
**AT10934: Audio Recorder with SAM G53-XPRO and
MEMS MIC XPRO**

SMART ARM-based Microcontrollers**Introduction**

This application note describes the audio recording application based on SAM G53 Xplained Pro, MEMS MIC Xplained Pro, and I/O1 Xplained Pro kits.

The recording process is based on MEMS MIC Xplained Pro using WM7220 MEMS microphone hardware with a PDM hardware interface controller on SAM G53.

PDM hardware itself converts 1-bit PDM data to PCM.

This application uses SD card from I/O1 Xplained Pro to store recorded audio file.

The application source code is available in Atmel web.

Features

- Audio Recorder Application
- 16KHz Stereo Audio Recording Support

Table of Contents

1	PDM Peripheral Overview	3
1.1	Description	3
1.2	Characteristics	3
1.2.1	Pulse Density Modulation Interface Block Diagram	4
2	Hardware Requirements	5
3	Software Requirements.....	5
4	Audio Recorder Application with SAM G53 Xplained Pro	6
4.1	Application Overview.....	6
4.2	Application Configuration	6
4.2.2	PDM Interface.....	7
4.2.3	Mono Left Channel Mode	7
4.2.4	Mono Right Channel Mode.....	7
4.2.5	Stereo Channel Mode.....	8
4.2.6	SD Card Interface	8
4.3	Flowchart.....	9
4.3.1	sysclk_init	11
4.3.2	sd_mmc_init	11
4.3.3	sd_mmc_test_unit_ready.....	11
4.3.4	Start_audio_recorder	11
5	Getting Started.....	13
5.1	Hardware Interface.....	13
5.1.1	Mounting Jumpers in MEMS MIC Xplained Pro.....	13
5.1.2	Connect Extension Boards to SAM G53.....	13
5.2	Software Interface	14
5.3	Programming the Application	16
5.4	Procedure.....	17
6	Result	17
7	User Configuration	18
7.1	Creating Project for Newer ASF Version	18
8	References.....	20
9	Revision History	21

1 PDM Peripheral Overview

1.1 Description

The Pulse Density Modulation Interface Controller (PDMIC) from SAM G53 is a PDM interface controller and decoder that support both mono/stereo PDM format. It integrates a clock generator driving the PDM microphones and embeds filters which decimate the incoming bit stream to obtain most common audio rates.

The PDMIC includes a DSP section containing a decimation filter, a droop compensation filter, a sixth-order low pass filter, a first-order high pass filter and an offset and gain compensation stage. Refer to datasheet for more information on Digital Signal Processing (Digital Filter).

Pulse-density modulation, or PDM, is a form of modulation used to represent an analog signal with digital data.

The audio signal coming from MEMS microphone is in the form of PDM signal. PDM uses only one bit to convey audio.

The PDM clock (PDMCLK) is used to sample the PDM bit stream. The SAM G53 device provides the Master clock to Microphone to transmit the PDM data.

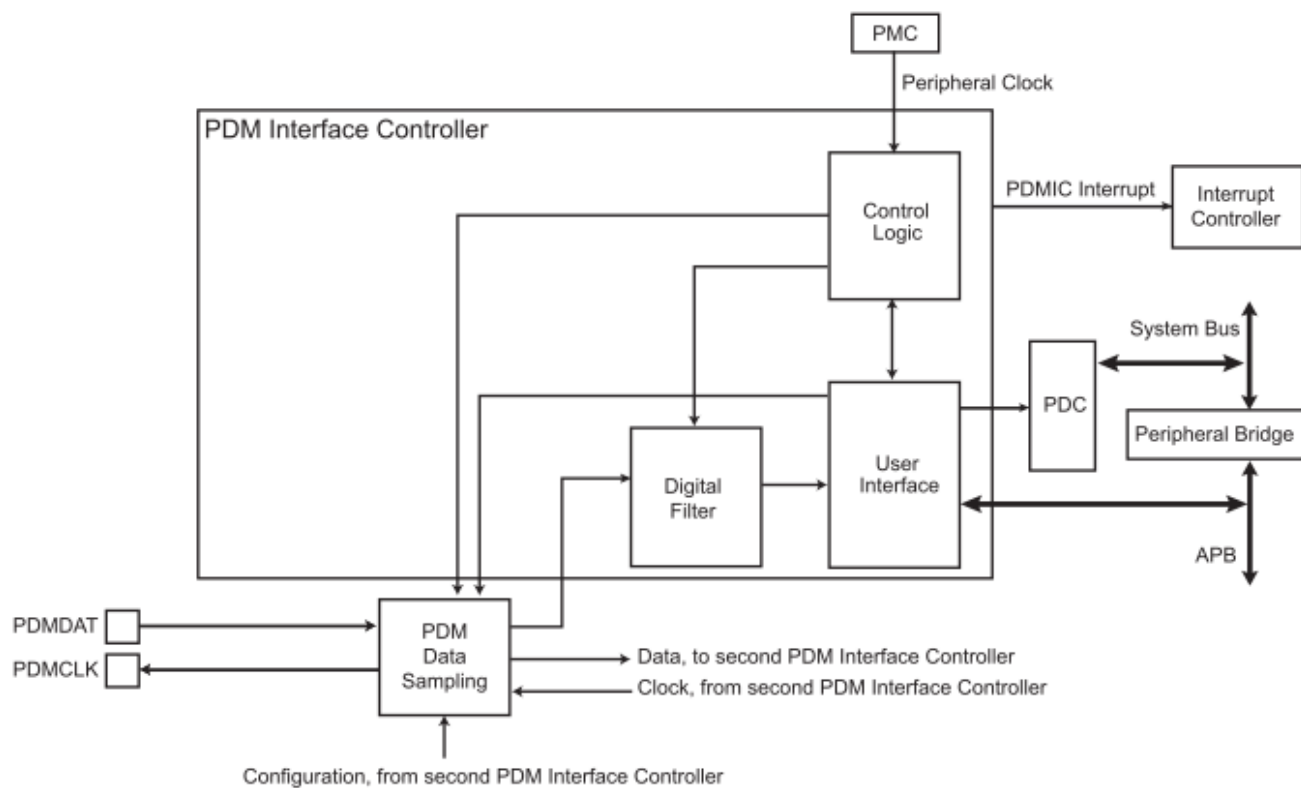
The one-bit data is asserted on the data line on either the rising or falling edge of the master clock. For PDM stereo operation, in which one microphone asserts the data line on the rising edge of the master clock which is routed to PDMIC0, while a second microphone asserts the data line on the falling edge which is routed to the PDMIC1.

The data coming from the microphone is sent to the decimation process where the 1-bit PDM data is converted to 16-bit PCM data and it is decimated by a factor of 64 or 128.

