Experiment 12

December 8, 2023

Introduction to DSP processors

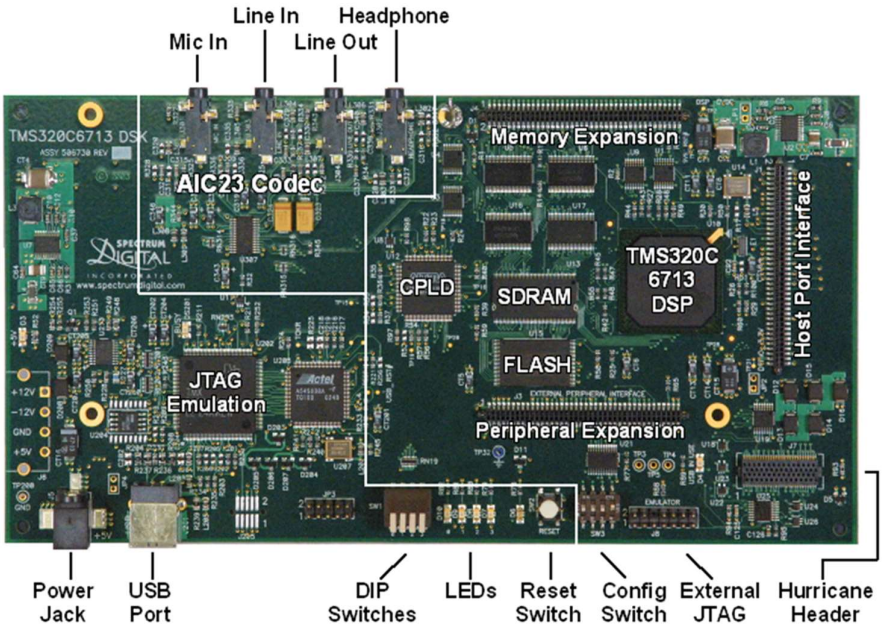
DATE: 27/11/2023

AIM: Familiarization of DSP Hardware and Software

THEORY:

TMS320C6713 DSP Starter Kit (DSK)

The C6713 DSK is a board that enables users to develop real-time DSP applications. The heart of the DSK is the Texas Instruments TMS320C6713 32-bit floating point Digital Signal Processor. DSP's differ from ordinary microprocessors in that they are specifically designed to rapidly perform the sum of products operation required in many discrete time signal processing algorithms. They contain hardware parallel multipliers, and functions implemented by microcode in ordinary microprocessors are implemented by high speed hardware in DSP's. Compared to fixed point processors, floating point processors are easier to program since issues like underflow, overflow, dynamic range etc can be ignored. The board is programmed using the TI Code Composer Studio (CCS) software, which connects to the board through a USB.



The DSK comes with a full complement of on-board devices that suit a wide variety of application environments. Key features include:

- A Texas Instruments TMS320C6713 DSP operating at 225 MHz.
- An AIC23 stereo codec

- 16 Mbytes of synchronous DRAM
- 512 Kbytes of non-volatile Flash memory (256 Kbytes usable in default configuration)
- 4 user accessible LEDs and DIP switches
- Software board configuration through registers implemented in CPLD
- Configurable boot options
- Standard expansion connectors for daughter card use
- JTAG emulation through on board JTAG emulator with USB host interface or external emulator
- Single voltage power supply (+5V)

