

**Arab Academy for Science, Technology and Maritime Transport**

**College of Engineering and Technology**

**Electronics and Communications**

Report

**The TIMS DSP-6713 Starter Kit (DSK)**

Prepared by:

*Aly Medhat Moslhi*

*Youssef Khaled Tolba*

Supervised by:

*Dr. Hesham Hamdy*

*Feb. 2021*

**Abstracts**

The recent developments in integrated circuits technology allowed the manufacture of cheap, reliable yet high-speed digital circuits that consumes less power than ever. The packaging size has significantly decreased that allowed the manufacture of complete digital systems on a single chip. That made the digital processing for a signal more feasible and cheaper. The DSK (DSP starter kit) is a low-cost stand-alone platform that enables development for the Texas instrument DSP family. The DSK comes with a full-on board component that serves a wide range of application environments. The DSK uses a Texas Instruments AIC23 (part #TLV320AIC23) stereo codec for the input and output of audio signals. In definition, a codec is either a hardware device or software which can encode or decode a digital stream. It can be used for audio or video content. The codec is compressing technology, there are codecs for data (PKZIP), for images (JPEG, PNG), video (MPEG-2, H.264), and audio (MP3). This report will discuss the basic usage for the DSK, and the specifications of the codec used in the kit.

**Table of Contents**

[Chapter 1 – Introduction 1](#_Toc66069771)

[Equalizers 1](#_Toc66069772)

[1) Parametric Equalizers 1](#_Toc66069773)

[2) Graphic Equalizers 1](#_Toc66069774)

[3) Other Equalization Filter Designs 2](#_Toc66069775)

[Multichannel Surrounded Sound 2](#_Toc66069776)

[1) Mono Sound Format 3](#_Toc66069777)

[2) Stereo Sound Format 3](#_Toc66069778)

[3) Surround Sound Format 3](#_Toc66069779)

[Chapter 2 – TIMS DSP-6713 4](#_Toc66069780)

[Main Key Features 4](#_Toc66069781)

[Other Key Features 5](#_Toc66069782)

[Chapter 3 – TIMS DSP-6713 File Setup 7](#_Toc66069783)

[Configuration of CCS for Kit 7](#_Toc66069784)

[Create New Project File on CCS 7](#_Toc66069785)

[Chapter 4 – Group Contribution 12](#_Toc66069786)

[Change Sampling Rate of TIMS DSP-6713 12](#_Toc66069787)

[Operate Mono and Stereo 12](#_Toc66069788)

[Software Implementation 15](#_Toc66069789)

[References 17](#_Toc66069790)

[Appendices 18](#_Toc66069791)

[Appendix A- MATLAB code 18](#_Toc66069792)

**Table of Figures**

[Figure 1 Mono, Stereo & Surround Sound Systems 2](#_Toc65939717)

[Figure 2 The TMS320C6713 DSP Codec Interface 4](#_Toc65939718)

[Figure 3 Tick the "6713" Box 7](#_Toc65939719)

[Figure 4 New CCS Project Window 8](#_Toc65939720)

[Figure 5 New Empty Project File 8](#_Toc65939721)

[Figure 6 Actual File on PC 9](#_Toc65939722)

[Figure 7 Edit Processor Options 10](#_Toc65939723)

[Figure 8 Edit Predefined Symbols 10](#_Toc65939724)

[Figure 9 Edit Basic Options 10](#_Toc65939725)

[Figure 10 Edit File Search Path 11](#_Toc65939726)

[Figure 11 Final Project Explorer 11](#_Toc65939727)

[Figure 12 DSP\_Config.h file display 12](#_Toc65939728)

[Figure 13 Stereo FIR Filter Example 13](#_Toc65939729)

[Figure 14 Mono-FIR Filter Example 14](#_Toc65939730)

# Chapter 1 – Introduction

## Equalizers

The term equalizer (EQ) has its origins in early telephone engineering when high-frequency losses over long distances had to be corrected so that the spectrum of the sound at the receiver matched the sound spectrum that was initially transmitted. Ideally, the system’s net frequency response has an equal response to all frequencies, and thus the term ‘equalization.’ Since then, the term has been used for any procedure that involves altering or adjusting the magnitude frequency response. Audio equalizers can be electronic circuits or software codes. It can be considered one of the most important methods of audio processing. The main function is to adjust the balance between different frequency components. Audio equalizers can be used for enhancing audio reproduction, including tone controls, loudspeaker, and headphone equalization, room-loudspeaker equalization, loudness equalization, and noise-based equalization. there are several types of audio equalizers which include. [1]

### Parametric Equalizers

The parametric equalizer is the most powerful and flexible of the equalizer types. Midrange bands in a parametric equalizer have three adjustments: gain, center frequency, and quality factor Q (or bandwidth). A parametric equalizer allows the operator to add a peak or a notch at an arbitrary location in the audio spectrum. At other frequencies, far away from the peak or notch, the parametric equalizer does not modify the spectral content, as its magnitude response, there is unity (0 dB). Adding a peak can be useful to help an instrument be heard in a complex mix, or to deliberately add coloration to an instrument’s sound by boosting or reducing a particular frequency range. Notches can be used to attenuate unwanted sounds, including removing power line hum (50 Hz or 60 Hz and sometimes their harmonics) and reducing feedback. [1]

### Graphic Equalizers

A graphic equalizer is a tool for independently adjusting the gain of multiple frequency regions in an audio signal. Common graphic equalizer designs can provide up to about 30 controls for manipulating the frequency response of each audio channel. Structurally, a graphic EQ is a set of filters, each with a fixed center frequency and bandwidth. The only user control is the commanded gain, or the amount of boost or cut, in each frequency band, which is often controlled with vertical sliders. A graphic equalizer can be implemented using either a cascade of equalizing filters or a parallel bank of bandpass filters. [1]

### Other Equalization Filter Designs

There are many other approaches to equalization, especially if the goal is less focused on interactive user control. For instance, equalizers may be designed to match an arbitrary magnitude and phase response, or closely match analog designs, or for the filters to be IIR yet still maintain a linear phase. From these techniques: Matched EQ and Optimal Design Techniques, Digital Equalizer Design Matching Analog Prototypes, and Linear Phase IIR Filters. [1]

## Multichannel Surrounded Sound

Surround soundtracks (or channels) were included in motion pictures, in the early 1950s, to provide a more realistic cinema experience. Later, the popularity of surround sound resulted in its migration from cinema halls to home theaters equipped with matrixed multichannel sound (e.g., Dolby). The below figure shows the three most common sound formats, i.e., mono, stereo, and surround.

