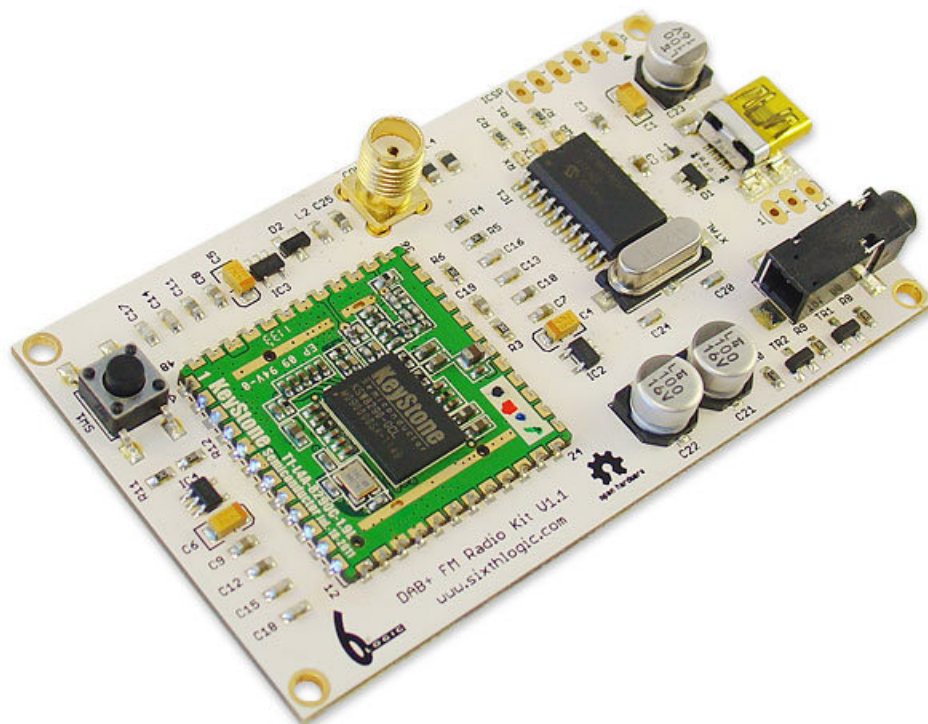


DAB+ FM Development Board User's Guide





DEVELOPMENT BOARD/KIT IMPORTANT NOTICE

Sixth Logic (SL) provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by SL to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and therefore may not meet the technical requirements of these directives or other related directives.

Should this evaluation board/kit not meet the specifications indicated in the User's Guide, the board/kit may be returned within 30 days from the date of delivery for a full refund.

SL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. SL disclaims all liability arising from this information and its use. The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies SL from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

SL currently deals with a variety of customers for products, and therefore our arrangement with the user is not exclusive. SL assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.

Please read the User's Guide and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages. For additional information on SL's environmental and/or safety programs, please contact the SL application engineer or visit www.sixthlogic.com. No license is granted under any patent right or other intellectual property right of SL covering or relating to any machine, process, or combination in which such SL products or services might be or are used.

FCC Warning

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by SL to be a finished end-product fit for general consumer use. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his/her own expense will be required to take whatever measures may be required to correct this interference.

Table of Contents

Chapter 1. Introducing the DAB+ FM Development Board	5
1.1 INTRODUCTION	5
1.2 DESCRIPTION OF THE DAB+ FM DEVELOPMENT BOARD / PRO.....	5
1.3 DEVELOPMENT BOARD OVERVIEW	6
Chapter 2. Quick Start Guide.....	7
2.1 INTRODUCTION	7
2.2 HIGHLIGHTS.....	7
2.3 SETTING UP THE HARDWARE.....	7
2.4 INSTALLING MICROCHIP’S CDC DRIVER	8
2.5 RUNNING THE SAMPLE PROGRAM	12
2.6 INITIAL AUTO SCANNING	15
2.7 BBE SOUND.....	15
2.7.1 BBE HD Sound Setting.....	16
2.7.2 PRESET EQ.....	16
2.7.3 ORIGINAL SOUND.....	16
2.8 SLIDESHOW (<i>Only on PRO Board that has T2_L4A_8650C module</i>).....	17
2.9 COMPILING THE SAMPLE PROGRAM	18
Chapter 3. Hardware Features and Descriptions.....	19
3.1 BOARD FEATURES.....	19
3.2 HARDWARE DESCRIPTIONS.....	19
Power Supply Circuit.....	19
Microchip PIC18F14K50	19
Reset Circuit	19
ICSP	19
EXT.....	19
Push Button (SW1).....	19
APPENDIX A – SCHEMATIC	20
APPENDIX B – MECHANICAL DRAWING	21



DOCUMENT REVISION HISTORY

Revision A (April 2012)

- Initial Release of this Document.

Revision B (December 2012)

- Updated BBE Sound
- Updated SlideShow for Pro Board
- Updated DLL functions list
- Added description for T2_L4A_8650C module

Chapter 1. Introducing the DAB+ FM Development Board



1.1 INTRODUCTION

The following sections provide an overview of the DAB+ FM Development Board and demonstrate how to use it.

The following topics are covered:

- Description of the DAB+ FM Development Board
- Using the DAB+ FM Development Board

1.2 DESCRIPTION OF THE DAB+ FM DEVELOPMENT BOARD / PRO

The DAB+ FM Development Board provides a platform for developing and evaluating DAB+ and FM receiver. The board contains a Keystone T1_L4A_8290C DAB/DAB+/FM module on a basic board and T2_L4A_8650C DAB/DAB+/FM/EPG/SlideShow module and a Microchip PIC18F14K40 microcontroller.

The T1_L4A_8290C module is an ultra low power DAB/DAB+/FM receiver module with the following features:

- ETSI EN 300 401 compliant receiver
- DAB/DAB+ sensitivity to -99dBm (typical)
- Decodes multiple audio services up to 256kbps without external RAM
- FM with RDS (RDS reception is subject to certain condition of power level and frequency deviation)
- Combined antenna input for FM / Band3
- Support DAB L-band reception
- BBE Sound
- Serial control interface
- RoHS compliant

The T2_L4A_8650C module is an ultra low power DAB/DAB+/FM/EPG/SlideShow receiver module similar to T1_L4A_8290C with extra EPG and SlideShow functions.

