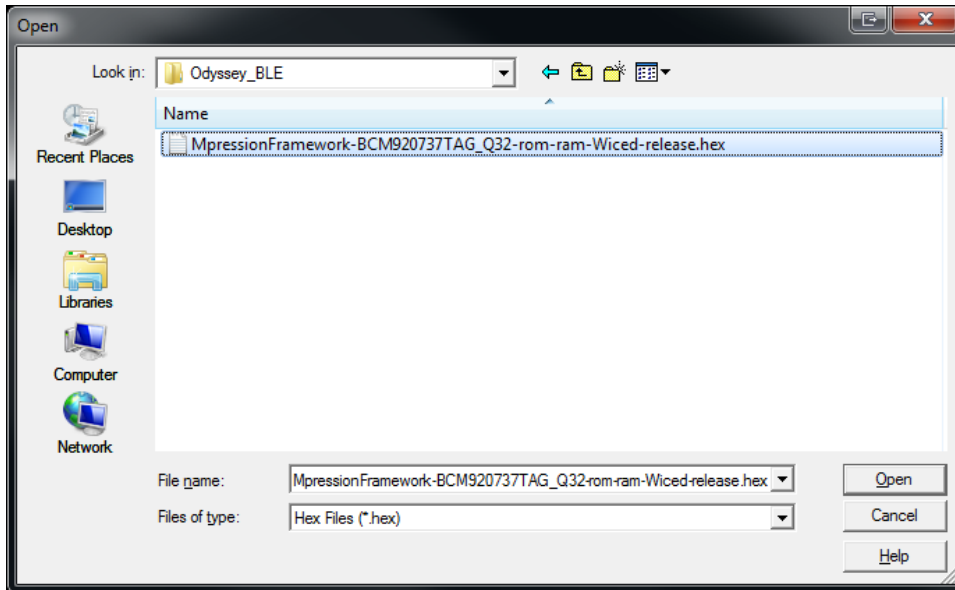


Figure 5-8 File Selection for BLE Device Firmware Update

- The BLE module will program in about 15 seconds and it will notify the user of success when complete. Press any key to close the window.

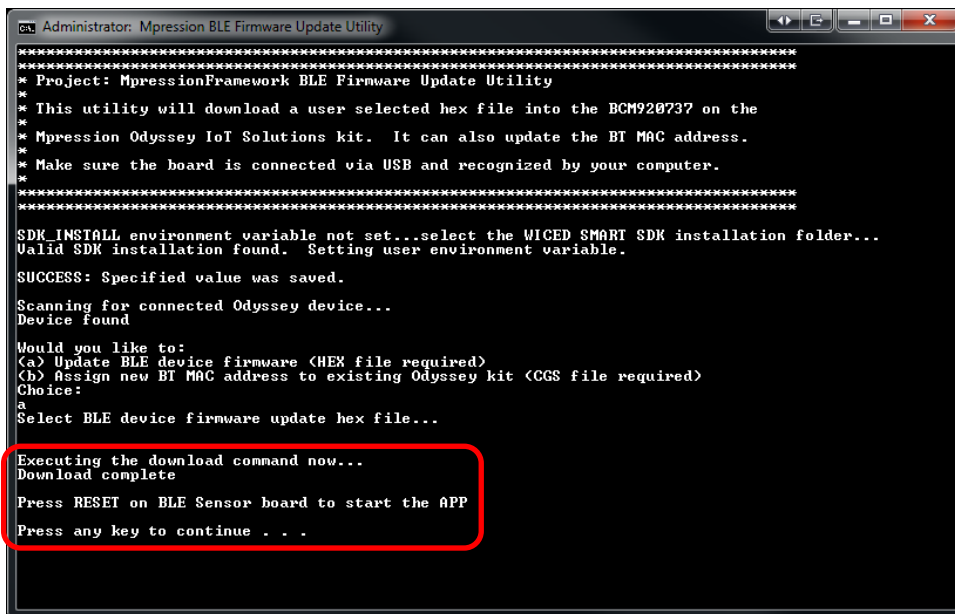


Figure 5-9 BLE Device Programming Status

- If the user chooses option “b” the script will prompt for the BLE device configuration cgs file. (Note: The cgs file name and directory path cannot exceed 260 characters in overall length.)

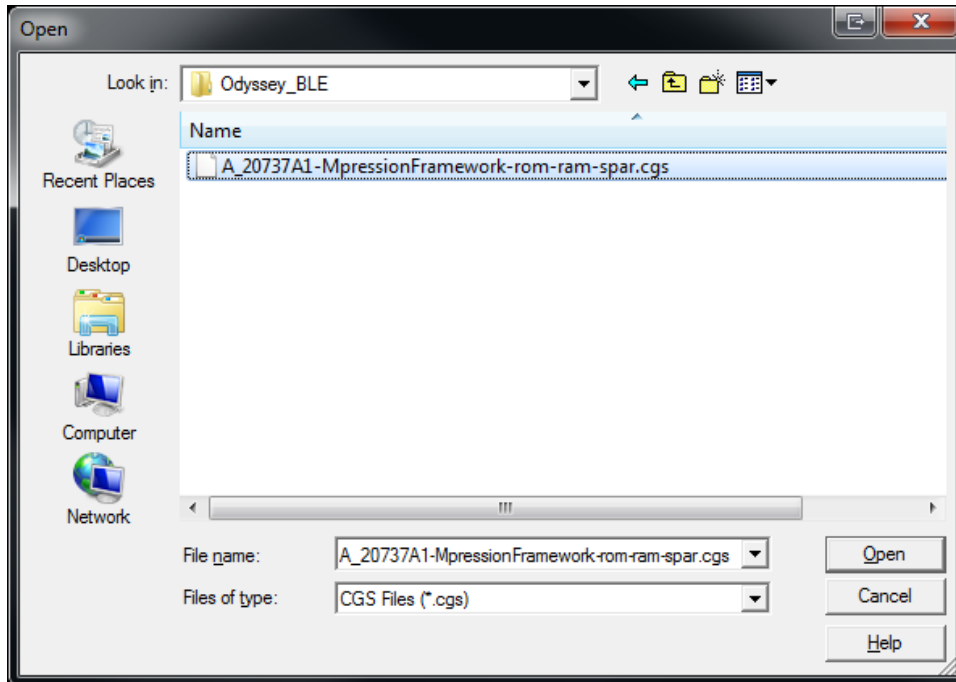


Figure 5-10 Configuration File Selection for BLE MAC Address Update

- The user will then be prompted to enter a new 12 digit hexadecimal MAC address. Enter the 12 digits with no spaces, colons, dashes or other delimiters between digits. i.e. 1234567890AB

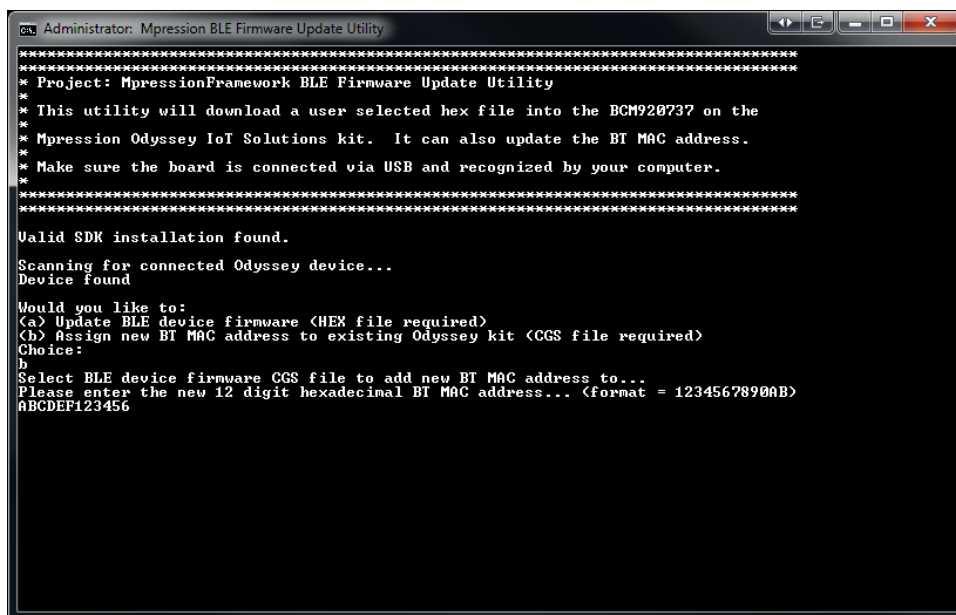


Figure 5-11 Setting New MAC Address

- The BLE module will program in about 15 seconds and it will notify the user of success when complete. Press any key to close the window.

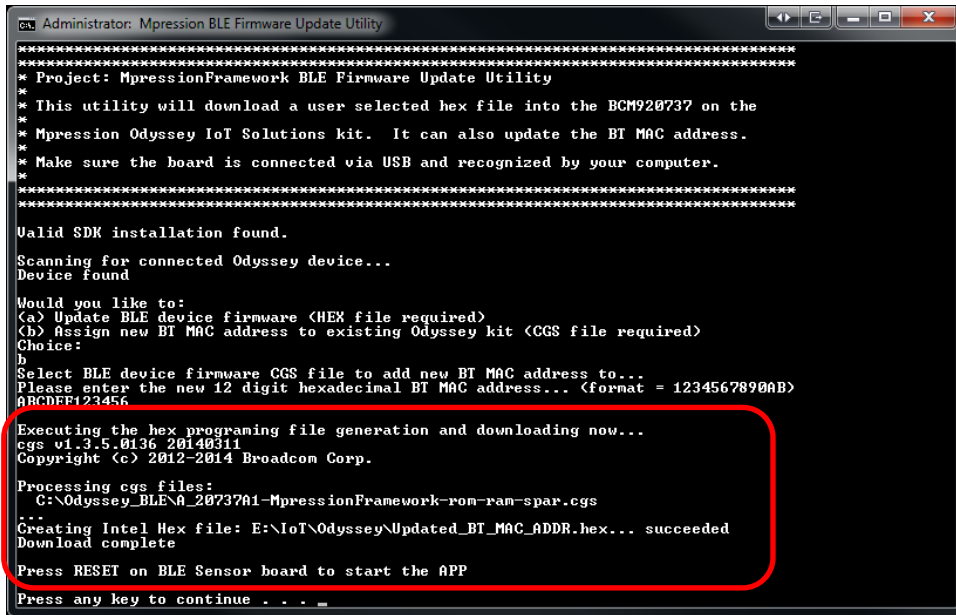


Figure 5-12 BLE Device Programming Status

- Programming is now complete. Unplug the USB cable and move the jumper on J4 to the pins closest to outside edge of the board (on pins 3 and 4).

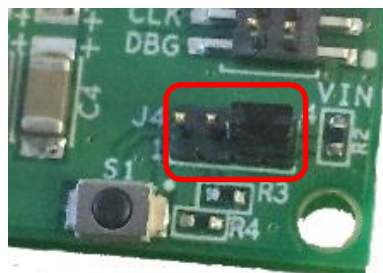


Figure 5-13 BLE Device Run Mode Jumper

## 5.2.2 WICED SDK

It is beyond the current scope of this document to detail programming the BLE device via the Broadcom WICED SMART SDK development tools.

Please see the Broadcom Community web site for additional details.

<http://community.broadcom.com/community/wiced-smart>

Macnica has also produced a series of Broadcom Bluetooth SMART videos to help bring a user up to speed.

- Bluetooth SMART Introduction and Concepts  
<http://www.macnica-na.com/vworkshops/bluetooth-smart-1>
- The WICED SMART Bluetooth Environment  
<http://www.macnica-na.com/vworkshops/bluetooth-smart-2>
- Build Your Own WICED SMART Bluetooth LE Device Application  
<http://community.broadcom.com/videos/1229>

## 6. Run Preprogrammed Demonstrations (Personalities)

---

### 6.1 Apply Power

The Odyssey MAX 10 FPGA and BLE Sensor Kit comes assembled with the MAX 10 FPGA board attached to the BLE sensor board. The power for this assembly is derived from the micro USB connector located on the BLE sensor board. This cable is not supplied with the kits.



Figure 6-1 Micro USB Cable

- To power-up the kit, simply insert a micro USB cable into the receptacle and the other end to a USB port supplying a standard 5 volts.