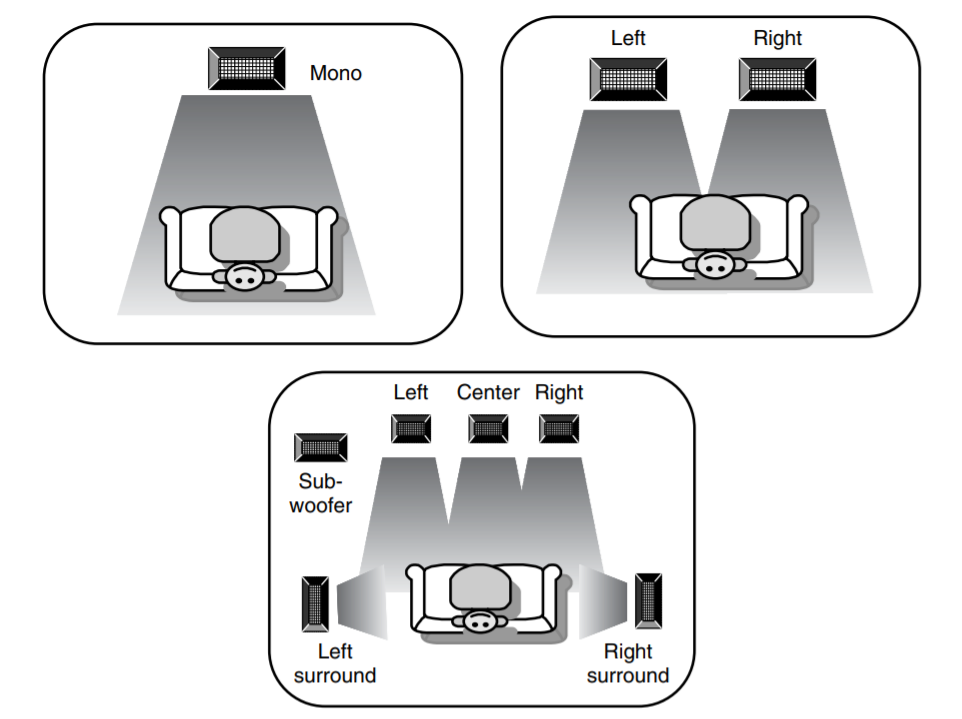


Figure 1 Mono, Stereo & Surround Sound Systems

### Mono Sound Format

Mono is a simple method of recording sound onto a single channel that is typically played back on one speaker. it is one of the most used formats still being used. Recording the audio through a single channel means that when you listen to the mono signal, whatever s that when you listen to the mono signal, whatever you hear on the right speaker will be played on the left speaker. this is because both speakers are playing the same signal channel. This is the recording technique found when recording an audio signal through mobile phones. [2]

### Stereo Sound Format

In stereo encoding, a two-channel recording is employed. Stereophonic sound reproduction is not simply twice mono. A second mono channel can be advantageous to provide increased coverage or even to give some arbitrary impression of difference (a second loudspeaker could, for example, be equalized differently). However, the point of adding a second channel is to specifically create an artificial soundstage between the two loudspeakers. [3]

### Surround Sound Format

Stereo provides a sound field in front, while the multichannel surround sound provides a multi-dimensional sound experience. The surround sound systems typically employ a 5.1-channel configuration, i.e., soundtracks are recorded using five main channels: left (L), center (C), right (R), left surround (LS), and right surround (RS). In addition to these five channels, a sixth channel called the low-frequency-effects (LFE) channel is used for the subwoofer. Since the LFE channel covers only a fraction (less than 150 Hz) of the total frequency range, it is referred to as the .1-channel. [4]

# Chapter 2 – TIMS DSP-6713

The TIMS DSP-6713 is a high-performance stereo audio codec with highly integrated analog functionality. The analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) within the TIMS DSP-6713 use multibit sigma-delta technology with integrated oversampling digital interpolation filters. Data-transfer word lengths of 16, 20, 24, and 32 bits, with sample rates from 8 kHz to 96 kHz, are supported. The ADC sigma-delta modulator features third-order multibit architecture with up to 90-dBA signal-to-noise ratio (SNR) at audio sampling rates up to 96 kHz, enabling high-fidelity audio recording in a compact, power-saving design. The DAC sigma-delta modulator features a second-order multibit architecture with up to 100-dBA SNR at audio sampling rates up to 96 kHz, enabling high-quality digital audio-playback capability, while consuming less than 23 mW during playback only. The TIMS DSP-6713 is the ideal analog input/output (I/O) choice for portable digital audio-player and recorder applications, such as MP3 digital audio players. [5]