The Microchip PIC18F14K50 provides a USB to Serial interface for the radio module to communicate with the host PC. Despite the sample code provided in this Development Board for communicating with the board is based on serial communication, user can analyse the serial communication and implements a driverless HID protocol by rewriting the firmware on the PIC18F14K50.

1.3 DEVELOPMENT BOARD OVERVIEW

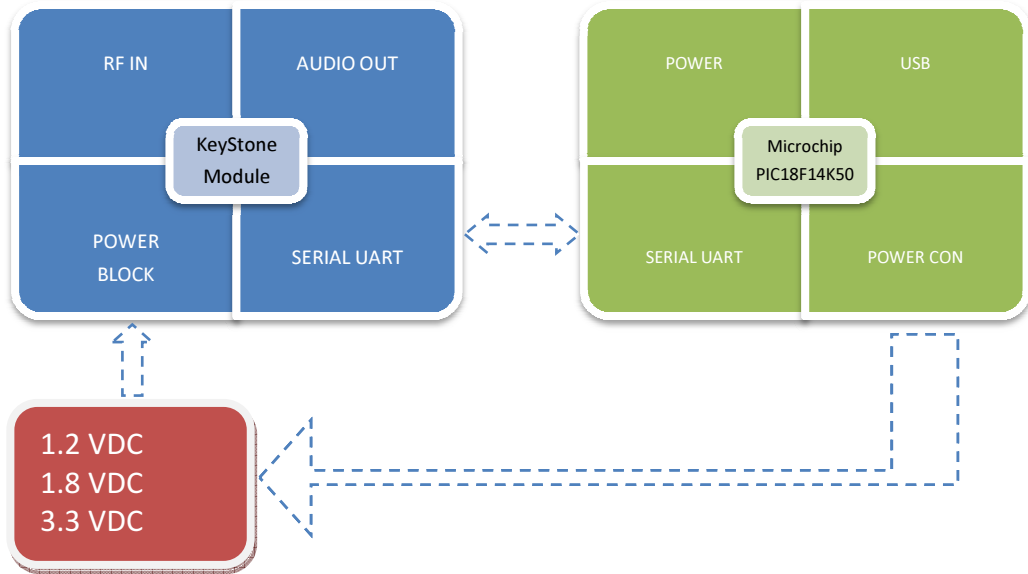


FIGURE 1.1: Development Board Block Diagram.

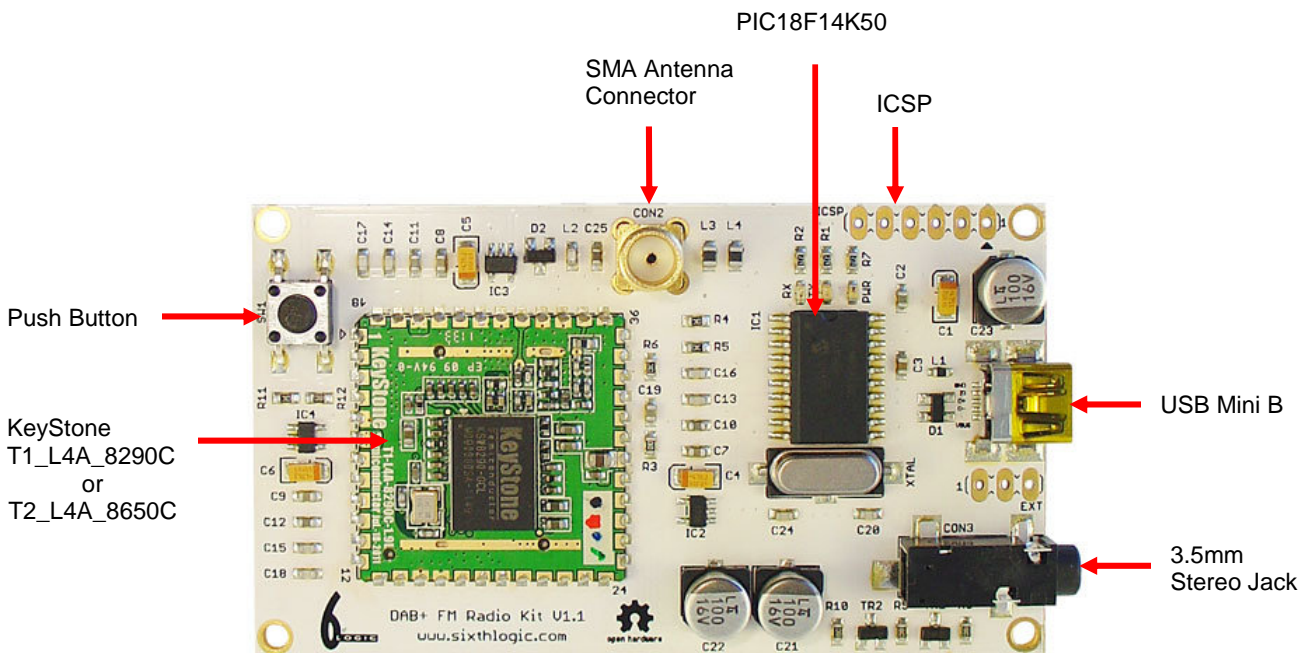


Figure1.2:Development Board Top View.

Chapter 2. Quick Start Guide

2.1 INTRODUCTION

This chapter explained the basic steps to get you started using the DAB+ FM Development Board.

