GC5016 GC Studio User’s Guide

User's Guide

TEXAS INSTRUMENTS

Literature Number: SLWU033A
May 2006–Revised November 2008

Contents

1 GC5016 Daughtercard and GC101 EVM Setup 7
2 GC Studio and the GC5016/GC101 Evaluation Hardware 7
2.1 GC Studio Software and the GC5016/GC101 Evaluation Hardware 8
2.2 GC Studio Projects 8
3 Creating a GC5016 Experiment with GC Studio 71
3.1 Creating a new GC Studio Project 72
3.2 GC5016 Projects included in the GC Studio Release 80
3.3 GC Studio References 81
4 Physical Description 81
4.1 PCB Layout 81
4.2 Parts List 84
4.3 Schematics 85

List of Figures

1 GC101 EVM and GC5016 Daughtercard 7
2 GC Studio Opening Display 10
3 GC Studio Open Project Display 11
4 GC Studio Open Project2 Display 12
5 GC Studio Project Display 12
6 GC Studio Project Run Display 13
7 JPEG Reference Graph in GC Studio Project Folder 14
8 GC Studio Graph Options 15
9 Start GC Studio 16
10 Select Chip Type 17
11 Select Project Path 18
12 Wizard Greeting Page 19
13 GC5016 Mode Selection 20
14 Simulated Clock Frequency Setup 21
15 Input Formatter Mode Selection 22
16 Output Formatter Mode Selection 23
17 Input Port Files 24
18 Input Port Files Navigation 25
19 CIC and FIR Interpolation Ratio Setup 26
20 Filter Design Analysis Tool 27
21 FIR Filter File Path 28
22 Mixer Frequency Setup 29
23 DIM Switch Setup 30
24 Memory Length 31
25 Project Descriptions 32
26 Project Complete 33
27 New Project View 34
28 GC Studio GC5016 Top Panel 35
29 GC Studio Detail Input, GC5016, and Output Panel 36
30 GC Studio Detail Input, GC5016, and Output Panel 37
31 GC Studio File Input, Selecting Load from File 38
32 GC Studio File Input, Load from File, Browse Box 38
33 GC Studio D.C. Input Formatter Variables 39
34 GC Studio Clock and Sync Variables 40
35 GC Studio D.C. — Channel Select 41
36 GC Studio D.C. — Output Formatter 42
37 GC Studio D.C. — Saving the DDC Output 43
38 GC Studio D.C. — Detailed Channel Block View 44
39 GC Studio D.C. — Detailed Complex Mixer 45
40 GC Studio D.C. — Detailed CIC 46
41 GC Studio DDC — Detailed PFIR 47
42 GC Studio DDC — Selecting PFIR Coefficients 48
43 GC Studio DDC — PFIR Coefficient Entry 49
44 GC Studio DDC — PFIR Select Coefficient File 49
45 GC Studio PFIR Coefficients Select from Files 50
46 GC Studio DDC Gain Display 51
47 GC Studio GC5016 Top Panel 52
48 GC Studio DUC Input, Channels, and Output 53
49 GC Studio Detail Input, GC5016, and Output Panel 54
50 GC Studio File Input, Selecting Load from File 55
51 GC Studio DUC Input Formatter Variables 56
52 GC Studio 57

53 GC Studio Clock and Sync Variables 58
54 GC Studio DUC Fs/8 Tone Input for Delay Calibration 59
55 GC Studio Clock and Sync Variables 60
56 GC Studio DUC — Selecting a DUC Channel 61
57 GC Studio DUC – Detailed DUC Channel View 62
58 GC Studio DUC — Detailed DUC Gain 63
59 GC Studio DUC – Detailed DUC PFIR Section 64
60 GC Studio DUC — Detailed DUC CIC Section 65
61 Studio DUC — Detailed DUC Complex Mixer Section 66
62 GC Studio DUC — Output Formatter section 67
63 GC Studio DUC — Write Output Memory to File 68
64 GC Studio GC5016 Choose New Project Location 72
65 GC Studio GC5016 Folder Locations 73
66 GC Studio GC5016 Plug in Selection 73
67 Stimulus Memory Selections 75
68 Response Memory Selections 76
69 GC101 Clock and Sync Selection 77
70 GC Studio and GC5016 Device Mode 78
71 53 GC Studio and GC5016 Device Mode 79
72 GC Studio and GC5016 Device Mode 79
73 Top Layer 1 81
74 Ground Plane Layer 2 82
75 Signal Layer 3 82
76 Signal Layer 4 83
77 Power Plane Layer 5 83
78 Bottom Layer 6 84

List of Tables

1 Frame Strobe Output Options 8
2 Output Channel Options 9
3 DUC Input Formatter Block — Data Input Format Modes 55
4 DUC Output Modes 66
5 GC Studio GC5016 Figure and Variable Programming 69
6 Bill of Materials for GC5016 EVM 84

GC5016 GC Studio User's Guide

1 GC5016 Daughtercard and GC101 EVM Setup

The GC5016 daughtercard combined with the GC101 motherboard provide a hardware evaluation platform for development of application specific settings for the GC5016. To operate the system, simply plug the GC5016 daughtercard into the DIMM socket on the GC101 motherboard, connect the parallel port interface cable to the GC101 and Host Computer (PC), and connect the power supply to the GC101.

Figure 1. GC101 EVM and GC5016 Daughtercard

2 GC Studio and the GC5016/GC101 Evaluation Hardware

GC Studio includes a software "plug-in" to control the GC101 motherboard, GC5016 daughtercard and the GC5016 device. Input data from a text file on the host computer (PC) is loaded into the GC101 input memory and stimulates the GC5016 input ports. Output data from the GC5016 is captured in the GC101 capture memory and transferred to the host computer. GC Studio configures the GC5016 internal registers using the interactive GUI.

Several GC Studio projects for the GC5016 are included in the installation. These projects can be used as the starting point for creating new experiments as desired.

2.1 GC Studio Software and the GC5016/GC101 Evaluation Hardware

The GC101 hardware works with the GCStudio and GC Scripter software packages in this version. Previous GC101_5016 software experiments must be converted to use the newer GC Studio format.

2.1.1 Required Elements

GC101 evaluation motherboard, the external 5V power supply for the GC101, and a ECP parallel port high speed cable.

GC Studio software (downloaded or provided on a CD).

GC5016EVM, Rev B, plug-in DIMM board.

Host Computer (PC) with BIOS supporting ECP mode for LPT1, Win2000 SP1, WinXP Home or WinXP Professional.

Administrator Group Privileges – under Windows O.S.

2.1.2 GC Studio Installation Instructions

If a previous version of GC Studio is installed, uninstall it first.

Double-click the file named “GC Studio_ Minimal_Setup_XX_XX_XXXXXX.exe” (where XX... is the version of the software) file on the CD or in the downloaded location on the host computer. The installation wizard will open. Follow the on-screen instructions. This loads the main GC Studio software.

Double-click the file named "GC5016_projects_XXXXXX.exe (where XX... corresponds to the GC5016 project version). The installation wizard will open. Follow the on-screen instructions and use the default settings. This loads the GC5016 projects in the proper directory to allow access by the GC Studio software.

After the installation is complete, access the program from the programs list from the computer Start menu.

2.2 GC Studio Projects

Several projects (sometimes referred to as experiments) are provided for the GC5016 in the GC Studio release. Users execute these existing projects or create new projects using the GC Studio GUI interface.

The following sections will create a GC Studio project. Along with the tutorial are editorial comments [ ] with provided explanations.

[Since the GC5016 can have receive-DDC, transmit-DUC, or transceive – DDC and DUC, there are more choices for configuring the GC5016. The GC101 and GC5016 plug in are configured with pseudo-variables (table 7 in the GC5016 datasheet) and some register field values (table 8 – 62 in the GC5016 datasheet).] The DDC and DUC configurations can only be used in one direction at a time.

The GC5016 daughtercard has two channel input ports (A & B), a Sync input port, and an option to use either output channels A & B or C & D, along with a Sync, Frame Strobe, and IFlag output ports. Switches 1 and 2 of dipswitch (S1) on the EVM selects which channels frame strobe is selected for the digital output. Switches 3 and 4 set the output channels to be used.

Table 1. Frame Strobe Output Options

SW1	SW2	Channel Fs [Frame Strobe] as D33	DESCRIPTION
ON	ON	D	used with DDC TDM experiments
OFF	ON	C	used with ChC DDC Interleaved IQ or ChCD Parallel IQ experiments
ON	OFF	B	used with ChA DDC Interleaved IQ or ChAB Parallel IQ experiments
OFF	OFF	A	used with DUC TDM experiments, DUC ChA Interleaved IQ or DUC ChAB Parallel IQ experiments

Table 2. Output Channel Options

SW3	SW4	OUTPUT PORTS SELECTED
ON	OFF	A and B
OFF	ON	C and D

Note: List the required switch settings into the Experiment notes for future reference on new experiments.

Some planning is needed for the GC Studio project, which helps to navigate the user screens and menus.

The four channels in the GC5016 can be programmed in pairs of channels. The term RECEIVE refers to DDC mode. The term TRANSMIT refers to DUC mode. The term SPLITIQ refers to channels that are paired, to increase the PFIR taps for the combined channel.

2.2.1 GC5016 Receive-DDC Studio Projects

RECEIVE applications – The GC5016 daughtercard can be used with an A and B receive input port. The input ports can be two separate real ports A and B, or a parallel complex port AB, where A is the I portion and B is the Q portion. In receive experiments, the input data source can be a user supplied decimal file, single column real, two column complex, or external data from an ADC or pattern generator. [In the case where a transceiver application is desired, and channels C and D are receive, the user configuration must be changed to use the A or B input port versus the C or D input port]. The GC5016 DDC outputs can be interleaved IQ, parallel IQ, or TDM. The selected channel Frame Strobe and divided channel clock are used to decode the output information.

When creating a RECEIVE – DDC application, the requirements include:

- Real or complex data file at IF frequency (can be multiple files if more than one input is used).
- Filter tap file for matching PFIR configuration in the .ANL file (this may be developed after the initial project is setup.

2.2.2 GC5016 Transmit-DUC Studio Projects

TRANSMIT applications – The GC5016 DIMM can be used with an A and B DUC input. The input ports can be TDM of all DUC channels onto the A input port; Interleaved IQ for the two channels A and B on the A and B input port; or the Parallel IQ input AB for splitIQ channel AB. Another DUC input uses the 8bitsI8bitsQ for each of the A and B channels. The DUC output is a full rate for real or parallel complex outputs. The DUC output is half rate for interleaved IQ outputs, and uses the I flag signal to determine the I or Q sample.

If you are creating a TRANSMIT – DUC application, the requirements include:

- Complex input file with zeros, Fs/8 Complex Tone, modulated data by channel.
- Filter tap file for matching PFIR configuration in the .ANL file (this may be developed after the initial project is setup.

The configuration is first developed with the zeros for all but channel A, and the Fs/8 tone in channel A. The Delay Parameter is adjusted for interleaved IQ, or TDM IQ input modes to get the Fs/8 tone + Channel A Mixer Frequency. The offset value changes for the PFIR configuration, input mode, and channel interpolation configuration. The calibration delay parameter is then used in the modulated data project.

2.2.3 GC5016 Transceiver Applications

TRANSCEIVER applications – A Transceiver application combines both a DDC and DUC application. The GC101/GC5016 daughtercard can only support either RECEIVE or TRANSMIT mode. The user can use the same general programming file where channels A and B are TRANSMIT, and channels C and D are RECEIVE. The user would have separate RECEIVE and TRANSMIT experiments for the TRANSCEIVER application.

The user needs to provide a SIGNAL SOURCE for the D.C.-RECEIVE input, or DUC-TRANSMIT input. This is provided from the Input Memory through a conversion that is done in GC Studio. A real input file is provided as a single column of scaled decimal values. A complex input file is provided as two columns of I and Q scaled decimal values. The D.C. Input file is at the over-sampled input rate. The DUC Input file is at the over-sampled Base-band input rate. [The user input file needs to be a modulo-4 number of samples, and is an input parameter during configuration].

The GC Studio / GC5016 displays show the appropriate inputs and outputs. These are determined from the user input and output programming parameters.

2.2.4 GC Studio Running a GC5016 Experiment

The next section illustrates the steps to run GC Studio with a specific experiment. In each experiment's folder there is an expected results JPEG file that shows the output graphs.

2.2.4.1 First Step — Installing the Software, Configuring the GC5016 DIMM, and Applying Power to GC101

The first step assumes that the GC Studio software has been installed on the users computer; the parallel cable has been connected to the GC101 and the users computer; the GC5016 DIMM has the switches set for 3,5,6,7, ON, and the GC5016 DIMM, GC101 have the 5V power supply plugged in, and the 3 LEDs on the lower left side of the GC101 are ON [2 green and one red].

2.2.4.2 Second Step — Start GC Studio

On the PC, go to "Start", select "All Programs", then Select GCStudio Folder, then select "GC Studio".

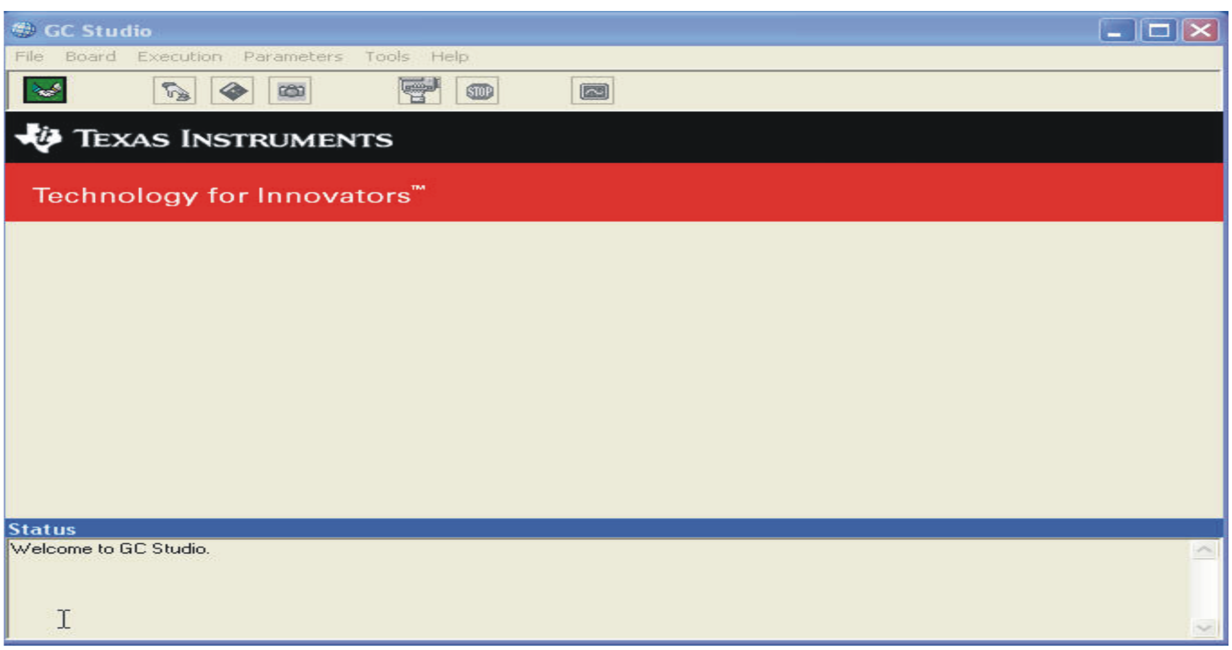


Figure 2. GC Studio Opening Display

2.2.4.3 Third Step — Select the User Project to Open

Select the File Menu with the Mouse and Select Open Project.