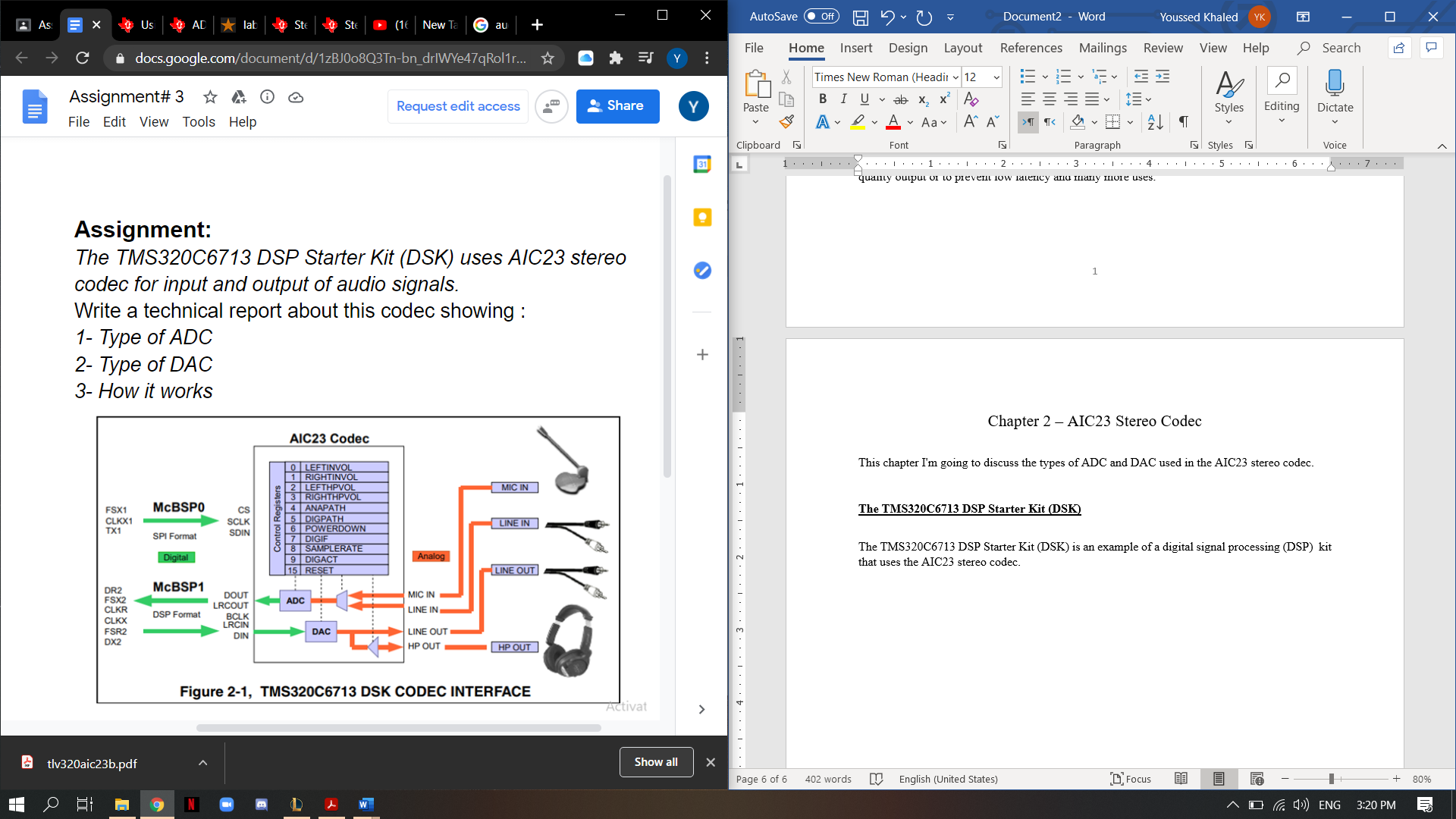


Figure 2 The TMS320C6713 DSP Codec Interface

## Main Key Features

While the TIMS DSP-6713 supports the industry-standard oversampling rates of 256 fs and 384 fs, unique oversampling rates of 250 fs and 272 fs are provided, which optimize interface considerations in designs using TI C54x digital signal processors (DSPs) and universal serial bus (USB) data interfaces. A single 12-MHz crystal can supply clocking to the DSP, USB, and codec. The TIMS DSP-6713 features an internal oscillator that, when connected to a 12-MHz external crystal, provides a system clock to the DSP and other peripherals at either 12 MHz or 6 MHz, using an internal clock buffer and selectable divider. Audio sample rates of 48 kHz and compact-disc (CD) standard 44.1 kHz are supported directly from a 12-MHz master clock with 250 fs and 272 fs oversampling rates. [5]

Low power consumption and flexible power management allow the selective shutdown of codec functions, thus extending battery life in portable applications. This design solution, coupled with the industry’s smallest package, the TI proprietary MicroStar Junior using only 25 mm2 of board area, makes powerful portable stereo audio designs easily realizable in a cost-effective, space-saving total analog I/O solution: the TIMS DSP-6713. [5]

The TIMS DSP-6713 uses AIC23 stereo codec serves as 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 also the codec operates to send 16-bit word length using SPI format, The top 7-bits of the control word should specify the register to be modified and the lower 9-bits should contain the register value. [5]

All audio data flows through the data channel. Many data formats are supported based on the three variables of sample width, clock signal source and serial data format.

As discussed above the TMS320C6713 DSP Starter Kit (DSK) uses the AIC23 stereo codec as its convertor unit, For analog to digital conversion a ADC sigma-delta modulator features third-order multibit architecture with up to 90-dBA signal-to-noise ratio (SNR) at audio sampling rates up to 96 kHz, enabling high-fidelity audio recording in a compact, power-saving design. And for digital to analog conversion a DAC sigma-delta modulator features a second-order multibit architecture with up to 100-dBA SNR at audio sampling rates up to 96 kHz, enabling high-quality digital audio-playback capability, while consuming less than 23 mW during playback only. [5]

## Other Key Features

1. Software Control Via TI McBSP-Compatible Multiprotocol Serial Port
   1. 2-wire-Compatible and SPI-Compatible Serial-Port Protocols
   2. Glueless Interface to TI McBSPs
2. Audio-Data Input/Output Via TI McBSP-Compatible Programmable Audio Interface
   1. I2S-Compatible Interface Requiring Only One McBSP for both ADC and DAC
   2. Standard I2S, MSB, or LSB Just
   3. Audio Master/Slave Timing Capability Optimized for TI DSPs (250/272 fs), USB mode
   4. Industry-Standard Master/Slave Support Provided Also (256/384 fs), Normal mode
   5. Glueless Interface to TI McBSPs
3. Integrated Total Electret-Microphone Biasing and Buffering Solution
   1. Low-Noise MICBIAS pin at 3/4 AVDD for Biasing of Electret Capsules
   2. Integrated Buffer Amplifier with Tunable Fixed Gain of 1 to 5
   3. Additional Control-Register Selectable Buffer Gain of 0 dB or 20 dB
4. Stereo-Line Inputs
   1. Integrated Programmable Gain Amplifier
   2. Analog Bypass Path of Codec
5. ADC Multiplexed Input for Stereo-Line Inputs and Microphone
6. Stereo-Line Outputs
   1. Analog Stereo Mixer for DAC and Analog Bypass Path
7. Volume Control with Mute on Input and Output
8. Highly Efficient Linear Headphone Amplifier
   1. 30 mW into 32 Ω from a 3.3-V Analog Supply Voltage
9. Flexible Power Management Under Total Software Control
   1. 23-mW Power Consumption During Playback Mode
   2. Standby Power Consumption <150 µW
   3. Power-Down Power Consumption <15 µW
10. Industry’s Smallest Package: 32-Pin TI Proprietary MicroStar Junior
    1. 25 mm2 Total Board Area
    2. 28-Pin TSSOP Also Is Available (62 mm2 Total Board Area)
11. Ideally Suitable for Portable Solid-State Audio Players and Recorders

# Chapter 3 – TIMS DSP-6713 File Setup

To work with this kit, one must first install a program called code composer studio (CCS), once the program is set up then the next step is to run it with the TIMS DSP-6713by connecting the Kit with the used computer via USB.

## Configuration of CCS for Kit

Once connected we must set the Target Configuration of the CCS to match that of the Kit this can be done by the following steps (View > Target Configuration ), then select “ New Target Configuration” and select the box with “6713” and save, next figure will help covey.

