Sample DSP Services Applications for the TMS320C6XXX DSK HPI Daughtercard (Board Revision 1.3a, Software Version 1.0.1.0)

The sample applications are contained in a single ZIP file. There are five (5) Code Composer Studio (CCS) projects, Digital_Input_Test, Digital_Output_Test, Digital_Voltmeter, Digital_VU_Meter, and DTMF, as described below. The ZIP archive should be extracted using folders to preserve the directory structure.

CCS or the winDSK6 program loader can be used to load the programs onto the DSK. If using CCS, the daughtercard must be reset after the DSP is reset by CCS to ensure the DSP's HPI port is a known state.

Digital_Input_Test

This project demonstrates the use of the digital inputs and the serial communication port. The state of all 16 digital inputs is repeatedly written out to the serial port. If left unconnected, the values will read 1 due to the pull-up resistors on the input pins.

Digital_Output_Test

This project demonstrates the use of the digital outputs. A single 1 is walked repeatedly across the 16 digital outputs, updating at a 1 kHz rate.

Digital_Voltmeter

This project demonstrates the use of the analog inputs and the serial communication port. The voltages read at all 8 analog inputs are repeatedly written to the serial port. The analog inputs have a range of 0-2.5V. Analog inputs 0 and 1 are also used to control the gain of the left and right codec line input channels. By connecting a variable voltage to analog inputs 0 or 1, the audio gain from line inputs to line/headphone output can be varied.

Digital_VU_Meter

This project demonstrates the use of the digital outputs and the codec line input. The digital outputs are used to display a bar graph of the audio input level, similar to a VU meter. High-efficiency LEDs with suitable resistors to limit the current to approximately 2mA can be connected between the digital outputs and ground to provide a visual indicator.

DTMF

This project demonstrates the use of serial communications. The serial port is used to receive characters that signal the application to produce a DTMF tone. The application accepts the characters 0-9, # and * as valid DTMF signals, and echoes them back to the terminal. Any other character is ignored, and an X is sent to the terminal.