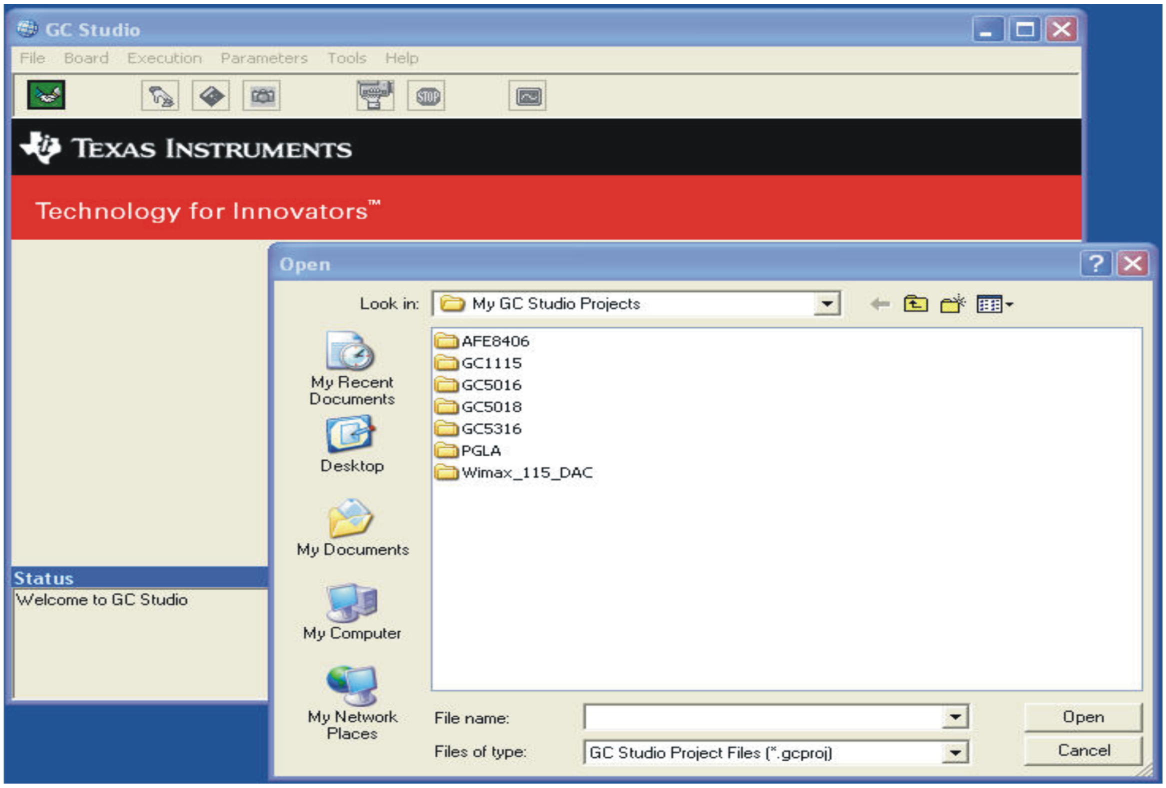


Figure 3. GC Studio Open Project Display

2.2.4.4 Fourth Step — Select the Experiment Folder

Navigate to the GC5016 folder. Select the D.C. folder. Select the DDC_diagconst_in folder. Select the DDC_SplitIQ_ParIQ_16bits_16pins_0sck_div folder. Inside this folder double click on the project file as shown in Figure 4.

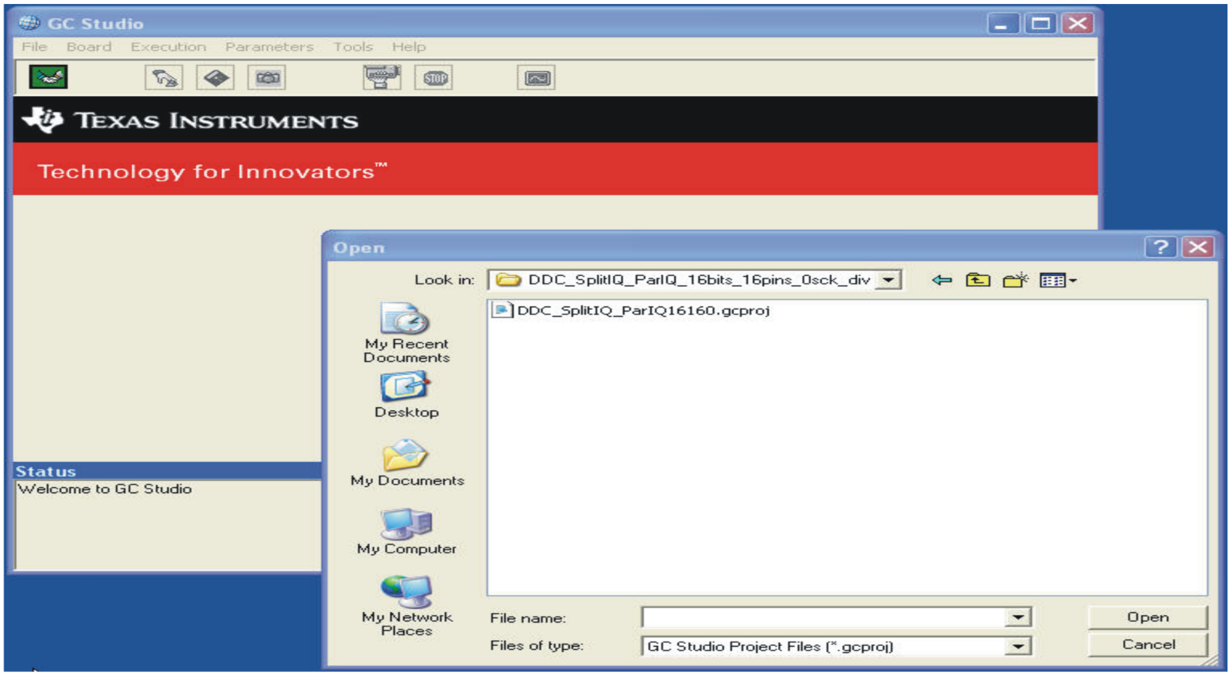


Figure 4. GC Studio Open Project2 Display

At this point, the GC Studio program loads the .gcprog selected file, and the user sees the GC Studio display, the Project Description, and the Graphs that are selected.

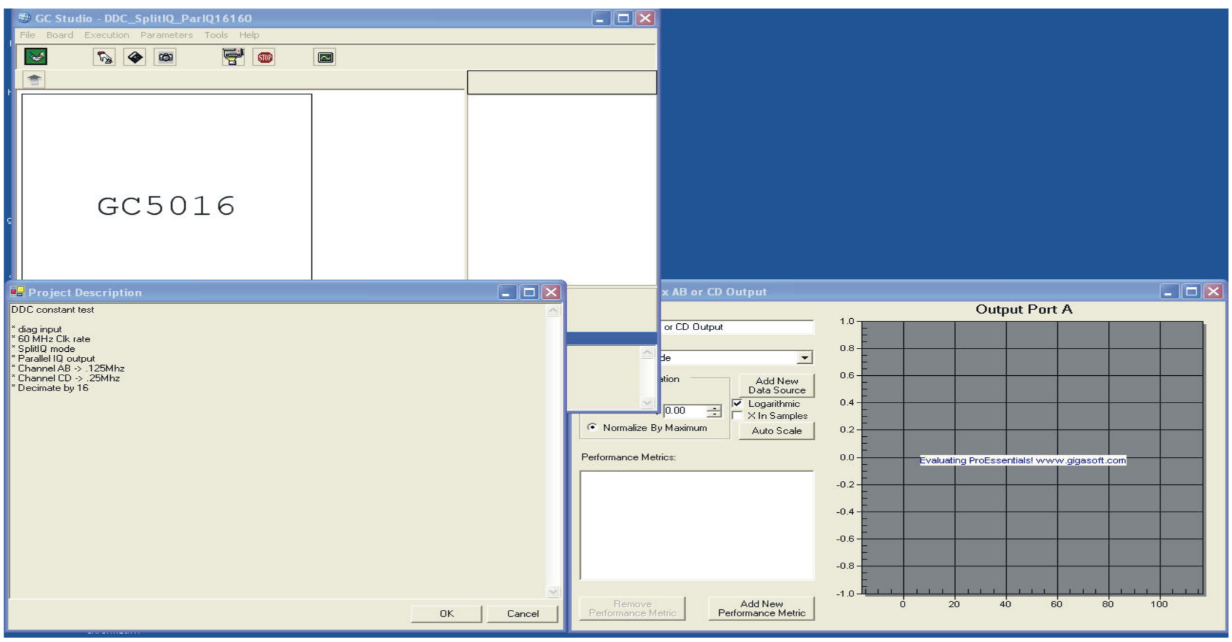


Figure 5. GC Studio Project Display

2.2.4.5 Fifth Step — The User Can Read and Close the Description

Note: In this case, Switches 1,2,3,4 can be used to select different outputs 1,2,4,5,6,7 ON output CD; 3,5,6,7 ON output AB.

On the Project Description panel, depress OK. This closes the Project Description.

Select the GC Studio panel, and find the red collared hand, depress the button to BUILD, LOAD/START, and CAPTURE the experiment. [The default installation of GC Studio has the GC Studio Settings Environment set to Load Automatically After Build (checked) and Capture Once Automatically After Load].

The Status at the bottom of the GC Studio panel updates to indicate the experiment progress. Input graphs are updated when the project opens. Output graphs update when the experiment has captured the data, converted this, and sends each output graph data. The panels have been re-arranged to capture the plot output.

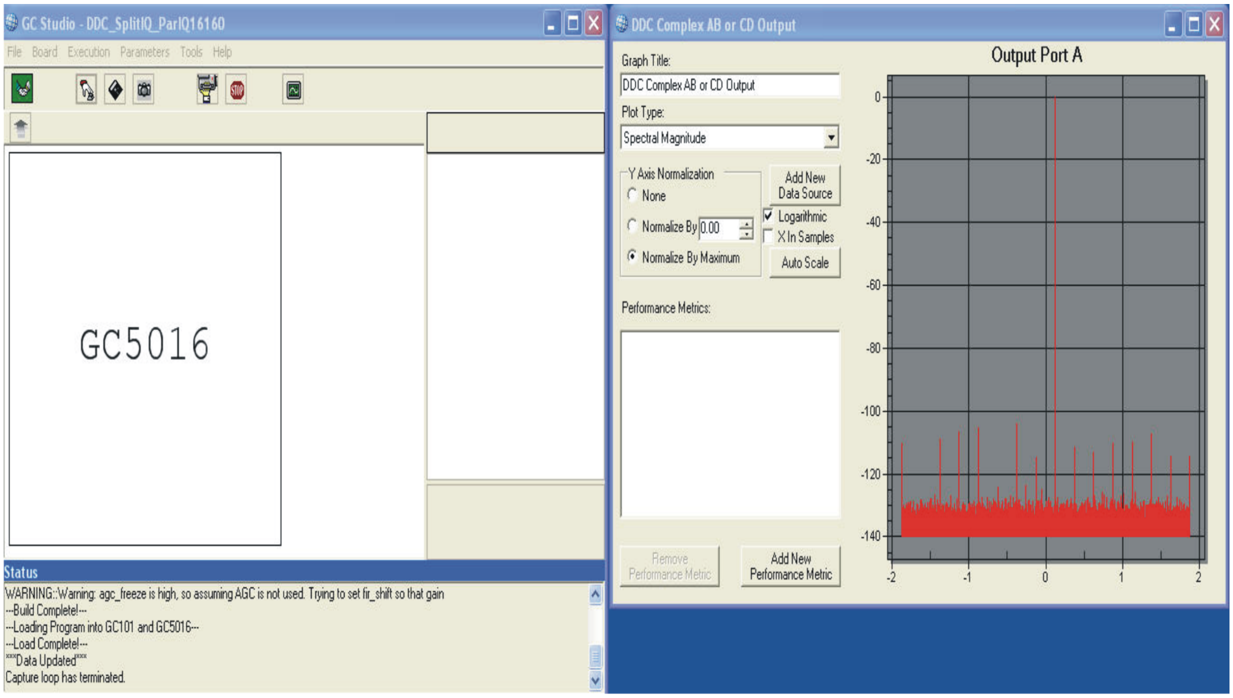


Figure 6. GC Studio Project Run Display

Navigate to the GC Studio experiment folder, go to the D.C. folder, select DDC_diagconst_in folder, select DDC_SplitIQ_ParIQ_16bits_16pins_0sck_div folder and open the JPEG output file "DDC_SplitIQ_ParIQ_16bits_16pins_AB.jpg.

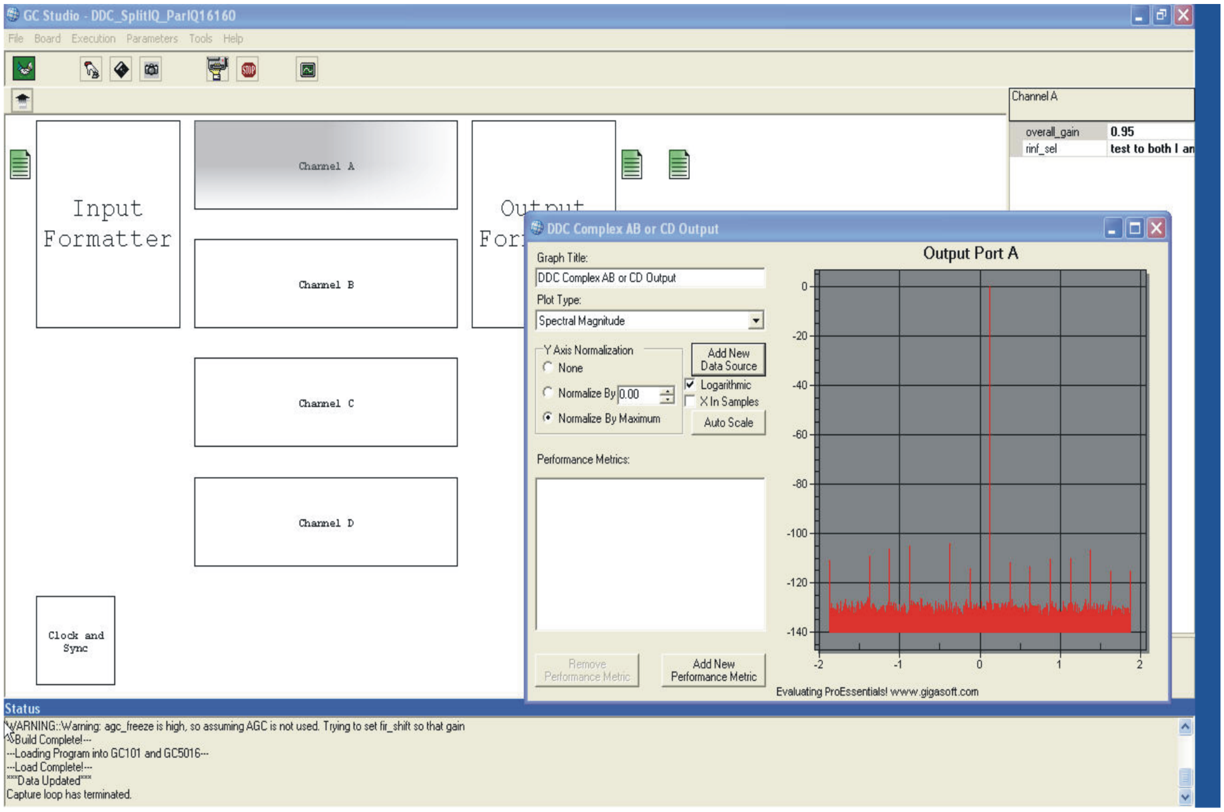


Figure 7. JPEG Reference Graph in GC Studio Project Folder

The plots have zoom features, and formatting. These are discussed in the GC Studio User's Guide document, which is included on the provided CD. Right clicking on the body of the graph brings up the options menu.