A user can use a laptop, PC or wall charger. Note, there are no LEDs or other indicators providing feedback that the kit is powered.

### 6.2 Launch Smartphone App

In order to run the demonstration personalities loaded into the flash on the kit as described in section 2.2.3, the user must have the Mpression Smartphone application installed on a Bluetooth Low Energy capable device.

- Launch the application which by default starts on the log in screen.

Note: The Smartphone app will verify that the device's Bluetooth radio is turned on. It will request permission to turn on the radio if it is off. Or, the user can manually ensure the radio is on before launching the app.

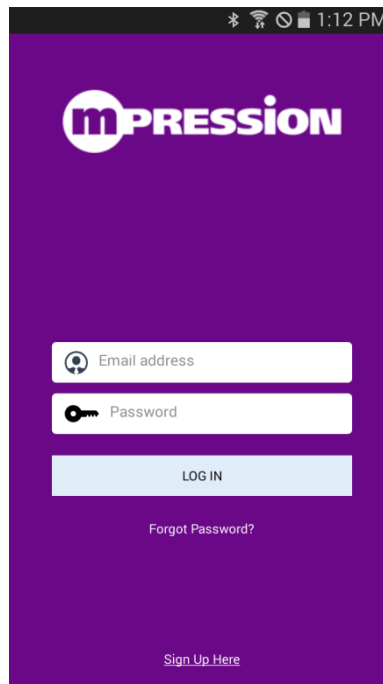


Figure 6-2 Login Screen

- If a user has not created an Mpression web-utility account, select the “Sign Up Here” link on the bottom of the page to create an account.
- If a user has previously created an account log in with the credentials used when the account was created.

Note: The application will stay logged in so the user does not have to re-enter credentials every time the application is launched unless explicitly logged out. However, if the user changes the password via the web based utility, the user will need to log out and log back in to the Smartphone app.

As users become more familiar and faster with the power-up, login and Smartphone application launch sequence, be sure to allow the Odyssey system to fully boot (approximately 7 seconds) before attempting to connect to the device over Bluetooth.

The next screen that appears will be the device discovery screen.

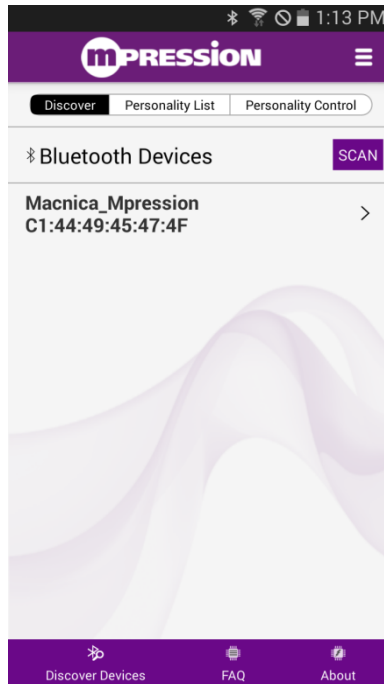


Figure 6-3 Device Discovery

Here a user will see the Odyssey kit in the list of available devices to connect to. If the user does not see it in the list, touch on the SCAN button to re-initiate the scanning sequence and verify the kit is powered. (Note: The BT MAC address may vary as compared to the above screen shot.)

## 6.3 Connect to Device

- Select the “Macnica\_Mpression” device to advance to the next screen where the list of preloaded personalities will display.

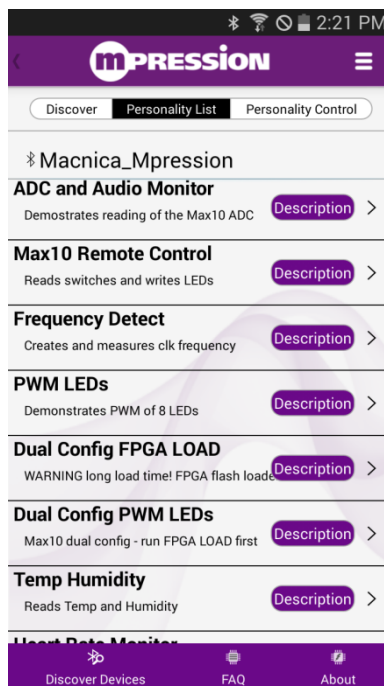


Figure 6-4 Personality List

## 6.4 Select Personality

There are 8 Odyssey kit personalities preloaded into the onboard flash that can be used to demonstrate the capabilities of the kit.

- A. **ADC and Audio Monitor** – This personality allows the user to read the value of the MAX 10 FPGA Analog to digital convert channels. The Odyssey MAX 10 FPGA board also reads the microphone and outputs the sound level on the LEDs.
  - B. **MAX 10 Remote Control** – This personality allows the user remote monitoring of the Odyssey MAX 10 FPGA board switches and allows remote control of the Odyssey MAX 10 FPGA board LEDs.
  - C. **Frequency Detect** – This personality uses the MAX 10 PLL to generate several output frequencies based on the input oscillator. It also has a frequency counter used to measure a generated clock or an external clock. LEDs output a binary representation of the detected frequency in 0.1Mhz increments.
  - D. **PWM LEDs** – This personality demonstrates how FPGA logic can be used to create a large number of PWM controllers, common in lighting and motor control applications. It is related to the MAX 10 Dual Config personality.
  - E. **Dual Config FPGA Load & Dual Config PWM LEDs** - This set of personalities is used to demonstrate the MAX 10 FPGA dual configuration capability. Dual Config FPGA LOAD includes a non-volatile loader which programs the MAX 10 FPGA so the configuration loads instantly at power on. Because the non-volatile loader takes a long time to run, a second personality, Dual Config PWM LEDs, is provided to control the FPGA after power-up, assuming the FPGA image was previously programmed with the non-volatile version of the Dual Config personality.
  - F. **Temp Humidity** – This personality measures temperature and relative humidity using the Silicon Labs Si7020 sensor.
  - G. **Heart Rate Monitor** – This personality measures heart rate using the Silicon Labs Si1147 UV/light/heart rate sensor.
- To select the personality of choice, simply touch the name of the personality and the information will be transferred via BLE and the Odyssey kit sensors and FPGA will be configured as required to enable the personality functions.

Depending on the personality chosen, the FPGA may or may not need to be configured. If the FPGA does require configuration, the single D1 green LED on the BLE sensor board will illuminate while the configuration process takes place.

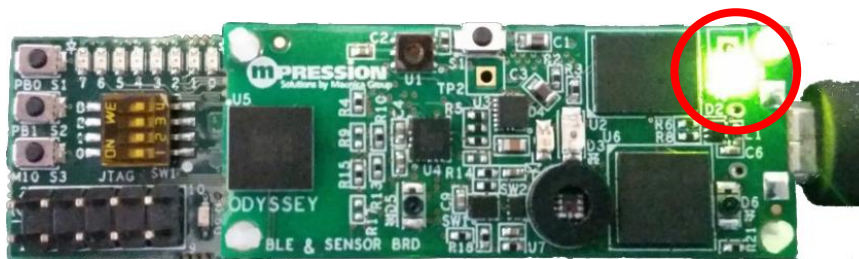



Figure 6-5 Activity LED

Take note that the Smartphone app will notify the user that the personality has successfully loaded, but this is only the phone app portion. Do not try to exercise any of the personality features until the green LED has turned off on the BLE sensor board.

- If a user chooses to change the personality of the kit, hit the back button, , on the Smartphone to return to the Personality List screen and choose another option.

## 6.5 Exercise Personality Features

Once a personality has fully loaded on the Smartphone and on the Odyssey kit, the user can interact with the personality features by simply tapping the button associated with the feature. Some features may require the user to input data while others are simply a command to be issued to the kit.

Each personality has a unique set of features to showcase the capabilities of the Odyssey MAX 10 FPGA and BLE Sensor Kit.



# 7. The Mpression Odyssey IoT Solutions Framework

By exercising the many preprogrammed personalities of the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit, a user can get a good feel for the grand functionality build into the kit. Before being able to create a custom personality a user first needs to have a general understanding of how all the moving parts of the Odyssey system work. This highly integrated infrastructure is referred to as the “Framework.” The basic function of each key component is outlined below.

## 7.1 The Smartphone

As the primary human interface to the Odyssey system, the Smartphone takes on the role of the controller for the entire system.

### 7.1.1 Communications link

The main link between the Smartphone (the client) and the Odyssey device (the server) is the Bluetooth Low Energy wireless radios. This link conforms to the Bluetooth specification v4.1 which defines the low energy requirements of a BLE device. The protocol in use is the Bluetooth SMART protocol stack.

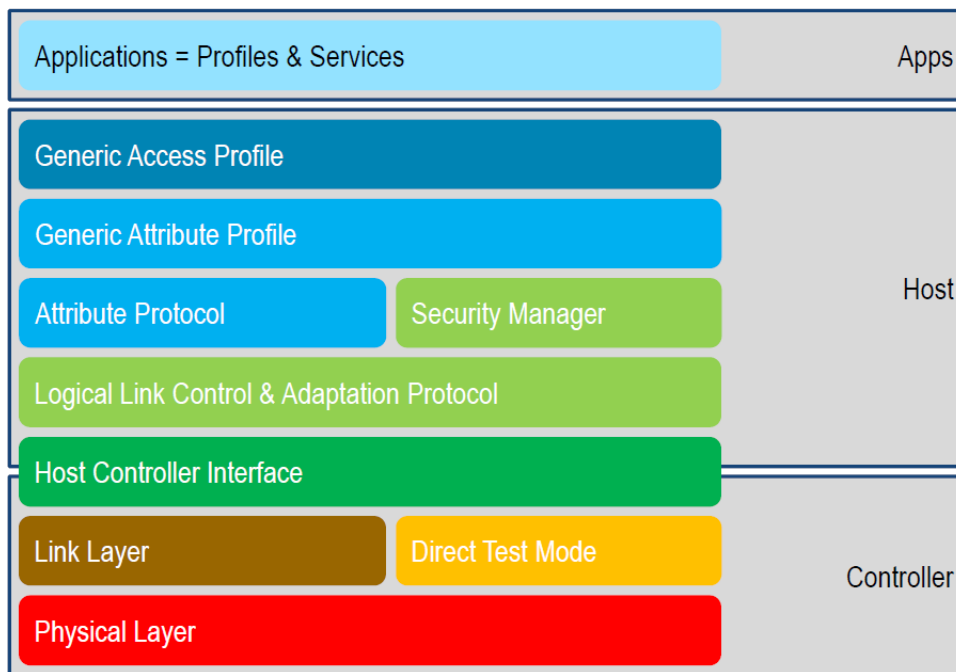


Figure 7-1 Bluetooth SMART Stack

### 7.1.2 The Application

As defined by the Bluetooth specification, a client Application interacts with a server’s Profile by means of the Attribute (ATT) protocol. As seen in Figure 7-1, immediately above the ATT protocol layer is the Generic ATT profile (GATT) layer. It is within this layer that a GATT data base, or the

Profile, is defined and used for transferring commands and data between the Smartphone and the Odyssey device. The actual GATT database is hosted by the BLE device as defined by specification. See section 7.2.2 below for details on the Odyssey Profile.

### 7.1.3 Personality control

A Personality is defined as a unique interaction with the Profile that sets up control of selected sensors, collects data from selected sensors and provides basic processing of the collected data. The personality control is presented graphically to the user via a Smartphone app that is built up of buttons, data entry and data return fields.

As part of the flexibility designed into the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit, the user can define a custom personality; A Smartphone app AND the unique method by which to interact with the Odyssey Profile. This is done by using an Internet cloud based Web Utility called the Smartphone App Configurator. This Web Utility creates all the necessary Smartphone App code as well as defining the Profile interactions without the user having to write a single line of code or needing to know how to manipulate the Bluetooth SMART protocol stack.

Once the Bluetooth LE communications link is established between the Smartphone and the Odyssey kit, the Smartphone app will query the Personalities programmed into the device flash. It will then access the user’s cloud based Personality database to create the Smartphone application GUI used to interact with the selected Personality.

## 7.2 The Bluetooth Low Energy Device

The Broadcom Bluetooth Low Energy device on the BLE sensor board is responsible for many base level interactions with the Smartphone, the EFM32 microcontroller and several sensors.