2.2 HIGHLIGHTS

Items discussed in this chapter include:

- Setting up the hardware
- Installing the Microchip's CDC driver
- Running the sample program

2.3 SETTING UP THE HARDWARE

- 1) Download the DAB+ FM Development Board's software package from the product website and then unzip the software package to a known folder.
- 2) Connect the supplied antenna to SMA connector on the board. The antenna has a magnetic base, sticking the antenna to a large metal surface, for example window frame, will produce a better reception. Users need to experiment to find a suitable location to place the antenna.
- 3) Connect a USB cable (Mini B) to the board, and the other side of the cable to the Host PC.
- 4) Ensure the three LEDs on the board light up.
- 5) Connect speakers or headphones to the 3.5mm stereo jack, possible to perform this step after the board is ready.

Handling the Board

When handling the board, it is important to observe the following precaution:



Without proper anti-static handling, the board can be damaged. Therefore use anti-static handling precautions when touching the board.



In order to avoid damaging the board, only operate the board from a USB port's 5VDC supply.

2.4 INSTALLING MICROCHIP'S CDC DRIVER

After the board is being plugged to the host PC, Windows should recognise the board as “CDC RS-232 Emulation Demo”. In the “Welcome to the Found New Hardware Wizard” window select “No, not this time” and then click “Next >” (Figure 2.1)

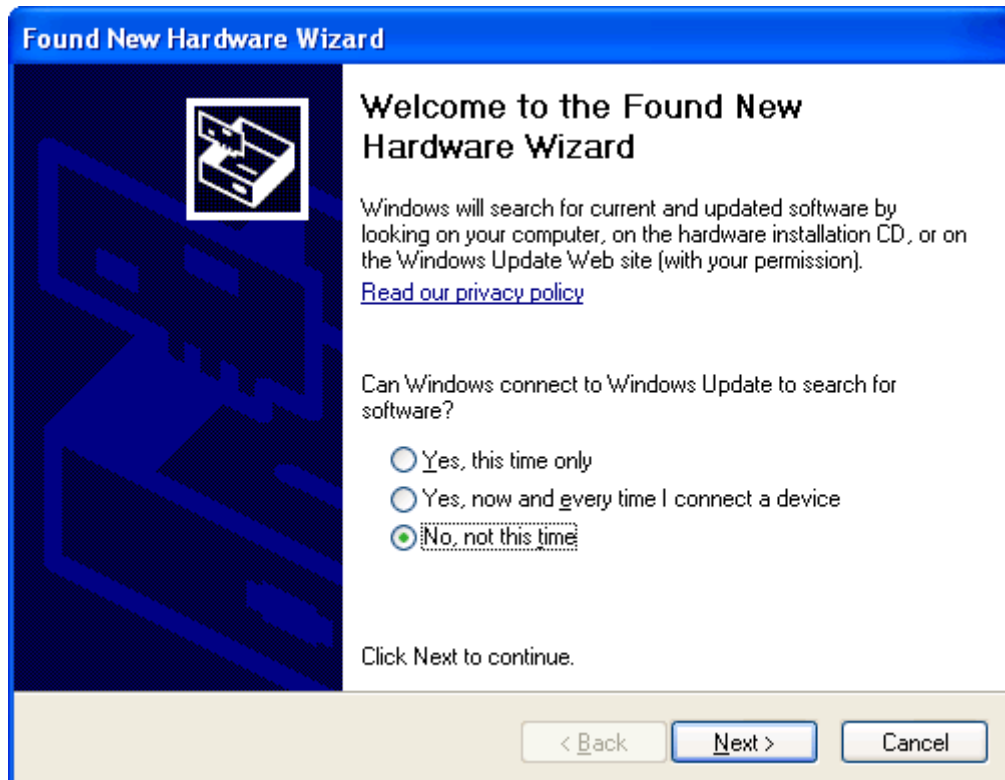


Figure 2.1: Found New Hardware Wizard Window.

The wizard will then prompt the user for a location from which to load the software for the communication port. Select “*Install from a list or specific location (Advanced)*” and click “*Next >*” (Figure 2.2)

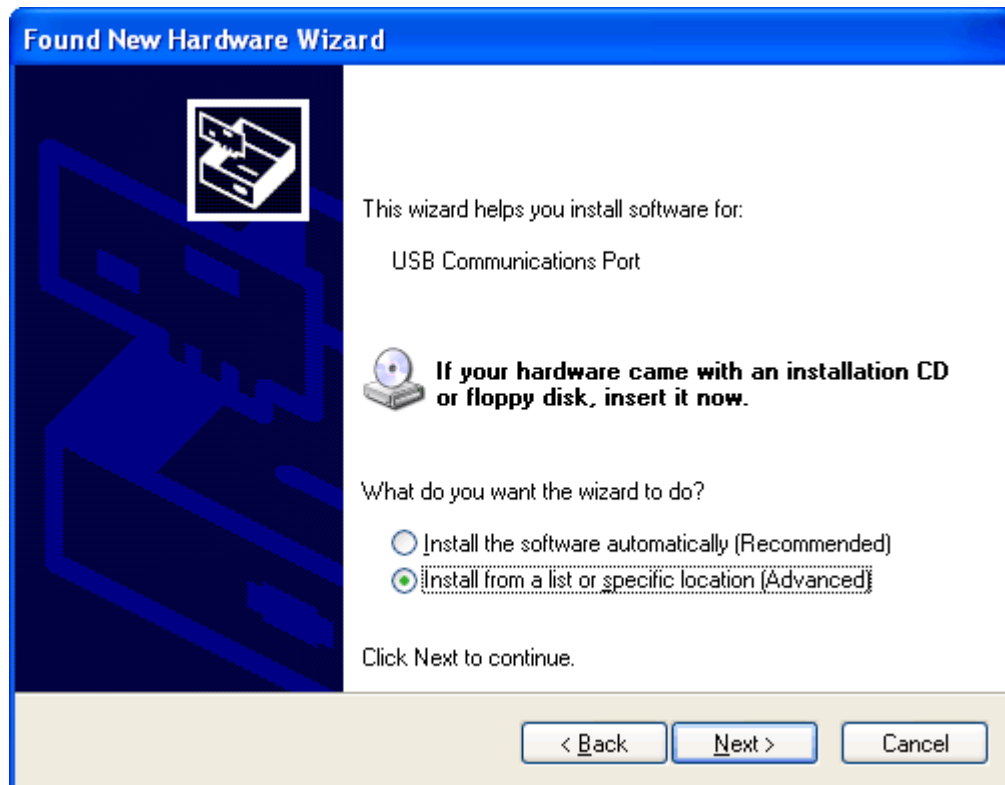


Figure 2.2: Select Software Location for Communication Port.