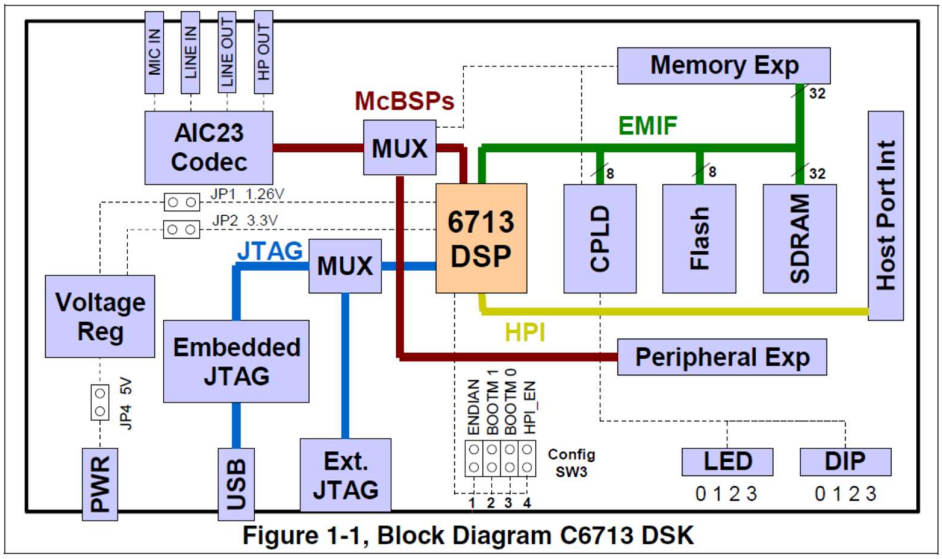


Figure 1-1, Block Diagram C6713 DSK

Functional Overview of the TMS320C6713 DSK

The DSP on the 6713 DSK interfaces to on board peripherals through a 32 bit wide EMIF (External Memory InterFace). The SDRAM, Flash and CPLD are all connected to the bus. EMIF signals are also connected daughter card expansion connectors which are used for third party add-in boards.

The DSP interfaces to analog audio signals through an onboard AIC23 codec and four 3.5 mm audio jacks (microphone input, line input, line output, and headphone output). The codec can select the microphone or the line input as the active input. The analog output is driven to both the line out (fixed gain) and headphone (adjustable gain) connectors. Multichannel Buffered Serial Port 0 (McBSP0) is used to send commands to the codec control interface while McBSP1 is used for digital audio data. McBSP0 and McBSP1 can be re-routed to the expansion connectors in software.

A programmable logic device called a CPLD is used to implement glue logic that ties the board components together. The CPLD has a register-based user interface that lets the user configure the board by reading and writing to its registers.

The DSK includes 4 LEDs and a 4 position DIP switch as a simple way to provide the user with interactive feedback. Both are accessed by reading and writing to the CPLD registers.

A 5V external power supply is used to power the board. On-board switching voltage regulators provide the +1.26V DSP core voltage and +3.3V I/O supplies. The board is held in reset until these supplies are within operating specifications.

Code Composer communicates with the DSK through an embedded JTAG emulator with a USB host interface. The DSK can also be used with an external emulator through the external JTAG connector.

Memory Map

The C67xx family of DSPs has a large byte addressable address space. Program code and data can be placed anywhere in the unified address space. Addresses are always 32-bits wide. The memory map shows the address space of a generic 6713 processor on the left with specific details of how each region is used on the right. By default, the internal memory sits at the beginning of the address space. Portions of the internal memory can be reconfigured in software as L2 cache rather than fixed RAM. The EMIF has 4 separate addressable regions called chip enable spaces (CE0-CE3). The SDRAM occupies CE0 while the Flash and CPLD share CE1. CE2 and CE3 are generally reserved for daughtercards.