1.2 Characteristics

- Multiplexed PDM Input Support
- 16-bit Resolution
- PDC Support
- Register Write Protection

1.2.1 Pulse Density Modulation Interface Block Diagram



2 Hardware Requirements

This application requires the following hardware:

- SAM G3 Xplained Pro board
- I/O1 Xplained Pro board with microSD card
- USB Cable (Standard A to micro-B USB)
- MEMS MIC Xplained Pro board

3 Software Requirements

The software needed for this application includes:

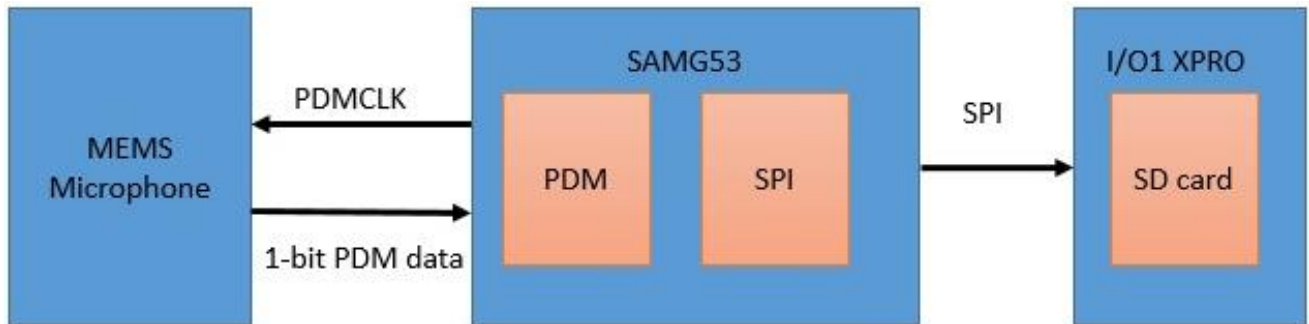
- Atmel Studio 6.2 or later

4 Audio Recorder Application with SAM G53 Xplained Pro

4.1 Application Overview

The application records audio data from MEMS microphone Xplained Pro via PDM interface once SW0 button is pressed in SAM G53 Xplained Pro kit.

Figure 4-1. Application Block Diagram



This application uses SD card from I/O1 Xplained Pro to store recorded audio data. SPI interface is used to write the audio data to SD card. FATFs is used to create file in SD card.

Once SW0 button is pressed in SAM G53 Xplained Pro, audio recording will get start, LED on I/O1 Xplained Pro Board is ON and the yellow LED0 at SAMG 53 Xplained Pro also blinks which indicates that audio is recording.

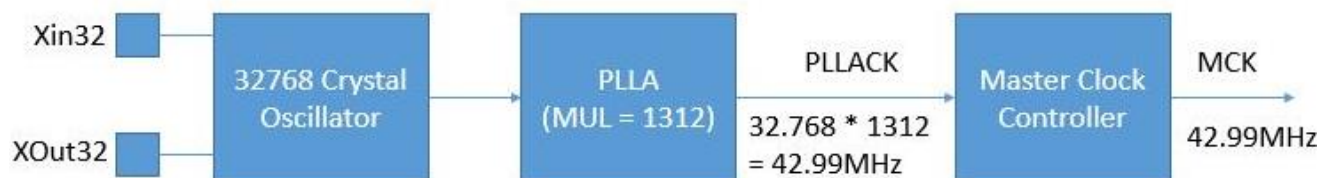
When SW0 button is pressed again, recording will stop and LED0 on I/O1 Xplained Pro board is OFF.

The recorded audio is stored as *.wav format (audio_0.wav) in SD card which is connected to I/O1 Xplained Pro board. Once the SD card is inserted in PC, the recorded audio can be directly played in any media application.

4.2 Application Configuration

The SAM G53 Xplained Pro kit contains one mounted 32.768kHz crystal. The device is operated using 32.768kHz Crystal Oscillator and the PLLA is configured to a value of 1312 and the resulting device operating frequency is 42.99MHz.

Figure 4-2. Clock Configuration



The modules used by this application are:

PDM: Converts audio data to 16-bit PCM data.

PDC: This is used to reduce CPU load when transferring data from buffers to SD card.

SPI: This is used to transmit data from SAM G53 to SD card.

Button (SW0): used to control application (start recording or stop recording).

4.2.2 PDM Interface

Pulse-density modulation, or **PDM**, is a form of modulation used to represent an analog signal with digital data.

An analog signal fed on the MEMS microphone can be directly sampled at a high sampling rate of 1.024MHz. The data coming from the microphone is sent to the decimation process where the 1-bit PDM data is converted to 16-bit PCM data and it is decimated by a factor of 64. ($1.024\text{MHz}/64 = 16\text{ KHz}$.) So, the resulted data is 16-bit/16KHz PCM signal.

Table 4-1. PDMIC Pin Description

EXT	Pin name	Description	Type
PA10	PDMCLK	Pulse Density Modulation Bit stream Sampling Clock	Output
PA09	PDMDATA	Pulse Density Modulation Multiplexed Data	Input

Warning: Since EDBG virtual COM port and PDM share same pins (PA09, PA10), EDBG COM port cannot be used.

4.2.3 Mono Left Channel Mode

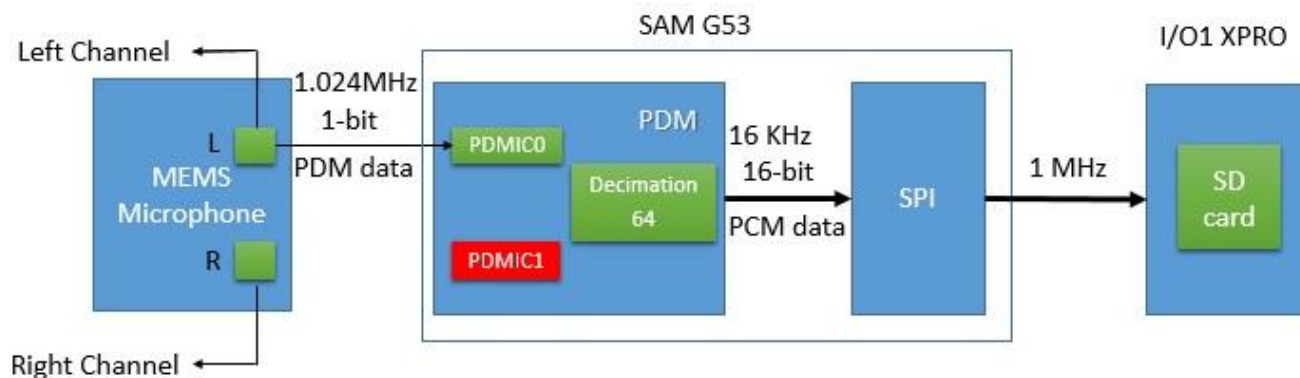
The device accepts 1-bit PDM audio data from MEMS microphone left channel.

The application is configured for PDMIC0 to get the data from MEMS microphone left channel.

For Mono Left Channel mode, the audio signal fed on the MEMS microphone transmits the 1-bit PDM data from MEMS left channel to PDMIC0.

`pdm_start_mono_recording_left_channel()` function configure PDMIC0 (PDM left channel) for Mono left channel.