The wizard now prompts the user for the location of the *.inf* file that Windows will use to automatically configure the binary driver files (*.sys* files) to create the Virtual COM port connection for the USB. Ensure that both “*Search for the best driver in these locations*” and “*Include this location in the search*” are both selected. Select “*Browse*” and navigate to the product software folder unzipped in step 2.3.1. If user unzip the product software into root of C driver, the driver files should be in the following folder:

```
C:\DABSoftware\inf
```

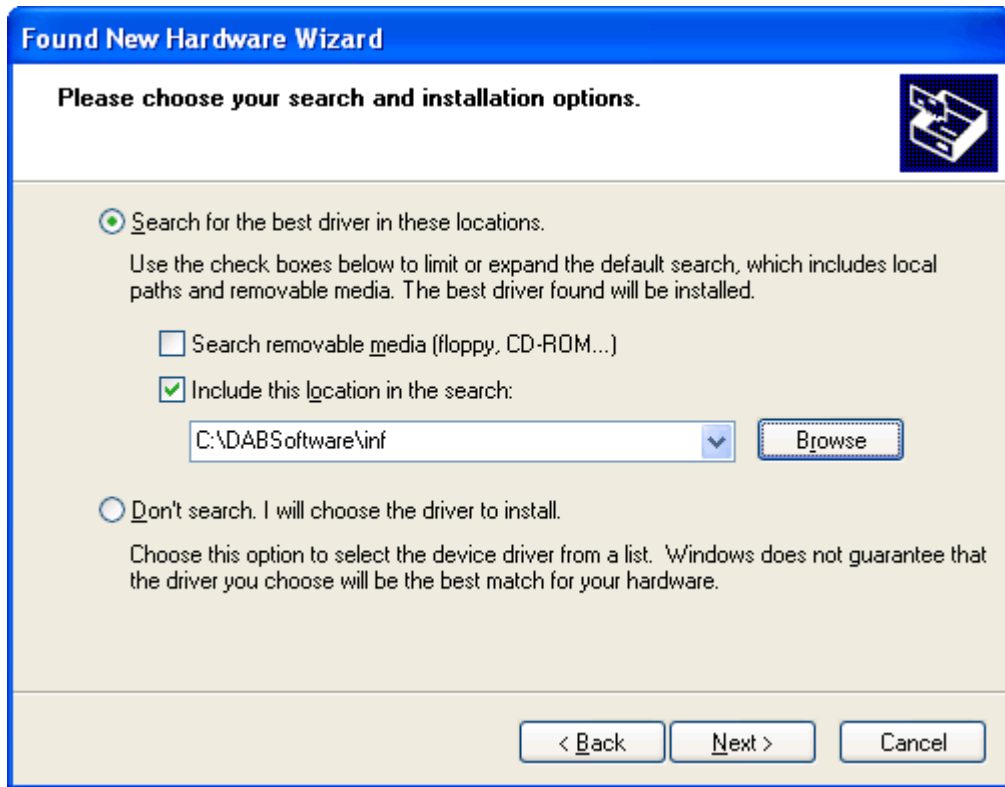


Figure 2.3: Directing Windows to the .INF file for the CDC Serial Emulation Driver.

Click “Next >”. The window should now begin loading the driver. If any warnings are issued (Figure 2.4), select “Continue Anyway”.

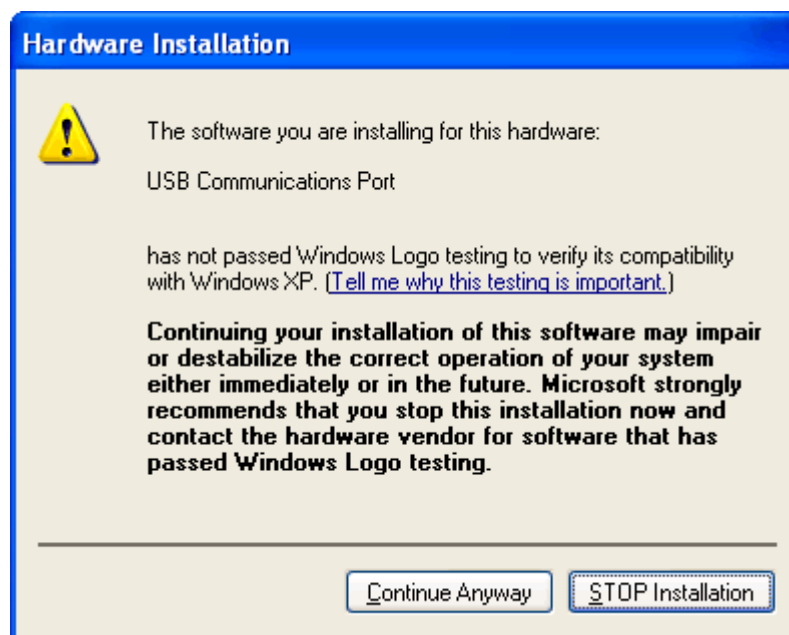


Figure 2.4: Directing Windows to the .INF file for the CDC Serial Emulation Driver.