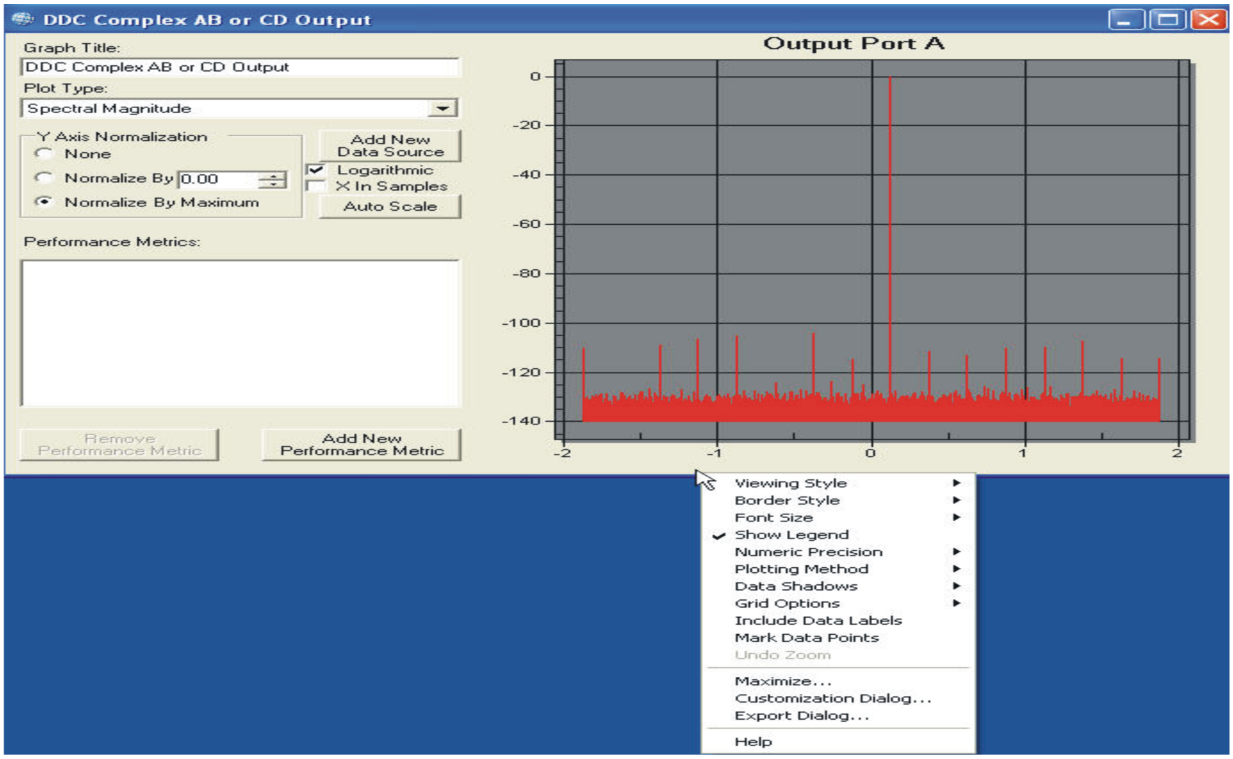


Figure 8. GC Studio Graph Options

There can be multiple plots, and depending on the PC display configuration, they can be off the screen. In this case, select the GC Studio Graph block from the Windows bottom task bar, right click on the desired graph, select Move, and using the keyboard arrows and the mouse move the graph to a visible area of the screen.

2.2.5 Use GC Studio to Create a New GC5016 Experiment

Users can create their own experiments from scratch. This section describes the procedures to create new experiments with GC5016 GCStudio setup wizard.

2.2.5.1 Start GC Studio.

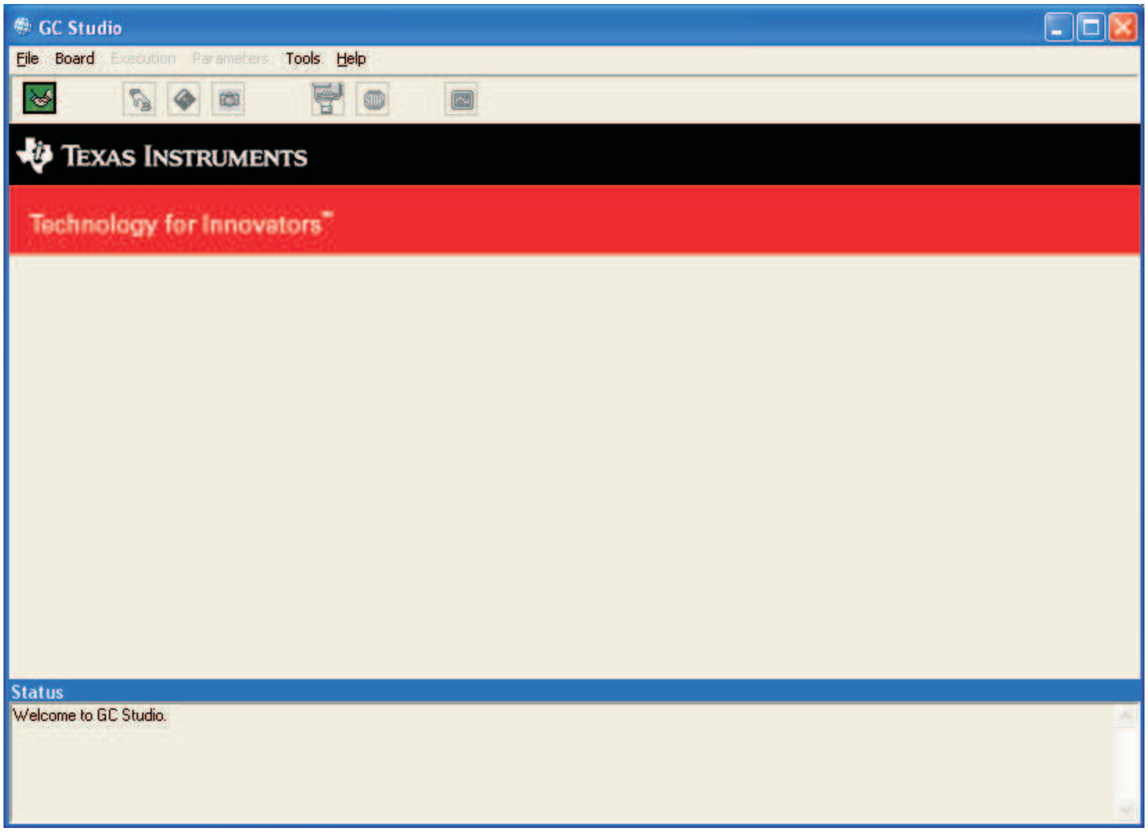


Figure 9. Start GC Studio

2.2.5.2 Select Chip Type and Project Path.

Go to File and click on New Project. Select “GC5016” and click “OK”.

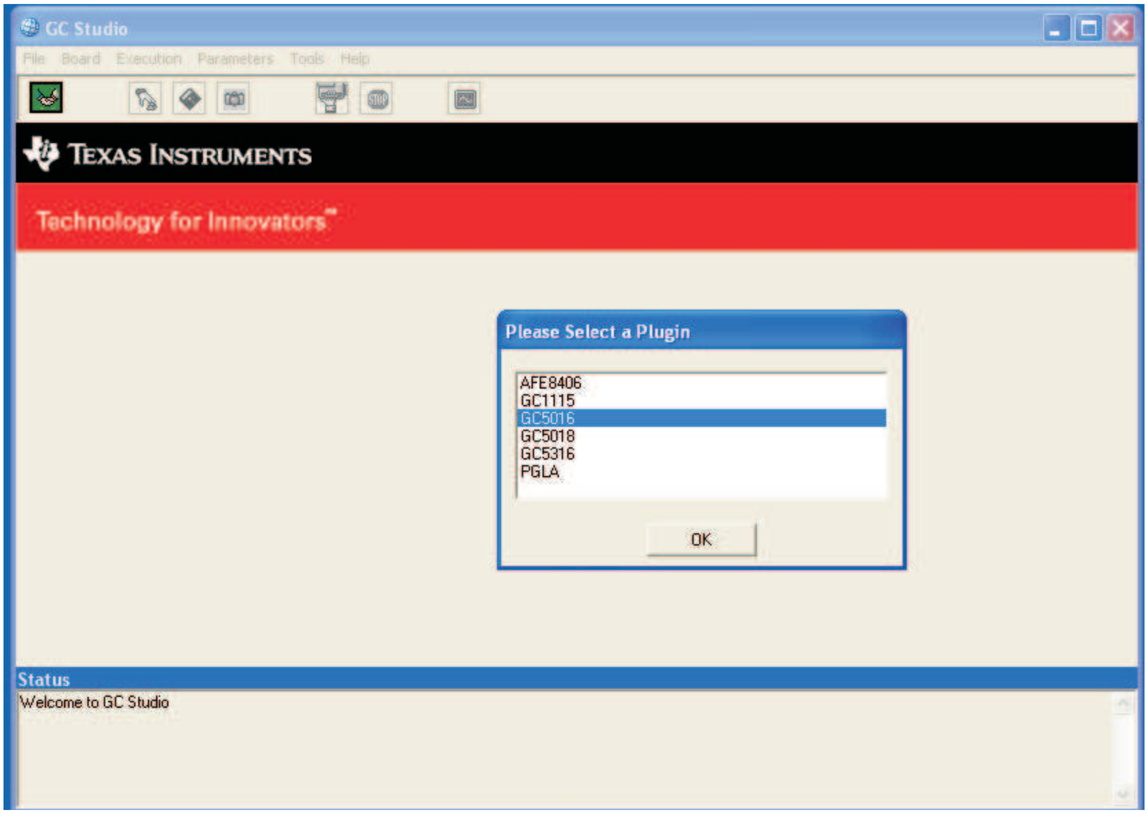


Figure 10. Select Chip Type

The following panel will be brought up. Enter the project name and path. The new experiment project will be created in this location with the name specified in the “Project Name” field. Click “Create” button when path is set.

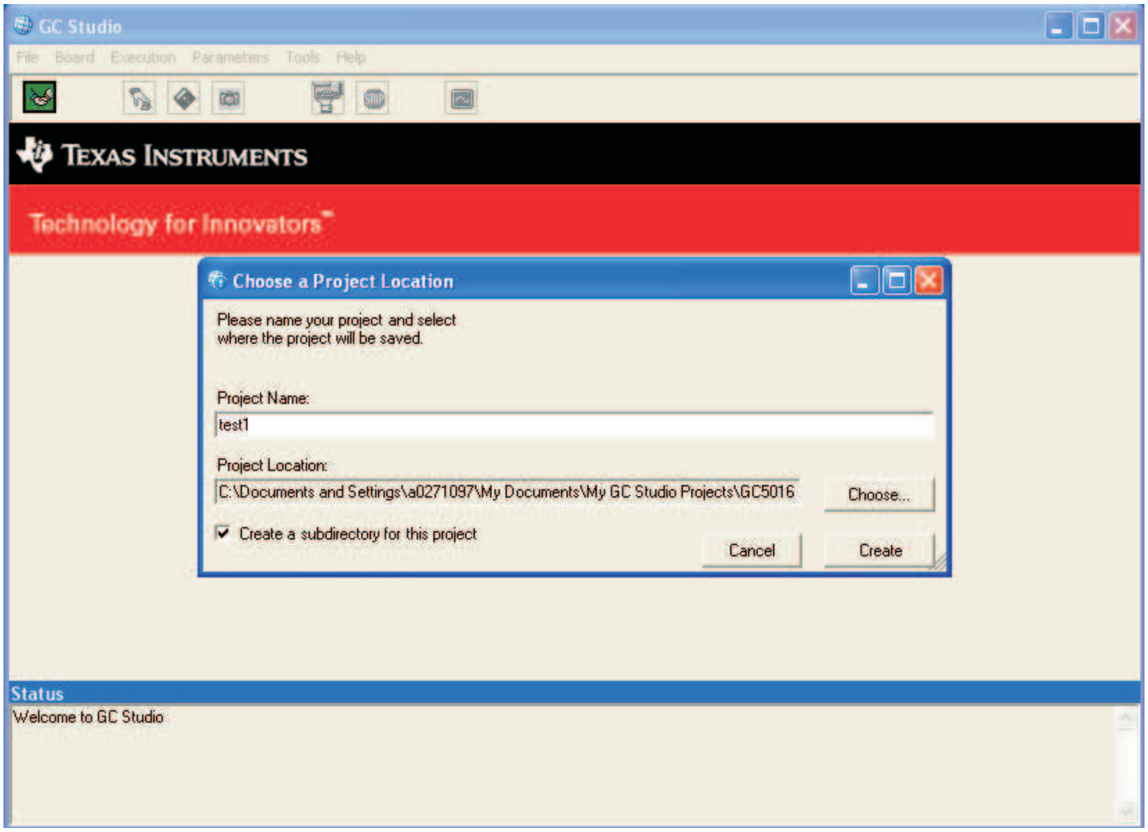


Figure 11. Select Project Path

2.2.5.3 Wizard Greeting Page

The wizard to setup GC5016 will be launched. Click “Next” to start setting up the project.

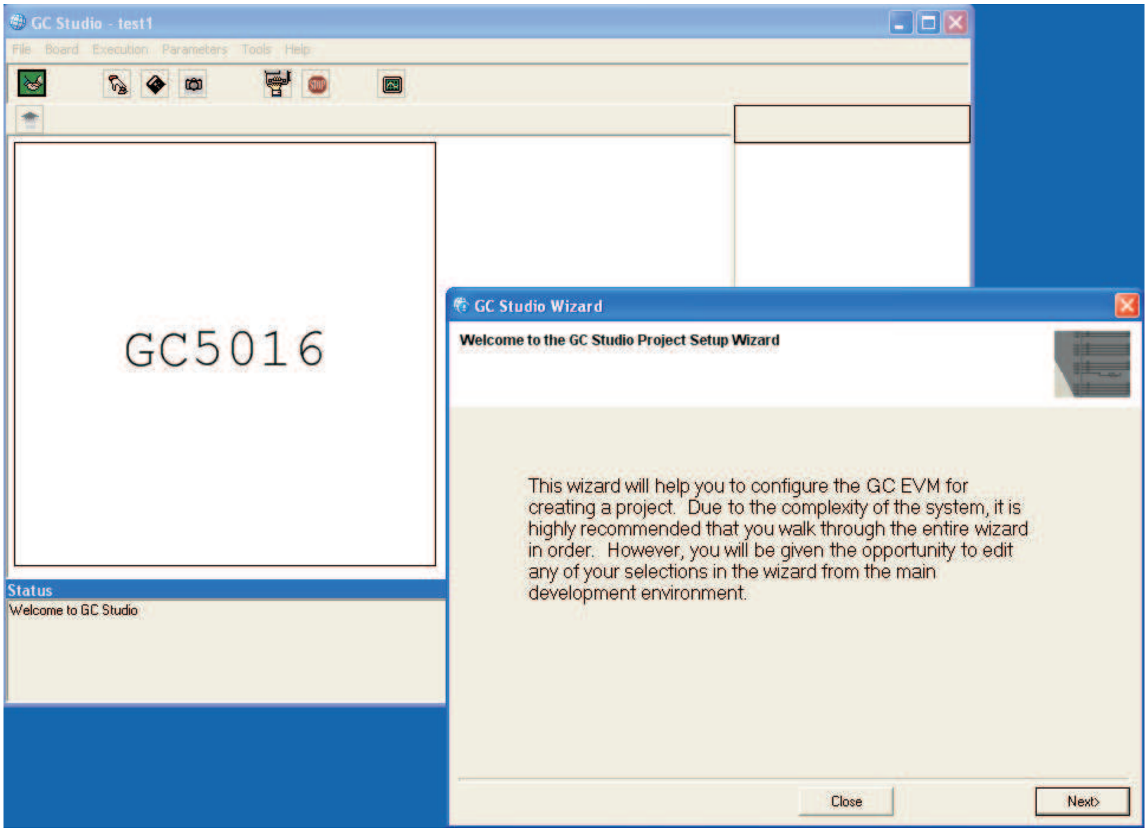


Figure 12. Wizard Greeting Page

2.2.5.4 GC5016 Mode Selection

User can set the GC5016 modes in this wizard. After they are selected, Click “Next”.