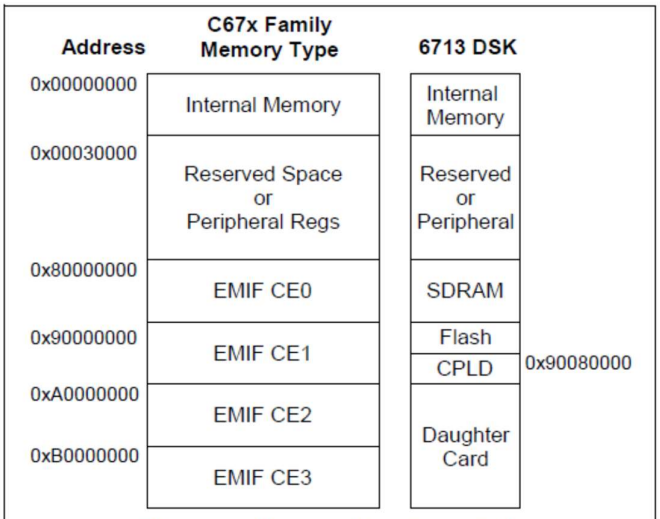


Figure 1-2, Memory Map, C6713 DSK

AIC23 Codec

The DSK uses a Texas Instruments AIC23 (part #TLV320AIC23) stereo codec for input and output of audio signals. The codec samples analog signals on the microphone or line inputs and converts them into digital data so it can be processed by the DSP. When the DSP is finished with the data it uses the codec to convert the samples back into analog signals on the line and headphone outputs so the user can hear the output.

The codec communicates using two serial channels, one to control the codec's internal configuration registers and one to send and receive digital audio samples. McBSP0 is used as the unidirectional control channel. It should be programmed to send a 16-bit control word to the AIC23 in SPI format. The top 7 bits of the control word should specify the register to be modified and the lower 9 should contain the register value. The control channel is only used when configuring the codec, it is generally idle when audio data is being transmitted, McBSP1 is used as the bi-directional data channel. All audio data flows through the data channel. Many data formats are supported based on the three variables of sample width, clock signal source

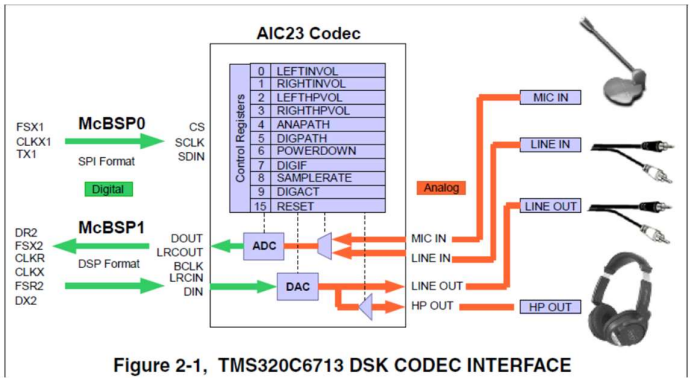


Figure 2-1, TMS320C6713 DSK CODEC INTERFACE

and serial data format.

The codec has a 12MHz system clock. The 12MHz system clock corresponds to USB sample rate mode, named because many USB systems use a 12MHz clock and can use the same clock for both the codec and USB controller. The AIC23 can divide down the 12 MHz clock frequency to provide sampling rates of 8, 16, 24, 32, 44.1, 48 and 96 KHz.

Software

Texas Instruments' Code Composer Studio (CCS) Integrated Development Environment (IDE) incorporates a C compiler, an assembler, and a linker. It is a development tool that allows users to create, edit and build programs, load them into the processor memory and monitor program execution. CCS communicates with the DSK via a USB connection. It supports real - time debugging has graphical capabilities. CCS is based on Eclipse, which is a Linux based open source software. CCSv7 and later does not require a paid license. The latest version is v11. But it does not support the debug probe on the 6713DSK. Older versions do not run on Windows 10. We will be using version 7.

A special Board Support Library (BSL) *dsk6713bsl32.lib* is supplied with the TMS320C6713 DSK. The BSL provides C-language functions for configuring and controlling all the on-board devices. The library includes modules for general board initialization, access to the AIC23 codec, reading the DIP switches, controlling the LED's, and programming and erasing the Flash memory. TI also provides a Chip Support Library (CSL) *csl6713.lib* that contains C functions and macros for configuring and interfacing with all the 'C6713 on-chip peripherals. The pre-compiled board support and chip support libraries are provided to you in the *dsplab* folder. The C source code for BSL functions are also provided. The folder also contains the required header files for using BSL and CSL functions.

On power on, a power on self - test (POST) program, stored by default in the onboard flash memory, uses routines from the board support library (BSL) to test the DSK. It tests the internal, external, and flash memory, the two multichannel buffered serial ports (McBSP), DMA, the onboard codec, and the LEDs. If all tests are successful, all four LEDs blink three times and stop (with all LEDs on). During the testing of the codec, a 1kHz tone is generated for 1 second.