**Graphical user interface, application

Description automatically generated**

Figure 3 Tick the "6713" Box

## Create New Project File on CCS

After finishing the previous step, it is a matter of creating your file to import and run it on the TIMS DSP-6713 Kit, to create a new project file first you must select Project > New CCS Project this will open a new window where you can name the project file as see fit and then select “empty project (with main.c)”.

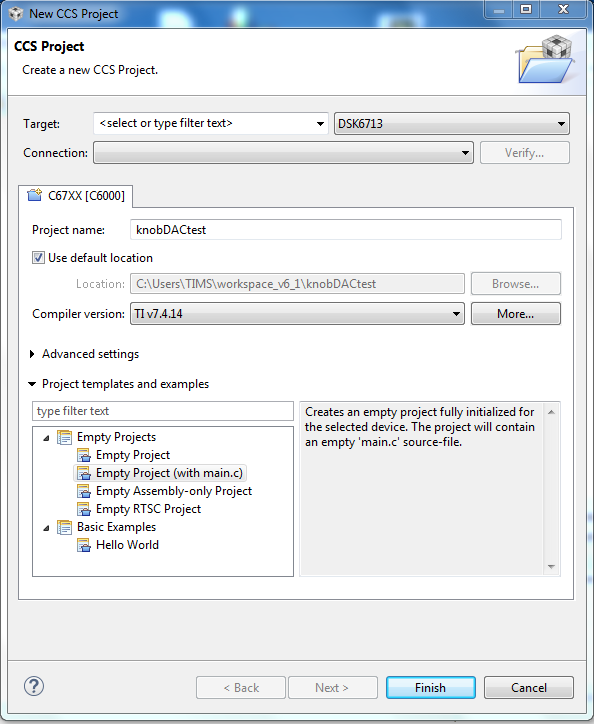


Figure 4 New CCS Project Window

After you click finish the file should appear as follows.

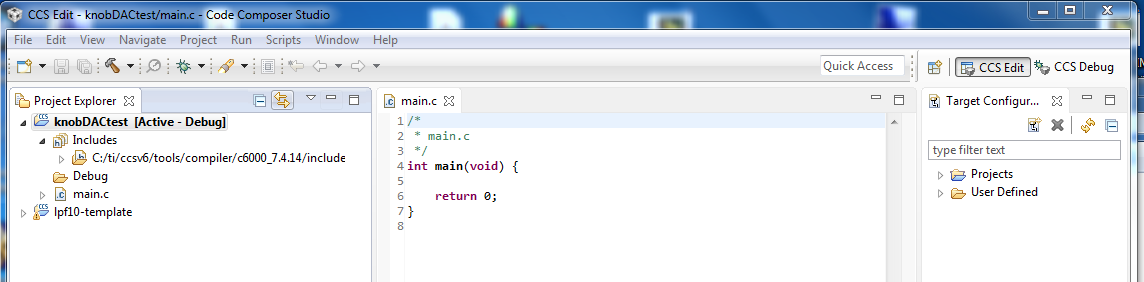


Figure 5 New Empty Project File

With the actual file on PC looking like this.

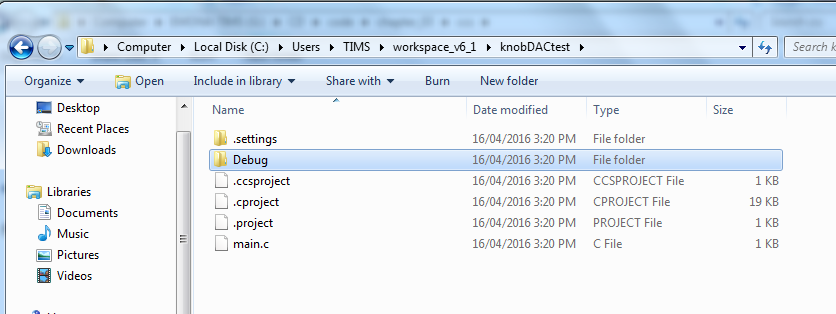
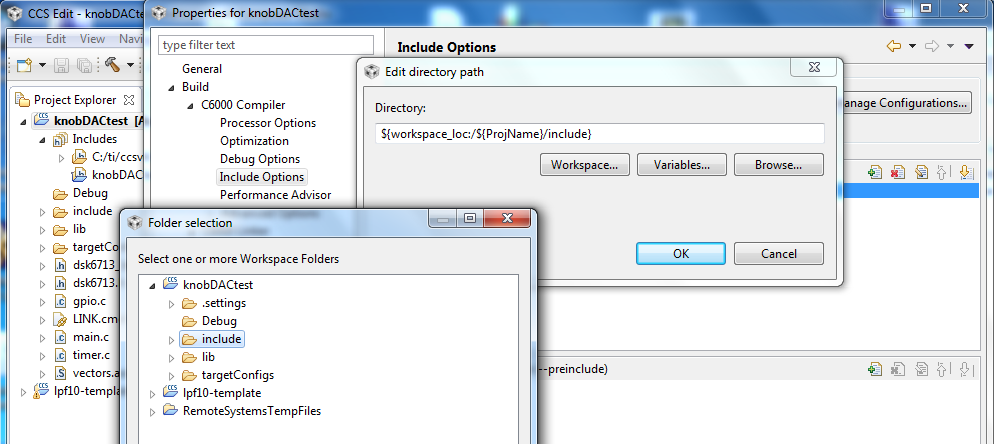


Figure 6 Actual File on PC

After that you should copy the following set of files to your project file ( include, lib , targetConfigs , gpio.c , timer.c , dsk6713\_led.h , dsk6713.h , LINK.cmd , vectors.asm ).

Then you must edit the build of the file to fit the used kit and the new files, start with Project Explorer > Show build settings > Include options > Add path and then select the include folder in your workspace folder, as follows.



Then in build settings edit the dialog as the following figures convey.