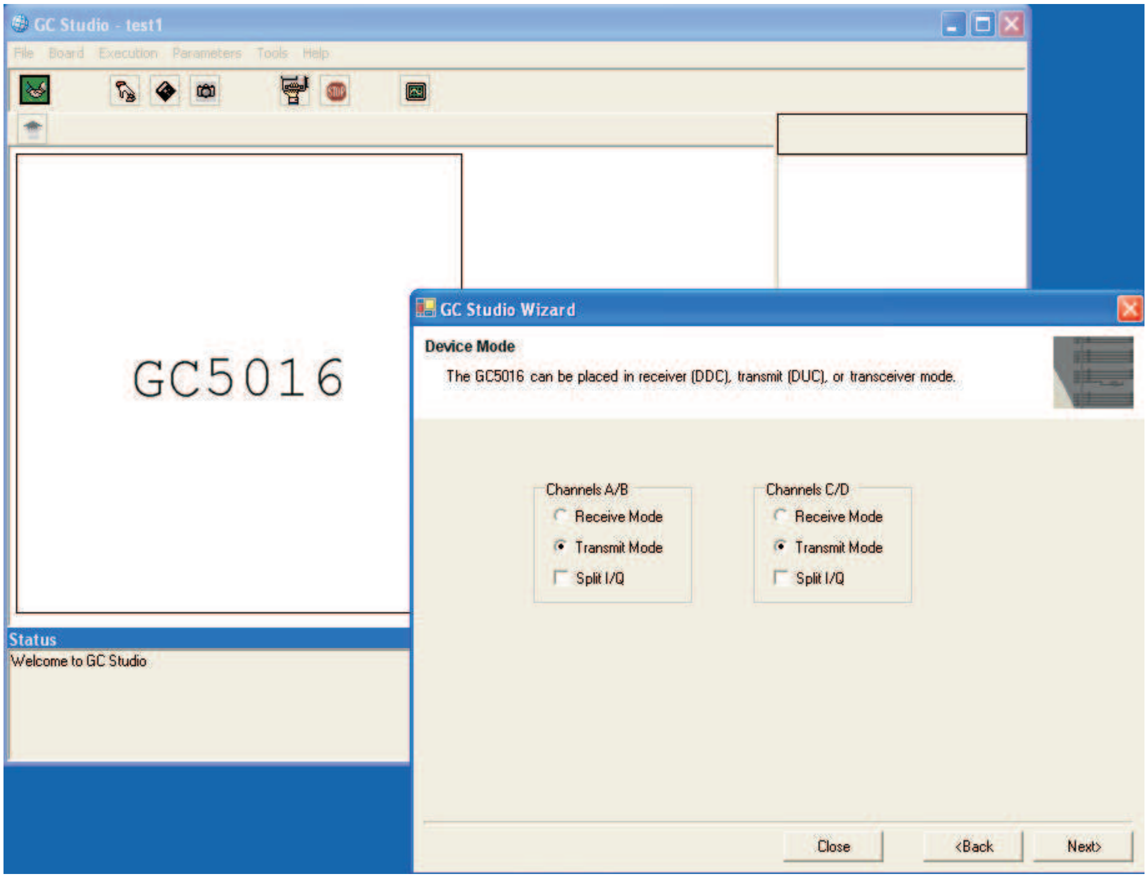


Figure 13. GC5016 Mode Selection

2.2.5.5 Simulated Clock Frequency Setup

Setup the Simulated clock Frequency in this page and Click “Next”.

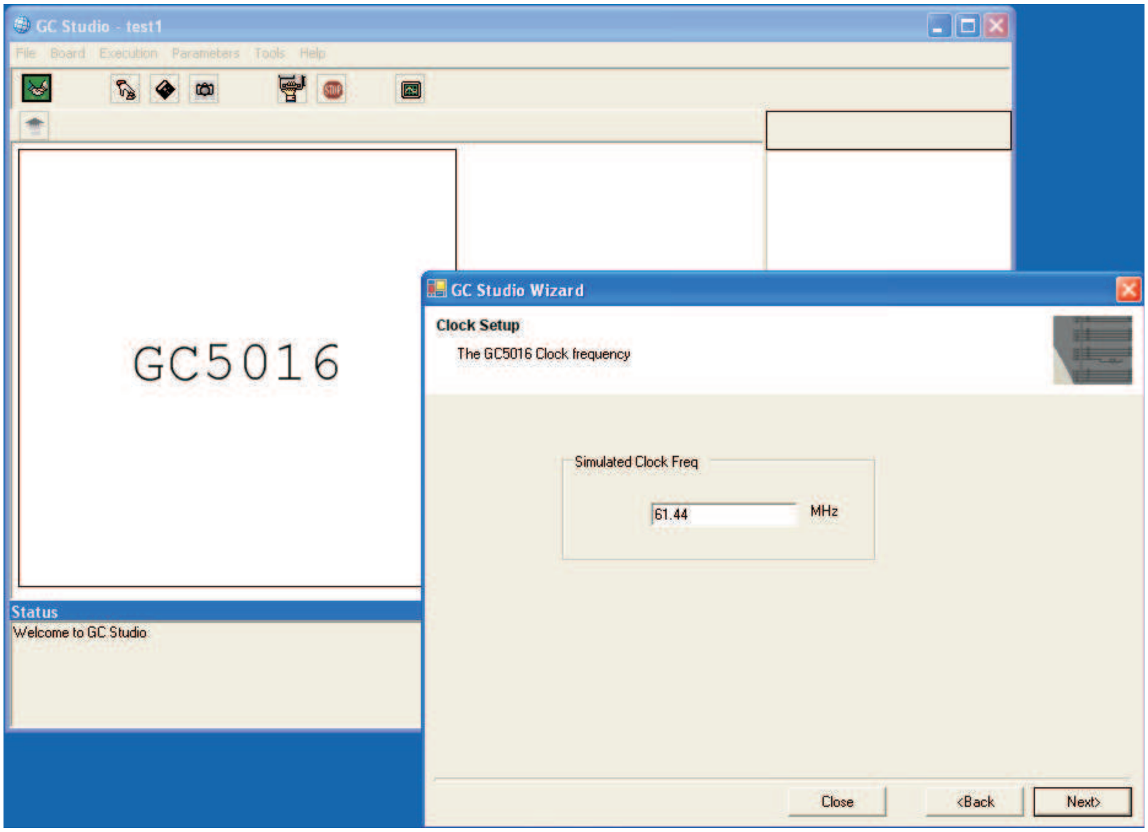


Figure 14. Simulated Clock Frequency Setup

2.2.5.6 Input Formatter Mode Selection

Select the proper input formatter modes from the drop down menu. This panel has a hyper link to TI website. Clicking on it loads the PDF format Input Formatter Mode application notes which has detailed description on how these different modes work. Click “Next” when it is selected.

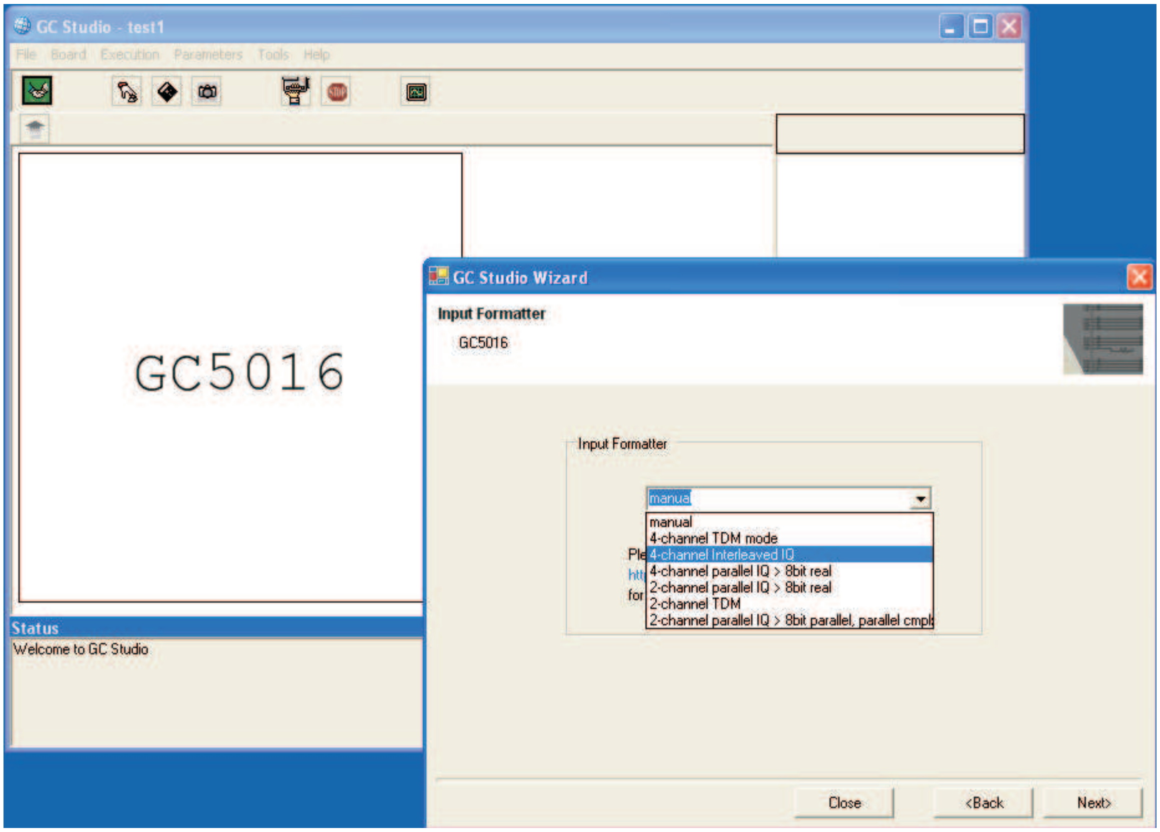


Figure 15. Input Formatter Mode Selection

2.2.5.7 Output Formatter Mode Selection

Select the proper output formatter modes from the drop down menu. This panel has a hyper link to TI website. Clicking on it loads the PDF format Output Formatter Mode application notes which has detailed description on how these different modes work. Click “Next” when it is selected.

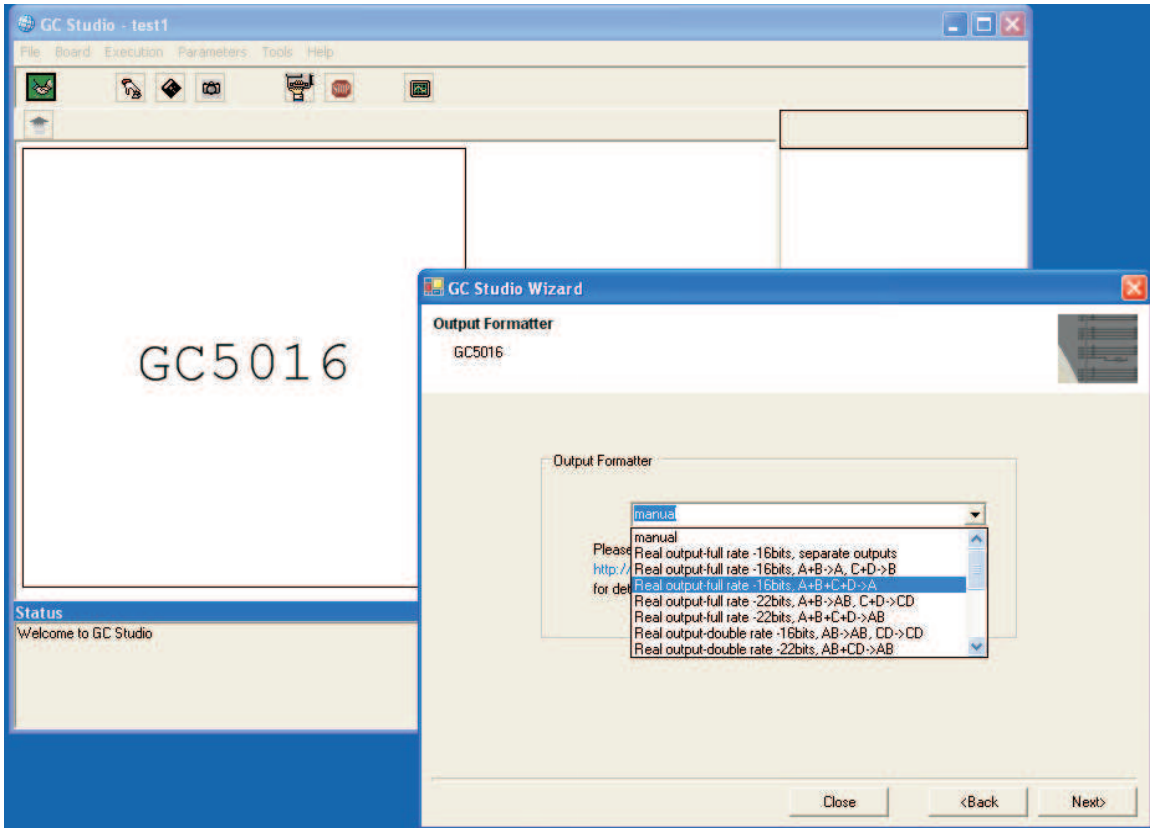


Figure 16. Output Formatter Mode Selection

2.2.5.8 Input Port Files

This panel allows user to specify the input baseband data path. User can type in the file path or click on the button next to the text box. Clicking on the button brings up a dialog window for user to navigate to the file.