CCS Project

A very readable and useful user guide for CCS is available online at

https://software-dl.ti.com/ccs/esd/documents/users_guide/index.html

All work in CCS is based on projects, which are typically a collection of files and folders required for an application to be run on the DSK. Project folders are stored and organized in workspace folder. A workspace is the main working folder for CCS. When CCS is launched, it will prompt for the workspace folder location.

A Code Composer Studio project comprises all of the files (or links to all of the files) required in order to generate an executable file. A variety of options enabling files of different types to be added to or removed from a project are provided. In addition, a Code Composer Studio project contains information about exactly how files are to be used in order to generate an executable file. Compiler/linker options can be specified.

To create a new CCS project, follow the steps below:

Go to menu *Project → New CCS Project…* or *File → New → CCS Project*.

In the New CCS Project wizard:

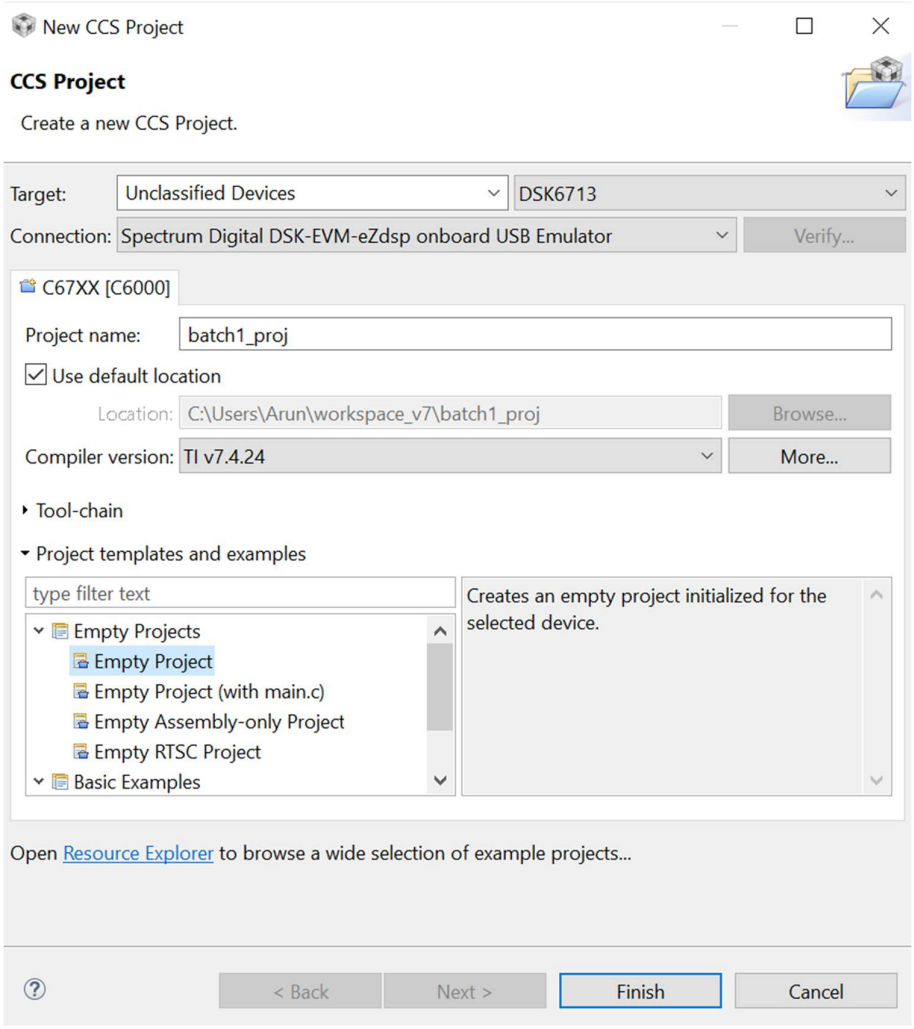
Type or select the *Target device*: Select *Unclassified Devices* and *DSK6713*

Connection: *Spectrum Digital DSK-EVM-eZdsp onboard USB Emulator*

This automatically creates a Target Configuration File *DSK6713.ccxml*. The Target Configuration File is a plain text XML file, with a .ccxml extension, that contains all the necessary information for a debug session: the type of Debug Probe, the target board or device , and (optionally) a path to a GEL (General Extension Language) script, which is responsible for performing device and/or hardware initialization.