Figure 4-3. Mono Left Channel



4.2.4 Mono Right Channel Mode

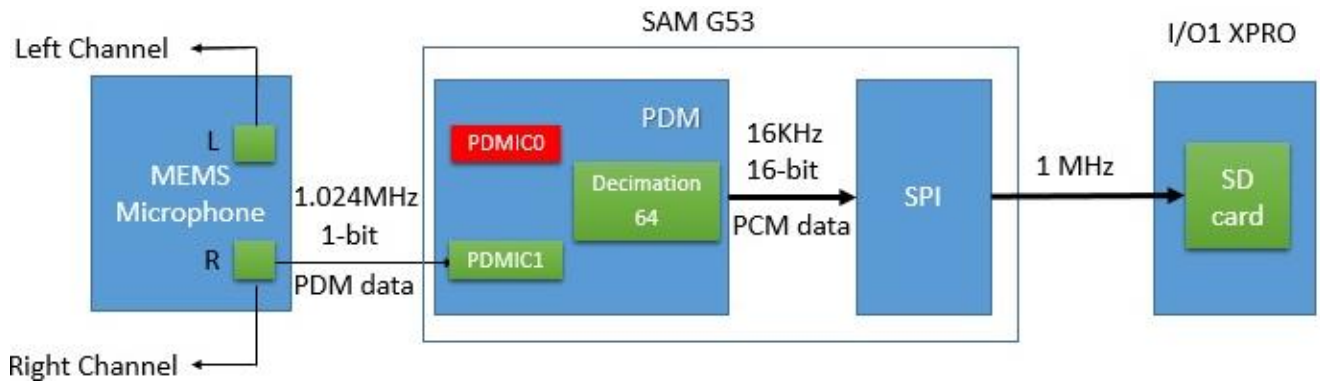
The device accepts 1-bit PDM audio data from MEMS microphone right channel.

The application is configured for PDMIC1 to get the data from MEMS microphone right channel.

For Mono Right Channel mode, the audio signal fed on the MEMS microphone transmits the 1-bit PDM data from MEMS right channel to PDMIC1.

`pdm_start_mono_recording_right_channel()` function configure PDMIC1 (PDM right channel) for Mono right channel.

Figure 4-4. Mono Right Channel

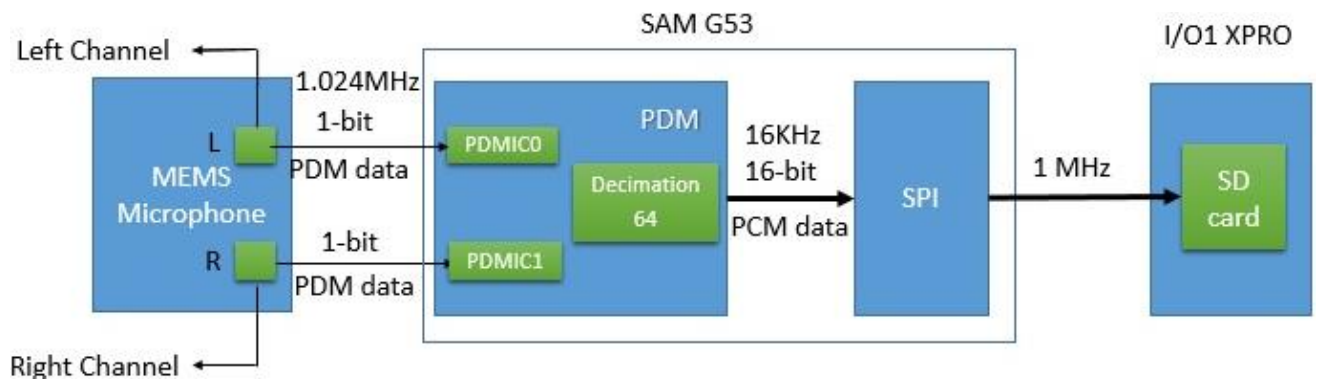


4.2.5 Stereo Channel Mode

The device accepts 1-bit PDM audio data from MEMS microphone left channel and right channel. When stereo mode is used, the microphone asserts the right channel data on the rising edge and asserts the left channel data on the falling edge.

The application is configured for PDMIC0, PDMIC1 to get the data from MEMS microphone right channel and left channel for stereo mode.

Figure 4-5. Stereo Mode



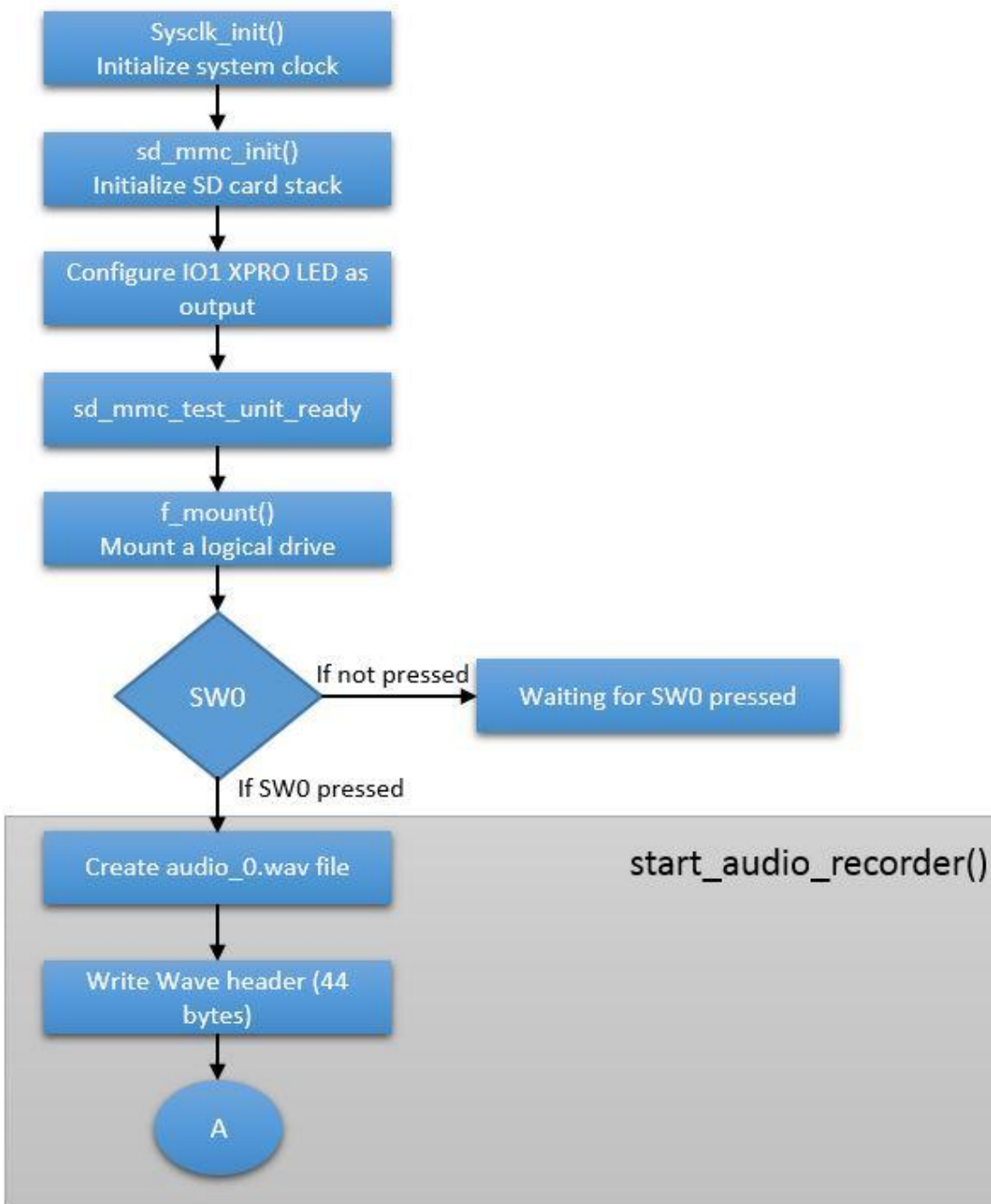
4.2.6 SD Card Interface

I/O1 Xplained Pro features a microSD card connector that connects to cards via SPI interface. MicroSD is used to store the converted audio file as WAVE format. The WAVE file format is a Microsoft and IBM audio file format standard for storing an audio bit stream on PCs.