The wizard should indicate that the software for the Communications port was successfully installed. Select “*Finish*”. (Figure 2.5)

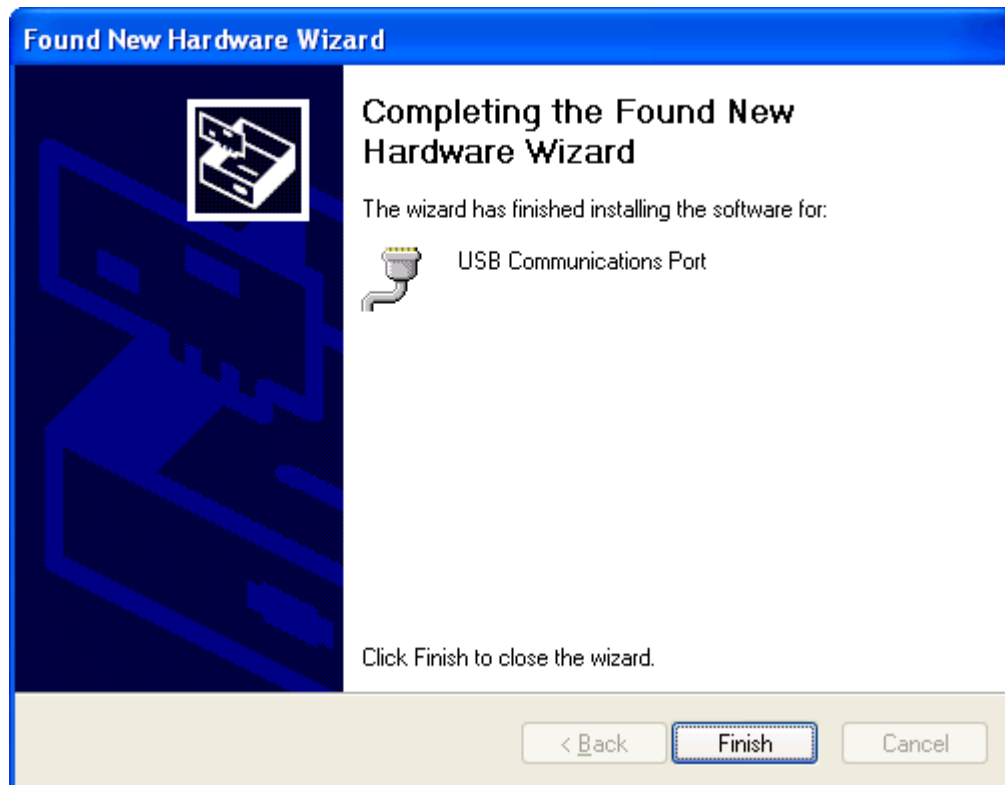


Figure 2.5: Successful Driver Installation Window.

Next, the virtual COM port will be checked in the Device Manager to identify the COM port number. Click on Window’s Start, Control Panel, System, Hardware, and finally Device Manager.

Expand the **Ports (COM and LPT)**. The new virtual COM port created by the *.inf* file should be found in the **Ports (COM and LPT)** drop-down list in the Device Manager window. Figure 2.6 shows the created COM port. The user’s COM port may differ.

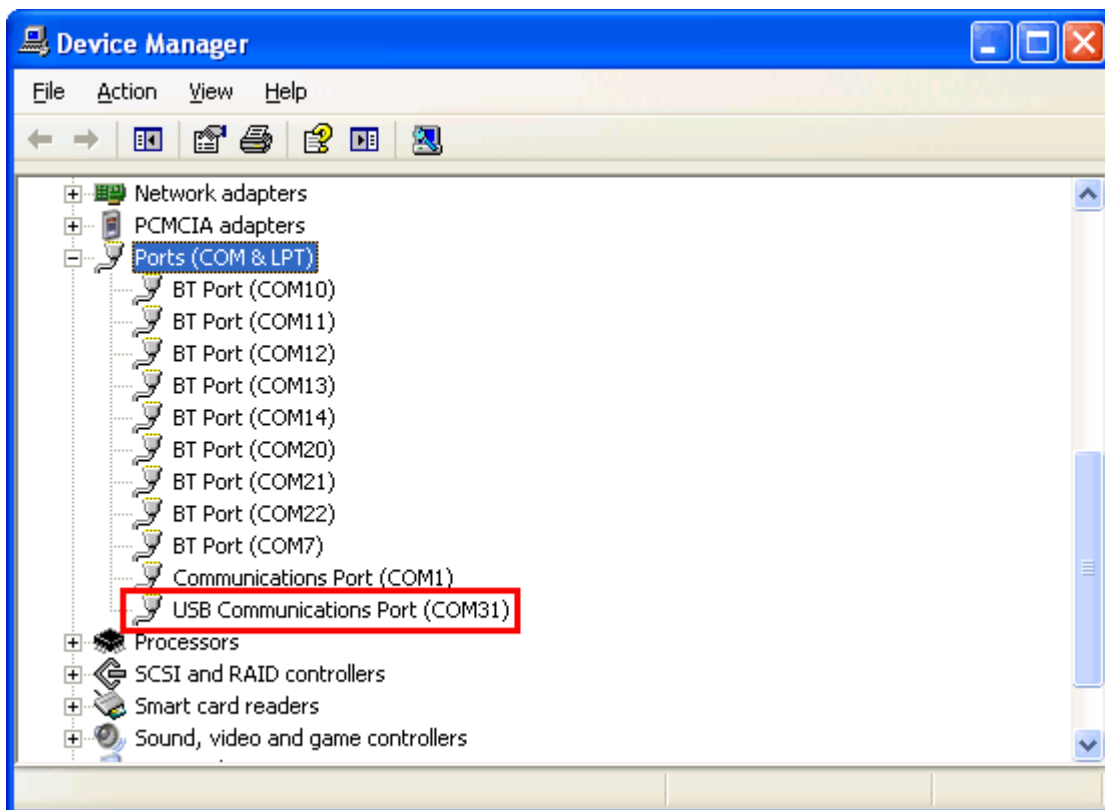


Figure 2.5: Device Manager showing COM port created.

Once the COM port driver has been successfully installed, the orange and the green LED will automatically turned off.

2.5 RUNNING THE SAMPLE PROGRAM

The sample program (VB.NET source code included) for the DAB+ FM Development Board requires Microsoft's .NET Framework 3.5 to work. The .NET Framework 3.5 can be found at:

<http://www.microsoft.com/download/en/details.aspx?id=21>

After installing the .NET Framework, navigate to the DAB software folder below:

C:\DABSoftware\TestKeyStoneRadio\bin\x86\Release

Execute **TestKeyStoneRadio.exe**

Figure 2.6 shows the default start up screen of the sample program.



