Atmel AVR32848: Android Accessory Demo

32-bit Atmel Microcontrollers

Features

- Control an accessory from an Android device
- Send data to and from an Android device to an accessory
- Supported development kits: EVK1104 and EVK1105
- Code examples for IAR and GCC Compiler

Introduction

Android smartphones represent nowadays (as of early 2012) more than 50% of a 650 million units per year market. Connecting an electronic device to an Android phone is becoming more and more popular. Examples range from sports equipment to medical devices, toys, audio systems, etc.

This document explains the usage of the demonstrator developed for the Android Open Accessory Protocol and based on Atmel microcontrollers. The examples are included in the Atmel Software Framework (ASF) to provide the customer with a quick and easy way to get started with developing an Android accessory. The use of the library is explained in the AVR4960 application note.

Deliveries

- App for Android phone
- Embedded firmware for EVK1104
- Embedded firmware for EVK1105

The demonstrator performs the following function

- Automatic launch of the Android App when connecting the development kit
- Displays the status of the kit Touch buttons on the phone screen
- Controls kit LEDs from the phone
- Displays on the kit LCD a message typed on the phone

Table of contents

1. Required tools, documents, and software for the demo

1.1 Required software

To use the demo several steps are necessary. The example is part of the huge collection of examples, drivers, services and more called Atmel Software Framework (ASF). ASF is an extension for the Atmel Studio 6 which provides an easy way to get started.

Atmel Studio 6 can be found here: http://www.atmel.com/Microsite/atmel_studio6/default.aspx

After installing Atmel Studio you should have access to the examples in the ASF. Chapter "Preparing the accessory" describes how to use the ASF to program the example code into the controller.

1.2 Required tools

The following section contains tools/hardware required to get started with the Android Open accessory. One compatible board is necessary to use as an accessory. The boards supported by Atmel are the EVK1105 and the EVK1104. Also necessary is a programmer/debugger suitable for the selected chip. The Atmel JTAGICE3 is recommended since it provides programming and debugging capabilities.

1.3 Required documents

This document covers all the information that are necessary to get started with the Android accessory demo. Additional information about the API of the Android accessory library can be found in the document AVR4960. Information about the Atmel Studio and the ASF can be found online at the Atmel homepage.

2. Getting started with the demo

2.1 Preparing the Android device

2.1.1 Compatible Android devices

To be compatible with the Android Accessory Protocol the phone has to run Android version 2.3.4 or later. Please note that due to the fact that the USB library has been back-ported from Android 3.1 to 2.3.4 the Android App only supports Android Version 2.3.4+. Please also note that the library is optional for Android 2.3.4 which means there are devices that are not compatible with Android Open Accessory even with Android version 2.3.4.

The device that has been tested and confirmed to work is the HTC Desire HD running Android Version 2.3.4.

2.1.2 Installing the Android application

To install the App on the Android device you need to download it from the Atmel homepage. The .apk file can be found here: http://www.atmel.com/Images/AVR32848.zip

The next step is to install an application from the Android market that allows you to install apps from your SD card. There are several Apps available for this purpose the one that has been tested is called "AppInstaller" by FunTrigger. After you copied the .apk file to the SD card of your Android device you can launch the "AppInstaller", navigate to the folder into which you copied the .apk file and install the Android accessory application. Please note that you have to allow installation of non-Market applications.

2.1.3 Using the Android application

The software is divided into the two main parts input from the accessory and output to the accessory.

Input panel

This panel shows all the incoming information from the Android device. There is no user interaction possible to this panel on the Android side. Only the accessory can change the values displayed here. The picture shows the input tab for the Atmel EVK1105. The input panel is chosen based on which kit is connected to the Android device. In simulation mode the App will display the input panel for the EVK1105.
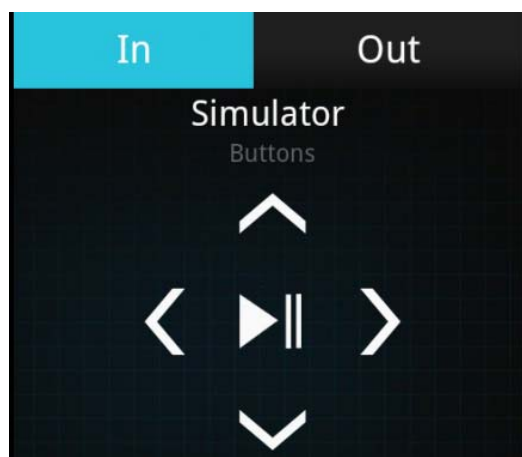
Output panel