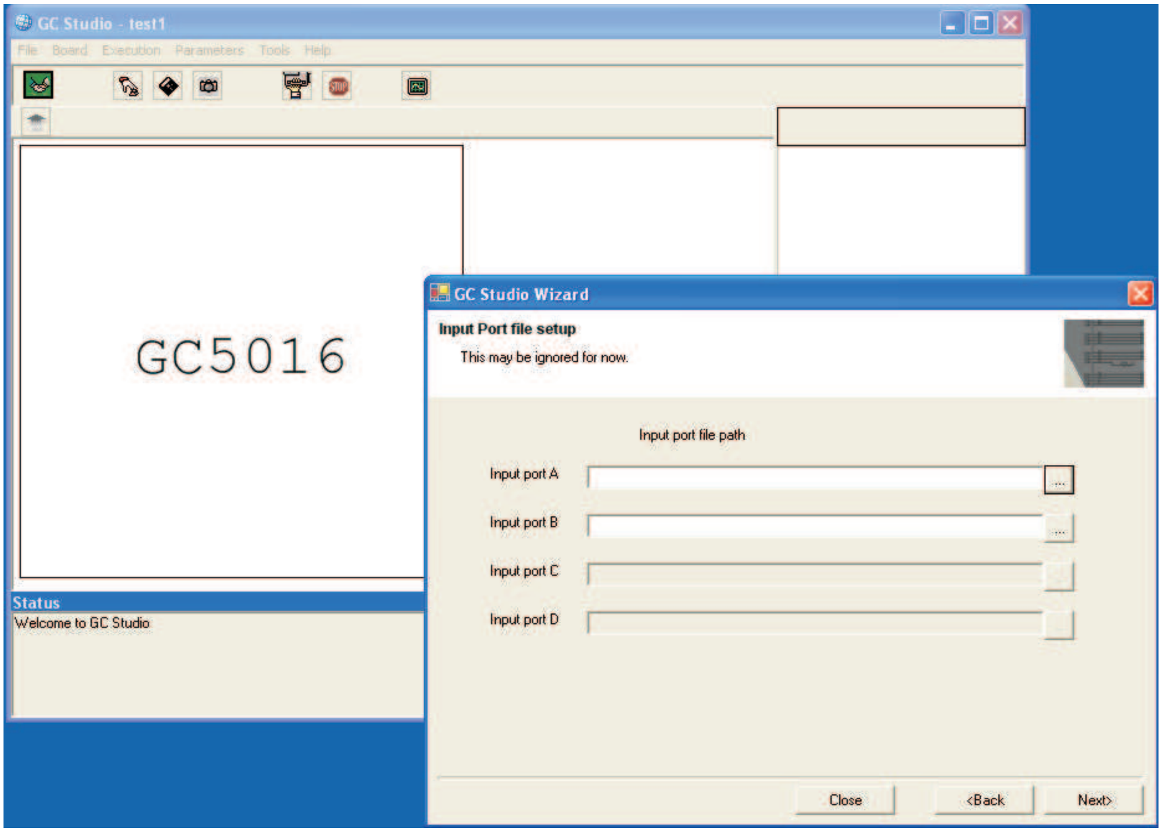


Figure 17. Input Port Files

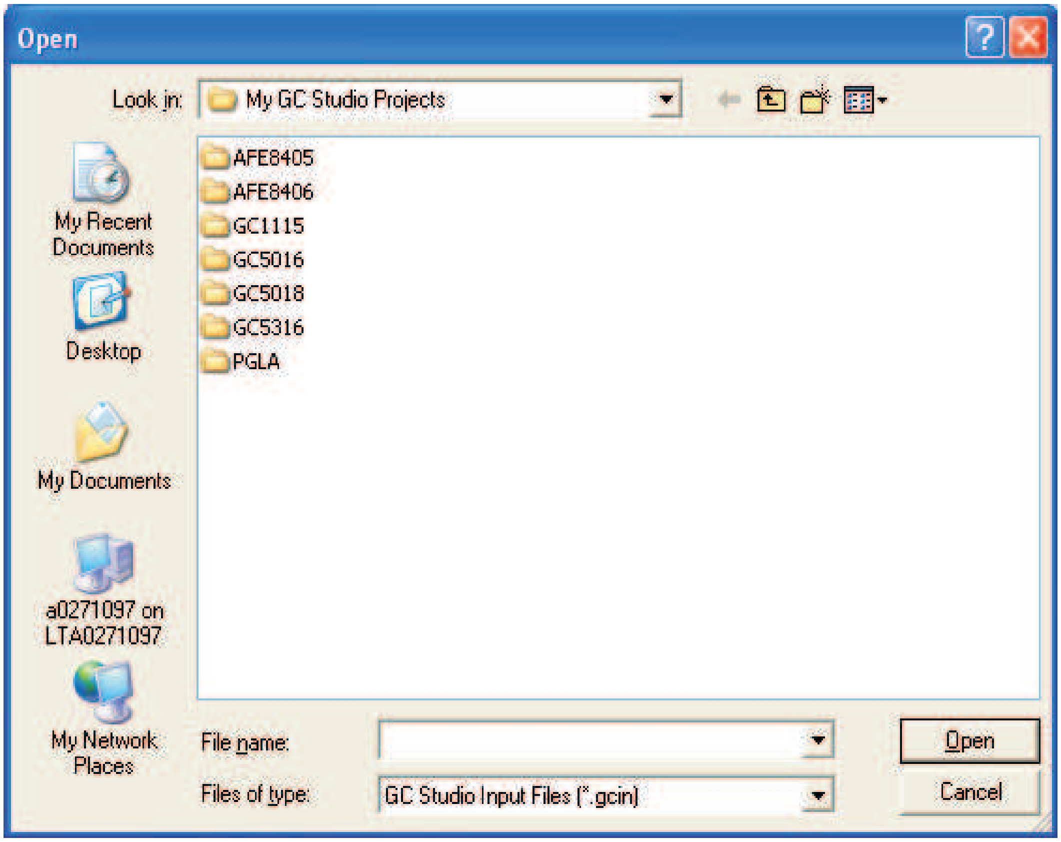


Figure 18. Input Port Files Navigation

2.2.5.9 CIC and FIR Interpolation Ratio Setup

User can setup the CIC and FIR interpolation ratio from this wizard.

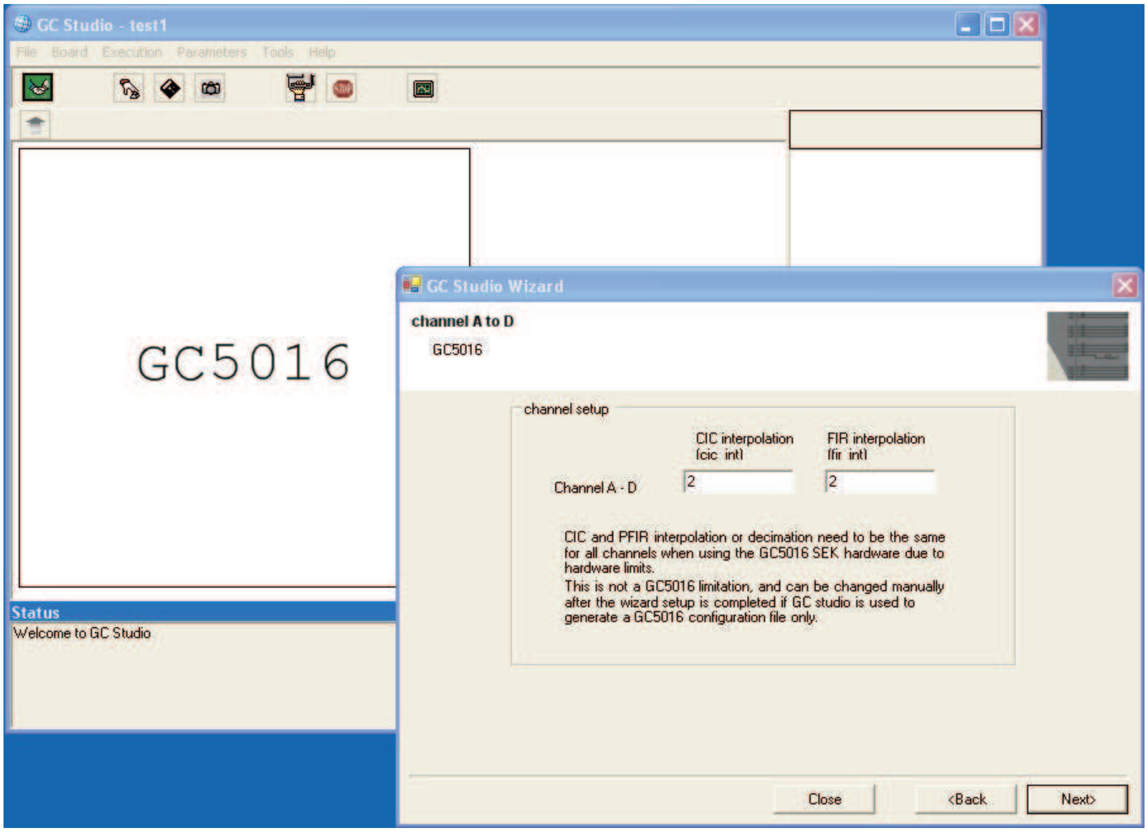


Figure 19. CIC and FIR Interpolation Ratio Setup

2.2.5.10 Filter Design Analysis Tool

This page helps user to design the CIC and FIR filter. Click on “Design Analysis” button. It will calculate the number of the CIC and FIR filter taps and print out the summary in the text box.

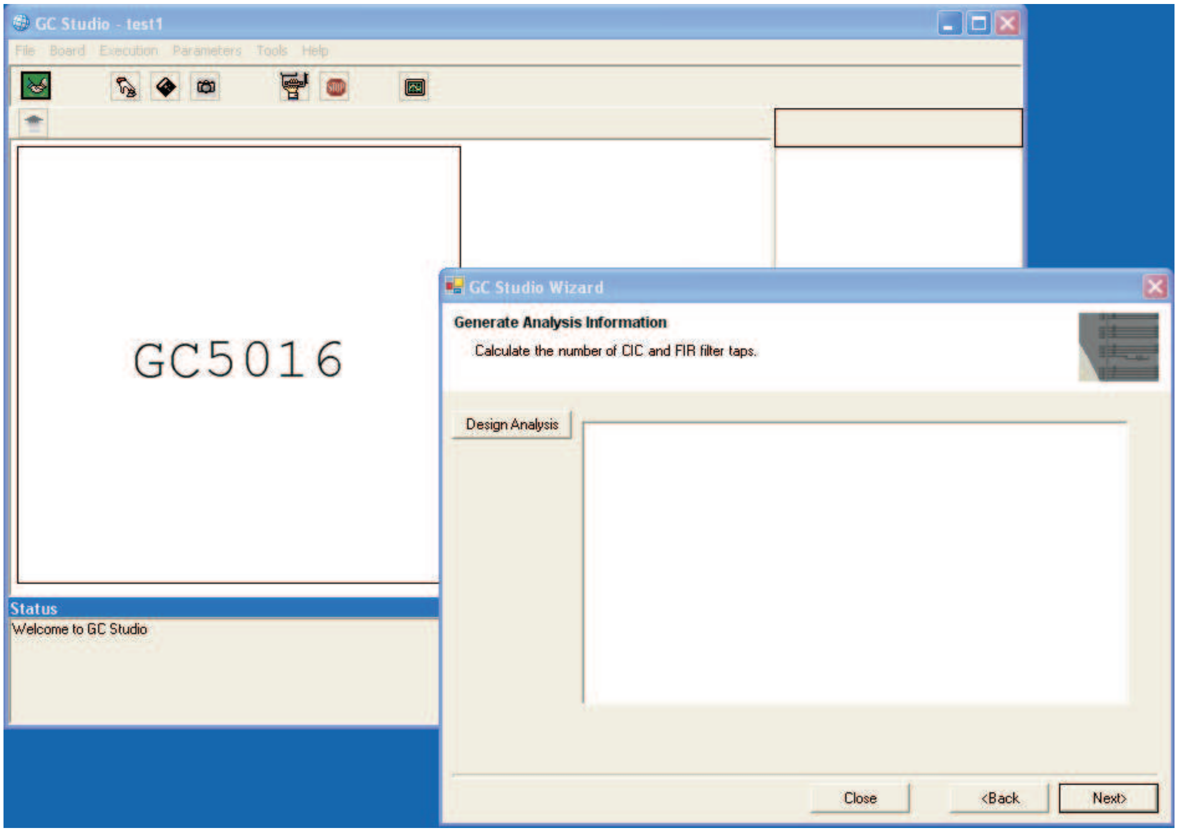


Figure 20. Filter Design Analysis Tool

2.2.5.11 FIR Filter File Path

Specify the FIR filter file path in this page. User can click on the button next to the text box to navigate to the file.

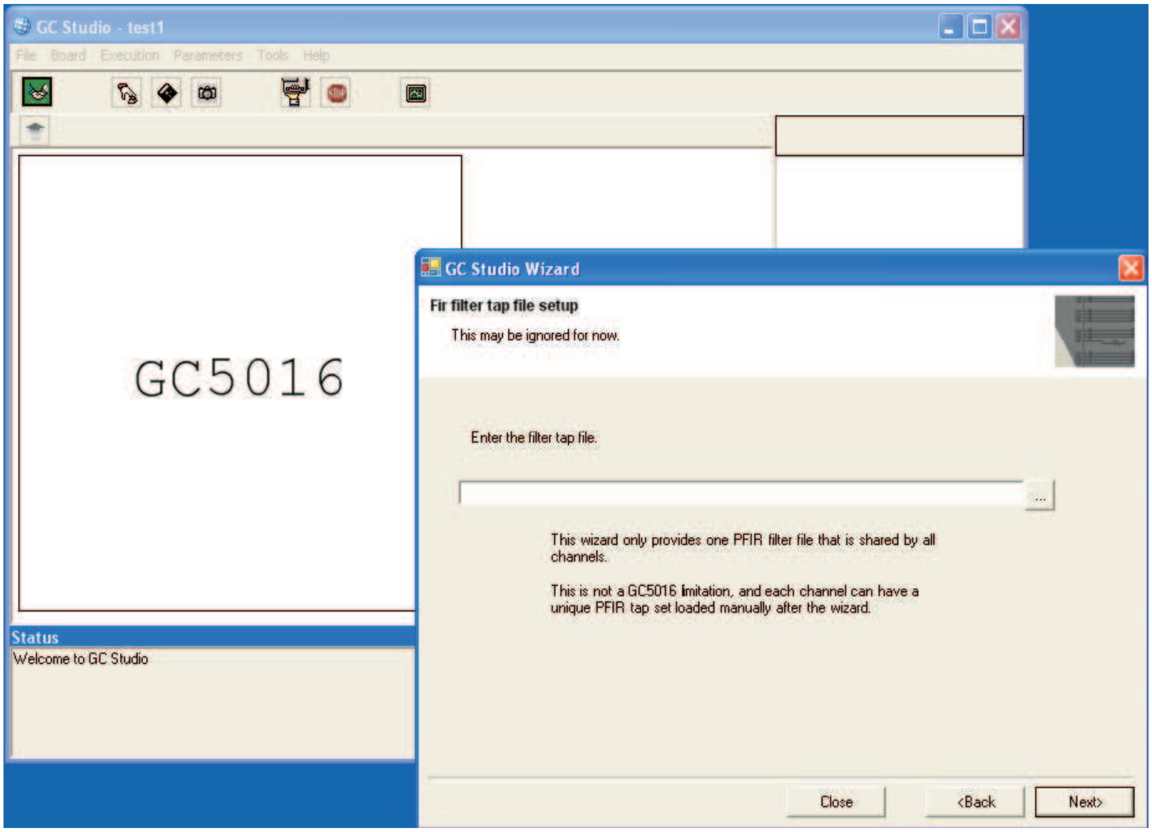


Figure 21. FIR Filter File Path

2.2.5.12 Mixer Frequency Setup

Setup the mixer frequency in this page.