Figure 2.6: DAB Sample Program Start up Screen

Click the green Power On button and select a COM port number created by the Hardware Wizard in step 2.4. The DAB+ FM Development Board will start communicating with the sample software. If everything is working properly, the orange and green LED will start to blink showing communication activities.

The DAB+ FM Development Board is designed and tested in Sydney, Australia. Therefore the stations around Sydney location get stored in the default memory while the board was under manufacturing test. In order to tune to user's DAB/DAB+ stations, double click on "DAB CHANNEL" word, the sample software will send a full rescan command to the board.

Figure 2.7 shows the basic navigation of the sample software and function of each button.



Figure 2.7: Basic Navigation and Function of each Button

1. Station Name. Drag the station name or frequency to Preset Button to save a channel.
2. RDS. Radio text.
3. DAB CHANNEL. Double click to force a full DAB channel rescan.
4. Total channels of DAB stations stored in the board.
5. Radio status.
6. DAB data rate.
7. Radio reception strength. Click on the tower icon to switch between stereo and mono.
8. Audio volume scale.
9. Radio ensemble name.
10. Preset buttons.
11. DAB stations list. Click station name to play.
12. Mute.
13. Genre.
14. Power on and off buttons.
15. Previous or Next channel.
16. Radio modes select, DAB or FM.
17. Volume control.
18. COM port selection.

2.6 INITIAL AUTO SCANNING

When the board is shipped outside of Sydney, Australia, the stored DAB stations will no longer be effective. Users need to perform initial auto scanning in order to receive the local stations of the country. Double click the word “DAB CHANNELS” to perform an auto scan.



Figure 2.8: Auto scanning

2.7 BBE SOUND

From the BBE Sound's website, BBE HD Sound is defined as

“BBE High Definition Sound compensates for phase and amplitude distortion, restoring the brilliance and clarity of the original content material. The BBE High Definition Sound process begins by applying a linear phase shift across the full frequency range of the signal, which allows the speaker system to reproduce the transients and harmonics in the correct order. The BBE process then compensates for speaker amplitude distortion by progressively boosting lower and higher frequencies, and doing so within the context of the phase correction process. This efficiently creates a fuller, richer sound without excess equalization.”

KeyStone working together with BBE Sound Inc, implemented BBE Sound in the core of the module.

2.7.1 BBE HD Sound Setting

BBE HD Sound feature in KeyStone’s module when properly tuned will deliver superior sound, however when it is not properly tuned or over tuned, the sound will cracked and distorted. Set the master volume to half then press the setup/eq button on the demo software to bring up the configuration screen. Follow the steps below to achieve a first good result.

- 1) Set Head Room to -12db
- 2) Click on BBE button to enable BBE
- 3) Slowly increase Hi to acceptable sound or slightly before it started to sound distorted.
- 4) Slowly increase Lo to acceptable sound or slightly before it started to sound distorted.
- 5) If you need more BASS, increase Mach Gain to 4db, anything over that, depending on the music will sound distorted.
- 6) Depending on your taste of BASS, adjusting Mach Freq, to 60hz, 90hz, 120hz or 150hz will have different low pass sound.
- 7) Surround, MP, HP Filter, HP Mode should be kept as low as possible or the sound will be distorted. Finally set the master volume to your taste.

2.7.2 PRESET EQ

Users can also select preconfigured EQ like, Bass Boost, Jazz, Live, Vocal and Acoustic.

2.7.3 ORIGINAL SOUND

If pure original sound is what you after, then switching both BBE and EQ off, and set head room to 0 will get you the original sound.

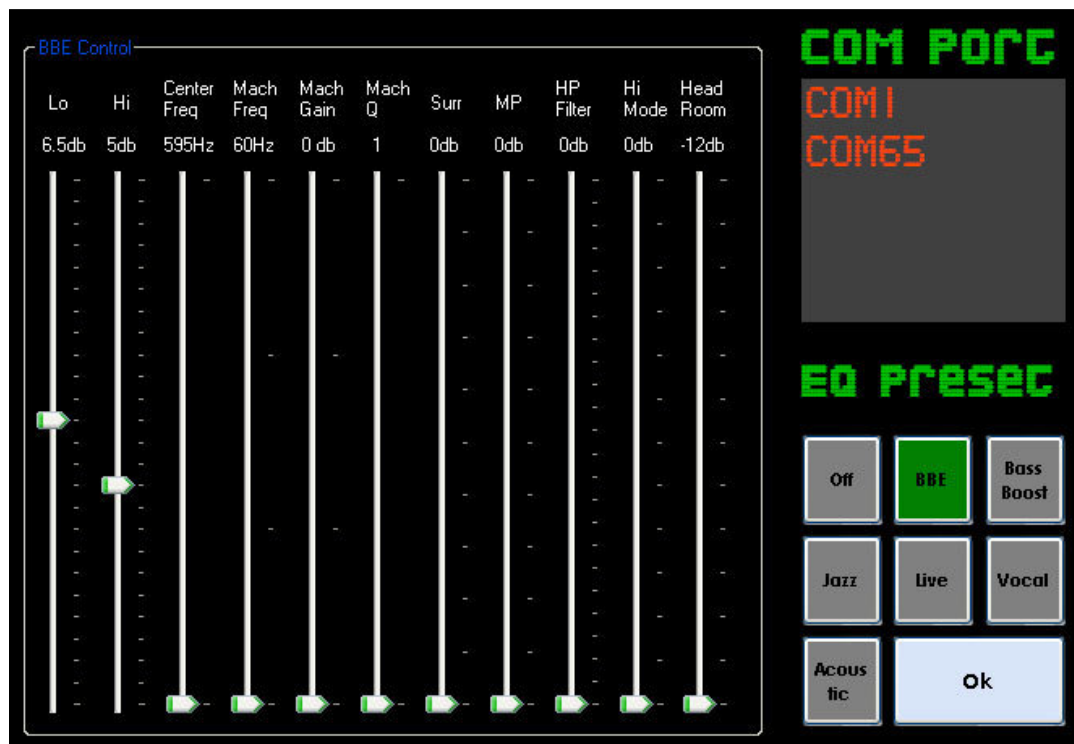


Figure 2.9: BBE HD Sound setting and configuration screen.