### 7.2.1 Communications link

As mentioned above, the main link between the Smartphone and the Odyssey device is the Bluetooth Low Energy wireless radios. All Odyssey control and data is transferred via this unencrypted link.

### 7.2.2 Odyssey Profile – The GATT Database

The Profile that is defined for the BLE device of the Odyssey system is comprised of the following 10 services and characteristics.

Table 9 Odyssey Profile

Service Name	Type	Number of Characteristics
GATT	SIG Required	0
GAP	SIG Required	2
Device Info	SIG Defined	2
Battery	SIG Defined	1
Profile Info	Custom	2
Personality	Custom	7
Button	Custom	2

Entry Filed Data	Custom	8
Return Field Data	Custom	8
Auto Update	Custom	2

It is beyond the scope of this document to go into depth on the specific implementation and use of each of the above. However, it is useful to understand how each unique personality utilizes the above services.

In particular, with respect to the Smartphone app and user control and data interaction, the above services take on the following roles.

- A. **Profile Info** - Provides information related to the currently load personality and the presence of the FPGA board.
- B. **Personality** - Provides the characteristics to control the loading of a personality and retrieve fundamental information of that personality. (revision, personality number, FPGA required, etc.)
- C. **Button** – Used to control the processing of a button that is pressed by a user on the Smartphone app. (trigger/scheduling and completion)
- D. **Entry Filed Data** – Used to carry the data entered into the entry field(s) of the Smartphone app to the BLE device (and then onto the FPGA if necessary)
- E. **Return Field Data** – Used to carry data collected by the BLE device (sensors or FPGA) to the Smartphone app.
- F. **Auto Update** – Used to automatically transfer data from the BLE device to the Smartphone without the need for a user request.

### 7.2.3 Command Interpreter

Along with the responsibility of maintaining the communications link (and the GATT database) the BLE device can also execute commands or interpret and forward them. Depending on the information received across the BLE link, the action to be taken may or may not be the responsibility of the BLE device itself. In the cases where the BLE device does not need to handle the command or data, it will forward the command and/or data onto the EFM32 microcontroller or FPGA for further processing. See section 7.5 below for details on the commands processed by the BLE device.

### 7.2.4 Sensor data processing

For sensor applications requiring simple command processing, the BLE will execute the command and immediately respond with the results. The sensors that the BLE handles data processing for are:

- Temperature
- Humidity
- Proximity
- UV
- Light
- Accelerometer
- FPGA

## 7.3 The EFM32 Microcontroller

The Silicon Labs EFM32 microcontroller acts as a secondary command and data processor. It is a slave device connected to the BLE device via the SPI bus. Code for this microcontroller is written in C using Silicon Labs' Simplicity Studio tool set. No operating system is implemented or required.

### 7.3.1 Communications link

The EFM32 microcontroller has a built in USB port that implements a virtual COM port to its host. Application code running on the EFM32 passes traffic between the USB port on the EFM32 and the UART of the EFM32 that is connected to the Broadcom BLE device UART. This allows USB download of code images, via the virtual COM port, to the BLE (from a PC script or WICED Smart SDK) and it also allows debug trace data from the BLE to pass through the EFM32 out the USB port.

An additional application running on the EFM32 implements a Console Menu. Through this menu a user can load personality and FPGA programming images to the flash. See the following sections for details.

### 7.3.2 Personality loading and FPGA Programming

One of the most critical functions the EFM32 handles is the transfer of the personality configuration data from the on-board flash to the appropriate devices. The unique GATT database interaction definition and the FPGA design programming file stored in flash (the personality data file) is retrieved, on command, as a result of a user choosing to change personalities. The flash is initially programmed over the USB connection from a host PC using the Console Menu interface. See section 8.4 below.

Local to the Odyssey kit, the BLE is programmed via the UART interface handled by the EFM32's virtual COM port.

The FPGA is also programmed by the microcontroller running application code called the "JAM player." This JAM player exercises GPIOs of the EFM32 connected to the JTAG port of the FPGA when attached, programming the FPGA with the image stored in on-board flash.

### 7.3.3 Flash Layout

The SPI flash on the BLE sensor board is used to hold up to 15 personalities that can be loaded as described in the previous section. The first ten personality locations (slots) are large enough to accommodate the inclusion of an FPGA design configuration file (<design>.jbc). The flash layout is as follows.

Table 10: Flash Memory Layout

Offset	Size (bytes)	Description
0x000000	192K	Personality 1 data file
0x030000	192K	Personality 2 data file
0x060000	192K	Personality 3 data file
...	...	...
0x1B0000	192K	Personality 10 data file
0x1E0000	8K	Personality 11 data file*
0x1E2000	8K	Personality 12 data file*
...	...	...
0x1E8000	8K	Personality 15 data file*
0x1EA000	88K	Reserved

\*Cannot contain an FPGA design configuration file

Each above personality data file entry contains the following information.

Table 11: Personality File Format

Offset	Name	Size (bytes)	Description
0x00000	Personality number	4	Globally unique number that identifies the personality
0x00004	Personality name	22	Personality name -- Null terminated string
0x0001A	FPGA required	1	Bits [1:0] 00: No FPGA required; Erase FPGA 01: FPGA required; Erase and program FPGA 10: No FPGA required; Don't erase FPGA 11: FPGA required; Don't erase but program FPGA Bits [7:2] Reserved
0x0001B	Revision	1	Personality revision number (optional)
0x0001C	Reserved	228	Reserved. Default 0xFF
0x00020	Personality size	4	Size, in bytes, of the Device Personality Flash File. If no FPGA is present, this will be 4096 bytes, otherwise it will vary depending on the FPGA .jbc file size. This file size does not include the checksum byte (if present).
0x00024	Reserved	224	Reserved. Default 0xFF
0x00100	Personality setup commands	512	Commands to execute when the personality is first loaded. Executed by the BLE module after the FPGA is loaded, if applicable.
0x00300	Background commands	256	Commands that execute continuously in the background.
0x00400	Background INT commands	256	Command to execute in response to an interrupt*
0x00500	Button 1 commands	256	Command to execute in response to a button press.
0x00600	Button 2 commands	256	Command to execute in response to a button press.
0x00700	Button 3 commands	256	Command to execute in response to a button press.
0x00800	Button 4 commands	256	Command to execute in response to a button press.
0x00900	Button 5 commands	256	Command to execute in response to a button press.

0x00A00	Button 6 commands	256	Command to execute in response to a button press.
0x00B00	Button 7 commands	256	Command to execute in response to a button press.
0x00C00	Button 8 commands	256	Command to execute in response to a button press.
0x00D00	Reserved	768	Reserved. Default 0xFF
0x01000	FPGA configuration	188K	Space to hold the <FPGA design>.jbc file used to program the FPGA

\* Not currently implemented

### 7.3.4 Command Interpreter

Any commands that are not processed by the BLE device that are passed on to the EFM32 via the SPI bus are handled by application code running on the EFM32. See section 7.5 below for details on the commands processed by the EFM32 microcontroller.

### 7.3.5 Sensor data processing

For sensor applications requiring several commands or significant processing, the BLE will issue a single command to the EFM32 for processing. The EFM32 will process and respond with the results. This offload processing is done by the EFM32 because it has a large code space and external SRAM attached. The sensors that the EFM32 handles complex data processing for are:

- Heart rate monitor
- Blood oximetry

## 7.4 The FPGA

The FPGA board includes an Altera MAX 10 FPGA. This reconfigurable logic device allows users to create an unlimited number of designs that can interact with the EFM32 microcontroller, the outside world and users directly.

### 7.4.1 Communications link

The MAX 10 FPGA communicates to the BLE sensor board via the I<sup>2</sup>C and JTAG interfaces. Since the FPGA is a device on the I<sup>2</sup>C bus, personalities can be created that control the MAX 10 FPGA opening an entirely new realm of design possibilities.

### 7.4.2 Custom logic

The MAX 10 FPGA is a blank slate of reprogrammable logic provided to the user to configure as needed. The tool set required to create a custom FPGA design is the Altera Quartus-II suite version 14.1 or higher and it is recommended that a user start from an existing Odyssey MAX 10 design and modify it. Once modified, a new FPGA programming file will be generated that a user will convert to the required format as described in section 7.3.2 above. It is beyond the scope of this document to detail how to design with the MAX 10 FPGA. Please refer to the abundant resources available from Altera and Macnica to make the most of the MAX 10 FPGA flexibility. Users may find the Altera Design Store a valuable resource for example designs that can be leveraged for custom designs. Included are the designs preprogrammed into the flash.

Altera Design Store: <https://cloud.altera.com/devstore/>

## 7.4.3 Programming

As covered in the previous sections, the FPGA will be primarily programmed by the EFM32 microcontroller over the JTAG interface. There is a header available on the FPGA board to program the FPGA using a stand-alone JTAG programmer as well.

## 7.4.4 Sensor data processing

With the limitless capabilities of the MAX 10 FPGA and the provided connectivity of the FPGA board, the MAX 10 FPGA can process data for:

- On-board MEMS Microphone
- On-board LEDs
- On-board Dip Switches
- Arduino Nano compatible boards
- Anything a user wants to connect to the expansion pins

## 7.5 Personality Commands

There are many personality commands available to the user that are built into the Framework enabling the features of the Mpression Odyssey MAX 10 FPGA and BLE Sensor Kit. This list of commands is accessible via the Web Utility when editing a personality.

Table 12: Personality Commands

Command Type	Processor	Implemented*
I <sup>2</sup> C Read/Write	BLE	Yes
SPI Read/Write	BLE	No
Get HRM	EFM32	Yes
CMD Stop	EFM32	Yes
Get UV	BLE	No
Get Proximity	BLE	No
Get Blood Oximetry	EFM32	No
LED On/Off	EFM32	No
Detect Motion	BLE	No
Get Diagnostics	EFM32	No
GET EFM Revision	EFM32	No
NOP	BLE	No
NOP Stop	BLE	Yes

\*At the time this document was updated, these are the currently available commands. The other commands will be added with future releases.

Each of the above commands may require a subset of the following data fields.

- I<sup>2</sup>C Device Address
- I<sup>2</sup>C Command
- SPI Command\*
- Register Address Size (in bytes)
  - ◇ 0 to 3
- Register Address (if non-zero Register Address Size)
- Number of Data Bytes

- ◇ 1 to 4
- Write Endianness\*
- Write Source
  - ◇ Smartphone app user data entry field 0-7 or fixed value\*
- Fixed Write Data
- Read Destination
  - ◇ Smartphone app user data return field 0-7 or autonomous field
- Read Endianness\*
- Wait Period (in msec)\*

\*At the time this document was updated, these items are not implemented or have limited functionality.

## 7.5.1 I<sup>2</sup>C

This is perhaps the most flexible personality command available. When using the FPGA board, the FPGA itself can be configured to be an I<sup>2</sup>C slave and receive commands from the Smartphone passed through the BLE device. When using the battery board, the I<sup>2</sup>C commands can be issued to an external device via the expansion pins.

- **Write** - Writes command/data onto I<sup>2</sup>C bus.

Supported data fields

- Device Address
- Register Address Size
- Register Address
- Number of Data Bytes
- Write Source
- Fixed Write Data

- **Read** - Reads data from the I<sup>2</sup>C bus

Supported data fields

- Device Address
- Register Address size
- Register Address
- Number of Data Bytes
- Read Destination



Table 13 I<sup>2</sup>C Devices

Device	Address	Data Sheet
Si7020 Humidity/ Temperature Sensor	0x40	<a href="http://www.silabs.com/Support%20Documents/TechnicalDocs/Si7020.pdf">http://www.silabs.com/Support%20Documents/TechnicalDocs/Si7020.pdf</a>
Si1147 Proximity/ UV/Ambient Light Sensor	0x60	<a href="http://www.silabs.com/Support%20Documents/TechnicalDocs/Si1145-46-47.pdf">http://www.silabs.com/Support%20Documents/TechnicalDocs/Si1145-46-47.pdf</a>
FPGA	Design Dependent*	See FPGA Designs

\* The factory installed personalities that include an FPGA design have the I<sup>2</sup>C address set to 0x30

## 7.5.2 Get HRM

This command initiates heart rate processing that includes a sequence of I<sup>2</sup>C transactions with the Silicon Labs Si1147 device. This processing detects the presence of a finger over the sensor LED to determine pulse rate and reports the rate data back to the Smartphone app via the Broadcom BLE device.