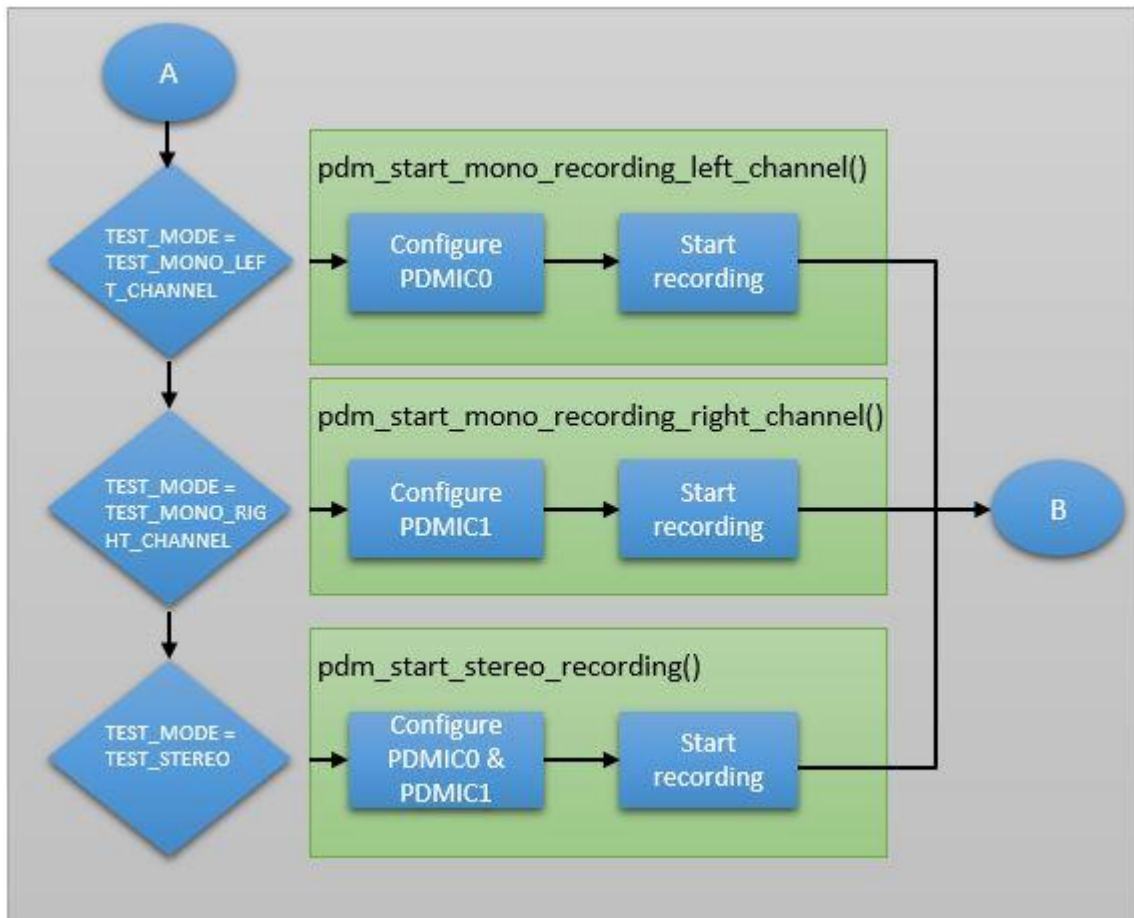
Refer to [WAVE format](#) for more information on WAVE PCM sound file format.

Once the conversion is completed i.e., when the 16-bit PCM data is ready, the raw data can be written to SD card using SPI interface at the rate of 1MHz.

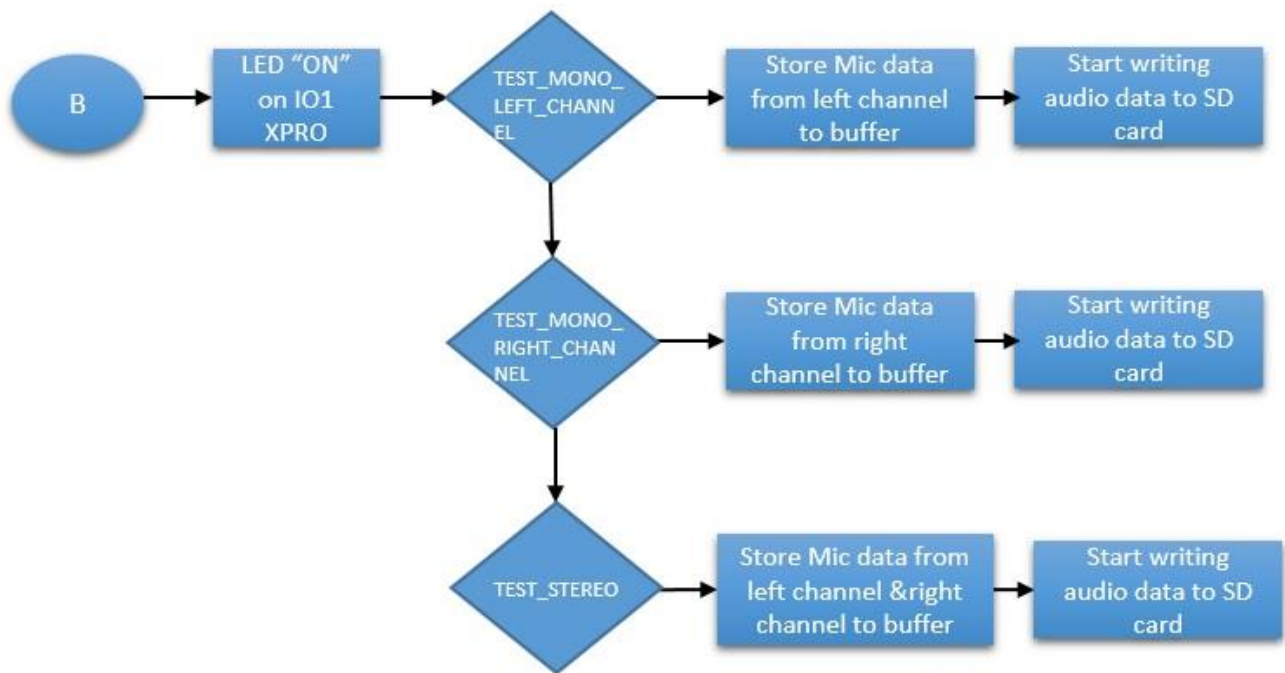
4.3 Flowchart



Continued...



Continued...



Note: Ensure that SD card is inserted in the slot before pressing SW0 button and SD card has to be inserted until next SW0 button.