2.8 SLIDESHOW (Only on PRO Board that has T2_L4A_8650C module)

Slideshow adds visual content (slides) to radio broadcasts on DAB or DAB+. It enhanced the digital radio with visuals using images in JPEG or PNG format.



Figure 2.10: Slideshow in the middle of the demo software.

2.9 COMPILING THE SAMPLE PROGRAM

The provided source code was written in VB.Net using Microsoft Visual Studio 2008. There are 40 functions in the `keystonecomm.dll` library. User could study the provided sample VB.NET source code and translate the sample program into C# or C/C++ as needed.

The following VB.NET declarations are required to start a new project:

```
Function CommVersion Lib "keystonecomm.dll" () As Int32
Function OpenRadioPort Lib "keystonecomm.dll" (ByVal port As String, ByVal usehardmute As Boolean) As Boolean
Function HardResetRadio Lib "keystonecomm.dll" () As Boolean
Function CloseRadioPort Lib "keystonecomm.dll" () As Boolean
Function PlayStream Lib "keystonecomm.dll" (ByVal mode As SByte, ByVal channel As Int32) As Boolean
Function StopStream Lib "keystonecomm.dll" () As Boolean
Function SetVolume Lib "keystonecomm.dll" (ByVal volume As SByte) As Boolean
Function IsSysReady Lib "keystonecomm.dll" () As Boolean
Function VolumePlus Lib "keystonecomm.dll" () As SByte
Function VolumeMinus Lib "keystonecomm.dll" () As SByte
Sub VolumeMute Lib "keystonecomm.dll" ()
Function GetVolume Lib "keystonecomm.dll" () As SByte
Function GetPlayMode Lib "keystonecomm.dll" () As SByte
Function GetPlayStatus Lib "keystonecomm.dll" () As SByte
Function GetTotalProgram Lib "keystonecomm.dll" () As Int32
Function NextStream Lib "keystonecomm.dll" () As Boolean
Function PrevStream Lib "keystonecomm.dll" () As Boolean
Function GetPlayIndex Lib "keystonecomm.dll" () As Int32
Function GetSignalStrength Lib "keystonecomm.dll" (ByRef biterror As Integer) As SByte
Function GetProgramType Lib "keystonecomm.dll" (ByVal mode As SByte, ByVal dabIndex As Int32) As SByte
Unicode Function GetProgramText Lib "keystonecomm.dll" (ByVal programText As String) As SByte
Unicode Function GetProgramName Lib "keystonecomm.dll" (ByVal mode As SByte, ByVal dabIndex As Int32, ByVal namemode As SByte, ByVal programName As String) As Boolean
Unicode Function GetPreset Lib "keystonecomm.dll" (ByVal mode As SByte, ByVal presetindex As SByte) As Int32
Unicode Function SetPreset Lib "keystonecomm.dll" (ByVal mode As SByte, ByVal presetindex As SByte, ByVal channel As Int32) As Boolean
Function DABAutoSearch Lib "keystonecomm.dll" (ByVal startindex As Byte, ByVal endindex As Byte) As Boolean
Unicode Function GetEnsembleName Lib "keystonecomm.dll" (ByVal dabIndex As Int32, ByVal namemode As SByte, ByVal programName As String) As Boolean
Function GetDataRate Lib "keystonecomm.dll" () As Int16
Function SetStereoMode Lib "keystonecomm.dll" (ByVal mode As SByte) As Boolean
Function GetFrequency Lib "keystonecomm.dll" () As SByte
Function GetStereoMode Lib "keystonecomm.dll" () As SByte
Function GetStereo Lib "keystonecomm.dll" () As SByte
Function ClearDatabase Lib "keystonecomm.dll" () As Boolean
Function SetBBEQ Lib "keystonecomm.dll" (ByVal BBEOOn As Byte, ByVal EQMode As Byte, ByVal BBELo As Byte, ByVal BBELo As Byte, ByVal BBECFreq As Byte, ByVal BBEMachFreq As Byte, ByVal BBEMachGain As Byte, ByVal BBEMachQ As Byte, ByVal BBESurr As Byte, ByVal BBEMp As Byte, ByVal BBEPF As Byte, ByVal BBELo As Byte, ByVal BBELo As Byte) As Boolean
Function GetBBEQ Lib "keystonecomm.dll" (ByRef BBEOOn As Byte, ByRef EQMode As Byte, ByRef BBELo As Byte, ByRef BBELo As Byte, ByRef BBECFreq As Byte, ByRef BBEMachFreq As Byte, ByRef BBEMachGain As Byte, ByRef BBEMachQ As Byte, ByRef BBESurr As Byte, ByRef BBEMp As Byte, ByRef BBEPF As Byte, ByRef BBELo As Byte, ByRef BBELo As Byte) As Boolean
Function SetHeadroom Lib "keystonecomm.dll" (ByVal headroom As SByte) As Boolean
Function GetHeadroom Lib "keystonecomm.dll" () As SByte
Function MotQuery Lib "keystonecomm.dll" () As Boolean
Unicode Sub GetImage Lib "keystonecomm.dll" (ByVal ImageFileName As String)
Function GetApplicationType Lib "keystonecomm.dll" (ByVal index As Int32) As SByte
Sub MotReset Lib "keystonecomm.dll" (ByVal mode As SByte)
```

The sample program was provided to enable user to quickly learn the basic functionalities of the board, it has a lot of room for improvements, for example, in order to speed up the responsiveness of the user interface, and user could move some of the functions to worker threads and assign priority to each function.

Please refer to <http://www.monkeyboard.org/86-technical-documents/81-keystone-dab-fm-module-programming-api> for the latest API.