Project Name: Give your project a name, such as *batch1_proj*. The default location will be a folder with the name of your project within the *workspace_v7* folder

In project templates, select Empty project

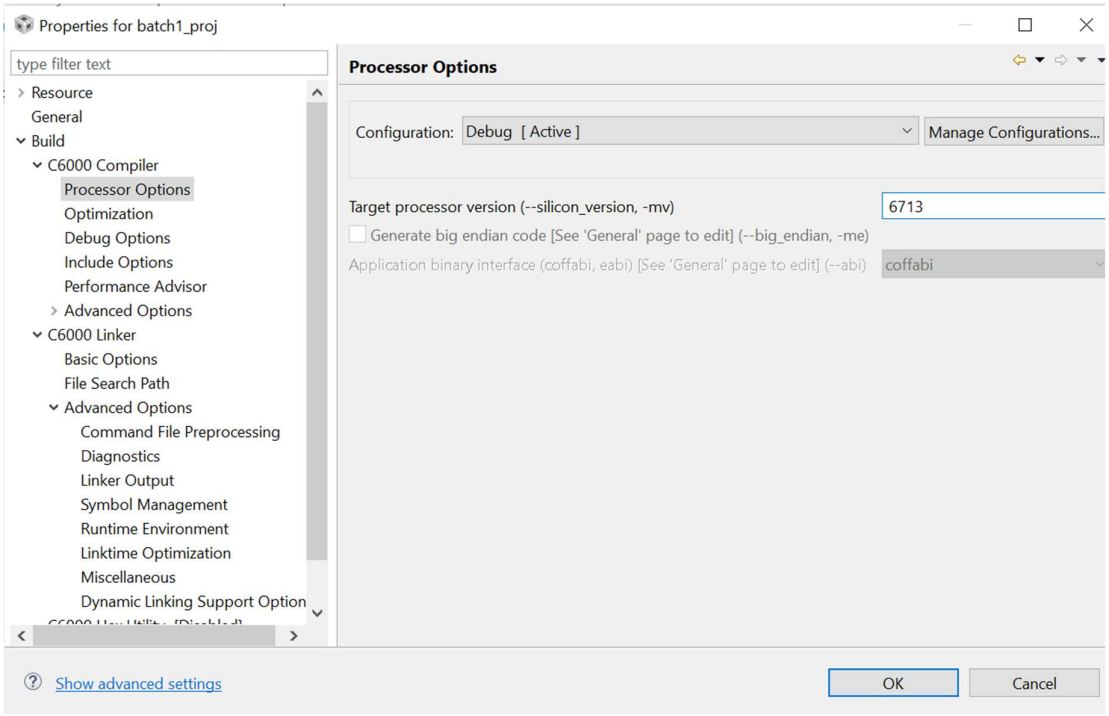


Click Finish

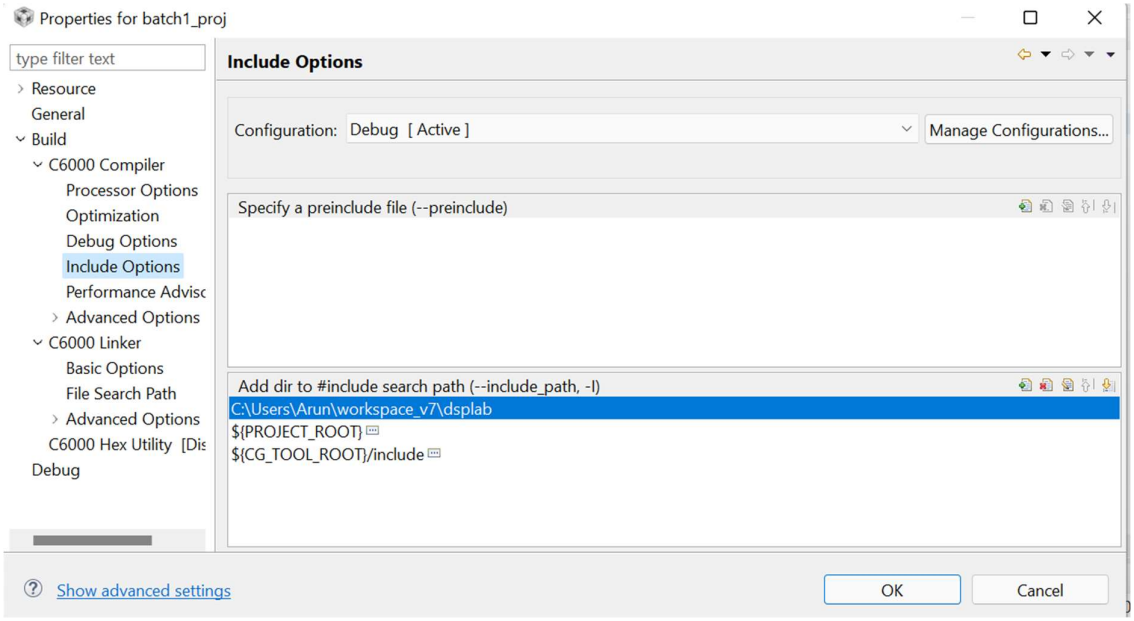