This panel gives options to send commands to the accessory to control the LEDs and the display. The dimmable LEDs can be controlled using a slider, the switchable LED is controlled using a button and the display can be used using a textbox and the keyboard build into Android. The corresponding figure shows the output panel for a connected EVK1105 board.
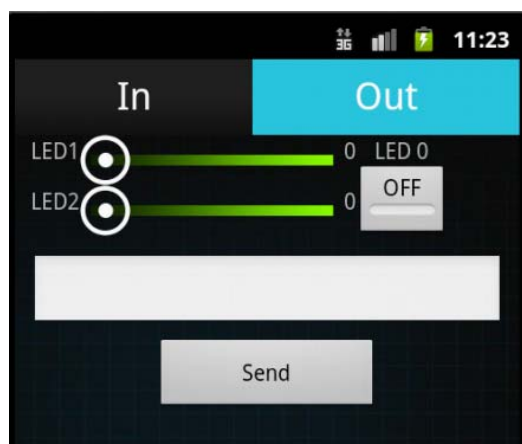
2.2 Preparing the accessory

As described in chapter "Required tools, documents and software" you need to have Atmel AVR Studio® 5.1 or Atmel Studio 6 installed to have access to the demo. To use the demo open AVR Studio 5.1 or Atmel Studio 6 and go to File -> New -> Example Project from ASF. Then to find the demo in the ASF provided for Atmel Studio type in the board name in the search window and look for the demo labeled "USB Host Android Accessory demo". Open the demo and compile it. Now you can either start debugging the application or program it to the microcontroller to play around.

2.2.1 Important notice - preparing the demo

To run the Android accessory demo it is necessary to supply the board with an external power supply that is able to provide more than 500mA@5V. The reason for this is that the kit must operate in USB Host mode as an accessory and therefore the kit must also provide 500mA to the Android device in addition to its own current consumption. The connector for an external power supply is located at different positions for the Atmel EVK1104 and the Atmel EVK1105. The instructions on how to connect the supply are shown in the next chapters.

2.2.2 Connecting an external power supply on the EVK1104

It is necessary to supply the power to J2 in order to use the demo.

Note: Make sure to supply enough power for the board AND the Android device!

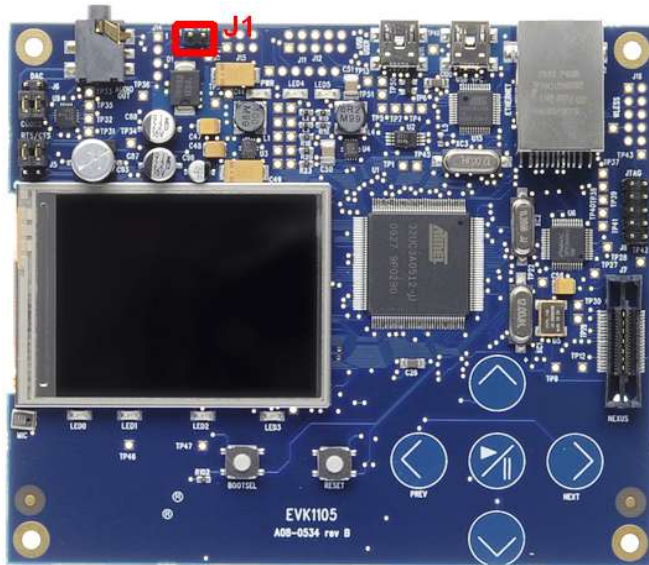**Figure 2-1.   EVK1104 power supply.**
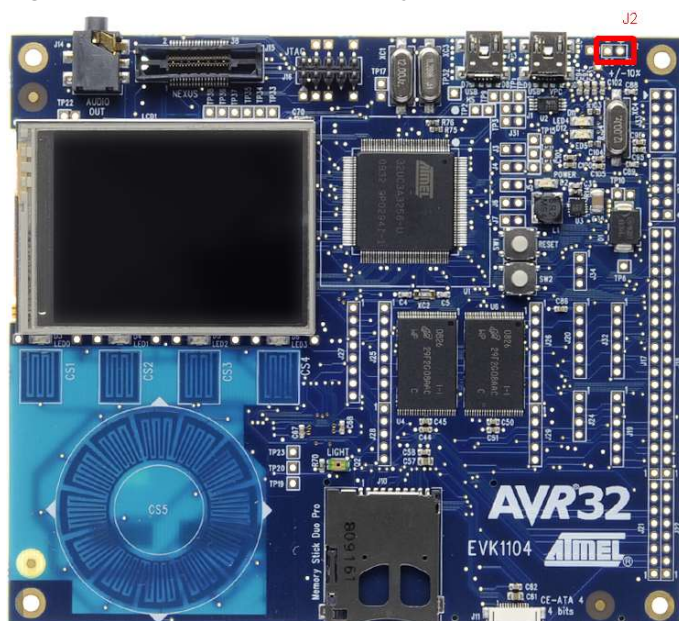
2.2.3 Connecting an external power supply on the EVK1105

It is necessary to supply the power to J1 in order to use the demo.

Note: Make sure to supply enough power for the board AND the Android device!

Figure 2-2. EVK1105 power supply.

2.3 Connecting the Android device to the development kit

To start the demo, connect the Android device to the left USB connector as shown in .

Figure 2-3. USB port to use on EVK1104.

Figure 2-4. USB port on EVK1105.