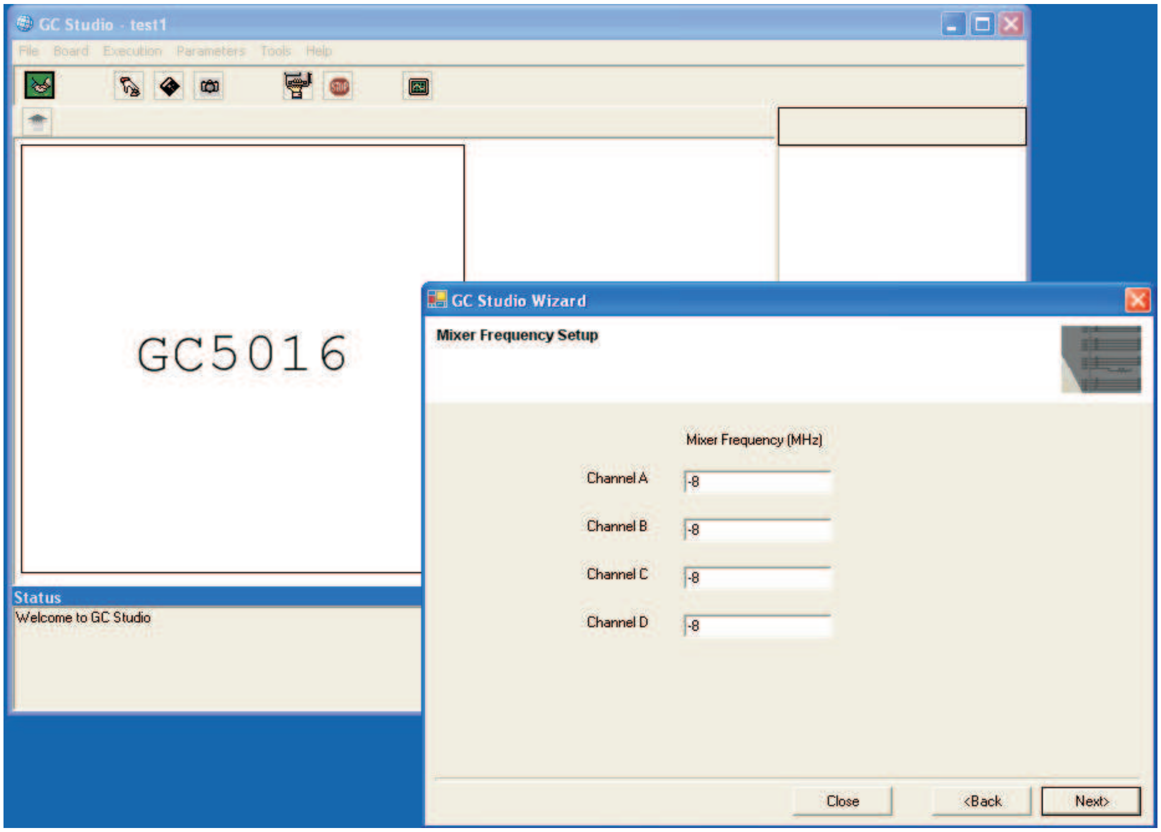


Figure 22. Mixer Frequency Setup

2.2.5.13 DIM Switch Setup

There is a DIM switch (S1) on the GC5016 daughter card with 8 switches. These switches should be set depending on the GC5016 mode. The following page gives a graphic view of the switch setup depending on the mode that user has set in the previous steps. User can set the switches on the daughter card based on this picture.

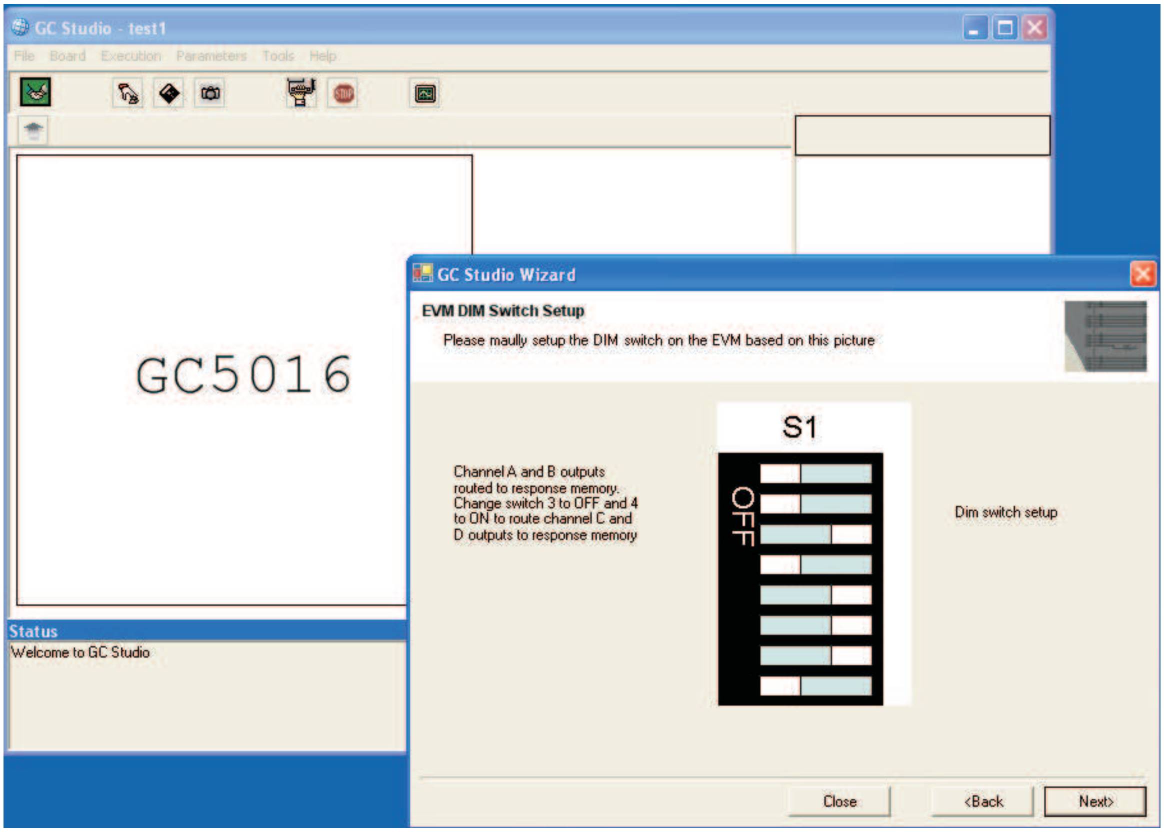


Figure 23. DIM Switch Setup

2.2.5.14 Memory Length

This page shows the baseband file length and the memory length.

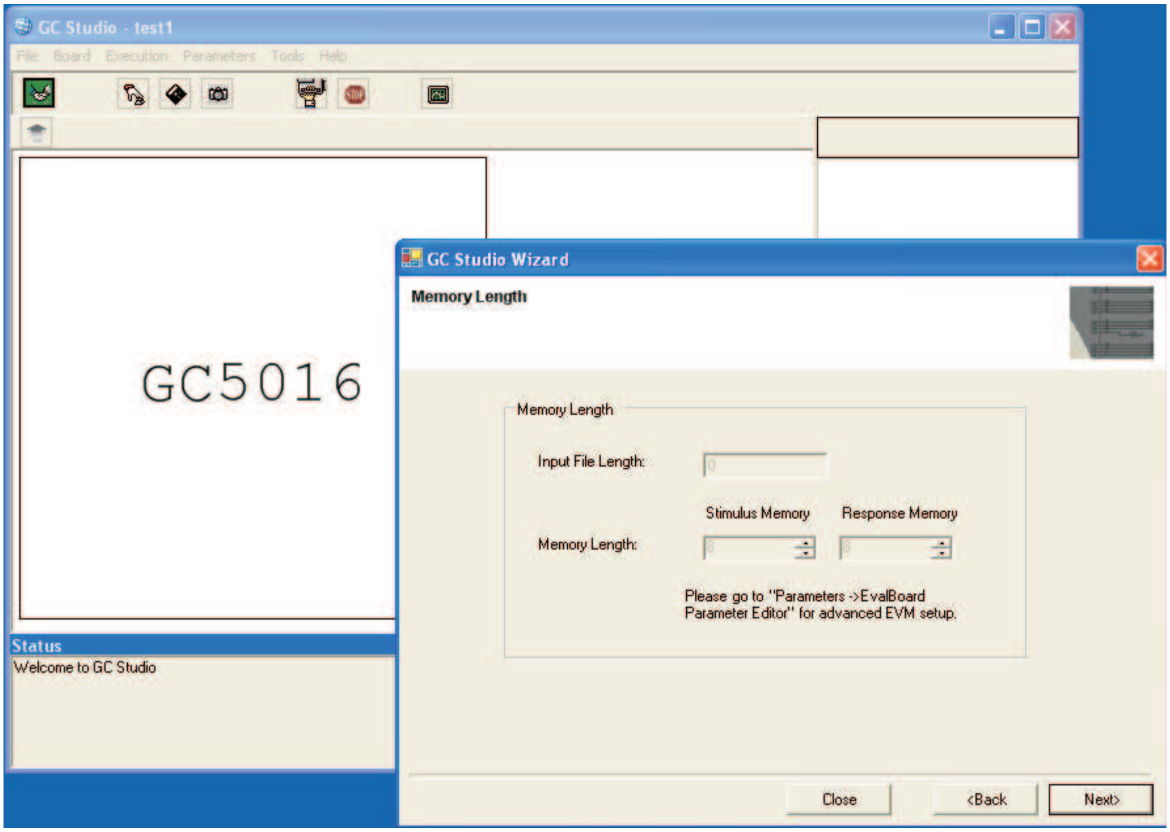


Figure 24. Memory Length

2.2.5.15 Project Descriptions

Users can add experiment descriptions in the text box. The descriptions will be saved as part of the experiment project and can be retrieved anytime later.

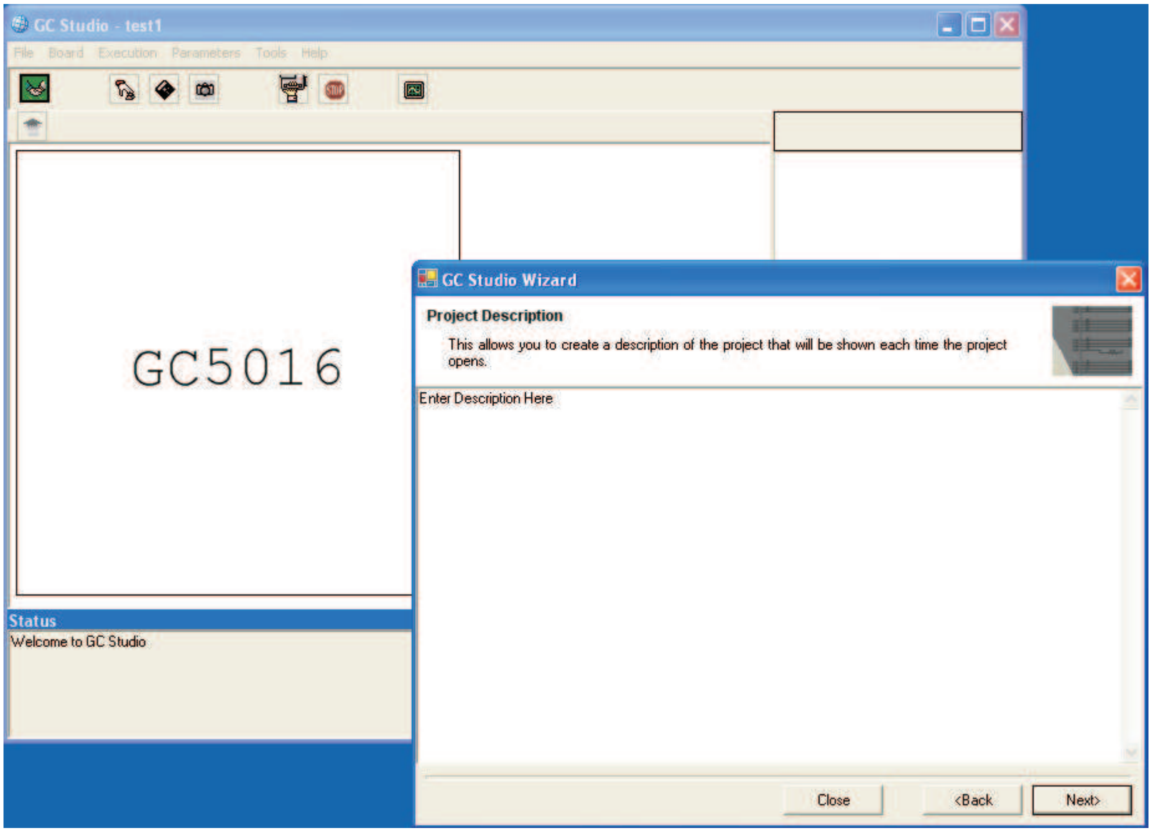


Figure 25. Project Descriptions

2.2.5.16 Project Complete

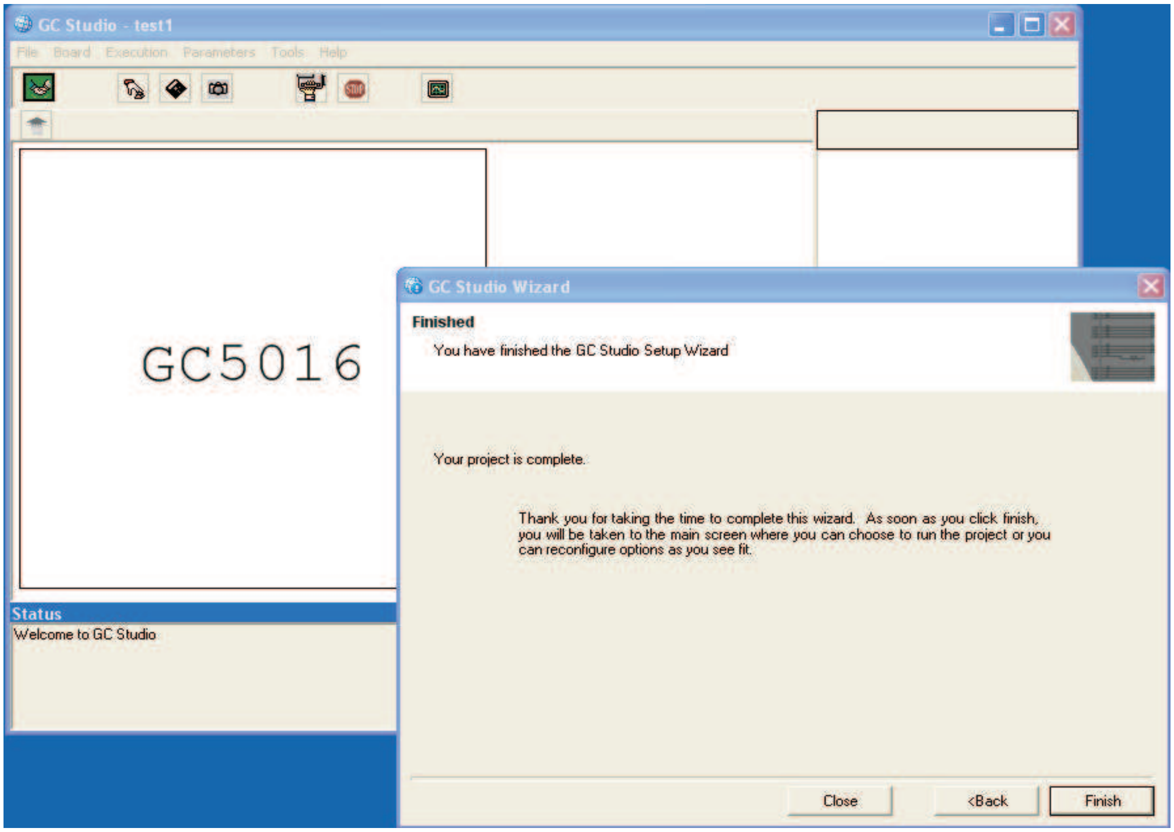


Figure 26. Project Complete

Clicking on “Finish” if the setup is done. Users can click on “Back” if they want to change anything. The following is the view of the new experiment project after “Finish” is clicked.

