

VS1011 APPNOTE: PLAYBACK

Microcontroller example code

All information in this document is provided as-is without warranty. Features are subject to change without notice.

Revision History			
Rev.	Date	Author	Description
1.10	2016-05-09	HH	Changed quick sanity check registers.
1.00	2012-11-28	HH	Initial version.

Contents

VS1011 AppNote: Playback Front Page	1
Table of Contents	2
1 Playback on VS1011	3
1.1 vs10xx_uc.h	3
1.2 player.h	4
1.3 player1011.c	5
1.4 Main Program	5
2 Latest Version Changes	6
3 Contact Information	7

List of Figures

1 Playback on VS1011

The VS1011 Datasheet tells how to play back files using VS1011 as a slave processor. This AppNote and software package provides the same information as generic microcontroller C code.

Note that this code is not written for any particular microcontroller. It has been written in standard C, and the few parts having to do with SPI bus communication that are environment-specific, have not been included.

To create your own program, read the included source code, make the modifications and additions suggested in this document, compile with your favourite microcontroller compiler, and run.

The rest of this Chapter introduces the files in this package.

1.1 vs10xx_uc.h

Contains symbols for VS10xx registers and register bits.

1.2 player.h

Definitions and prototypes for the player. The following functions are declared:

int VSTestInitHardware(void)

Initializes microcontroller for VS10xx operations. You need to add the microcontroller-specific code to this file in `player1011.c`.

Returns 0 on success, non-zero on failure.

int VSTestInitSoftware(void)

Makes a software reset for VS1011 and initializes it for use, including loading the patches package.

Returns 0 on success, non-zero on failure.

int VSTestHandleFile(const char *fileName, int record)

Plays back a given file or records to it, depending on whether `record` is 0 or 1. (Note: VS1011 has no recording support.)

u_int16 ReadSci(u_int8 addr)
void WriteSci(u_int8 addr, u_int16 data),
int WriteSdi(const u_int8 *data, u_int8 bytes)

You need to provide these functions that read and write from the VS1011's Serial Control and Data SPI Interfaces. For details on how to implement these functions, see Application Note *Connecting VS10xx SPI Buses*, available at <http://www.vlsi.fi/en/support/applicationnotes.html>

void SaveUIState(void)
void RestoreUIState(void)
int GetUICommand(void)

You need to provide these functions if you want your player to have a simple demonstration user interface.

`SaveUIState()` saves the user interface environment before execution, and `RestoreUIState()` restores it. In many cases these may be implemented as empty functions.

GetUICommand() should return -1 if there are no messages and -2 if forceful cancel is requested. By returning 63 (ASCII code for '?'), you get the list of other options from the player. See `player1011.c` for details.

1.3 `player1011.c`

Contains the implementation for VS1011 playback.

See the source code for details.

1.4 Main Program

Your main program, which is not included in this package, should look something like this:

```
int main(...) {
    if (VSTestInitHardware() || VSTestInitSoftware()) {
        printf("Failed initializing VS10xx, exiting\n");
        exit(EXIT_FAILURE);
    }

    /* ... */

    /* Playback example. You can call these functions many times in a row
       because they leave VS1011 in a stable state. */
    VSTestHandleFile("MyPlayFile.mp3", 0);

    /* ... */
}
```

2 Latest Version Changes

Version 1.10, 2016-05-09

The following source code changes have been made:

- Changed quick sanity check registers in this package from registers SCI_HDAT0 and SCI_HDAT1 to SCI_AICTRL1 and SCI_AICTRL2.
- Added chip type recognition to this package. Now code will refuse to run if the chip is not a VS1011.

Version 1.00, 2012-11-28

Initial version.

3 Contact Information

VLSI Solution Oy
Entrance G, 2nd floor
Hermiankatu 8
FI-33720 Tampere
FINLAND

URL: <http://www.vlsi.fi/>
Phone: +358-50-462-3200
Commercial e-mail: sales@vlsi.fi

For technical support or suggestions regarding this document, please participate at
<http://www.vsdsp-forum.com/>

For confidential technical discussions, contact
support@vlsi.fi