Supported data field

- Read Destination

## 7.5.3 CMD Stop

Commands the EFM32 to terminate interactions with the Si1147 device and perform a graceful shut down. This terminates the collection of heart rate measurements.

Supported data fields

- None required

## 7.5.4 NOP Stop

Perform no operation. Do not execute any further commands associated with personality set-up, buttons pressed, interrupts or background commands.

Supported data fields

- None required

# 8. The Web Utility - Create A Custom Personality

As part of the feature rich development environment of the Odyssey MAX 10 FPGA and BLE Sensor Kit, a user can create a personality similar to those preprogrammed. By using an internet cloud based utility a user can define a custom sensor configuration, FPGA control and usage scenario.

## 8.1 Sign in to Smartphone App Web Utility

Once logged in, a list of private and public personalities will be provided. See prerequisites section 2.2.4 above. Each custom personality created will first be listed in the “My Personality List” which remains visible only to the user account until explicitly requested to publish it to the “Public Personality List.”

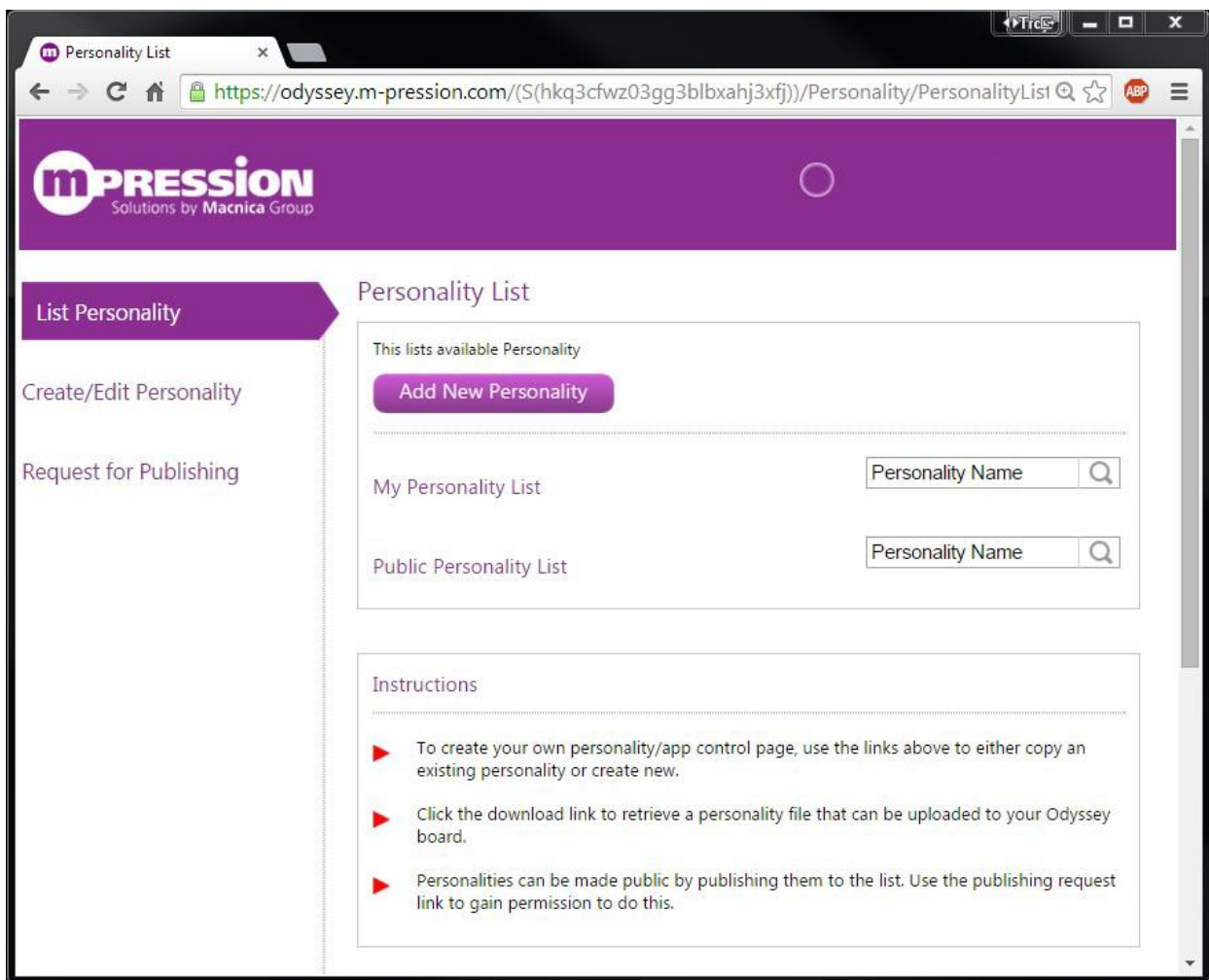


Figure 8-1 Private and Public Personality List

## 8.2 Create New Personality

- Click on the “Add New Personality” button to open the personality editor page. Or, alternatively, a user can select “Copy & Edit” to the right of an existing personality to create a new custom personality by using an existing one as a starting point. This is an excellent way to see how to use many of the features built into the factory loaded personalities.

### Edit Personality

Please do not enter confidential information or files when creating personalities (this form). This information is not secure. Please see the [FAQs](#) for more information

Personality Details(GUID:902)
Revision#:1

Personality Name

Description URL
  Description

Teaser Field

Personality Instructions

Autonomous Field  ON  
 --Type--  --Displa--

Comparison Indicator 1  ON  
 > Greater than

Comparison Indicator 2  ON  
 > Greater than

Comparison Indicator 3  ON  
 > Greater than

Comparison Indicator 4  ON  
 > Greater than

FPGA Required

Personality Setup Commands

Background Commands

Repeat Delay  seconds

Background Interrupt Commands

Sl#	ButtonText	Entry Field	Return Field	Control Preview	Visible	Actions
1					<input type="checkbox"/>	edit
2					<input type="checkbox"/>	edit
3					<input type="checkbox"/>	edit
4					<input type="checkbox"/>	edit
5					<input type="checkbox"/>	edit
6					<input type="checkbox"/>	edit
7					<input type="checkbox"/>	edit
8					<input type="checkbox"/>	edit

Figure 8-2 Personality Editor

Each of the entry fields has a description and usage helper pop-up window accessible by placing the mouse over the icon. Below is the correlation of the editor entry fields to the Smartphone app screen.

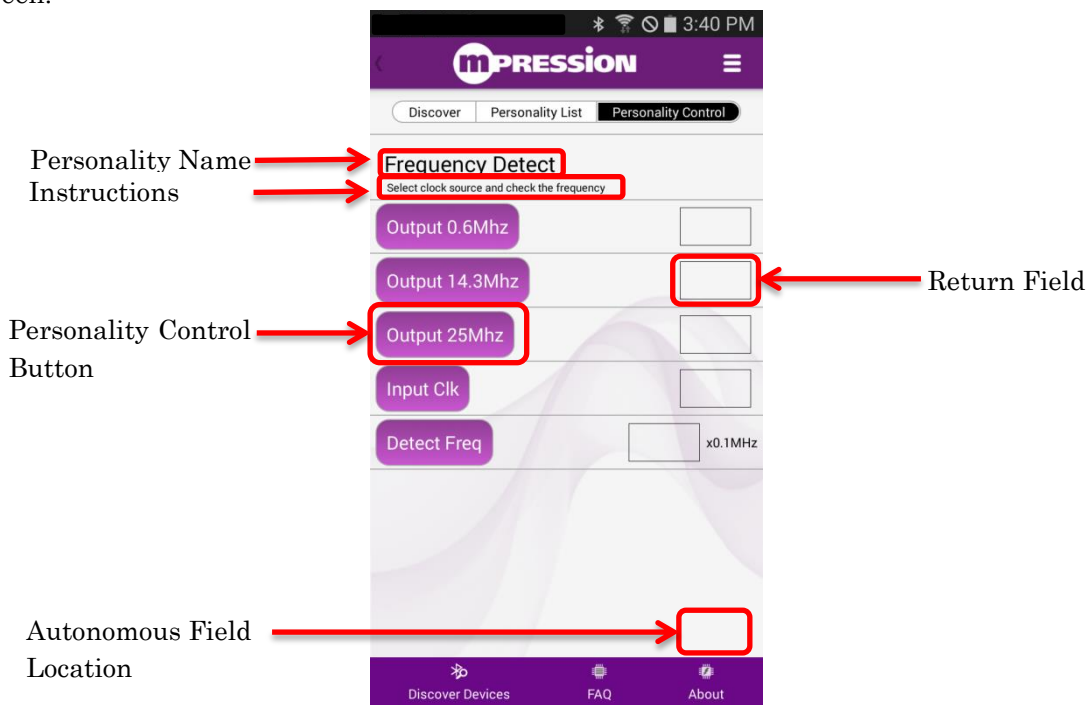


Figure 8-3 Smartphone App Fields

### 8.2.1 Text Fields

The “Personality Name”, “Description”, “Teaser Field” and “Personality Instructions” fields allow a user to enter any character string (or URL for Description) to help identify and instruct the use case of the custom personality.

## 8.2.2 Autonomous Field

Enable this Return field by sliding the ON/OFF switch appropriately if the personality will be returning a data value repeatedly and without user request. The user may enter a short character string to identify the autonomous field and also define what data type to display the return field.

## 8.2.3 Comparison Indicator Fields (Not implemented)

## 8.2.4 FPGA Required Field

In many use cases it will be necessary to program the FPGA with a design to complete functionality requirements. The FPGA design can be programmed into the MAX 10 FPGA by one of two methods.

- By means of a personality load (Recommended)  
When using a personality to load the FPGA a user can easily switch between personalities that have different FPGA designs and the FPGA will be automatically reprogrammed.
- Directly via the JTAG header on the MAX 10 FPGA board  
If this method is used, the Odyssey Framework has no awareness of the FPGA design. It is beyond the scope of this document to detail this method.

The Web Utility Personality Editor or the Console Menu must associate FPGA design images, using the JAM “jbc” (Jam Byte Code) file format, with a personality. The image is programmed into the FPGA via the FPGA JTAG pins using software that is running on the EFM32 microcontroller called a JAM player. A valid JAM (<design>.jbc) file can be created by converting a MAX 10 FPGA SOF file that is generated as part of the Quartus II tool flow as described in section 7.4 above. Note: This works for POF-to-JBC format conversions as well.

- Open the Quartus II Programmer (see section 2.2.7C)
- Click on “Add File” and browse to the directory with the SOF file and select it. Click “Open”
- From the “File” menu, click on “Create JAM, JBC, SVF, or ISC file...”