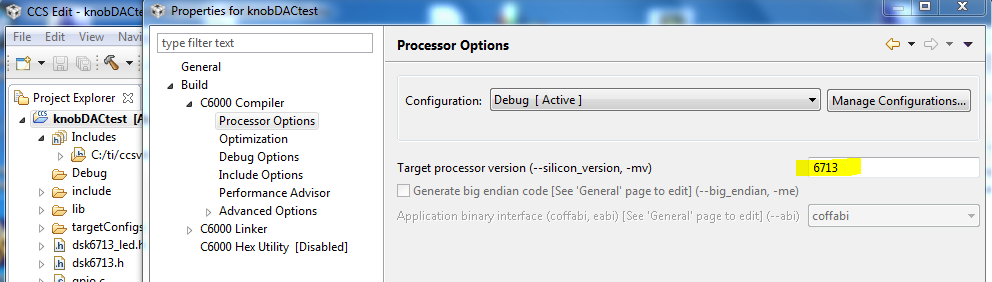


Figure 7 Edit Processor Options

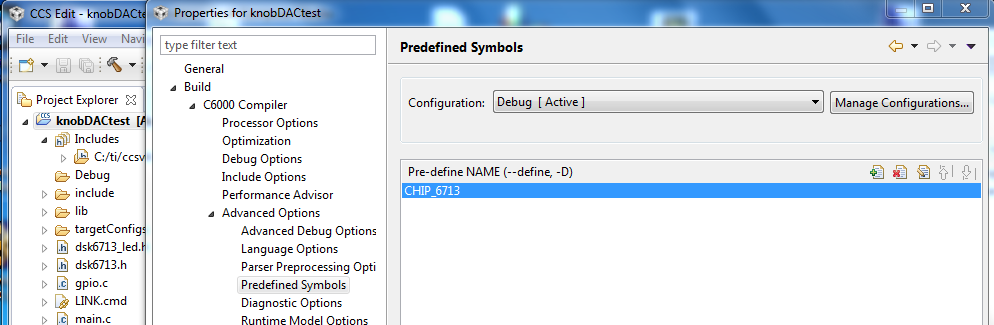


Figure 8 Edit Predefined Symbols

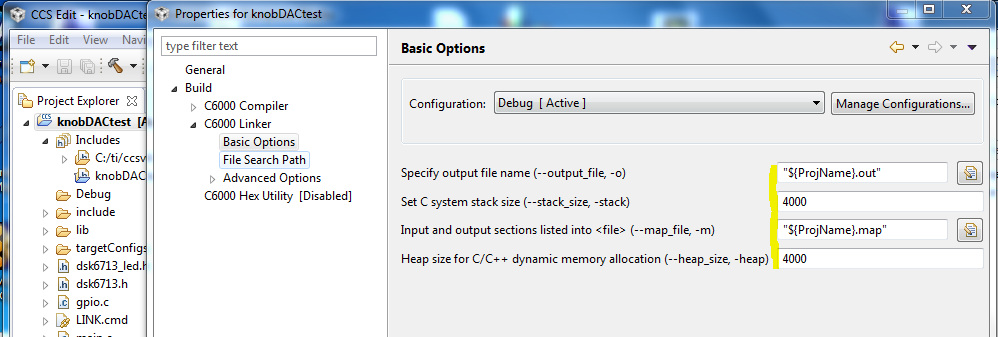


Figure 9 Edit Basic Options

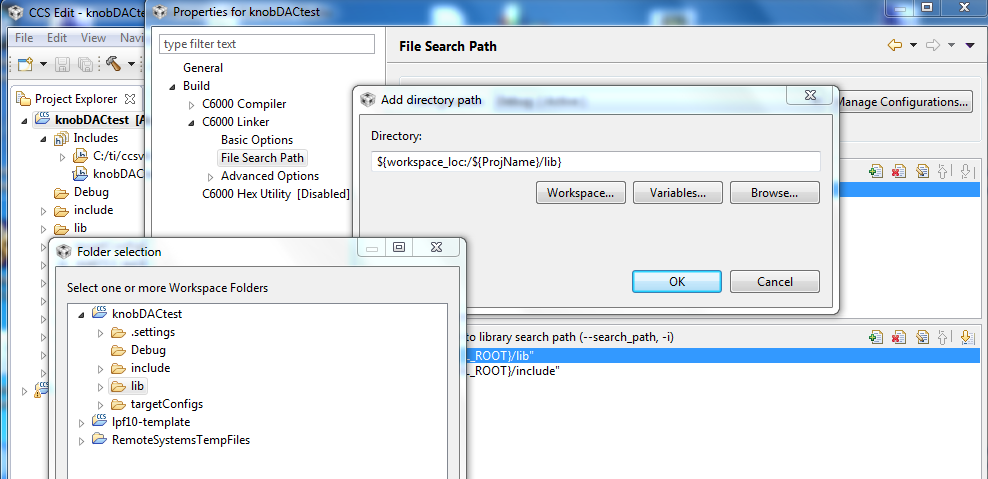


Figure 10 Edit File Search Path

After finishing all build edits you click on build icon and your final project explorer should look as follows and you can start writing and compiling your code.

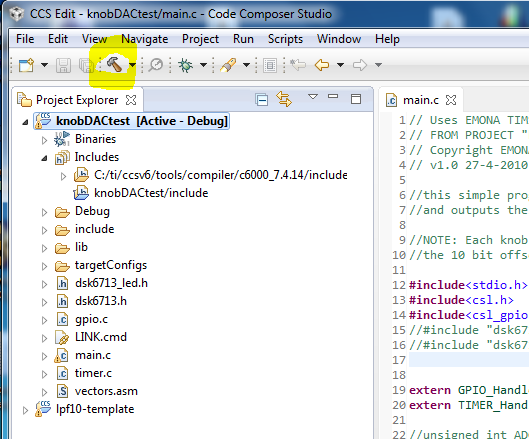


Figure 11 Final Project Explorer

# Chapter 4 – Group Contribution

## Change Sampling Rate of TIMS DSP-6713

The first goal was to try and change the sampling rate of the TIMS DSP-6713 and as mentioned before in chapter 2 the maximum sampling rate that can be archived by the TIMS DSP-6713 is 96 KHz.

This can be change from the DSP.Config.h file tab as shown in the figure below.