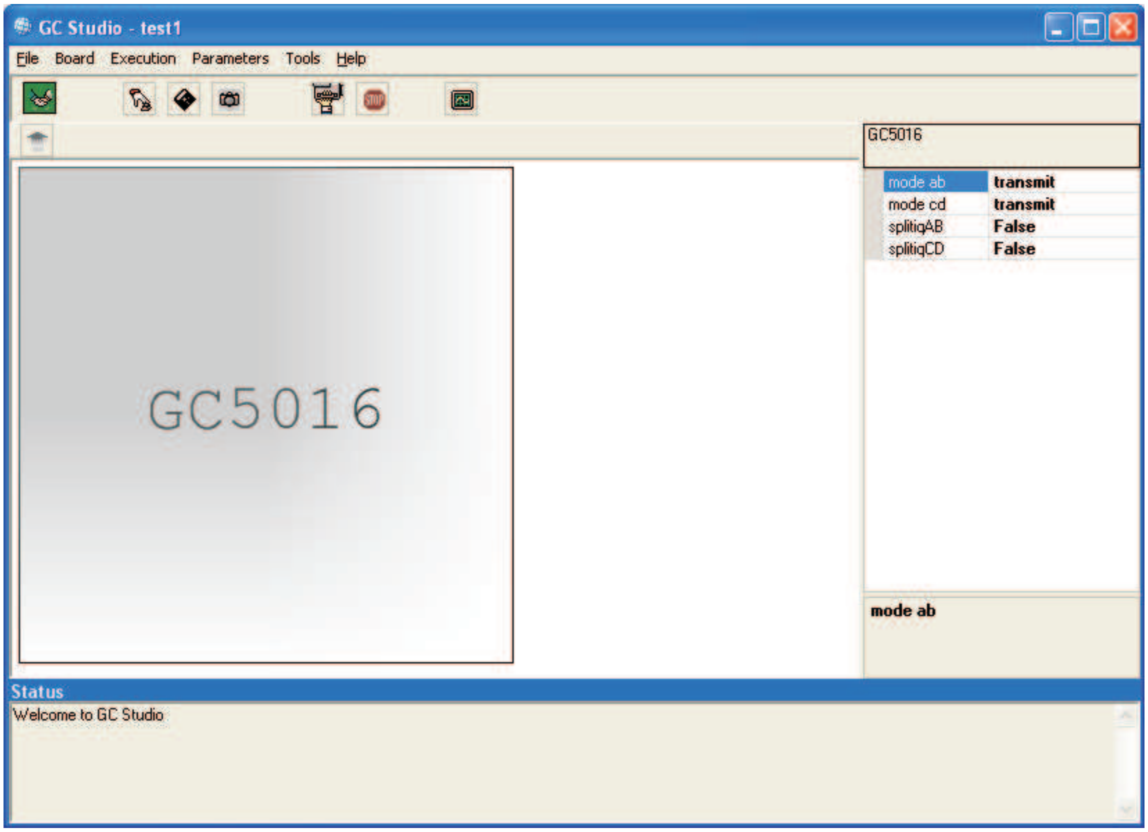


Figure 27. New Project View

2.2.6 GC5016 D.C. Parameters in GC Studio Experiments

2.2.6.1 General Variable Information

When selecting a variable on the Right portion of the GC Studio display, there is additional information at the bottom right of the display. The other references and application notes has additional information. In general there are top level variables, 'pseudo-variables' listed in table 7 of the GC5016 datasheet. These are not register settings but engineering variables for configuration of the GC5016. The Guide for using CMD5016, GC5016 Input Output Mode Application Note, GC5016 Gain and AGC application note and reference experiments should be consulted for additional information. The next level of variables are the fields in each of the GC5016 microprocessor registers. These are further described in the GC5016 datasheet tables 8-62, and can be referenced in the cmd5016 experiments, or block diagrams with variables assigned to their functional blocks:

- TABLE 7 – Engineering Programming Values for GC5016 (use Programming guide)
- TABLES 8-62 – register fields within the GC5016 Microprocessor registers

This release of the GC5016 software uses mostly the table 7 values, and sets the register values for the user. There are two choices for providing more register controls:

1. Using the Tools->Options->Debug you can Save the Device Programming to a file. You can modify the values in the file. Using the same screen, uncheck the Save Device Programming, and check the Use alternate Device Programming Script. This includes the low level register changes desired.
2. The user can override registers using the GC Scripter after running the experiment first. In this case, the user needs to manually capture the Output file after running the script commands. Consult the GC Scripter documentation for writing scripts using the Tools -> Script Editor menu options.

2.2.6.2 Top Panel

Depending on the User controls at the top level, there are different elements displayed. Select the GC Studio panel, and click on the box labeled GC5016.

The figures show the menu screens and variables for the D.C. example. The experiment name is shown in the top frame of the screen as a title.

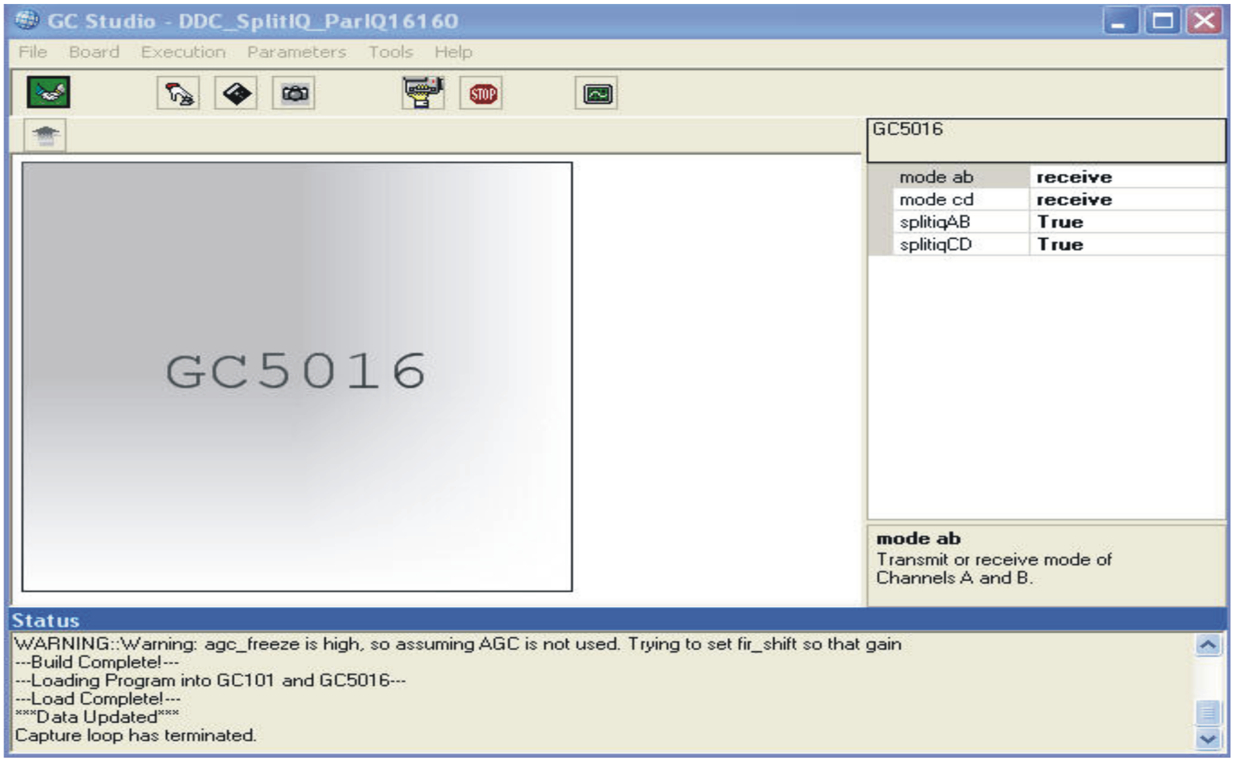


Figure 28. GC Studio GC5016 Top Panel

2.2.6.3 GC5016 Input, Channels, and Output

Selecting the GC5016 box, and double clicking opens the detail view. (See Figure 29) This shows the experiment view of the Input Stimulus, GC5016 programmed for experiment, and Output Monitor memory.

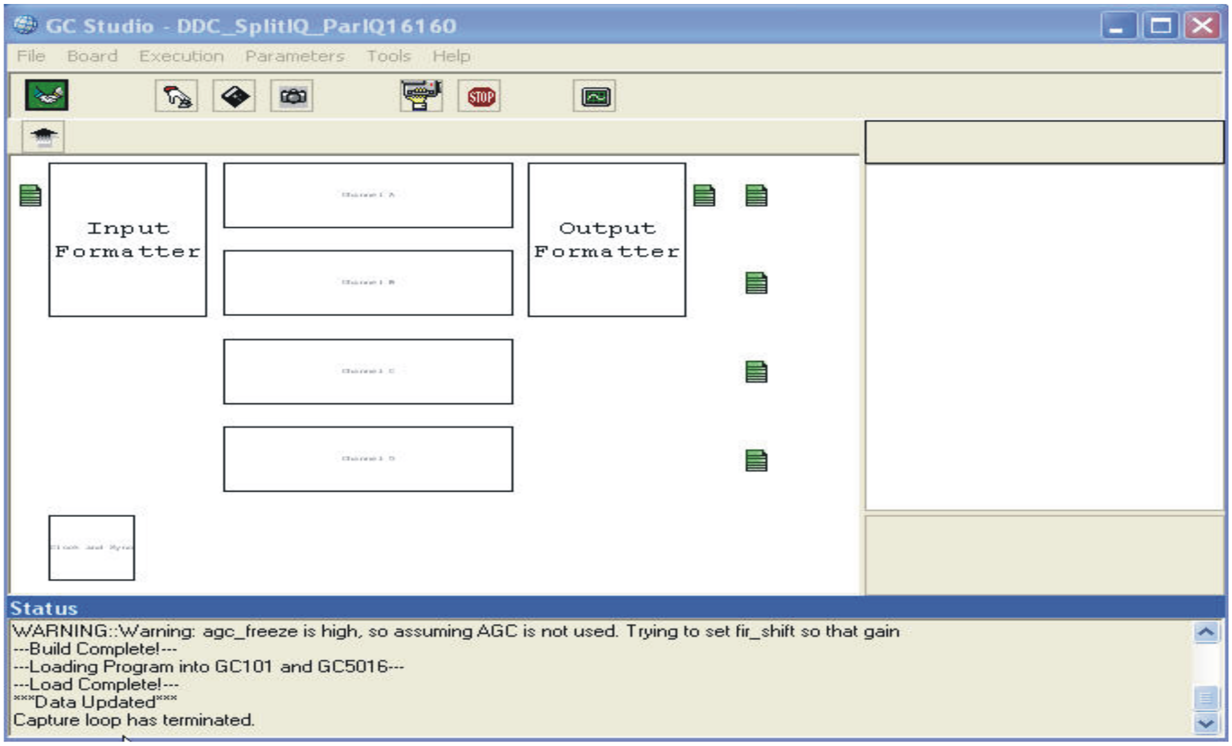


Figure 29. GC Studio Detail Input, GC5016, and Output Panel

2.2.6.4 Input Files

On the left edge, the green boxes are the input files to be loaded into the input memory. There can be from 1 to 4 Green boxes, one per channel. Click on a green box, and on the upper right, the Load From File can be used to enter, or with the ….box you can browse for an input file.

The input file is a decimal column file. A Real input has a single column. A Complex input has two (I and Q) columns.

Once you have a entered a value into a variable, it is tagged as user modified.

[It is good practice to put experiment's data files in with the project folder.]

Real File
Real-Decimal

Complex File
In-phase-Decimal <sp> Quadrature-Decimal

Note: The user would select the LoadFrom File browse box (…) to navigate the file. See Figure 31 for the Navigation button. See Figure 32 for the File Browse Box displayed.

Note: In the lower right corner of the screen, when selecting an object or variable name, context sensitive help is printed in this section.

Note: At the bottom of the display is a Status log. This displays the current status. Errors need to be identified and fixed. If there are warnings, you can check example experiment's JPEG files to see if they are expected warnings.

Note: The Tools->Options->Debug->Use Alternate Stimulus Memory Image – can input a hexadecimal file with no formatting. This is useful if you suspect that the input file conversion is in error. The Dump Contents of Stimulus Memory is the post Input File Conversion hexadecimal file.