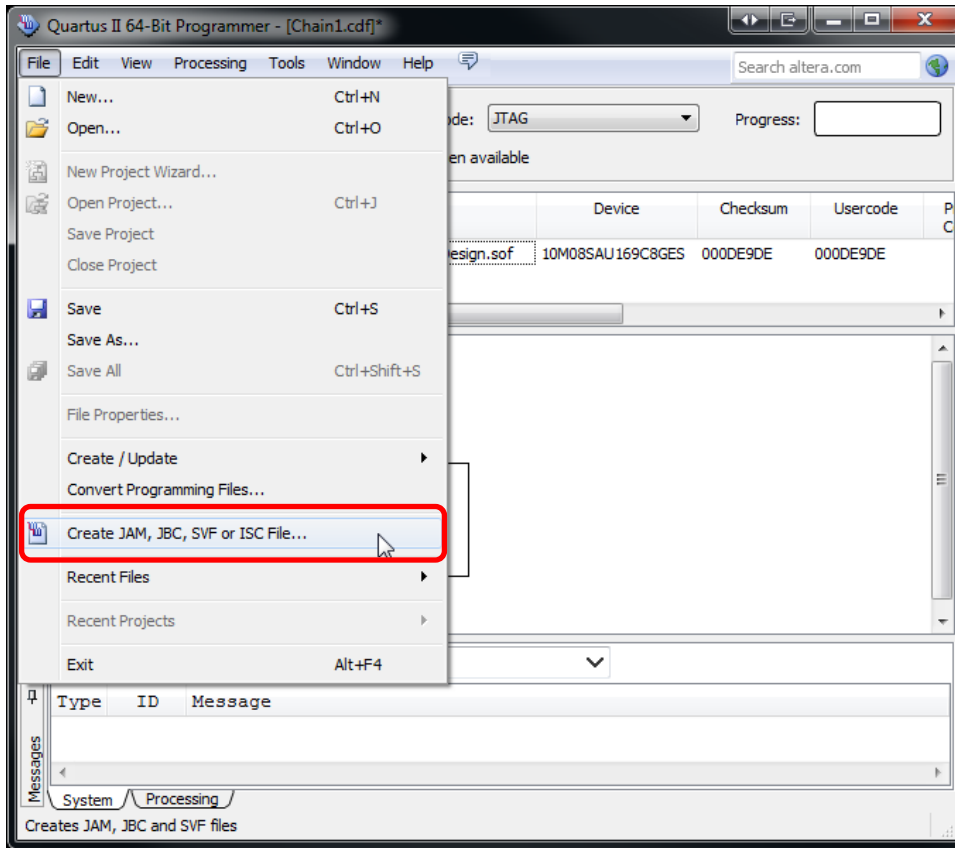


Figure 8-4 Create JAM File Selection

- In the pop-up window, name the JBC file and make sure the file format is set to .jbc. Feel free to use any filename desired. Click “OK”

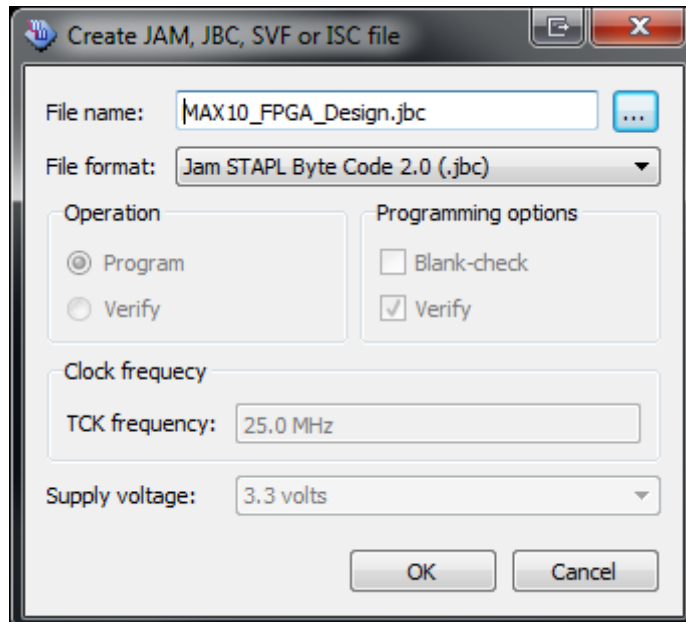


Figure 8-5 Create JBC

- Message should pop-up indicating successful creation of the JAM file.

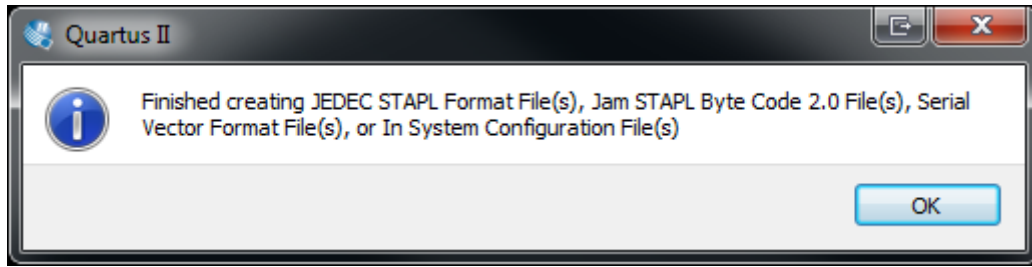


Figure 8-6 Successful JAM File Creation

Once a valid JAM file has been generated, the file needs to be associated with a specific personality.

Note: If the user has selected the “FPGA Required” option in the Personality Editor, the user **MUST** supply an FPGA JAM file. If it is missing, an incorrect file size will result.

During the development cycles, the user can manually update the JAM file in the flash by means of the Console Menu. See section 8.4.1 below.

- In the “FPGA Required” drop-down menu section of the Web Utility (just under the image of the Smartphone), select “FPGA Required, but just run the JAM file” option.

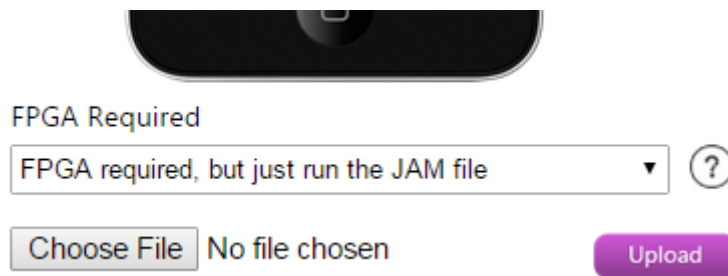


Figure 8-7 FPGA Required Selection

Notice that the FPGA section now has “Choose File” and “Upload” options showing.

- Browse to the FPGA <design>.jbc file.

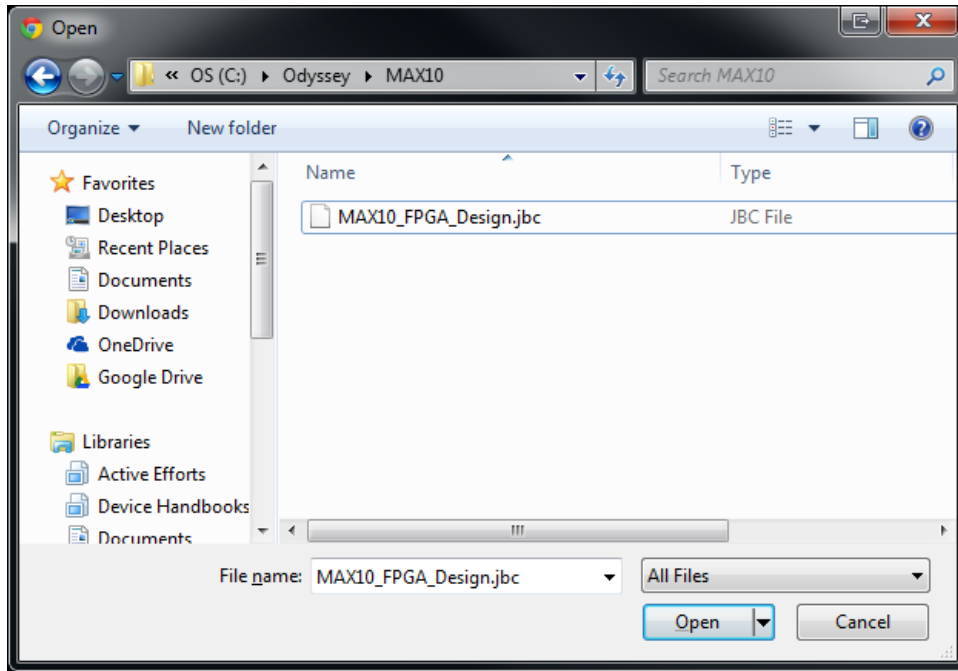


Figure 8-8 FPGA JAM (JBC) File Selection

- Click on the “Upload” button to have the file uploaded, associated and appended to the current personality file.



Figure 8-9 Upload JAM File

## 8.2.5 Personality Setup Commands Field

Run when a personality is first loaded. This will occur after the FPGA has been loaded (if applicable). Up to 16 commands are allowed.

## 8.2.6 Background Commands Field

Run on a 1 second interval once the personality is loaded. Up to 5 commands are allowed.



## 8.2.7 Background Interrupt Commands Field (Not Implemented)

## 8.2.8 Personality Controls Fields

Run on demand via a button press once the personality is loaded. Up to 8 controls are allowed. Each of the 8 controls can be easily enabled or disabled by checking the “Visible” box. The Personality Control editor is made accessible by clicking on the “edit” link to the right of each of the 8 available controls.

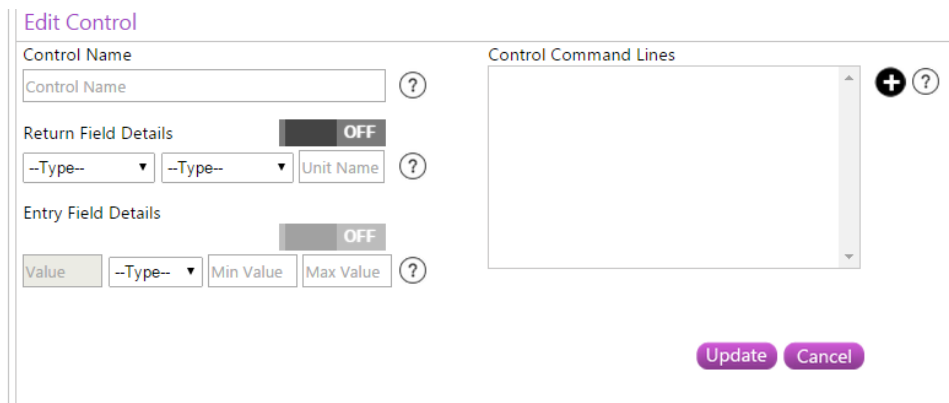


Figure 8-10 Personality Control Edit Fields

The “Control Name” field is a character string that will appear on the Smartphone app button that user will press to initiate the commands as defined in the “Control Command Lines” field. If the command requires either a data entry field or a data return field, those are enabled by sliding the ON/OFF switch respectively. The return data size, format and name will be required as well as the entry data format and limits dependent on the fields enabled.

Each of the above “Command” entry fields are populated by clicking the icon adjacent to the command field of choice. The list of available commands, as discussed in section 7.5 above, is displayed as below.

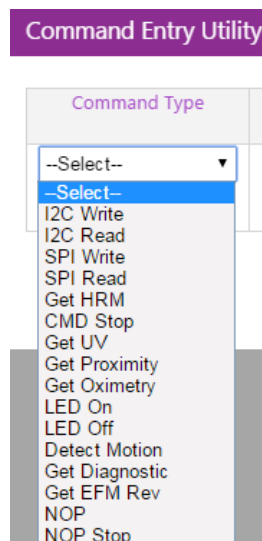


Figure 8-11 Available Commands

Recall that each command may require additional data fields to be populated.

Command Type	I2C Device Address / SPI CMD	Register Address / Register address / I2C CMD byte/ Size	Register address / I2C CMD byte/	No of Data Bytes	Write Endian	Write Source	Fixed Write Data	Read Destination	Read Endian	Wait Period(ms)
-Select-	0x	-Select-	0x	-Select-	-Select-	-Select-	0x	-Select-	-Select-	

Save Cancel

Figure 8-12 Command Specific Data Fields

Here is an example creating a custom heart rate monitor personality with a few of the above fields populated. This personality will automatically return the current heart rate in the autonomous field as well as retrieve an instantaneous heart rate reading when the user pushes control button 1.

### Edit Personality

**Please do not enter confidential information or files when creating personalities (this form). This information is not secure.**

**Please see the [FAQs](#) for more information**

Personality Details(GUID:911)
Revision#:3

Personality Name ?

Description URL  Description

 ?

Teaser Field ?

Personality Instructions ?

Autonomous Field ON ?

uint8
BPM
decimal

Comparison Indicator 1 OFF ?

value
> Greate
And
> Greate
value

Comparison Indicator 2 OFF ?

value
> Greate
And
> Greate
value

Comparison Indicator 3 OFF ?

value
> Greate
And
> Greate
value

Comparison Indicator 4 OFF ?

value
> Greate
And
> Greate
value


FPGA Required ?

Personality Setup Commands + ?

Background Commands + ?

Get HRM X

NOP Stop X



Repeat Delay  
 seconds

Background Interrupt Commands  
+ ?

### Personality Controls

Sl#	ButtonText	Entry Field	Return Field	Control Preview	Visible	Actions
1	Get HR		1	BPM	<input checked="" type="checkbox"/>	<a href="#">edit</a>
2				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
3				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
4				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
5				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
6				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
7				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>
8				<input type="text"/>	<input type="checkbox"/>	<a href="#">edit</a>

#### Edit Control

Control Name  
 ?

Return Field Details ON  
   ?

Entry Field Details OFF  
    ?

Control Command Lines  
 X ?