4.3.1 sysclk_init

This function initializes System clock. The device is operating at 32768Hz frequency Crystal Oscillator and PLLA is configured with a value of 1312, so the resulting device operating frequency is $32768 * 1312 = 42.99\text{MHz}$.

4.3.2 sd_mmc_init

This function initializes SD/MMC card stack and also this function initializes SPI in Master mode and enable it. Then the application configures LED on I/O1 Xplained Pro as output. This LED is used to indicate whether recording is ongoing or not.

4.3.3 sd_mmc_test_unit_ready

This function tests the memory state and initializes the memory. This function waits for plugging a microSD card in slot. This is used to check the media status.

FAT file system driver that provides an interface for accessing a FAT file system located on a memory device. Once the SD card is ready, the application waits for SW0 button pressed.

4.3.4 Start_audio_recorder

If SW0 is pressed, the application uses FAT file system to create an audio0.wav file in SD card to write the audio data in file. Once audio0.wav file is created in SD card, a wave header of 44 bytes is written to audio0.wav file.

The application configures PDMIC0 or PDMIC1 or both according to the mono left channel or mono right channel or stereo mode respectively.

Once SW0 button is pressed in SAM G53 Xplained Pro, audio recording will get start and LED on I/O1 Xplained Pro Board is ON which indicates that audio is recording.

When SW0 button is pressed again, recording will stop and LED on I/O1 Xplained Pro board is OFF.

The recorded audio is stored as *.wav format (audio_0.wav) in SD card which is connected to I/O1 Xplained Pro board. Once the SD card is inserted in PC, the recorded audio can be directly played in any media application.

5 Getting Started

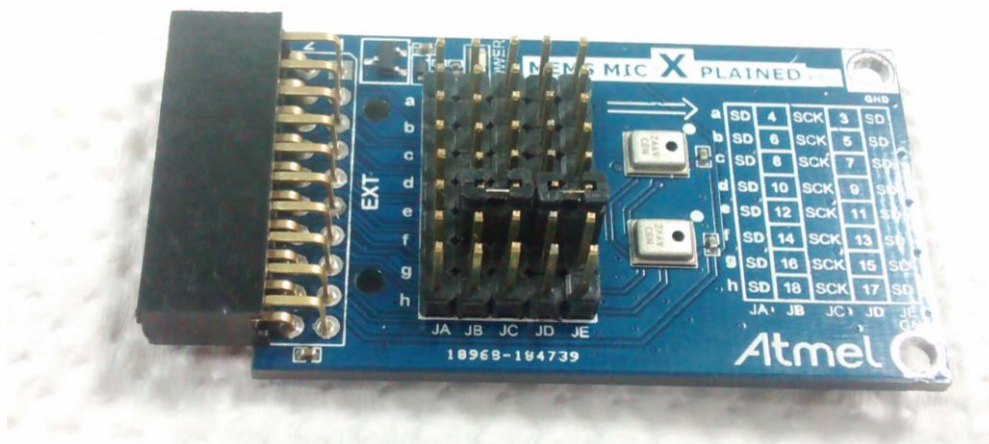
5.1 Hardware Interface

5.1.1 Mounting Jumpers in MEMS MIC Xplained Pro

In the MEMS MIC Xplained Pro, the columns JB, JD are EXT pins whereas JA, JE are MEMS mic DATA line and JC is MEMS MIC clock (SCK) line.

SAM G53 pins PA09 (PDMDATA), PA10 (PDMCLK) are routed on pin13, pin14 on EXT3 of SAM G53 Xplained Pro. Mount jumpers between JD and JE to connect MEMS data line on pin13, JB and JC to connect MEMS clock line on pin14 as shown in [Figure 5-1](#).

Figure 5-1. MEMS MIC Xplained Pro

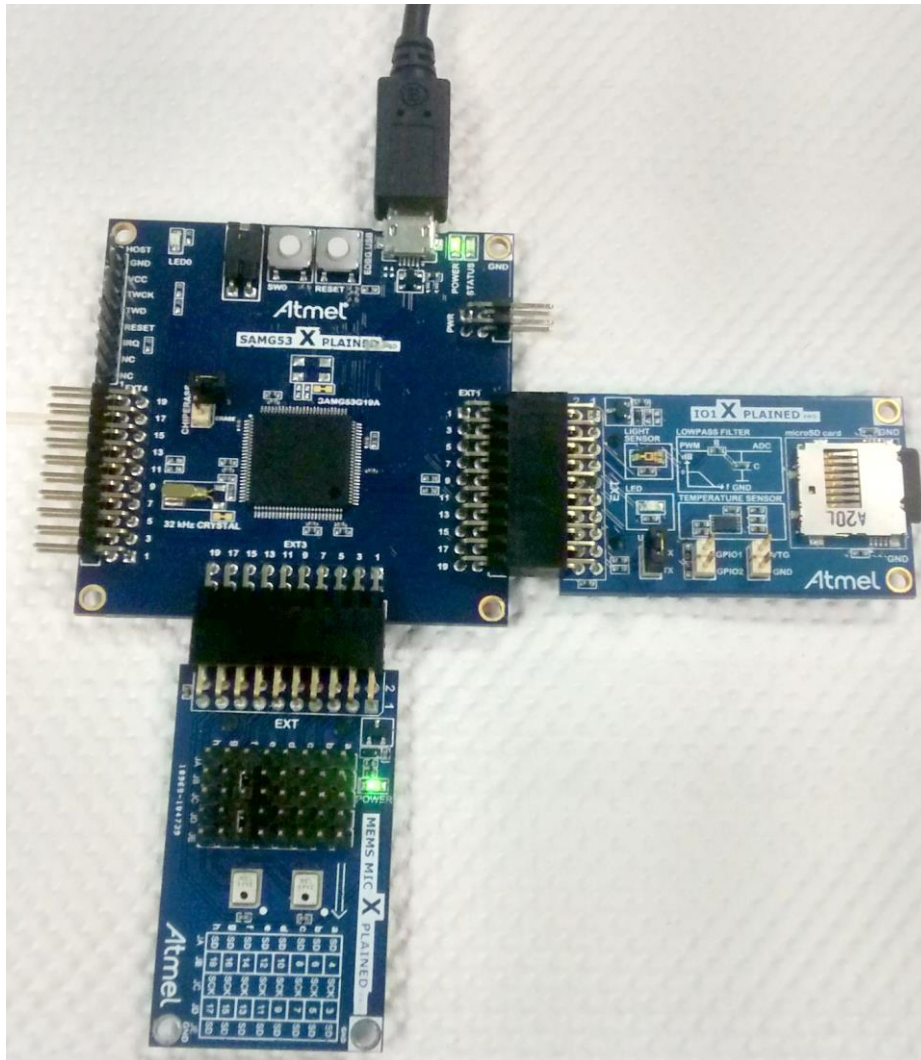


5.1.2 Connect Extension Boards to SAM G53

Connect pin1 of MEMS Mic Xplained Pro to pin1 in EXT3 G53 Xplained Pro and connect I/O1 Xplained Pro to EXT1 as shown in [Figure 5-2](#).

Plug the micro USB cable to SAM G53 EDBG USB and power the target. Ensure that SD card is inserted to I/O1 Xplained Pro.

