

Generic SDK Build on Raspberry Pi

Draft Version 1.0
June 2015



Table of Contents

1. Purpose.....	6
2. Generic SDK Build System.....	6
3. Building Generic SDK	6
3.1 Native compilation	6
3.2 Cross Compilation	6
4. Building Microsemi ALSA Driver for Pi.....	7
3.3 Testing HBI Driver	7
3.3.1 Procs interface.....	7
3.3.2 Sample Apps	7
4 Troubleshooting	8



List of Tables

Table 1 Abbreviations used in this document

Hata! Yer işareti tanımlanmamış.

List of Figures

No table of figures entries found.

Document Owner	Shally Verma
Version	1.0
Date	June 15

Document Change History			
Date	Version	Changed by	Change Description
15 June 2015	1.0	Shally Verma	Re-formatted Generic SDK Build/Setup document on Pi as per Microsemi template. Added history section to keep track of changes. Document modified with latest build instructions. Env.sh file is redundant and no need to source any more
11 Aug 2015	1.0	Shally Verma	Added SPI HBI driver build and run section for Pi
3 june 2016	1.0	Shally Verma	Update sample apps build. Section for native and cross-compilation
9 Dec 2016	1.0	Shally Verma	Updated document with links to compile SDK natively on raspberry and using cross compiler



1. Purpose

This document describes Generic Software Development Kit build and setup for Raspberry Pi Platform.

2. Generic SDK Build System

Development source code for Generic SDK is available on svn at http://aussvn01/svn/apps/vproc_sdk/trunk

There are two primary files inside the root directory of Generic SDK : Makefile and Makefile.globals.

All the environment variables and build option setting is done in Makefile.globals where as Makefile act as Master makefile which contains all rules for building different targets.

3. Building Generic SDK

Microsemi VPROC Generic SDK can be both natively or cross compiled.

3.1 Native compilation

- Please refer to ZLS38100_SDK_Build_Guide_Raspberry.docx for instructions on native compilation of sdk

3.2 Cross Compilation

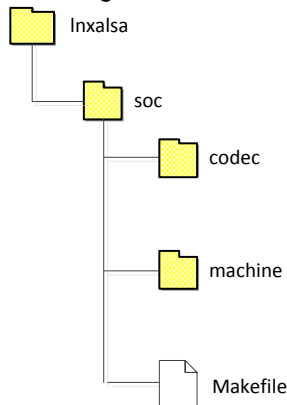
- Please refer to SDK_Build_Guide_Raspberry_CrossCompiler.docx for cross compilation of sdk



4. Building Microsemi ALSA Driver for Pi

Microsemi ALSA Driver for Pi is located at /trunk/platform/raspberry/driver/sound/lnxalsa and comply ASoC architecture (<http://www.alsa-project.org/main/index.php/ASoC>).

Code organization of Microsemi ALSA driver is something like this:



“soc” directory contains the ALSA driver code for codec (i.e. zl380xx) device and a machine driver to register Microsemi Vproc device as sound card to Pi.

Microsemi ALSA driver currently support device tree based build procedure only. Thus a device tree source file Microsemi-dac-overlay.dts is available which identifies Microsemi DAC as device to Pi.

Please refer further to build guides on detailed instructions on enabling device tree overlays

3.3 Testing HBI Driver

3.3.1 Procfs interface

Applicable to linux based system. HBI driver can be tested using HBI linux driver procfs interface (enabled/disabled based on HBI_ENABLE_PROCFS option). For details of these interface, please refer to HBI_Linux_Driver_Specification.docx

3.3.2 Sample Apps

Number of sample applications are available in /apps directory.

Some apps built by default. Some need enabling TEST macro during compilation.

Example to build “direction-of-arrival”, do ‘make apps TC_DOA=1

4 Troubleshooting

Q. Audio not coming out of speaker on “aplay”

A. This may be because of loose i2s connections between pi and vproc device Or clock not coming properly to vproc as Microsemi vproc device configured as i2s slave.