Update Cancel

Save Cancel

#### Command Entry Utility

Command Type	I2C Device Address / SPI CMD	Register Address Size	Register address / I2C CMD byte	No of Data Bytes	Write Endian	Write Source	Fixed Write Data	Read Destination	Read Endian	Wait Period(ms)
Get HRM	0x	0	0x	--Select--	--Select--	Entry Field	0x 0	Return Field 0	--Select--	0

Update Cancel

Figure 8-13 Complete Custom Personality

- Once a user has completed creating and populating all the fields, buttons and commands desired, save the personality and it will populate to the “My Personality” list.

## Personality List

This lists available Personality

[Add New Personality](#)

---

My Personality List

Personality Name	Modified On	Actions
Read Heart Rate	1/29/2015 7:03:23 PM	<a href="#">Edit</a>   <a href="#">Copy &amp; Edit</a>   <a href="#">Download</a>   <a href="#">Delete</a>

Public Personality List

Figure 8-14 Saved Custom Personality

## 8.3 Download Custom Personality

With the new custom personality complete, the next step is to download the Web Utility generated personality configuration file to the PC.

- Simply click on the “Download” link for the new personality in the “My Personality List” and save the GATT\_xxx.bin file to the PC and note the location where it is saved. The name of the personality file can be changed as desired.

## 8.4 Load Personality to Flash

Now that the custom Odyssey personality configuration is accessible on the PC, the user needs to load it to one of the 15 personality “slots” available on the Odyssey kit. This is achieved via the USB cable connected to the Odyssey and loading the binary configuration file using a serial communication protocol. Be sure to have the USB virtual serial COM port drivers installed before connecting the Odyssey kit. See section 2.2.6 for details.

- Power up the Odyssey kit with the USB cable connected to the PC. Upon successful connection, a virtual serial COM port will be opened between the PC and the device. (Wait for the Windows device connection tone.) Use the Device Manager to determine the COM port number assigned to the Odyssey kit.
- Open a Tera Term session using the above COM port and establish a connection using the following settings:

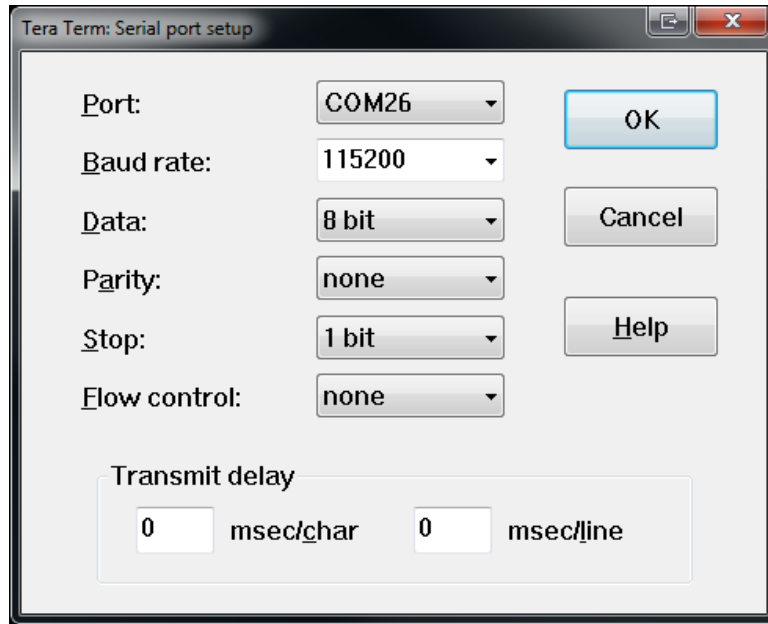


Figure 8-15 COM Port Settings

- To initiate the Console Menu system built into the Odyssey kit for maintenance purposes press the M10 (S3) switch on the MAX 10 FPGA board located nearest the JTAG connector. The D1 green LED will illuminate on the BLE sensor board just as it did in Figure 6-5 from above.

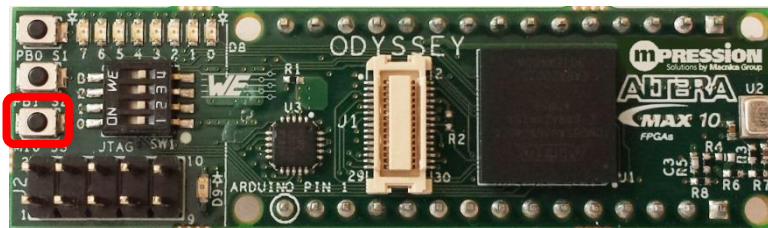


Figure 8-16 MAX 10 Present/Console Menu Push-button

- Press the “Enter” key to display the menu options

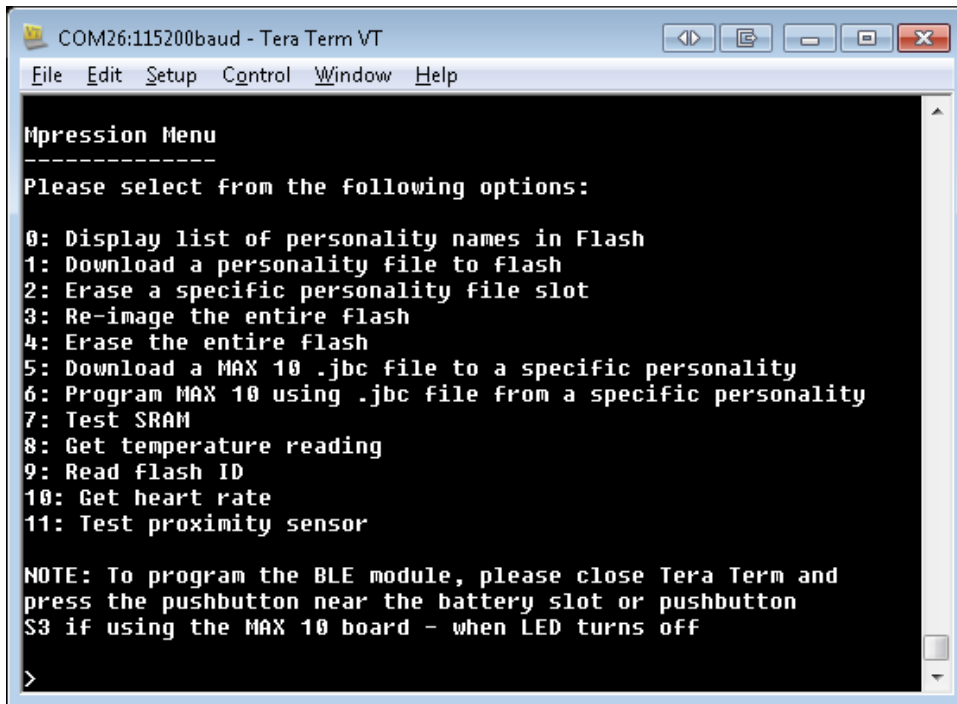


Figure 8-17 Console Menu

- If the console output is skewed, change the **New-line** characteristics of the Tera Term session under the **Setup** -> **Terminal** pull-down menu. Select either **LF** or **Auto**.

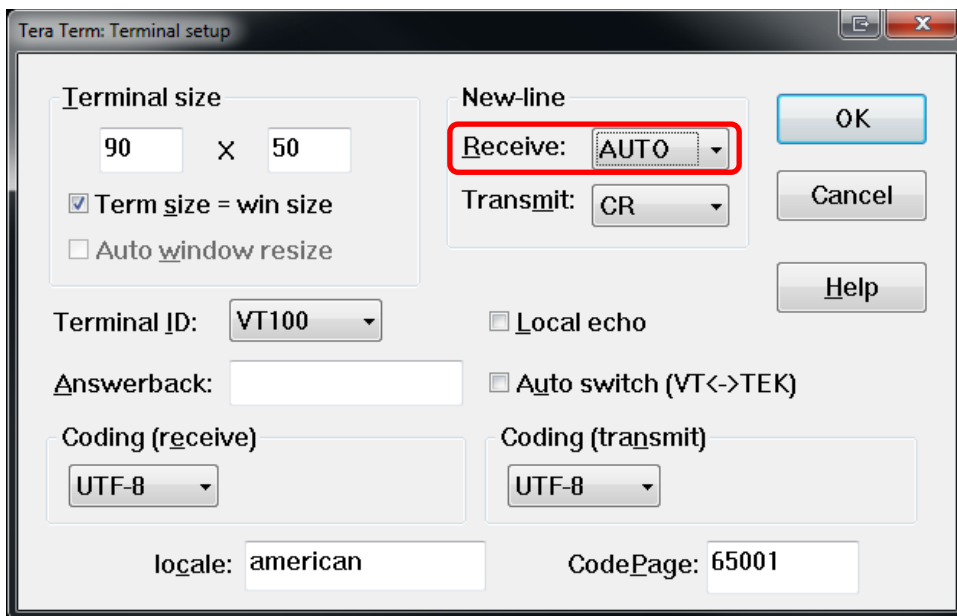
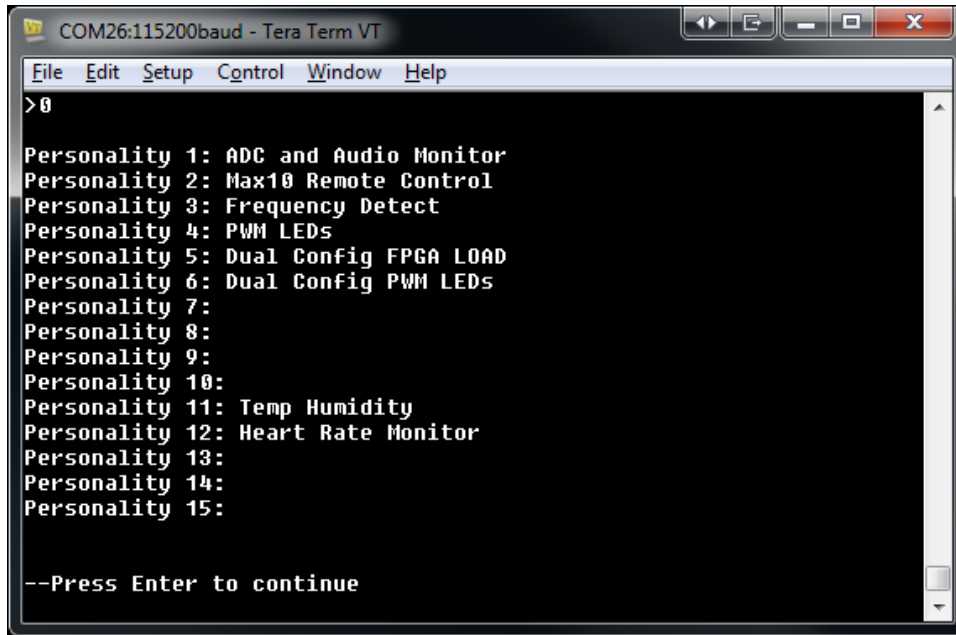


Figure 8-18 Tera Term New-line Setting

- Select option “0” to display the current list of personalities loaded into the flash and then press “Enter” to return to the main menu.



```

COM26:115200baud - Tera Term VT
File Edit Setup Control Window Help
>0
Personality 1: ADC and Audio Monitor
Personality 2: Max10 Remote Control
Personality 3: Frequency Detect
Personality 4: PWM LEDs
Personality 5: Dual Config FPGA LOAD
Personality 6: Dual Config PWM LEDs
Personality 7:
Personality 8:
Personality 9:
Personality 10:
Personality 11: Temp Humidity
Personality 12: Heart Rate Monitor
Personality 13:
Personality 14:
Personality 15:

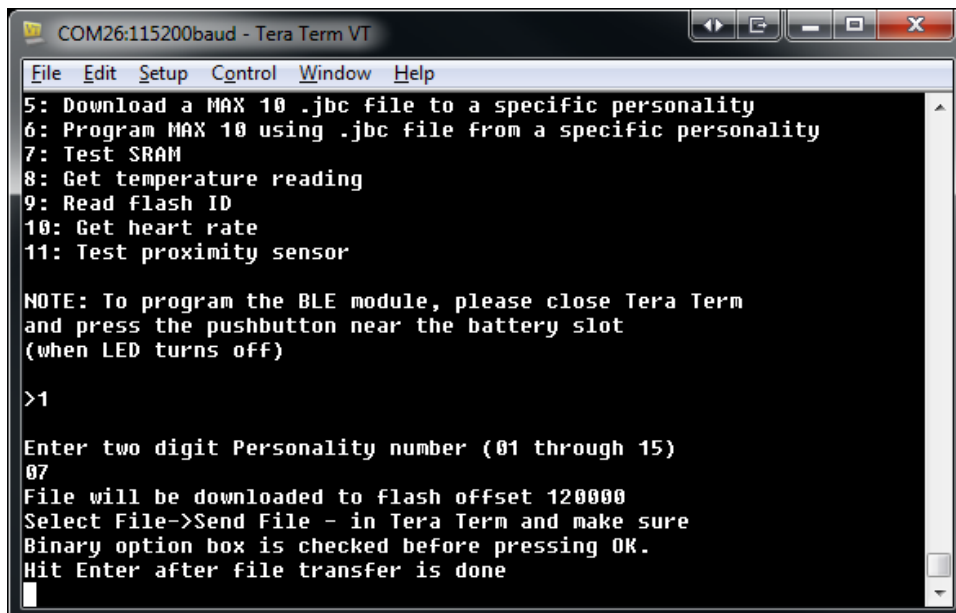
--Press Enter to continue

```

Figure 8-19 Flash Resident Personalities

- Now select option “1” and select an available flash location, “07” in this case, to upload the custom personality configuration to the device at slot 7 in the flash.

