Cerebot Voice Recorder Reference Design

Revision: December 13, 2011

www.digilentinc.com

1300 NE Henley Court, Suite 3
Pullman, WA 99163
(509) 334 6306 Voice | (509) 334 6300 Fax

Cerebot32MX4/7 Voice Recorder

Overview

This project demonstrates the use of the pmodlib C library in conjunction with the PmodDA2, PmodMIC and PmodAMP for the Cerebot32MX4 microcontroller. It also includes the use of the PmodSF for the Cerebot32MX7 microcontroller. Both of these function by storing sound recorded via the PmodMIC for playback through the PmodDA2 and PmodAMP.

Functional Description

Cerebot32MX4

The project built for the Cerebot32MX4 relies on storing the sound data directly into RAM and playing it back. As a result, the length of recording is very low. But it serves to show how the PmodDA2, PmodMIC, and PmodAMP can be used in conjunction.

Audio is captured by the 12-bit analog-to-digital controller on the PmodMIC and then sampled at the rate specified in the setup.h file (default is 12000 samples a second). This is stored into a number of buffers allocated during initialization of the project. The actual capture process relies on pressing and holding down BTN1 onboard the Cerebot32MX4. Once BTN1 is released the sampling will stop until pressed again.

Since the Cerebot32MX4 is limited to two SPI ports, only onboard memory is used to store audio data. Once the buffer allocated fills, then pressing and holding BTN1 will not add anymore audio data to the buffer. To empty the buffer, audio must be played but pressing and holding BTN2.

Audio playback is done by the 12-bit digital-to-analog controller on the PmodDA2. It then passes the analog signal to the PmodAMP for playback through either of two 1/8-inch audio jacks. The audio is sampled at the rate specified in the setup.h file (default is 12000 samples a second). The audio data is pulled from the buffers allocated during initialization. If the buffers are empty, no sound will be played.

Source code for the recorder is located in the directory titled "RecorderDemo_460". The project was created using MPLAB IDE v8.80 and written entirely in C utilizing the Microchip and pmodlib libraries. Processor configuration settings are located in setup.h.

Cerebot32MX7

Similar to the 32MX4 setup, the 32MX7 has the additional of the PmodSF for storage of audio data. All audio data captured by the PmodMIC is stored into the PmodSF sequentially. Playback through the

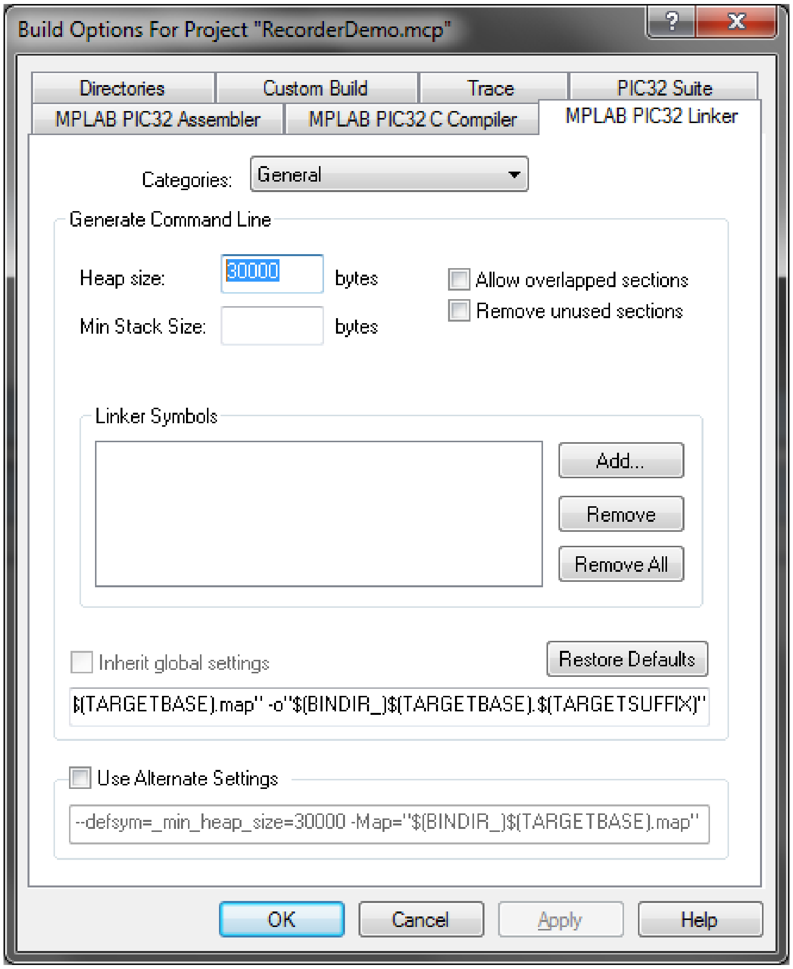
PmodDA2 and PmodAMP pulls audio data from the PmodSF up till the last address recorded to in the PmodSF's memory.