The screen will change according to the enumeration process. A successful connection will be indicated by a textbox labeled "Atmel Android Accessory Demo" on the top of the screen. In case the enumeration fails for some reason the screen will show a message "This Device is not compatible". The enumeration can fail for one of the following reasons:

- Wrong Android version
- Not an Android device

If the enumeration is successful the Android device will ask you to launch the App for the demo if it is already installed. If it is not installed yet it will provide a link to the App on the Atmel website.

After installing the app/giving the App the rights to communicate using the USB port the App provides two tabs. The first one is labeled "In" displays the name of the board and the status of the touch buttons. The second one labeled "Out" provides functionality to control the LEDs and the screen on the accessory.

3. Android accessory application

3.1 Overview

Currently there are two fully running demos for this application which provide a nearly equivalent set of functions for two different boards. Supported boards are the Atmel EVK1105 and the Atmel EVK1104. The shared functionality will be explained in the corresponding chapter with a section covering the differences.

4. Core functionalities of the demo

The main functionalities of this demo are:

- Control LEDs on the boards
- Write text to the display on the development boards
- Read status of the touch buttons on the board and display the status on the Android device

4.1 Common features

4.1.1 Controlling the LEDs

There are four LEDs on each of the supported boards. Three of them can be controlled by the Android application and one LED indicates a connected USB OTG cable. Two LEDs, controlled by the Android application, can be dimmed using a PWM and one can be toggled.

4.1.2 Reading the buttons

Any change to the status of the buttons is reported to the Android app. Both boards use an external touch to check the buttons. However since one of the chips uses interrupts to keep track of pressed buttons and the other one is polling the buttons the method is described in the board specific section.

4.1.3 Writing text messages to the display

Both boards are using the same display module which is used to display text messages from the Android application. The firmware makes use of the display driver for the et024006 module included in the ASF. It provides the basic functionality to initialize the display and display strings sent from the Android device.

4.2 Atmel EVK1105 specific implementation

4.2.1 LEDs

The LEDs are used in the following manner:

LED	Function
LED0	Indicates connected USB OTG cable
LED1	Can be toggled using the Android App
LED2	Can be dimmed by adjusting the duty cycle of the PWM using the Android App
LED3	Can be dimmed by adjusting the duty cycle of the PWM using the Android App

4.2.2 Buttons

The EVK1105 uses the QT™1081 to read the touch buttons. The chip monitors the buttons independently from the main MCU and issues an interrupt when a button is pressed.

4.3 Atmel EVK1104 specific implementation

4.3.1 LEDs

The LEDs are used in the following manner:

LED	Function
LED0	Can be dimmed by adjusting the duty cycle of the PWM using the Android App
LED1	Can be dimmed by adjusting the duty cycle of the PWM using the Android App
LED2	Can be toggled using the Android App
LED3	Indicates connected USB OTG cable

4.3.2 Buttons

The EVK1104 uses the QT60168 to read the touch buttons. The chip monitors the buttons independently from the MCU and the status of the buttons is read by the MCU via SPI periodically.

4.3.3 Light sensor

A light sensor is mounted on the EVK1104 and is connected to one of the ADC channels of the MCU. The raw ADC measurement result is periodically sent to the Android device.

5. Android application

5.1 Adding your own features

If you want to adjust the Android app to your personal needs you can download the Atmel source code here: http://www.atmel.com/Images/AVR32848.zip

Instructions on how to program your App can be found here:

http://developer.android.com/guide/index.html

The Android Accessory Protocol is explained here:

http://developer.android.com/guide/topics/usb/adk.html

6. Troubleshooting

If you encounter issues while running the demo please see the information given below. If you encounter a problem that is not covered contact android@atmel.com or file a bug report on this web page: http://asf.atmel.com/bugzilla/

6.1 Determine the Android version and update Android

If you have troubles connecting the accessory to the phone your Android device may run an old Android version. You can find the Android version by going to "Settings -> About Phone -> Software Information/Android Version" or similar entries depending on the Device you are using.

The update function can usually also be found in the "Settings -> About Phone" settings. For the HTC Desire HD it can be found in "Settings -> About Phone -> Software updates".

6.2 The Android App does not work anymore after pushing the reset button on the accessory

It is not possible to detect a short interrupt of the USB connection on the Android side. The accessory however will try to re-enumerate the Android device which will be successful but the Android accessory connection remains desynchronized. To prevent this make sure to disconnect the Android device from the accessory for at least 1 second either by pressing the reset button for this amount of time or disconnect the USB plug.

6.3 Closing/pausing the App and restarting it does not work

This is a confirmed bug on the Android side which is still to be fixed by the Android OS.

Appendix A. Revision history

Doc. Rev.	Date	Comments
32192A	5/2012	Initial document release

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan G.K.
16F Shin-Osaki Kangyo Building
1-6-4 Osaki
Shinagawa-ku, Tokyo 141-0032
JAPAN
Tel: (+81)(3) 6417-0300
Fax: (+81)(3) 6417-0370