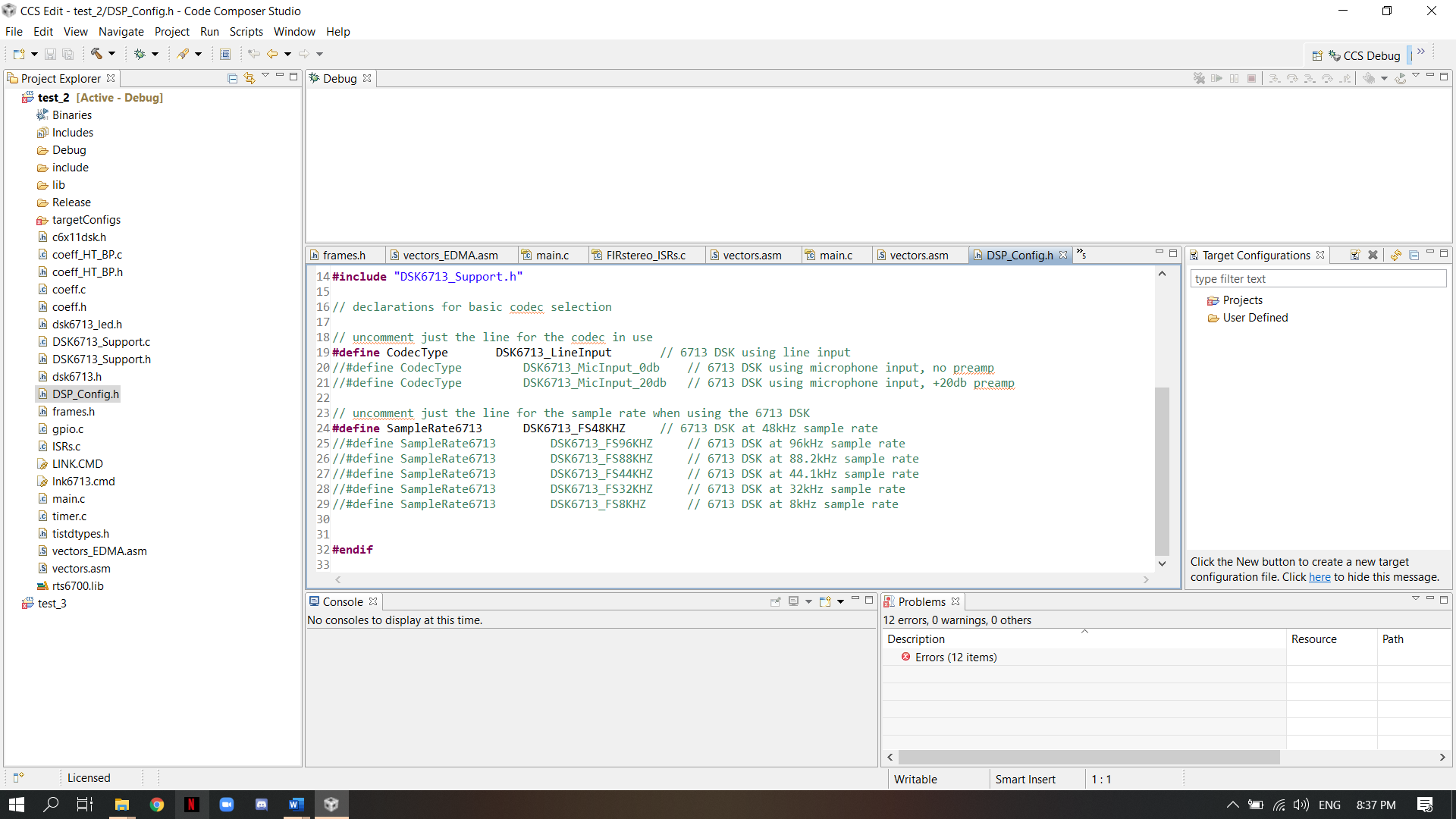


Figure 12 DSP\_Config.h file display

As shown in the figure, the sampling rate can be changed by uncommenting the already written lines the can set the sampling rate to 48 KHz, 96 KHz, 88 KHz, 44 KHz, 32 KHz, and 8 KHz but this does not mean that we can change them to other rates but these are the recommended rates.

## Operate Mono and Stereo

After changing the sampling rate, the next thing we did was to test the TIMS DSP-6713 by using the example code found on the CD that came with the TIMS DSP-6713, the two examples we used were the mono-FIR filter and the stereo-FIR filter.