Your project should now show in the Project Explorer window.

First, we need to set some properties for our project. Right-click on the project and go to Properties (Or from menu *Project > Properties*)

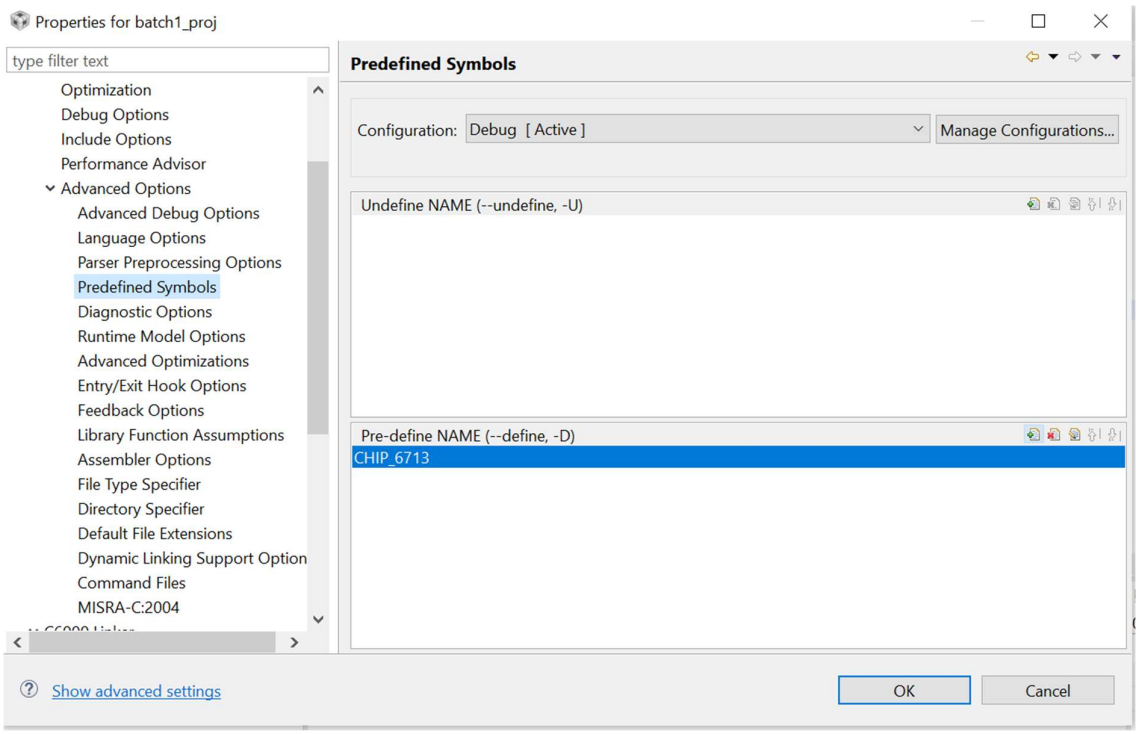
Under *Build>C6000 Compiler>Processor Options*, set *Target processor version* as 6713