Note: The BLE sensor board contains enough flash memory to hold 10 personalities that include an FPGA design programming file and 5 additional personalities without FPGAs. Slots 1-10 can be used for personalities with or without FPGAs and slots 11-15 are only for personalities **without** FPGA designs included.



```

COM26:115200baud - Tera Term VT
File Edit Setup Control Window Help
5: Download a MAX 10 .jbc file to a specific personality
6: Program MAX 10 using .jbc file from a specific personality
7: Test SRAM
8: Get temperature reading
9: Read flash ID
10: Get heart rate
11: Test proximity sensor

NOTE: To program the BLE module, please close Tera Term
and press the pushbutton near the battery slot
(when LED turns off)

>1

Enter two digit Personality number (01 through 15)
07
File will be downloaded to flash offset 120000
Select File->Send File - in Tera Term and make sure
Binary option box is checked before pressing OK.
Hit Enter after file transfer is done

```

Figure 8-20 Upload to Flash Personality #7

- Use the Tera Term “File -> Send File” pull-down menu to select the GATT\_XXX.bin file saved to the PC from section 0 above. **Be sure to select the “Binary” option to ensure a successful transfer.** A window will briefly open up showing the transfer. Depending on the size of the

configuration file (usually very small unless there is an FPGA configuration file included) there may not be time to enough to read the details of the transfer.

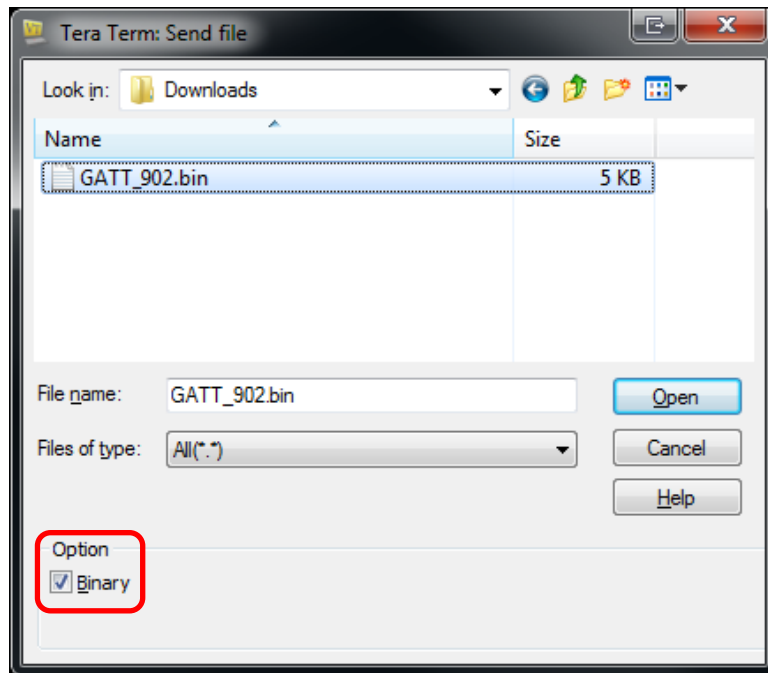


Figure 8-21 Send File to Device

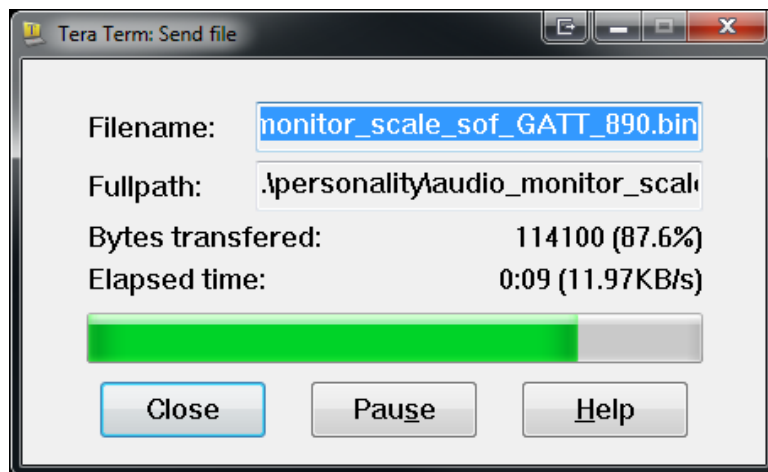
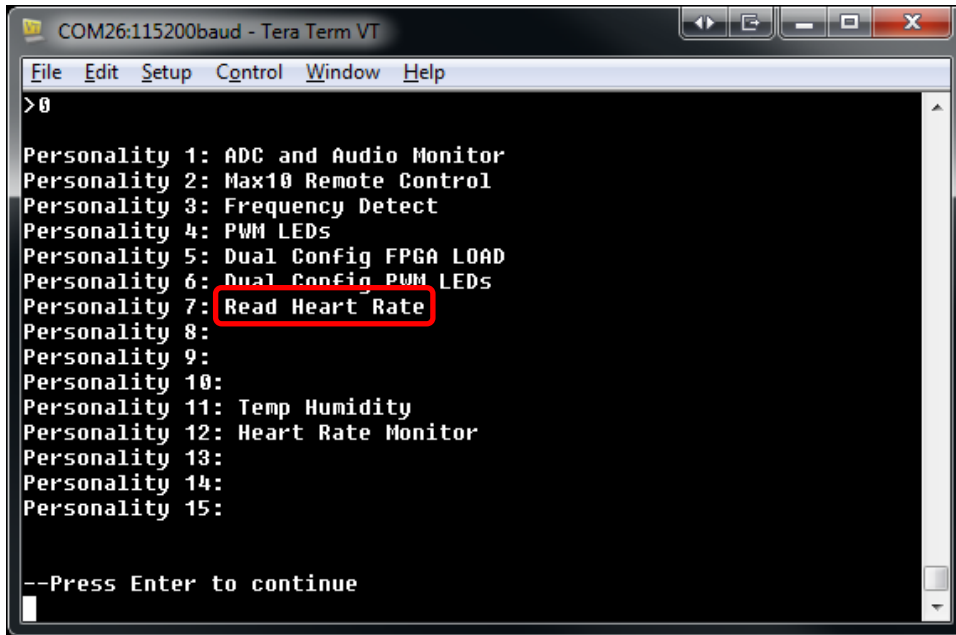


Figure 8-22 File Being Sent To Device

- When the transfer is complete, return to the serial console and press the “Enter” key. Once again select menu option “0” to view the flash resident personalities. The new custom personality should be visible.





```
COM26:115200baud - Tera Term VT
File Edit Setup Control Window Help
>0
Personality 1: ADC and Audio Monitor
Personality 2: Max10 Remote Control
Personality 3: Frequency Detect
Personality 4: PWM LEDs
Personality 5: Dual Config FPGA LOAD
Personality 6: Dual Config PWM LEDs
Personality 7: Read Heart Rate
Personality 8:
Personality 9:
Personality 10:
Personality 11: Temp Humidity
Personality 12: Heart Rate Monitor
Personality 13:
Personality 14:
Personality 15:
--Press Enter to continue
```

Figure 8-23 Custom Personality Loaded into Flash

## 8.4.1 Update FPGA design with Console Menu

If the user chooses to update an FPGA design JAM file via the Console Menu, the user must first load the associated personality, inclusive of a JAM file, into an open flash location. See section 8.4 above. Keep in mind, the FPGA programming JAM file will only be programmed into the MAX 10 FPGA if the associated personality has the “FPGA Required” option set. Note: Personalities with FPGA images are only allowed in slots 1 thru 10. Slots 11-15 are too small in terms of allocated flash memory space.

- Access the Console Menu by means of the steps outlined in section 8.4 below.
- Use Console Menu option “0” to list the personality slots to see which are populated. Be sure to have a valid personality loaded into a slot (1-10) before attempting to load the FPGA JAM file
- Load JAM file to the chosen personality slot with option menu 5

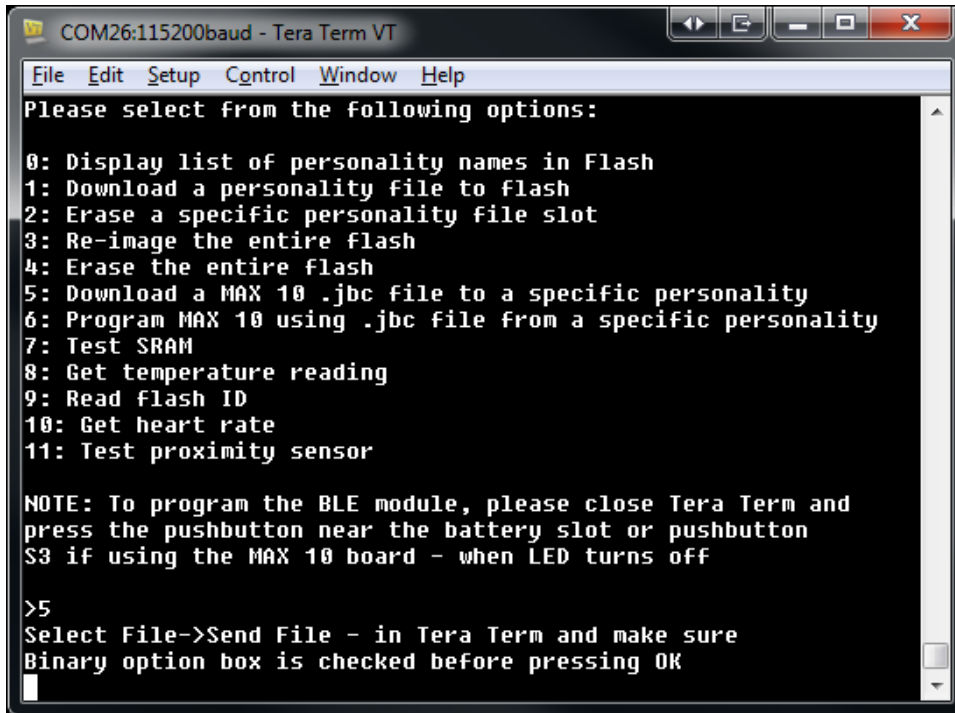


Figure 8-24 Load JAM File

- Use the Tera Term “File -> Send File” pull-down menu to select a <design>.jbc file. **Be sure to select the “Binary” option to ensure a successful transfer.** A window will briefly open up showing the transfer.

