



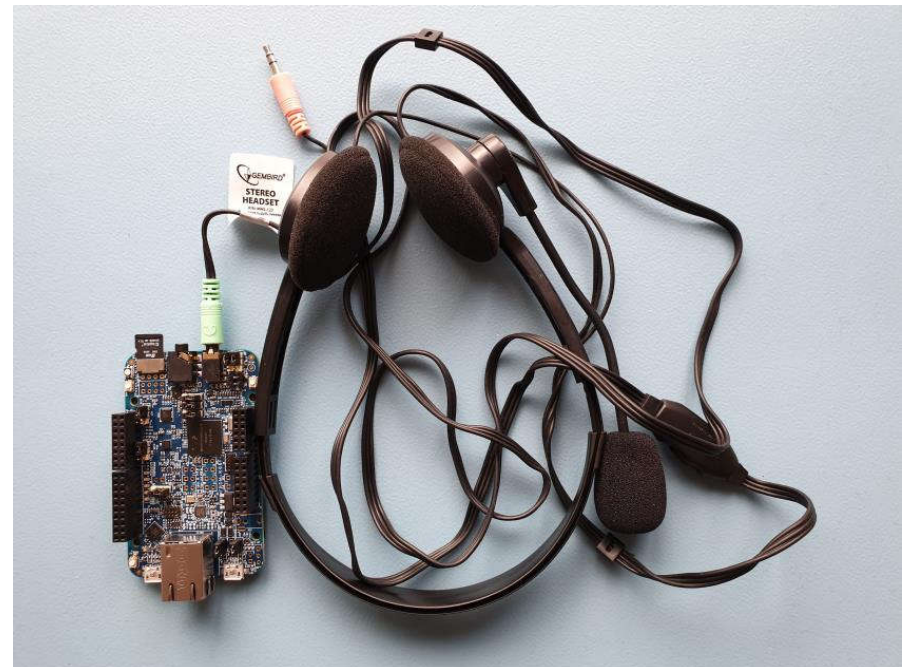
MASARYK
UNIVERSITY
Czech Republic

PV198 – One-chip Controllers

Music Player

Content

1. Audio Codec
2. I2S
3. WAV file format
4. Application



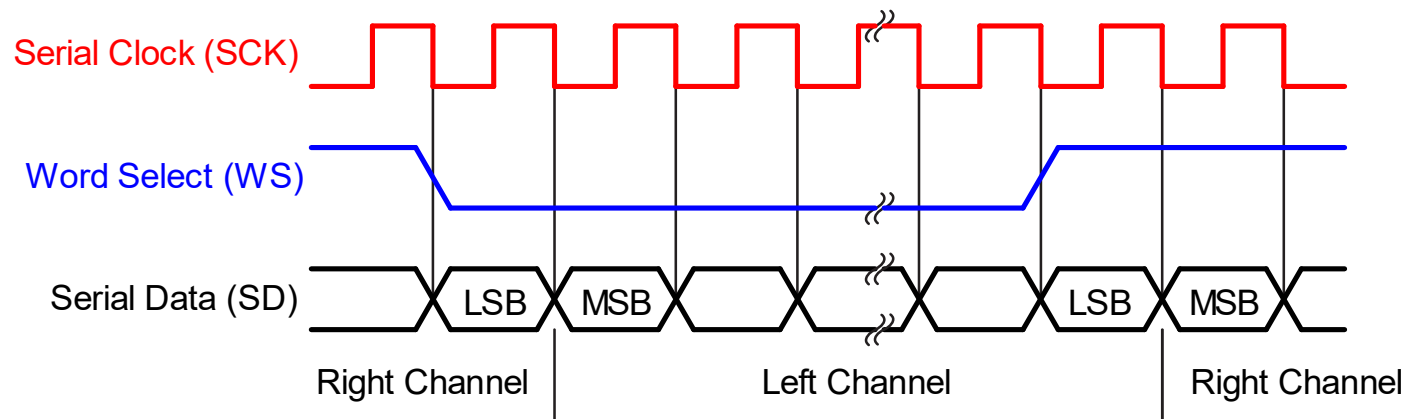
Audio Codec

- Dialog Semiconductor – DA7212 – [link](#)
- 24-bit audio
- Low-power
- Beep generator – 2 Sine wave generators
- Bidirectional I2S