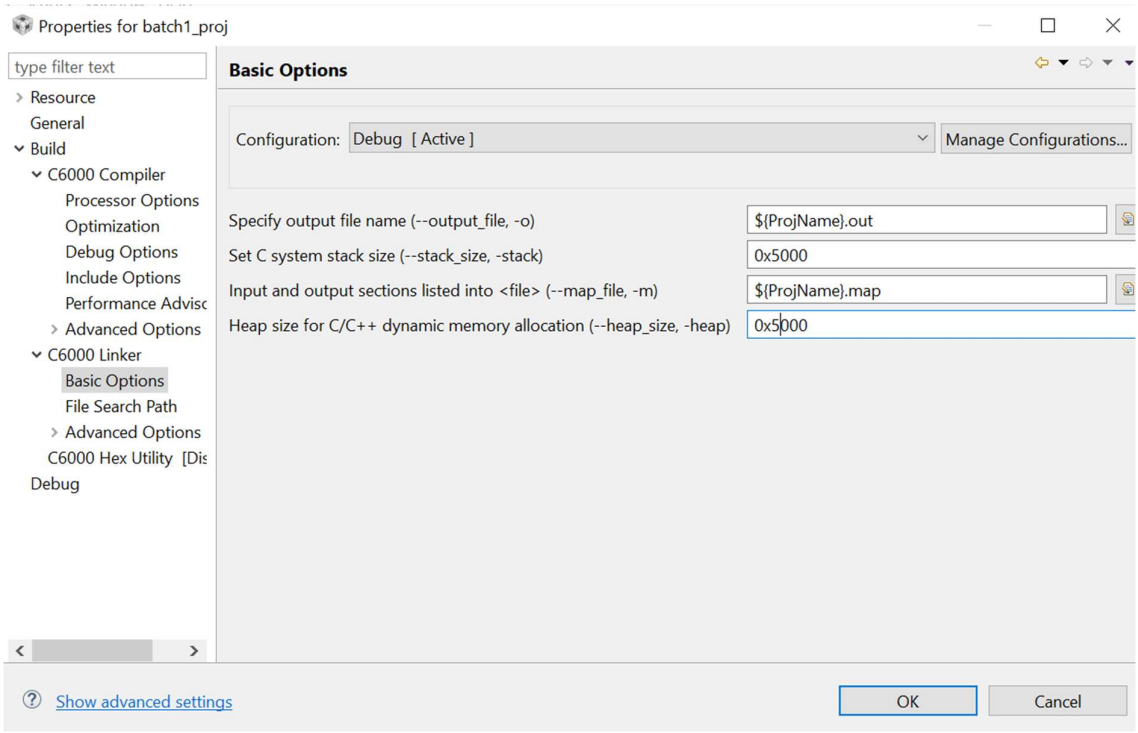
Under *Include Options*, go *to Add dir to #include search path*, click on the file icon with a green + mark and Browse to the folder *dsplab* provided to you and click OK. This folder contains the header files for the board support library and chip support library functions.