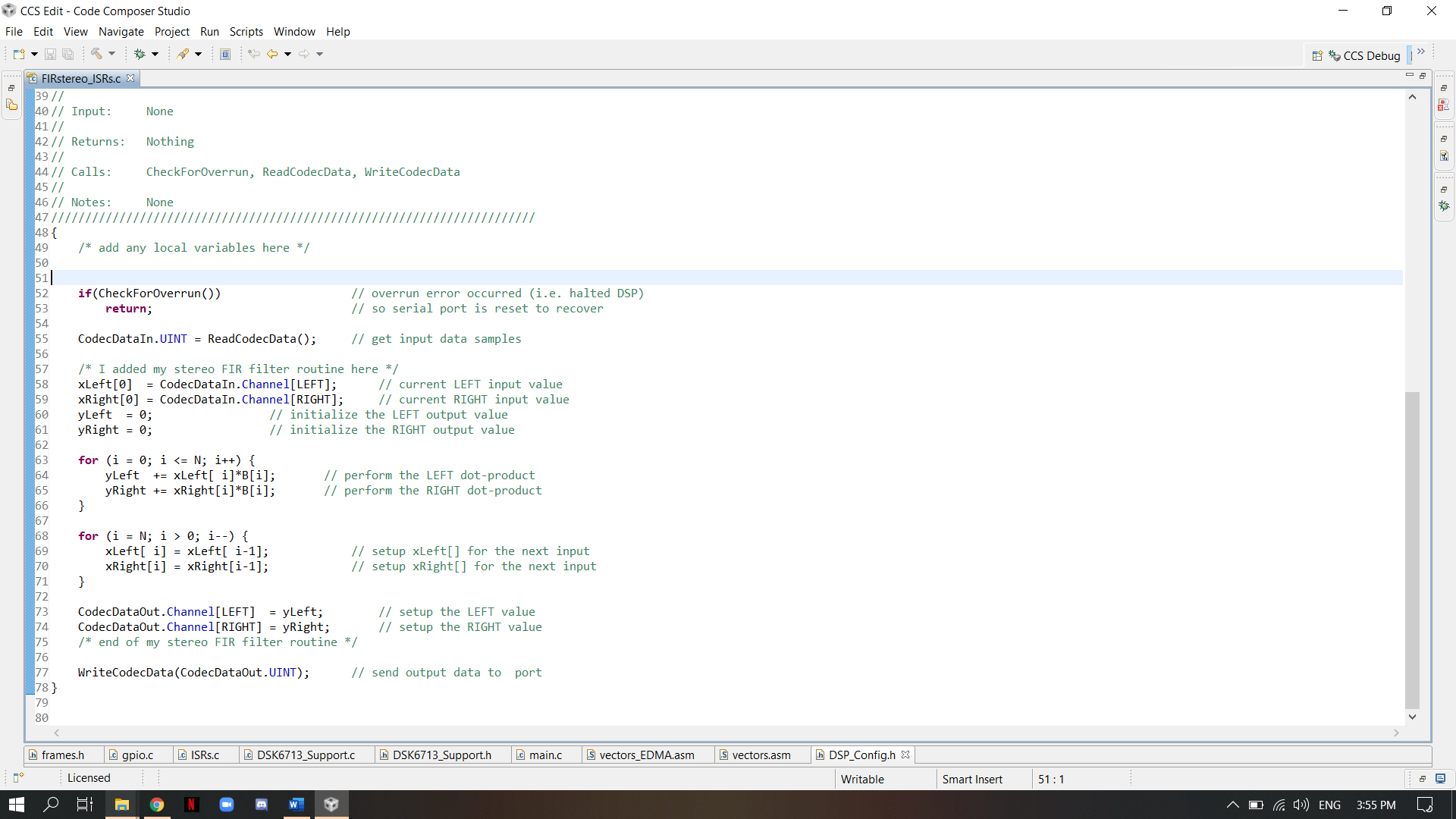


Figure 13 Stereo FIR Filter Example

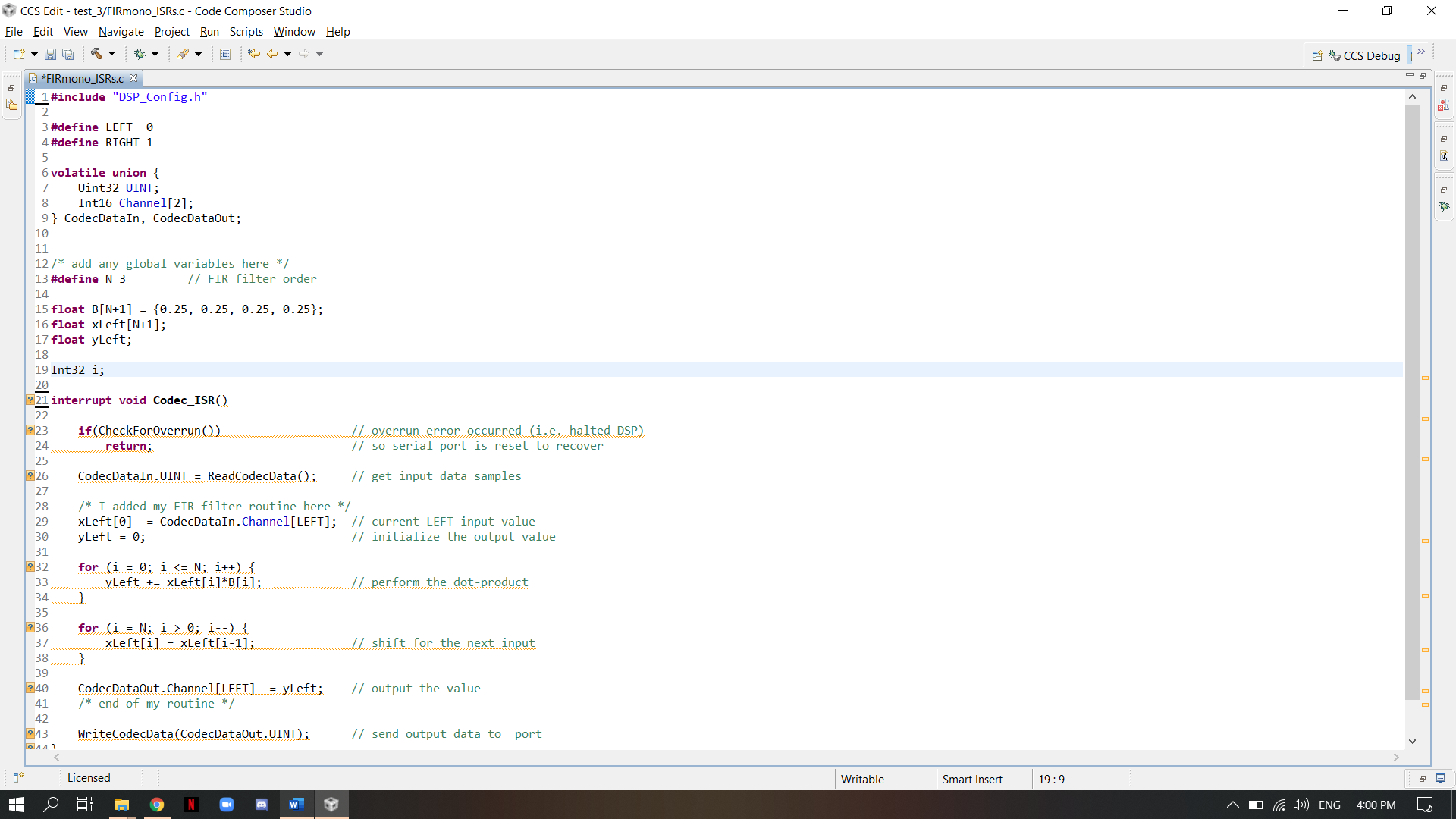


Figure 14 Mono-FIR Filter Example

Unfortunately, due to the TIMS DSP-6713 is a hardware oriented device we could not show the result of the codes in this report thus the only viable option was to mimic the example codes on a software program that does the same function and compare them, we used MATLAB for its one of the best software that can mimic and adjust signals.

## Software Implementation

This Software code involves the creation of a 7-channel equalizer primarily by using the Butterworth filter approximation method. The mathematics modeling program MATLAB was used to model the equalizer. A stereo sound file was input into the program, where 1 low-pass filter, 1 high-pass filter, and 5 band-pass filters divided the stereo sound signal into 7 different frequency channels. Each filtered channel was then adjustable including a dB gain variable to change the gain for each respective channel, and then the results were recombined into two stereo channels. The stereo channels were then saved to a new stereo sound file. The results either will sound will work as on a mono filter or as a stereo filter.

The 7-channel equalizer filter for this MATLAB code can be modeled using basic principles of low-pass, band-pass, and high-pass filters. The Butterworth approximation method is used to filter these 7 individuals’ channels and was accomplished using the mathematics modeling program MATLAB.

The first channel is obtained by a 3rd order LPF with a cutoff frequency of 64Hz giving the first channel range from 0Hz to 64Hz. The second channel is obtained from the 1st bandpass filter with the order of 2, the range of this bandpass filter is from 64Hz to 500Hz. The third channel is obtained from the 2nd bandpass filter with the order of 5, the range of this bandpass filter is from 500Hz to 1000Hz. The fourth channel is obtained from the 3rd bandpass filter with the order of 6. The range of this band is from 1kHz to 4kHz. The fifth channel is obtained from the 4th bandpass filter with the order of 14, the range of this bandpass filter is from 4kHz to 8kHz. The sixth channel is obtained from the 5th bandpass filter with the order of 25, the range of this bandpass filter is from 8kHz to 16kHz. The 7th channel (last one) is obtained from the high pass filter with the order of 20 and a cutoff frequency of 16kHz. The gain for each channel was then calculated by applying the fast Fourier transform (FFT) on all 7 filtered channels individually and then converting the FFTs’ amplitudes to dB gain values.