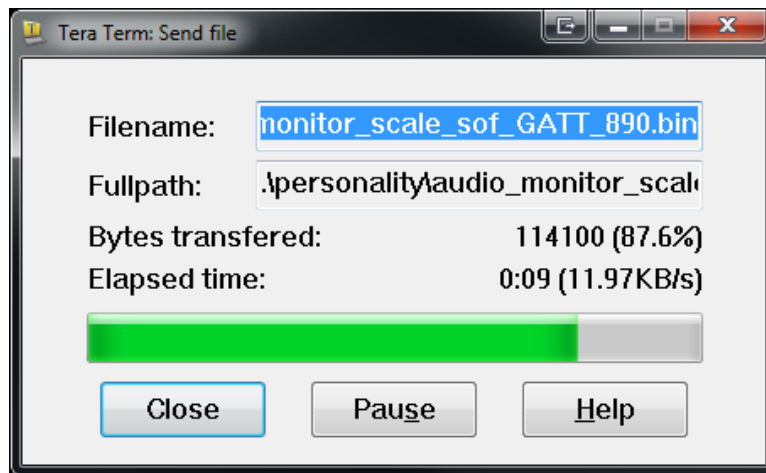
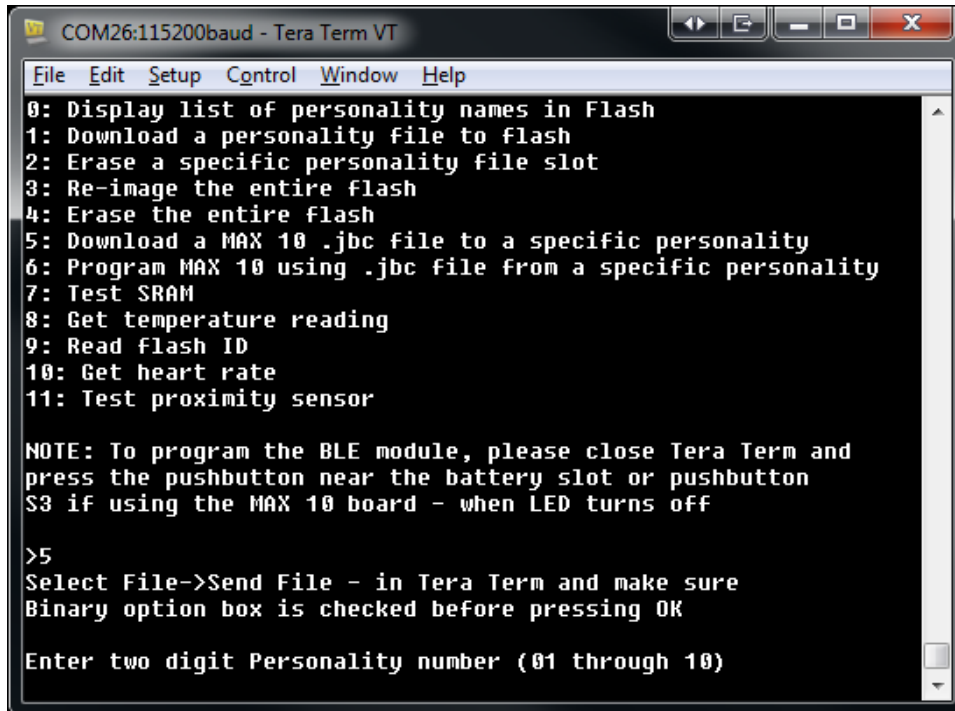


Figure 8-25 File Being Sent To Device

- After the JAM file has been sent to the Odyssey, select the personality to associate with the new FPGA programming file.



```
COM26:115200baud - Tera Term VT
File Edit Setup Control Window Help
0: Display list of personality names in Flash
1: Download a personality file to flash
2: Erase a specific personality file slot
3: Re-image the entire flash
4: Erase the entire flash
5: Download a MAX 10 .jbc file to a specific personality
6: Program MAX 10 using .jbc file from a specific personality
7: Test SRAM
8: Get temperature reading
9: Read flash ID
10: Get heart rate
11: Test proximity sensor

NOTE: To program the BLE module, please close Tera Term and
press the pushbutton near the battery slot or pushbutton
S3 if using the MAX 10 board - when LED turns off

>5
Select File->Send File - in Tera Term and make sure
Binary option box is checked before pressing OK

Enter two digit Personality number (01 through 10)
```

Figure 8-26 Select Personality to Associate JAM File

The new FPGA design programming file is now loaded into the flash on the BLE sensor board and associated with the personality. Now when the user selects this personality from the list of personalities on the Smartphone app, the FPGA will be programmed with the desired design.

## 8.5 Run Custom Personality

Now that the custom personality has been created and loaded onto the Odyssey MAX 10 FPGA and BLE Sensor Kit, the user can interact with it.

- Cycle power to the kit by unplugging and plugging back in the USB cable. (The activity LED on the BLE sensor board will turn off)
- On the Smartphone App, select the “Discover Devices” button on the lower-left corner of the screen to return to the “Discover” screen.
- Once again, select the “Macnica\_Mpression” device from the list to open the “Personalities List” screen. Notice the custom personality is now one of the available options in the list.

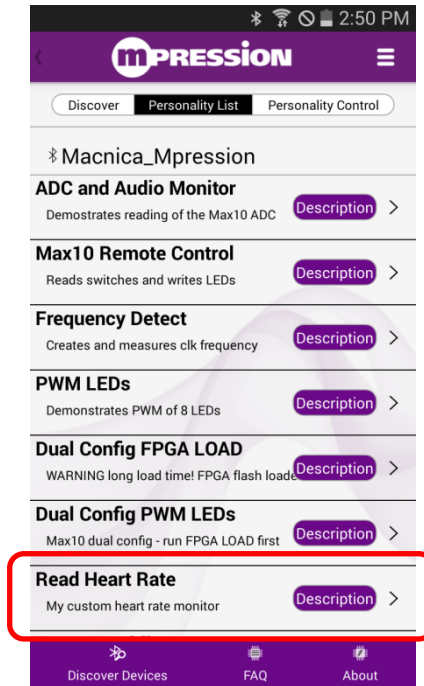


Figure 8-27 Custom Personality Available

- Select the new personality to download and open the “Personality Control” screen.

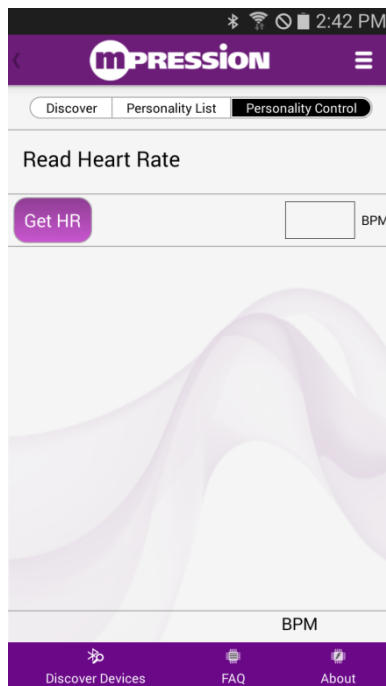


Figure 8-28 Custom Personality Control Screen

- If the user has followed this example and created a heart rate monitor, place a finger gently on top of the O-ring on the BLE sensor board enough to cover both the sensor LEDs of the heart rate detection circuit.

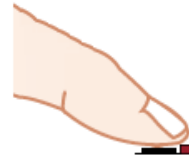


Correct finger position

Incorrect finger position



Sensor LEDs



Sensor LEDs

Figure 8-29 Heart Rate Sensor Finger Position

Notice the Autonomous field will automatically update as we designed in the Personality Editor. Also, notice that when the user presses the **Get HR** button, the return data field is updated with the current heart rate as well.

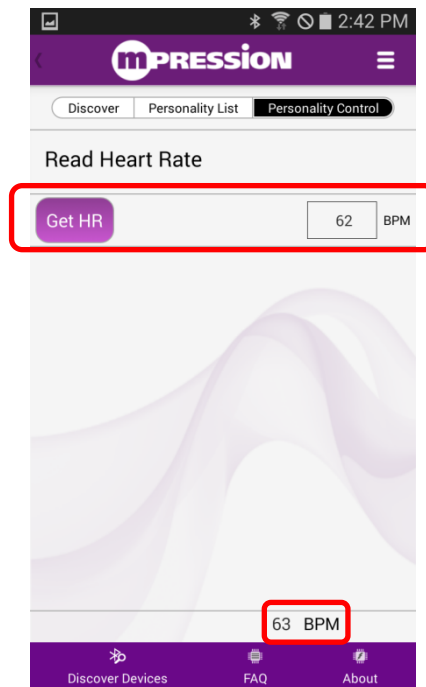


Figure 8-30 Personality Data Fields Active

## 9. Tips and Tricks

---

### 9.1 Consult FAQs

<http://www.m-pression.com/odyssey-faq>

### 9.2 Cut standoffs

The standoffs holding the boards together can be modified to be easier to work with. If a user takes sharp cutters and cuts near the very tips of the standoffs, a user can get the boards on and off more quickly and without damaging force.

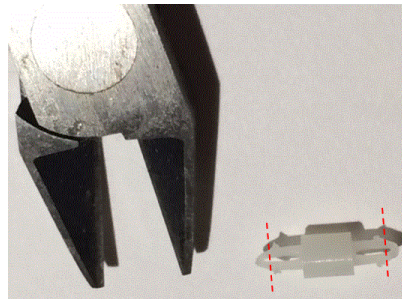


Figure 9-1 Clip Locking Standoffs

### 9.3 Stand-alone MAX 10 FPGA board

Bread-boarding the MAX 10 FPGA kit using solderless breadboards has several advantages. These inexpensive bread boards give the MAX 10 board kit stability and flexibility for probing and testing, as well as options for powering the board independently.

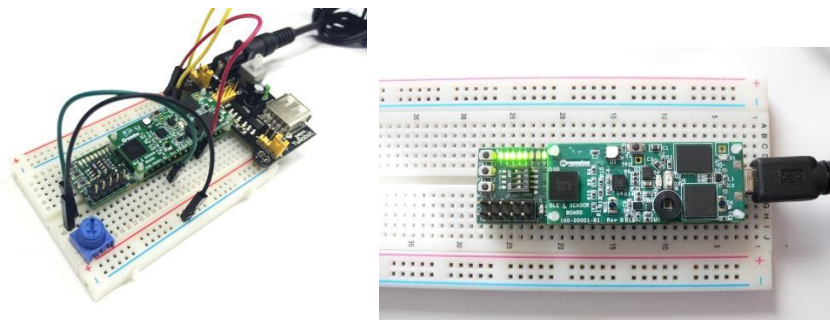


Figure 9-2 Stand-alone MAX 10 FPGA Board

### 9.4 Getting leverage for the FPGA I<sup>2</sup>C interface

Leverage the Odyssey MAX 10 designs to get the I<sup>2</sup>C interface going quickly. The example designs are based on the Opencores I<sup>2</sup>C core ([www.opencores.org](http://www.opencores.org)) that is widely used. A user can have alternate designs or revisions that pinout the I<sup>2</sup>C signals to the FPGA board DIP interface (under side) pins so there is access at the breadboard.

FPGA project and source code files for the Odyssey example personalities are posted on the software downloads page: <http://www.m-pression.com/solutions/boards/odyssey/odyssey-downloads>

Schematics and bill of materials for the kit may be found in the documentation section of the quick start page: [http://www.m-pression.com/solutions/boards/odyssey-fpga#quick\\_start](http://www.m-pression.com/solutions/boards/odyssey-fpga#quick_start)

For assessing I<sup>2</sup>C interaction, users can take advantage of a couple of inexpensive debug tools:

The Bus Pirate ([http://dangerousprototypes.com/docs/Bus\\_Pirate\\_v3.5](http://dangerousprototypes.com/docs/Bus_Pirate_v3.5)) is an inexpensive board with I<sup>2</sup>C software that can generate custom commands just by typing them in or using scripts. There are also low cost analyzers by Saelae for I<sup>2</sup>C and SPI (<http://www.saleae.com>). If the battery board is to be used, there are I<sup>2</sup>C test points available as well.

## 9.5 Startup delay

If a user connects to the Odyssey board over Bluetooth shortly after applying power, it may be getting interrogated before it has finished its boot cycle. It is best to wait 5-7 seconds after power-on to connect to the device. It is often a good idea to exit the Smartphone app, kill it from background memory, and restart it.

## 10. Document Revision History

---

Date	Revision	Changes
February 11, 2015	1.0	<ul style="list-style-type: none"><li>• Initial Document</li></ul>
February 23, 2015	1.1	<ul style="list-style-type: none"><li>• Update all sections, added Framework details, Tips &amp; Tricks.</li></ul>
March 6, 2015	1.2	<ul style="list-style-type: none"><li>• Updated Tips &amp; Tricks</li></ul>
March 9, 2015	2.0	<ul style="list-style-type: none"><li>• Removed all references to 2 separate kits.</li></ul>
March 26, 2015	2.1	<ul style="list-style-type: none"><li>• Added regulatory statements</li></ul>