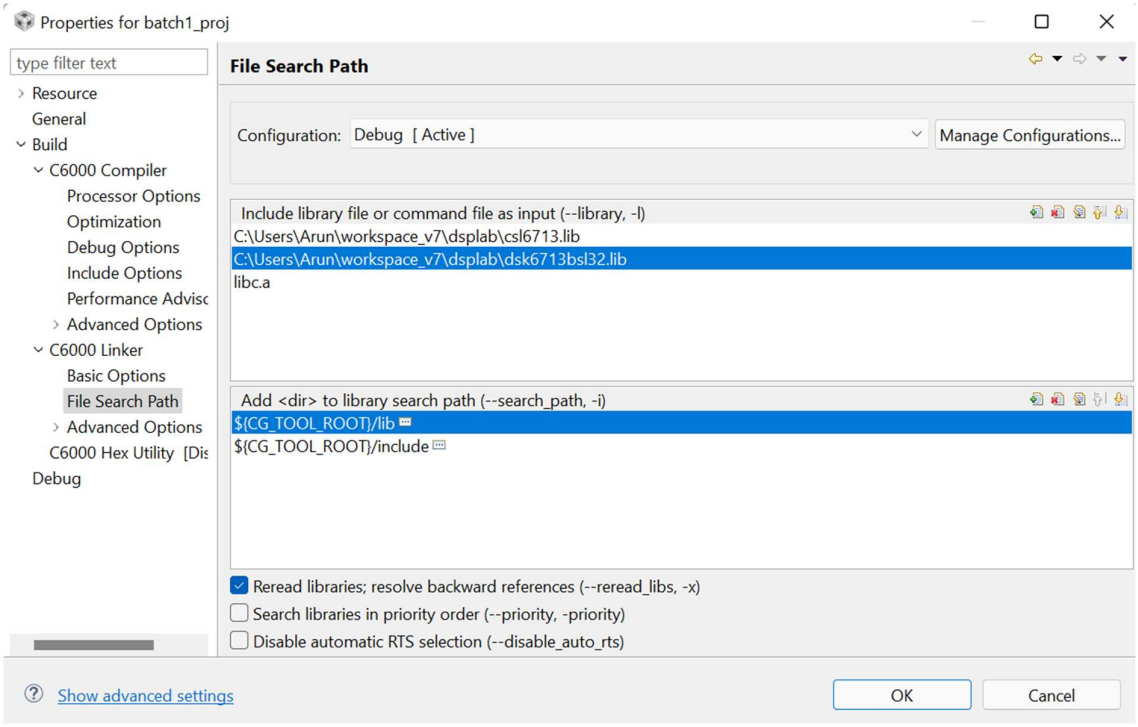
Under *Advanced Options>Predefined Symbols*, in the *Pre-define NAME* window, click on the file icon with green + , and enter CHIP_6713. This symbol is used for conditional compilation. If you don't do this step, you will have to type the line #define CHIP_6713 in your source file.



Under *C6000 Linker> Basic options*, enter a suitable value (eg: 0x5000) as the size for stack and heap



Under *C6000 Linker>File search path*, *Include library file or command file as input* window should already contain the file *libc.a* which is the standard C library. We need to add the chip support and board support libraries. Click on the file icon with green +, browse to *dsplab* folder, select the file *csl6713.lib*, click Open, then OK. Similarly add the file *dsk6713bsl32.lib*


Properties for batch1_proj
type filter text
Resource
General
Build
C6000 Compiler
Processor Options
Optimization
Debug Options
Include Options
Performance Advisc
Advanced Options
C6000 Linker
Basic Options
File Search Path
Advanced Options
C6000 Hex Utility [Dis
Debug
File Search Path
Configuration: Debug [Active]
Manage Configurations...
Include library file or command file as input (--library, -l)
C:\Users\Arun\workspace_v7\dsplab\csl6713.lib
C:\Users\Arun\workspace_v7\dsplab\dsk6713bsl32.lib
libc.a
Add <dir> to library search path (--search_path, -i)
${CG_TOOL_ROOT}/lib
${CG_TOOL_ROOT}/include
Reread libraries; resolve backward references (--reread_libs, -x)
Search libraries in priority order (--priority, -priority)
Disable automatic RTS selection (--disable_auto_rts)
Show advanced settings
OK
Cancel