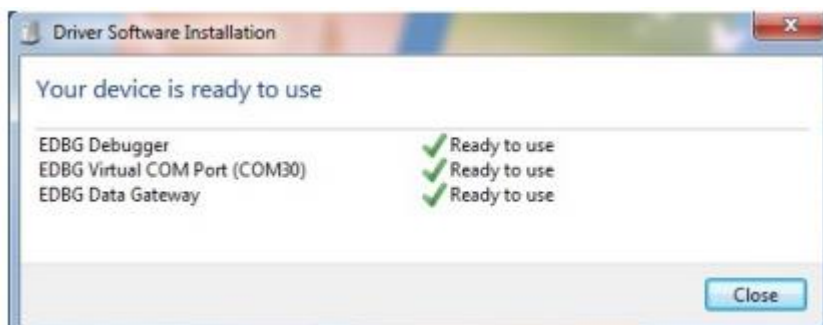
Figure 5-2. Hardware Setup



5.2 Software Interface

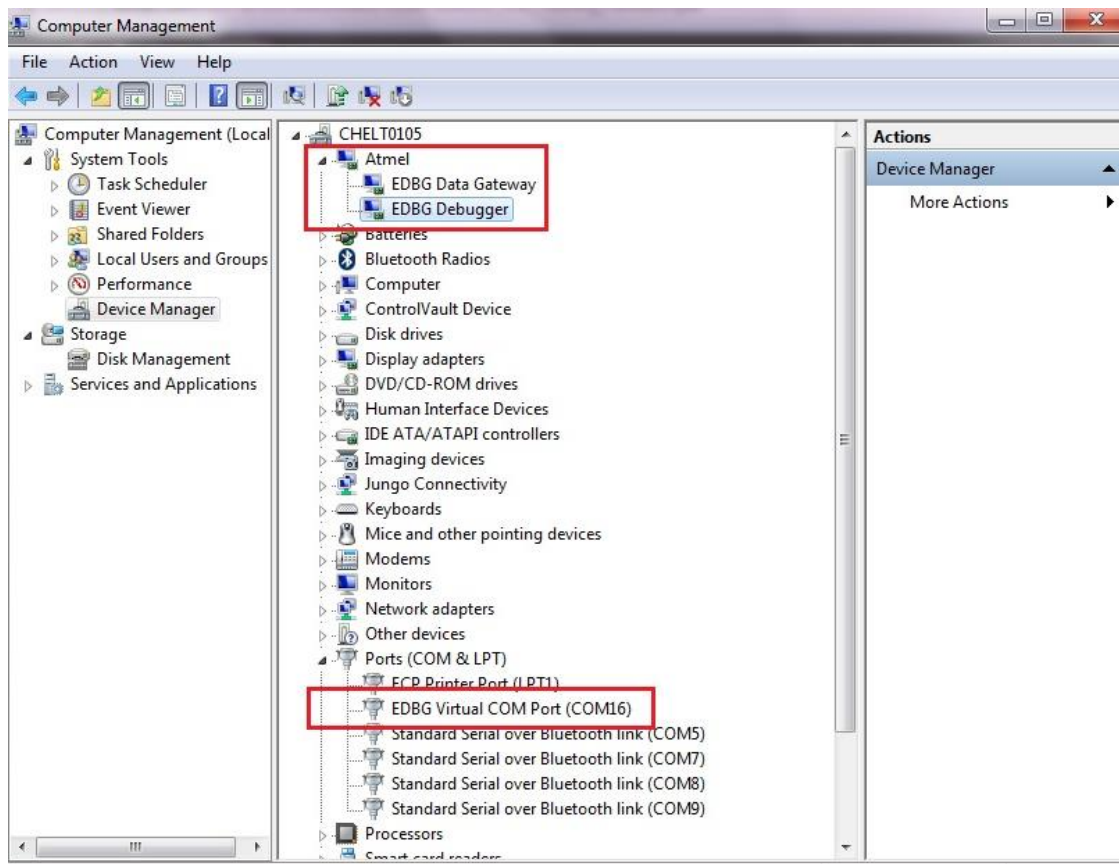
There is one USB port on the SAM G53 Xplained Pro board - EDBG USB. Once the SAM G53 Xplained Pro kit is connected to the PC, the Windows® task bar will pop-up a message as shown in [Figure 5-3](#).

Figure 5-3. SAM G53 Xplained Pro Driver Installation



If the driver installation is proper, EDBG will be listed in the Device Manager as shown in [Figure 5-4](#).

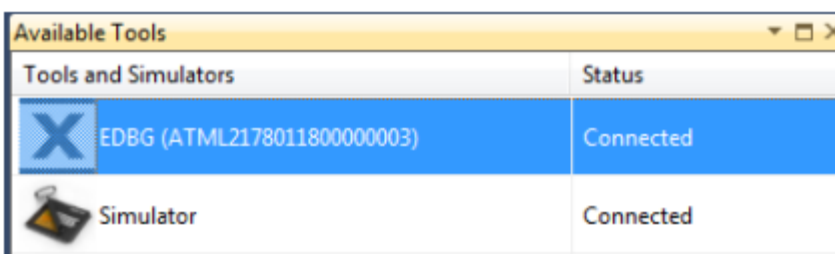
Figure 5-4. Successful EDBG Driver Installation



To ensure that the EDBG tool is getting detected in Atmel Studio:

Open Atmel Studio 6.2, go to 'View' → 'Available Atmel Tools'. The EDBG should get listed in the tools as "EDBG" and the tool status should display as "Connected" as shown in [Figure 5-5](#). This indicates that the tool is communicating properly with Atmel Studio.

Figure 5-5. EDBG Under Available Atmel Tools



If the tool does not get displayed in 'Available Atmel Tools', disconnect the tool and reconnect again. Right click on the tool in the 'Available Tools' list, click on "Upgrade". This will check whether the firmware in the tool is up to date. Click on "upgrade" to upgrade the firmware of the tool to latest version.