I2S Overview

- I2S – Inter-IC Sound – [link to specification](#), [Application Note](#)
- Serial bus for digital audio data in PCM format
- 3 signals:
 - Serial clock (SCK)
 - Word select (WS) / Frame sync (FS) – left/right channel select – toggled after each sample
 - Serial data (SD) – the data samples

I2S Overview



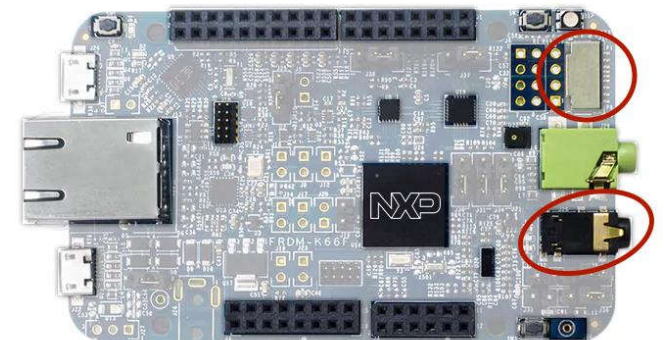
By wdwd - Own work, CC BY 3.0, <https://commons.wikimedia.org/w/index.php?curid=16579640>

WAV file format

- <http://soundfile.sapp.org/doc/WaveFormat/>

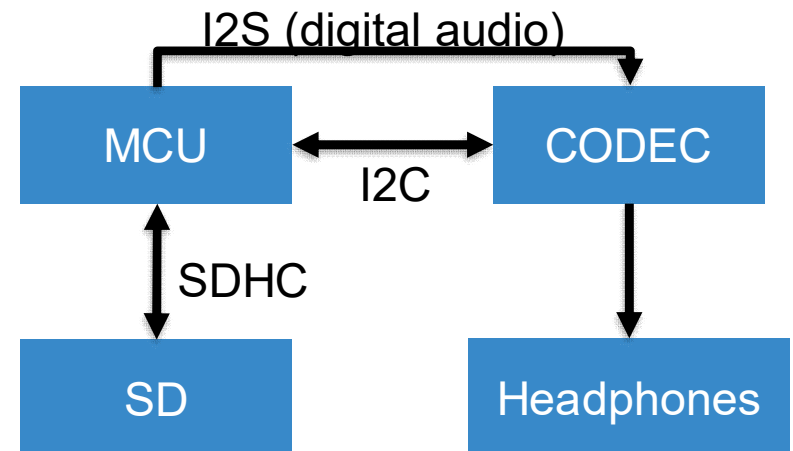
Application – Overview

- The application reads .wav files from the root folder of Micro SD card and plays them via 3.5mm audio jack.
- Input:
 - .wav file(s) in the root folder of Micro SD card
- Output:
 - Analog sound via 3.5mm audio jack
- Limitations:
 - Micro SD card must use FAT file system
 - PCM format of .wav files is supported only



Application

- Peripherals used:
 - I2S – for audio data transfer to audio codec
 - SDHC – for SD card operations, e.g., .wav files read operation
 - I2C – for audio codec configuration, e.g., sound bit-width, etc.
 - PIT – for timing of tone generation
- Schematics:



Application – Pins configuration

SD card

#	Peripheral	Signal	Route to	Label	Identifier	Direction	GPIO initial state	GPIO interrupt	Slew rate	Open drain	Drive strength	Pull select	Pull enable	Passive filter	Digital filter
E4	SDHC	CMD	SDHC0_CMD	SDHC0_CMD	SDHC0_CMD	Input/Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
D2	SDHC	DATA, 0	SDHC0_D0	SDHC0_D0	SDHC0_D0	Input/Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
D3	SDHC	DATA, 1	SDHC0_D1	SDHC0_D1	SDHC0_D1	Input/Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
E2	SDHC	DATA, 2	SDHC0_D2	SDHC0_D2	SDHC0_D2	Input/Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
E3	SDHC	DATA, 3	SDHC0_D3	SDHC0_D3	SDHC0_D3	Input/Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
D1	SDHC	DCLK	SDHC0_DCLK	SDHC0_DCLK	SDHC0_DCLK	Output	n/a	n/a	Fast	Disabled	Low	Pullup	Enabled	Disabled	n/a
B3	GPIO	GPIO, 10	PTD10	SD_CARD_DETECT	SDCARD_CARD_DETECTION	Input	n/a	Interrupt/...	Fast	Disabled	Low	Pullup	Enabled	Disabled	Disabled

Console

#	Peripheral	Signal	Route to	Label	Identifier	Direction	GPIO initial state	GPIO interrupt	Slew rate	Open drain	Drive strength	Pull select	Pull enable	Passive filter	Digital filter
E10	UART0	RX	UART0_RX	U7[4]/UART0_RX	DEBUG_UART_RX	Input	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled	n/a
E9	UART0	TX	UART0_TX	U10[1]/UART0_TX	DEBUG_UART_TX	Output	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled	n/a

CODEC

#	Peripheral	Signal	Route to	Label	Identifier	Direction	GPIO initial state	GPIO interrupt	Slew rate	Open drain	Drive strength	Pull select	Pull enable	Passive filter
C5	I2S0	MCLK	I2S0_MCLK	J1[9]/LLWU_P10/FB_AD9/I2S0_MCLK	Not Specified	Not Specified	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled
B3	I2S0	TX_BCLK	I2S0_TX_BCLK	J1[1]/J37[3]/I2S_TX_BCLK	I2S_TX_BCLK	Not Specified	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled
B4	I2S0	TX_FS	I2S0_TX_FS	J1[3]/J38[3]/I2S_TX_WCLK	I2S_TX_WCLK	Not Specified	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled
B11	I2S0	TXD0	I2S0_TXD0	J1[5]/U20[C5]/I2S_TXD	I2S_TXD	Output	n/a	n/a	Fast	Disabled	Low	Pulldown	Disabled	Disabled
C7	I2C1	SCL	I2C1_SCL	J2[20]/U20[D8]/I2C1_SCL/DA7212_CLK	I2C1_SCL	Input/Output	n/a	n/a	Fast	Enabled	Low	Pullup	Enabled	Disabled
B7	I2C1	SDA	I2C1_SDA	J2[18]/U20[C9]/I2C1_SDA/DA7212_SDA	I2C1_SDA	Input/Output	n/a	n/a	Fast	Enabled	Low	Pullup	Enabled	Disabled

Application – Peripherals configuration

- **SDHC** – board example reused
- **I2S** – configured via Peripherals tool however the peripheral properties (channels number, sample rate, data width, and interrupts) are configured in the application based on the .wav files' properties
- **PIT** – configured via Peripherals tool

Application – Source Code

- Main loop:
 - Configure the devices
 - Start PIT to play tones sequence via I2C
 - Start file reading, repeat the following sequence:
 - ☒ Fill the data read buffers
 - ☒ Wait until a buffer becomes empty after it is transmitted via I2S
- Interrupts:
 - Used board demo I2C communication interrupts
 - PIT uses interrupts for tones playing
 - I2S uses interrupts for FIFO status notifications

Application

- Parse WAV file header and setup SAI and Codec format correctly by implementing function:
 - `uint8_t processWavHeader(FILE* file)`

Homework

- Update the project such: when all songs have been played, it will start playing all again on repeat