BTN3 is used to bulk erase the PmodSF's memory.

Source code for the recorder is located in the directory titled "RecorderDemo_795". The project was created using MPLAB IDE v8.80 and written entirely in C utilizing the Microchip and pmodlib libraries. Processor configuration settings are located in setup.h.

Configuration

Project – Standard PmodLib configuration applies. In addition, since we are utilizing the BufferLib library, we must allocated memory to the heap for dynamic allocation. To do so you must right click on your project and select **Build Options…** then on the **MPLAB PIC32 Linker** tab fill in **30000** into the **Heap Size:** field.



Build Options For Project "RecorderDemo.mcp"

Categories: General

Generate Command Line

Heap size: 30000 bytes

Min Stack Size: bytes

□ Allow overlapped sections

□ Remove unused sections

Linker Symbols

□ Inherit global settings

```
$(TARGETBASE).map" -o"$(BINDIR_)$(TARGETBASE).$(TARGETSUFFIX)"
```

□ Use Alternate Settings

```
--defsym=_min_heap_size=30000 -Map="$(BINDIR_)$(TARGETBASE).map"
```

Pmod IO/Ports - IO ports and channels have been defined at macros for easy configuration in cerebot_robot_remote.c under //IO PORT DEFINITIONS, changes in hardware location must be reflected in this section.

Port Bitrates:

Cerebot32MX4

SPI1	- PmodMIC:	12500000
SPI2	- PmodDA2:	20000000

Cerebot32MX7

SPI1	- PmodMIC:	12500000
SPI1A	- PmodSF:	625000
SPI3A	- PmodDA2:	20000000

Sample Rate:

Timer1

Setup to have a 12000 ticks/sec tick rate, Timer2 is used to output audio data to the PmodDA2 at a regular interval. The data is pulled from the buffers allocated during initialization. If no data is found in the buffer, nothing is output to the PmodDA2.

Timer2

Setup to have a 12000 ticks/sec tick rate, Timer1 is used to sample the PmodMIC for audio data at a regular interval and store it into the buffers allocated during initialization. If space is not available in the buffers, the pulled data is ignored.

Assembly:

The table below details the electrical connections between the different components of the recorder. All jumpers are in their default positions.

Cerebot 32MX4				
Header	Description	Connections		
J1	PmodMIC (SPI1)	Pin	Peripheral Header	Peripheral Pin
		SS	PmodMIC - J1	1
		SDO	PmodMIC - J1	2
		SDI	PmodMIC - J1	3
		SCK	PmodMIC - J1	4
		GND	PmodMIC - J1	5
		VCC	PmodMIC - J1	6

JB	PmodDA2 (SPI2)	Pin	Peripheral Header	Peripheral Pin
		1	PmodDA2 - J1	1
		2	PmodDA2 - J1	2
		3	PmodDA2 - J1	3
		4	PmodDA2 - J1	4
		5	PmodDA2 - J1	5
		6	PmodDA2 - J1	6

Cerebot 32MX7				
Header	Description	Connections		
JD	PmodMIC (SPI1)	Pin	Peripheral Header	Peripheral Pin
		1	PmodMIC - J1	1
		2	PmodMIC - J1	2
		3	PmodMIC - J1	3
		4	PmodMIC - J1	4
		5	PmodMIC - J1	5
		6	PmodMIC - J1	6
JE	PmodDA2 (SPI3A)	Pin	Peripheral Header	Peripheral Pin
		1	PmodDA2 - J1	1
		2	PmodDA2 - J1	2
		3	PmodDA2 - J1	3
		4	PmodDA2 - J1	4
		5	PmodDA2 - J1	5
		6	PmodDA2 - J1	6
JF	PmodSF (SPI1A)	Pin	Peripheral Header	Peripheral Pin
		1	PmodSF - J1	1
		2	PmodSF - J1	2
		3	PmodSF - J1	3
		4	PmodSF - J1	4
		5	PmodSF - J1	5
		6	PmodSF - J1	6

PmodDA2				
Header	Description	Connections		
J2	PmodDA2 Analog Output	Pin	Peripheral Header	Peripheral Pin
		1	PmodAMP - J1	1
		2	PmodAMP - J1	2
		3	PmodAMP - J1	3
		4	PmodAMP - J1	4
		5	PmodAMP - J1	5
		6	PmodAMP - J1	6

Parts List

Digilent
(1) Cerebot 32MX4 or Cerebot 32MX7
(1) PmodDA2
(1) PmodMIC
(1) PmodSF (32MX7 only)
(1) PmodAMP
(1) 6 Pin Cable Connector
(1) 6 Pin Cable