5.3 Programming the Application

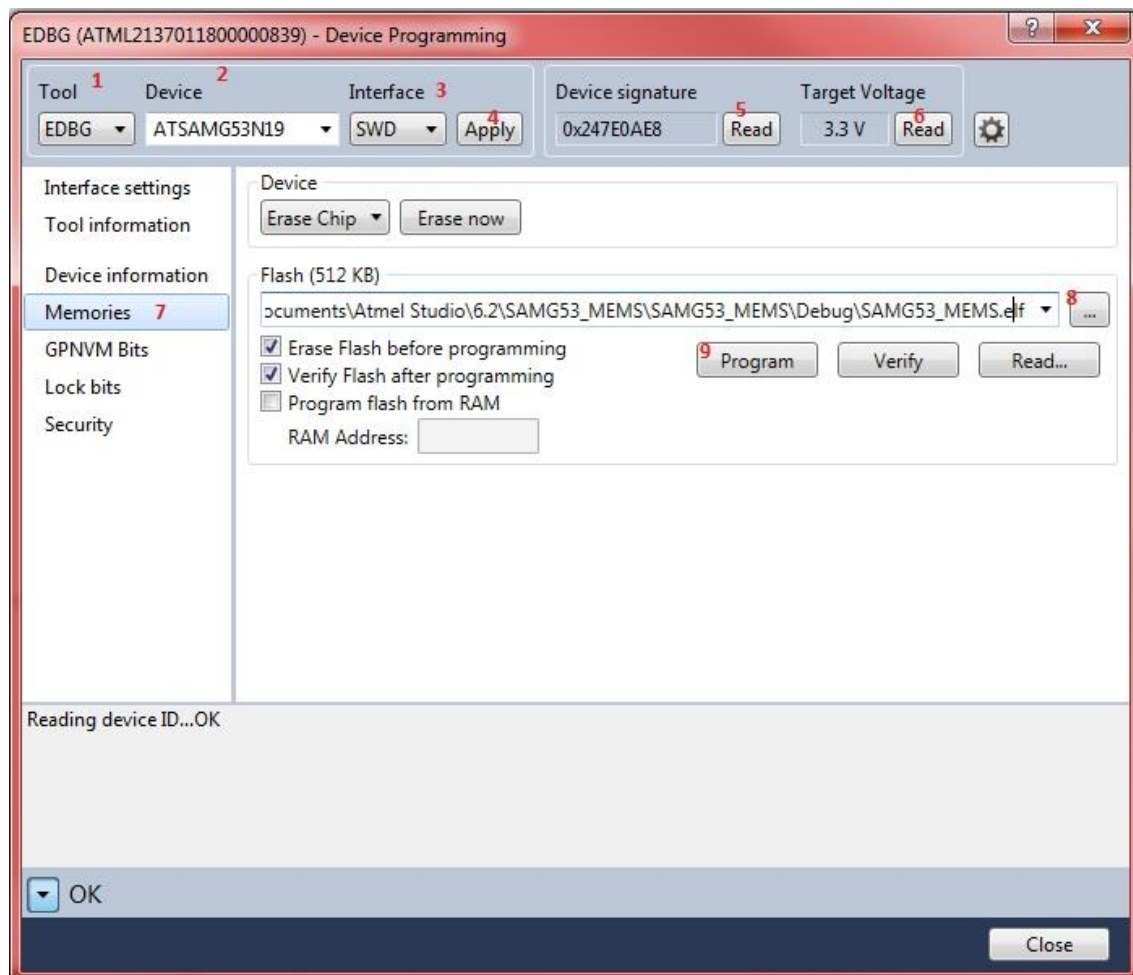
The firmware corresponding to this application comes as a zip file which will be available in web. The steps below explain the execution of this application.

1. Open the application which is available in the following path: `..\SAM_G53_MEMS\SAMG53_MEMS.atsln`.
2. Go to File → New → Example Project → Atmel_Application note.
3. Go to 'Build → Build Solution' to compile the project. The application occupies following resources when optimization is O1:

Program Memory Usage	:	18908 bytes	3.6 % Full
Data Memory Usage	:	16368 bytes	16.7 % Full

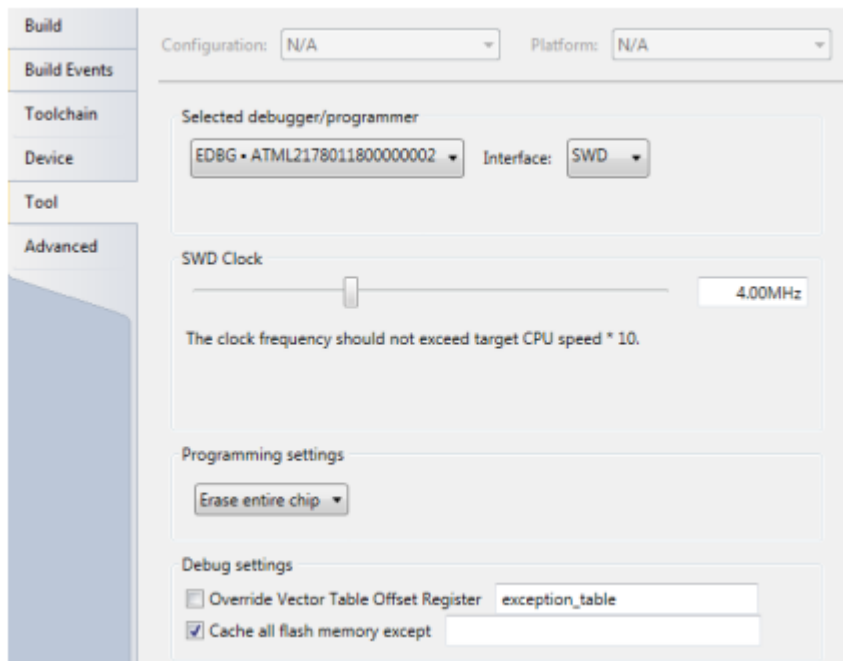
4. Once it is compiled successfully, go to Tools → Device Programming window.
5. Select appropriate tool, device and interface type and click Apply to connect to the kit. Check Device, Signature, and Target Voltage to ensure proper connection.
6. Now go to 'Memories' Tab. Browse the '*.hex/elf' file location and click program to flash the device as shown in [Figure 5-6](#).

Figure 5-6. Device Programming



7. To debug the code, right click on the project in the solution explorer window → Go to Project Properties.
8. Go to 'Tools' tab. Select EDBG as Debugger/programmer with 'SWD' Interface as in [Figure 5-7](#).

Figure 5-7. Debug Setting



5.4 Procedure

Once the application is programmed in SAM G53 Xplained Pro, press SW0 button in SAM G53 Xplained Pro and record the audio or music. The LED on SAM G53 Xplained Pro will glow which indicates audio is recording in SAM G53 Xplained Pro board. Once the recording is done, press SW0 again to stop recording.

6 Result

Now, The SD card in I/O1 Xplained Pro board has audio_0.wav file. The audio file can be played back in any media application such as PC (computer).

7 User Configuration

The application code is configured for Mono left channel mode as default. To configure the application for Mono right channel mode, modify the line in example.c.

```
#define TEST_MODE      TEST_MONO_RIGHT_CHANNEL
```

To configure the application for STEREO, modify the line in example.c to:

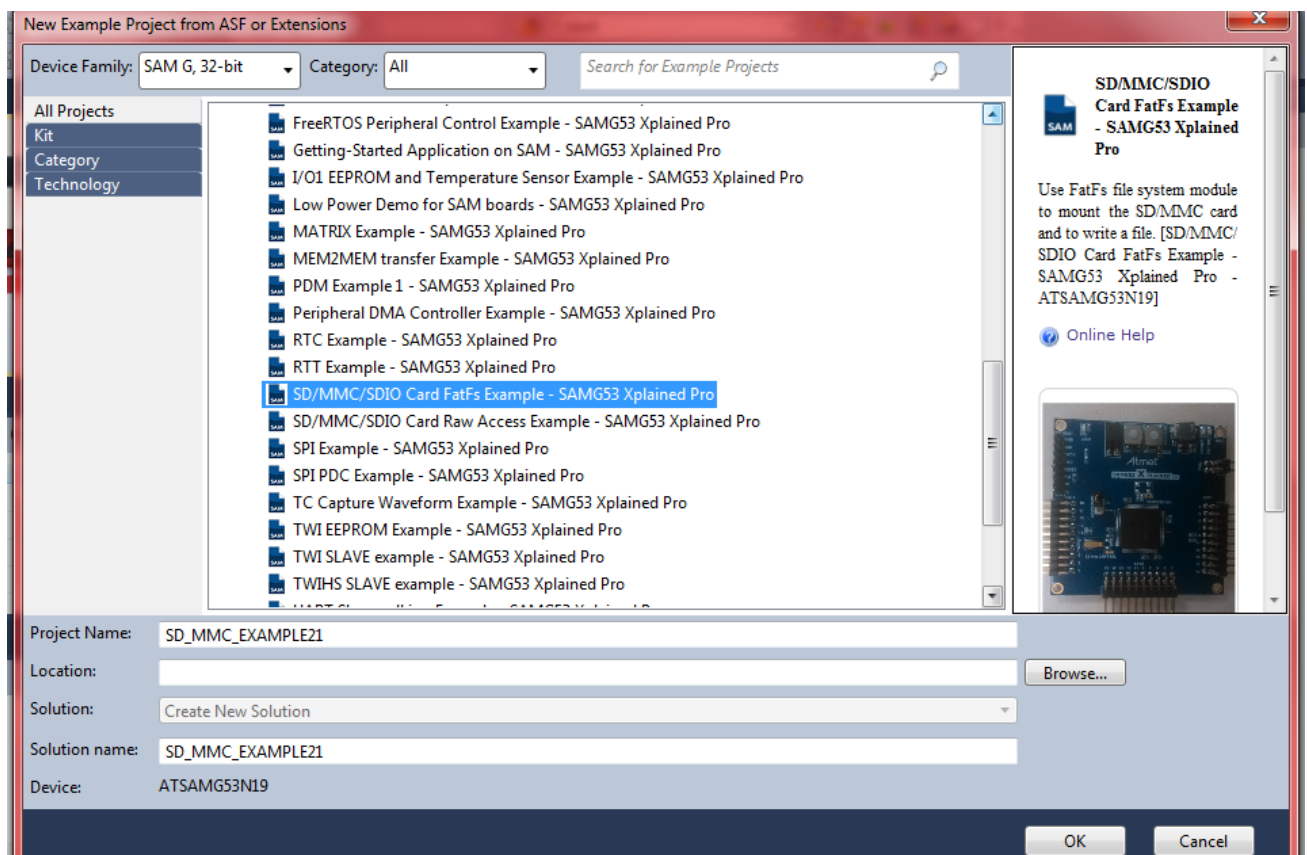
```
#define TEST_MODE      TEST_STEREO
```

7.1 Creating Project for Newer ASF Version

The firmware available with this application note is developed from ASF 3.20.1. The user can create a project in newer ASF version by following the below steps.

1. Launch Atmel Studio 6.2 or later.
2. Go to File → New → Example Project.
3. Filter the ASF project by selecting Device Family SAM G, 32-bit.
4. Select SD/MMC/SDIO Card FATFs Example – SAM G53 Xplained Pro as shown in Figure 7-1, rename the Project Name as SAMG3_MEMS and create the project.

Figure 7-1. New ASF Example Project



5. Go to ASF → ASF Wizard and add PDC (Peripheral DMA Controller Example (Driver)).
6. Replace example.c file with the example.c available with the .zip file.
7. Add `#define CONF_BOARD_PDM` in conf_board.h file and comment `#define CONF_BOARD_UART_CONSOLE`.

8. Modify `#define CONFIG_PLL0_MUL 1465` as `#define CONFIG_PLL0_MUL 1312` in order to operate the device at 42.99MHz.
9. Go to `spi_master.c` file in the following path `..\src\ASF\Common\services\spi\sam_spi\spi_master.c` and modify the function `spi_write_packet` as shown below in order to transfer SPI data via PDC.

```
#include "pdc.h"
Pdc *g_p_spi_pdc;
status_code_t spi_write_packet(Spi *p_spi, const uint8_t *data,
size_t len)
{
    uint32_t timeout = SPI_TIMEOUT;

    g_p_spi_pdc = spi_get_pdc_base(p_spi);
    pdc_packet_t pdc_next_packet;
    pdc_next_packet.ul_addr = (uint32_t)data;
    pdc_next_packet.ul_size = len;
    pdc_tx_init(g_p_spi_pdc, &pdc_next_packet, NULL);

    /* Enable the RX PDC transfer requests */
    pdc_enable_transfer(g_p_spi_pdc, PERIPH_PTCR_TXTEN);

    timeout = SPI_TIMEOUT * len;
    while(!(p_spi->SPI_SR & SPI_SR_TXBUFE)) {
        if (!timeout--) {
            return ERR_TIMEOUT;
        }
    }
    pdc_disable_transfer(g_p_spi_pdc, PERIPH_PTCR_TXTDIS);

    return STATUS_OK;
}
```

8 References

The following list contains links to the most relevant documents and software for MEMS MIC Xplained Pro:

1. [Xplained Pro products](#)²- Atmel Xplained Pro is a series of small-sized and easy-to-use evaluation kits for 8- and 32-bit Atmel microcontrollers. It consists of a series of low-cost MCU boards for evaluation and demonstration of features and capabilities of different MCU families.
2. [MEMS MIC Xplained Pro User Guide](#)³- Hardware User Guide in PDF.
3. [MEMS MIC Xplained Pro Design Documentation](#)⁴- Package containing schematics, BOM, assembly drawings, 3D plots, and layer plots, etc.
4. [Atmel Studio](#)¹ - Free Atmel IDE for development of C/C++ and assembler code for Atmel microcontrollers.
5. [WAVE format](#) - more information on WAVE PCM sound file format.

¹ <http://www.atmel.com/atmelstudio>

² <http://www.atmel.com/XplainedPro>

³ http://www.atmel.com/Images/Atmel-42427-MEMS-MIC-Xplained-Pro_User-Guide.pdf

⁴ http://www.atmel.com/Images/Atmel-42427-MEMS-MIC-Xplained-Pro_User-Guide.zip

9 Revision History

Doc Rev.	Date	Comments
B	06/2015	Table 4-1 has been corrected. Corrected documentation typos.
A	04/2015	Initial document release.



Atmel Corporation 1600 Technology Drive, San Jose, CA 95110 USA T: (+1)(408) 441.0311 F: (+1)(408) 436.4200 | www.atmel.com

© 2015 Atmel Corporation. / Rev.: Atmel-42423B-MCU-Audio-Recorder-with-SAM-G53-XPRO-and-MEMS-MIC-XPRO_Application-Note_062015.

Atmel®, Atmel logo and combinations thereof, Enabling Unlimited Possibilities®, and others are registered trademarks or trademarks of Atmel Corporation in U.S. and other countries. ARM®, ARM Connected® logo, and others are the registered trademarks or trademarks of ARM Ltd. Windows® is a registered trademark of Microsoft Corporation in U.S. and or other countries. Other terms and product names may be trademarks of others.

DISCLAIMER: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN THE ATMEL TERMS AND CONDITIONS OF SALES LOCATED ON THE ATMEL WEBSITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS AND PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and products descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

SAFETY-CRITICAL, MILITARY, AND AUTOMOTIVE APPLICATIONS DISCLAIMER: Atmel products are not designed for and will not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death ("Safety-Critical Applications") without an Atmel officer's specific written consent. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Atmel products are not designed nor intended for use in military or aerospace applications or environments unless specifically designated by Atmel as military-grade. Atmel products are not designed nor intended for use in automotive applications unless specifically designated by Atmel as automotive-grade.