Chapter 3. Hardware Features and Descriptions

3.1 BOARD FEATURES

- KeyStone T1_L4A_8290C DAB/DAB+/FM radio module (Basic Board)
- KeyStone T2_L4A_8650C DAB/DAB+/FM/SLS/EPG radio module (PRO Board)
- Microchip PIC18F14K50 USB Flash Microcontroller
- 12 Mhz crystal
- standard ICSP programming port for programming the PIC18F14K50
- EXT port logic signal for controlling external audio chip power and mute
- status LED, RX (orange) and TX (green)
- power LED (red)
- Bootloader push button
- 1.2VDC, 1.8VDC, 3.3VDC LDO
- 3.5mm stereo jack
- SMA antenna connector
- Dimensions : 78mm x 45mm

3.2 HARDWARE DESCRIPTIONS

Power Supply Circuit

The board is typically powered by 5VDC from USB. The PIC18F14K50 is drawing power directly from the USB. From the USB's 5VDC, the power is further distributed into 1.2VDC, 1.8VDC and 3.3VDC LDO to the KeyStone T1_L4A_8290C module. The power of these three LDO is controlled by the **$\overline{\text{SHDN}}$** pin.

Microchip PIC18F14K50

This microcontroller is used to provide a Virtual Serial Port to the KeyStone module. It is flashed with a customised CDC RS-232 Emulation Demo firmware. Other than performing the standard serial emulation, it also control the power, reset and audio shunt of the KeyStone module.

Reset Circuit

Reset of the module is performed by **$\overline{\text{RESET}}$** pin emulated by **DTR** of the virtual serial port.

ICSP

The ICSP programming port is the standard Microchip's In-Circuit Serial Programming port and is compatible with Microchip's PICKit.

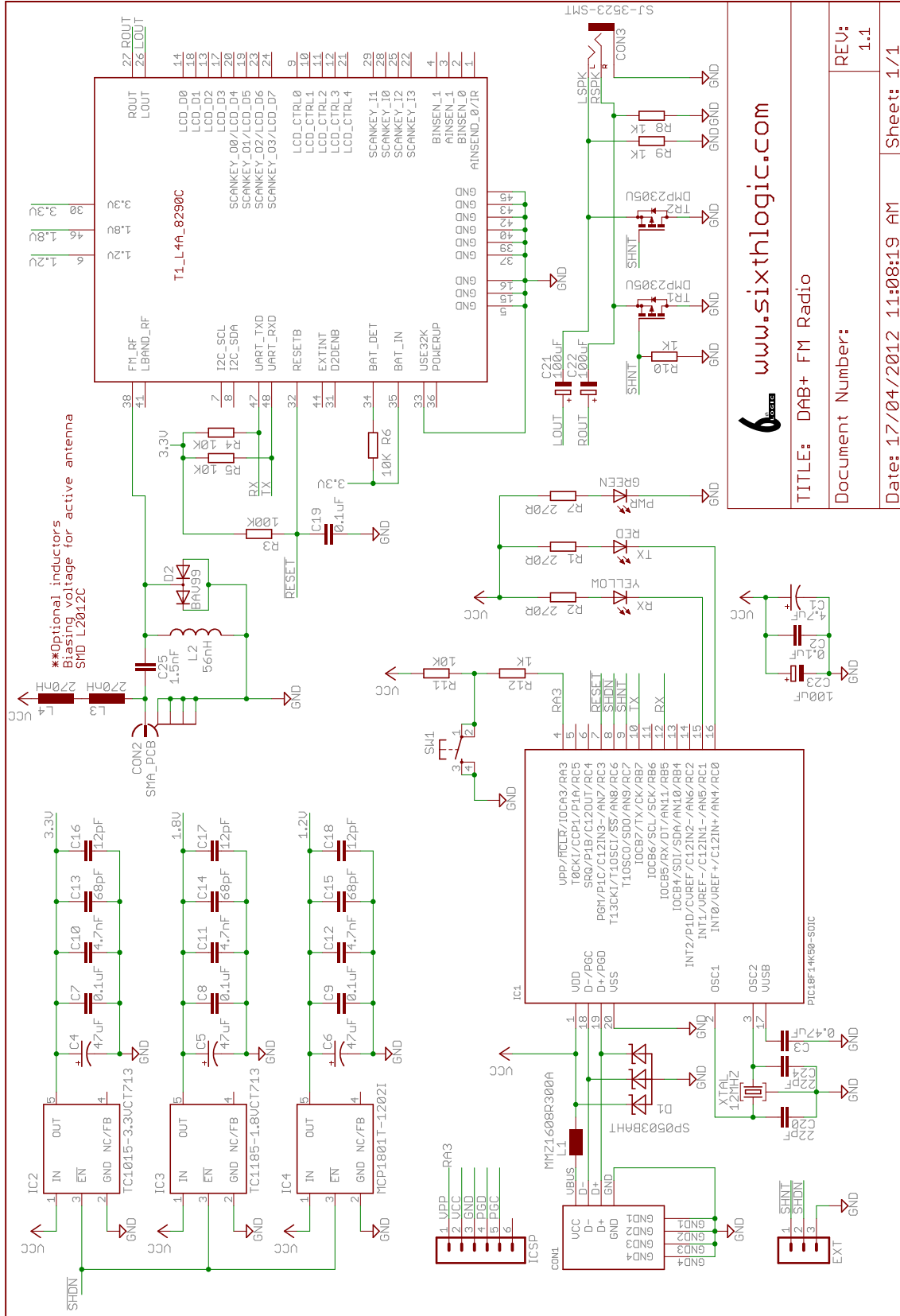
EXT

EXT port has two 5V logic signals, **$\overline{\text{SHDN}}$** and **$\overline{\text{SHNT}}$** . They are used to control external audio circuitry's power and mute.

Push Button (SW1)

This button, when pressed during a power-up, will trigger the PIC18F14K50's USB HID bootloader in programming mode.

APPENDIX A – SCHEMATIC



www.sixthlogic.com

TITLE: DAB+ FM Radio

Document Number:

Date: 17/04/2012 11:08:19 AM

Sheet: 1/1

REV: 1.1