Then the left and right channel of the stereo sound is separated in the code. This allows separate control of the amplitude of the left channels and the right channels signals for each filtered channel. for example, the code splits the left and right channels of the 32 Hz to 64 Hz filtered channel. This process was repeated for every other filtered channel. The gain control was done by setting a variable equal to the desired gain adjustment, where the default is 0 dB for no gain change. Then, the dB value is converted back into the original amplitude unit for the stereo sounds signal so it can be scaled correctly.

After the gain control of the left and right signals separately for each channel of the seven filtered channels, the signal was restored by adding all 7 of the filtered channels together after adjusting the gain to the desired level. The variable calculated by converting the dB value back into the unit used by the stereo sound signal was used to scale each filtered channel by multiplying each respective one together.

Finally, the output of the code can be chosen to be a stereo type or monotype. By default, the code was set to have stereo output, yet it can output a mono sound by simply uncommenting the last two lines in the code. The conversion from stereo sound to mono sound was done by calculating the mean of the left and right channel of the stereo signal, then output this means to be played for both the left and right channels. Figure 15 shows the plot of the 7-channel equalizer designed using MATLAB when the gain of all the channels was set to 0 dB.

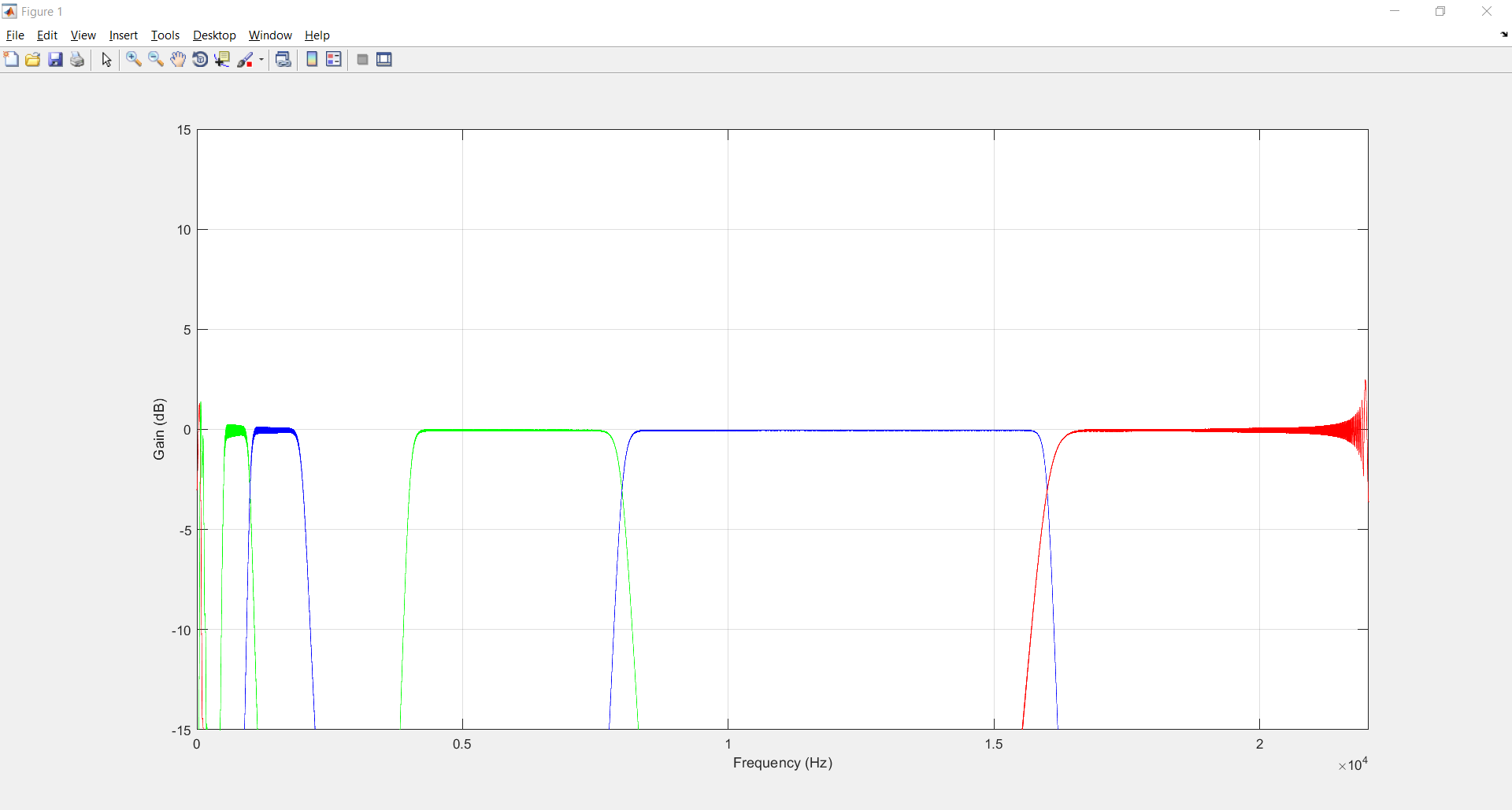


Figure the plot of the 7-channel equalizer at 0 dB

These filters were designed using basic principles of low-pass, band-pass, and high-pass filters. The Butterworth approximation method was used to find a transfer function that would appropriately filter 7 individual channels of the input stereo sound signal, and MATLAB was used to execute this function. The dB gain was adjusted after separating each channel into separate stereo components, and then the channels were recombined per left and right stereo components and the new filtered stereo sound signal was written. Finally, the output was determined to be either stereo sound or mono sound.

# References

|  |  |
| --- | --- |
| [1] | Vesa Välimäki, Joshua D. Reiss, "All About Audio Equalization: Solutions and Frontiers," 2016. |
| [2] | "https://hookeaudio.com/blog/binaural-3d-audio/difference-mono-stereo-surround-binaural-3d-sound/," 31 10 2017. [Online]. Available: https://hookeaudio.com/blog/binaural-3d-audio/difference-mono-stereo-surround-binaural-3d-sound/. [Accessed 15 2 2021]. |
| [3] | M. M. T. Richard Hallum, "Stereophony – a series of perspectives," 2017. |
| [4] | Audio signal processing and coding, A John Wiley & Sons, 2007. |
| [5] | s. d. incorportaion, "TMS320C6713 DSK". |
| [6] | D. A. Rodriguez, "Design of a 10-Band Digital Equalizer," NORWICH UNIVERSITY, 2014. |

# Appendices

## Appendix A- MATLAB code