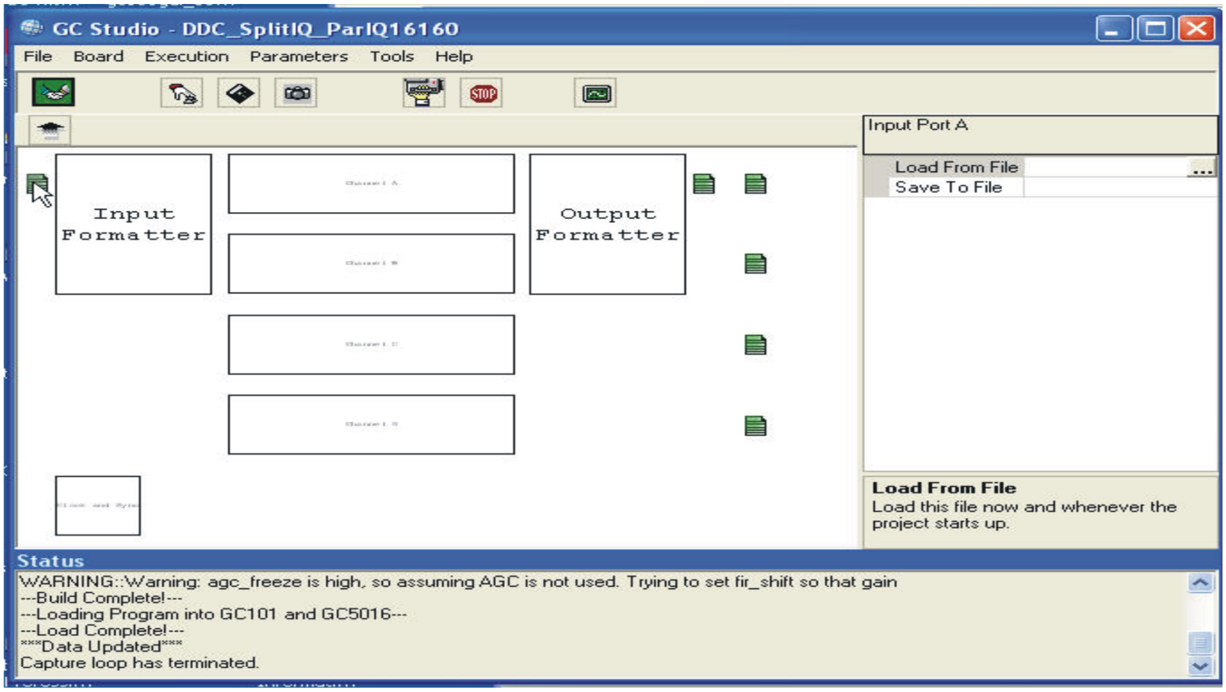


Figure 30. GC Studio Detail Input, GC5016, and Output Panel

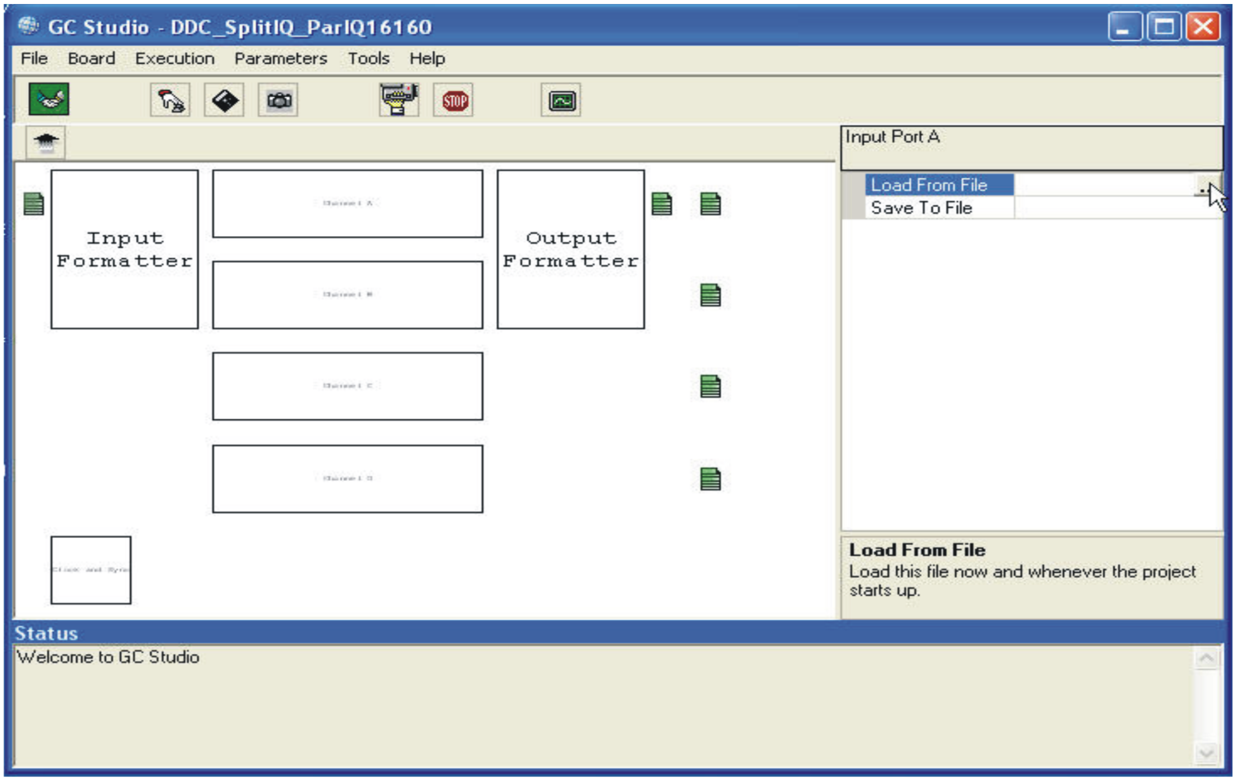


Figure 31. GC Studio File Input, Selecting Load from File

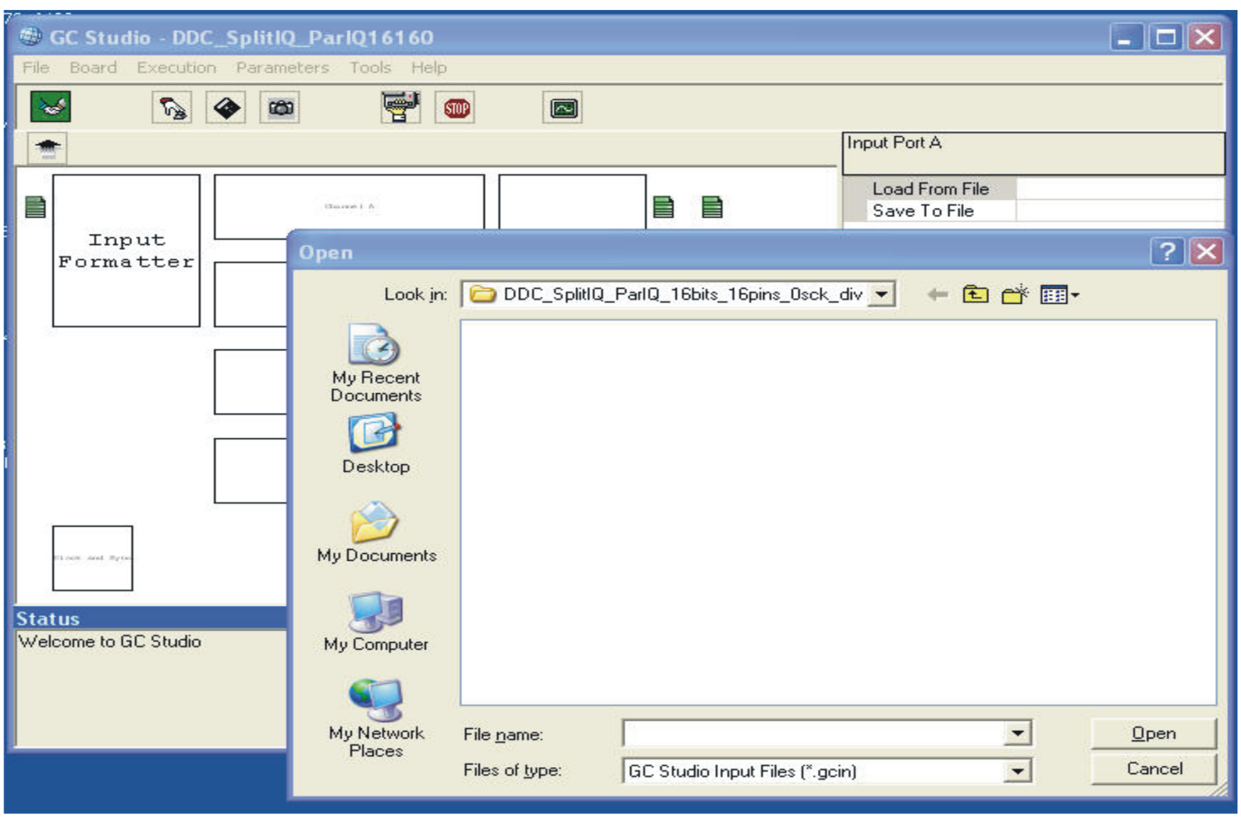


Figure 32. GC Studio File Input, Load from File, Browse Box

2.2.6.5 D.C. Input Formatter

The D.C. Input Formatter block selects a real or complex D.C. input and the rate of that port. Click on the Input Formatter Block to view the parameters that are available to set (See Figure 33). High-lighting each of the controls along the right shows a small help display on the bottom left (See Figure 33).

The GC5016 has individual channel selection for the rin_cmplx & rin_rate. This is simplified on the GC Studio to a general control for all channels.

Real input, rin_cmplx = 0, rin_rate = 1, zpad = 0

Parallel Complex input (two input ports), rin_cmplx = 1, rin_rate = 1. zpad = 0

rate real input, rin_cmplx = 0, rin_rate = 1, zpad = 1, zpad_sync = (1, or 4)

The GC5016 Datasheet and IO Mode Application Note describes all of the options.

Note: The Input Formatter selection for rin_cmplx and rin_rate changes the desired format for the input file.

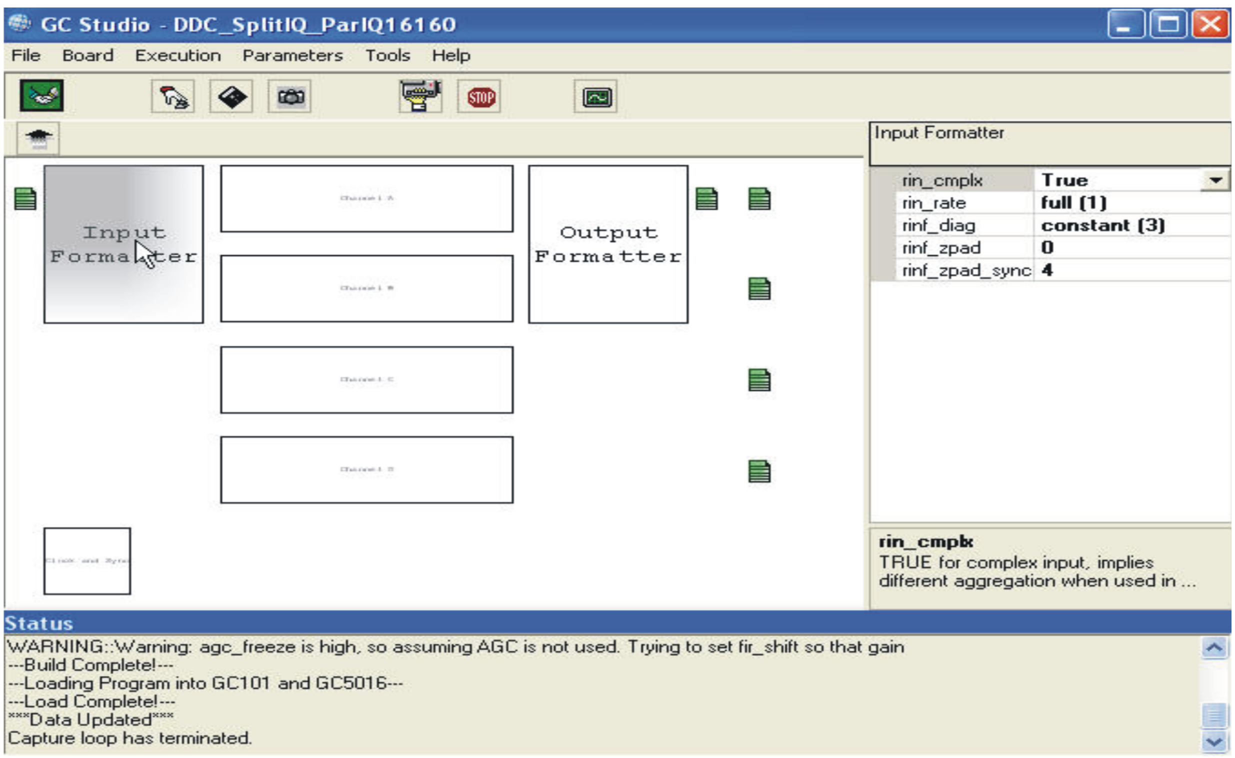


Figure 33. GC Studio D.C. Input Formatter Variables

2.2.6.6 Clock and Sync

The Clock and Sync section has two variables, as shown in Figure 34. The fck is the main Clock rate for the GC5016. The sync_mode is initially used to set the sync_index values for the different sections of the GC5016 sync sections.

The Fck or Ck is the desired clock rate of the GC5016. It does not need to be the same as the GC101 clocking, unless external data input or output is needed.

The common sync_mode values are:

- 0 — used for SIA external or memory synchronization
- 9 — used for One Shot Pulse start

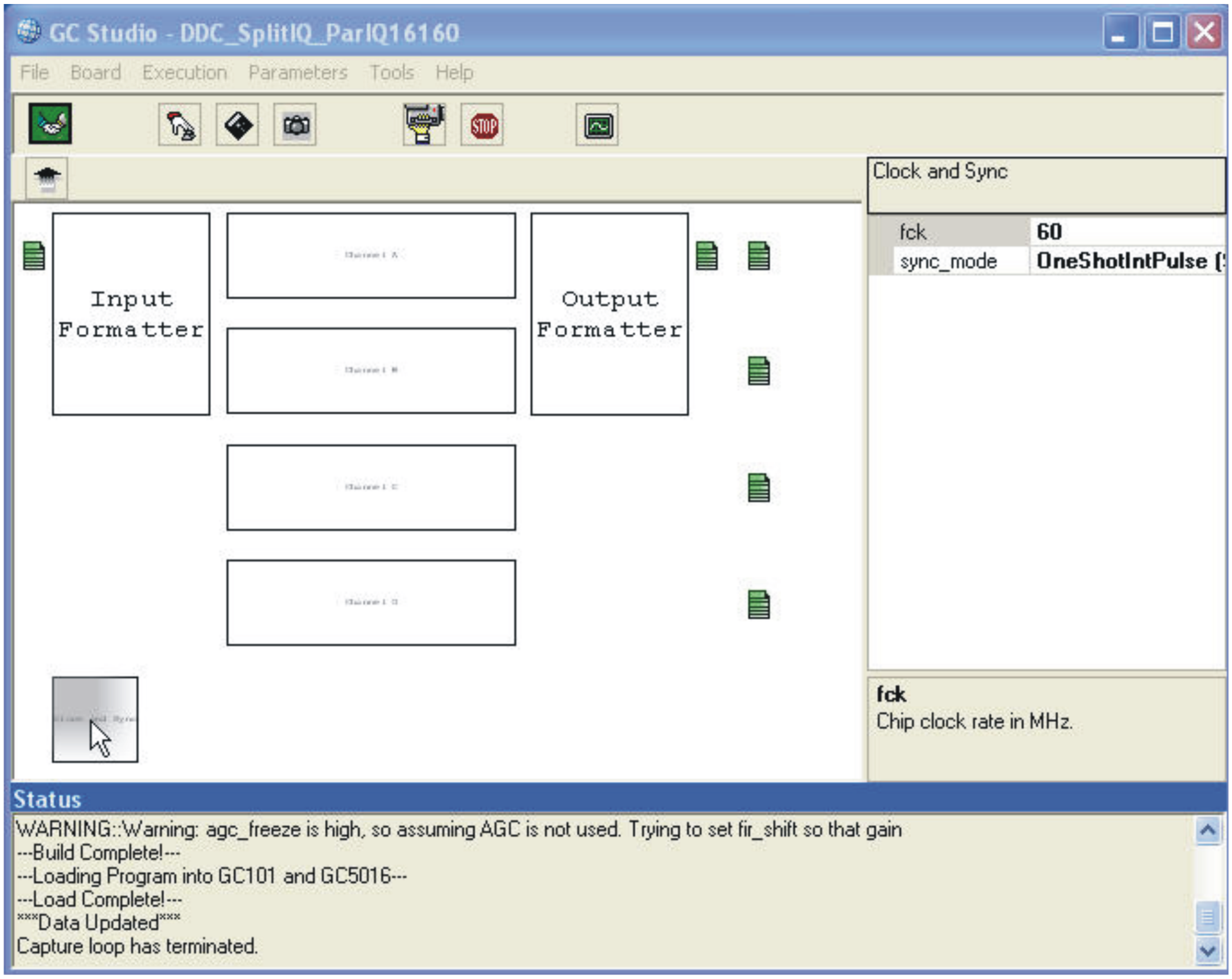


Figure 34. GC Studio Clock and Sync Variables

2.2.6.7 D.C. Channel Selection, Top Level D.C. Channel Block

There are up to 4 D.C. channel blocks. The channel assignment is based on the top level variables, splitiqAB and splitiqCD. The channels are in top to bottom order A..D.

There are 2 channel variables, overall_gain, rinf_sel. Click on one of the channel blocks. The variables are shown in Figure 35.

Overall gain – this follows the D.C. description for the gain equation, in that the value is linear.

This is typically adjusted so the baseband output is the proper signal level for the demodulator.

- Agc_inuse – set to 1.0
- D.C. fixed gain (complex input) – set to 1.0
- D.C. fixed Gain (real input) – set to sqrt(decimation) or 2.0

Rinf_Sel – This variable is shown to the left of the mixer in the block diagram with variables.

- Internal x4000 constant for I and Q – 8
- Real, Parallel Complex – 4), 4(I),1(Q)

Note: Other values for overall_gain are discussed in the gain application note, and the GC5016 datasheet. The Rinf_sel values are normally set through the cmd5016 file as shown in Table 1 of the datasheet.

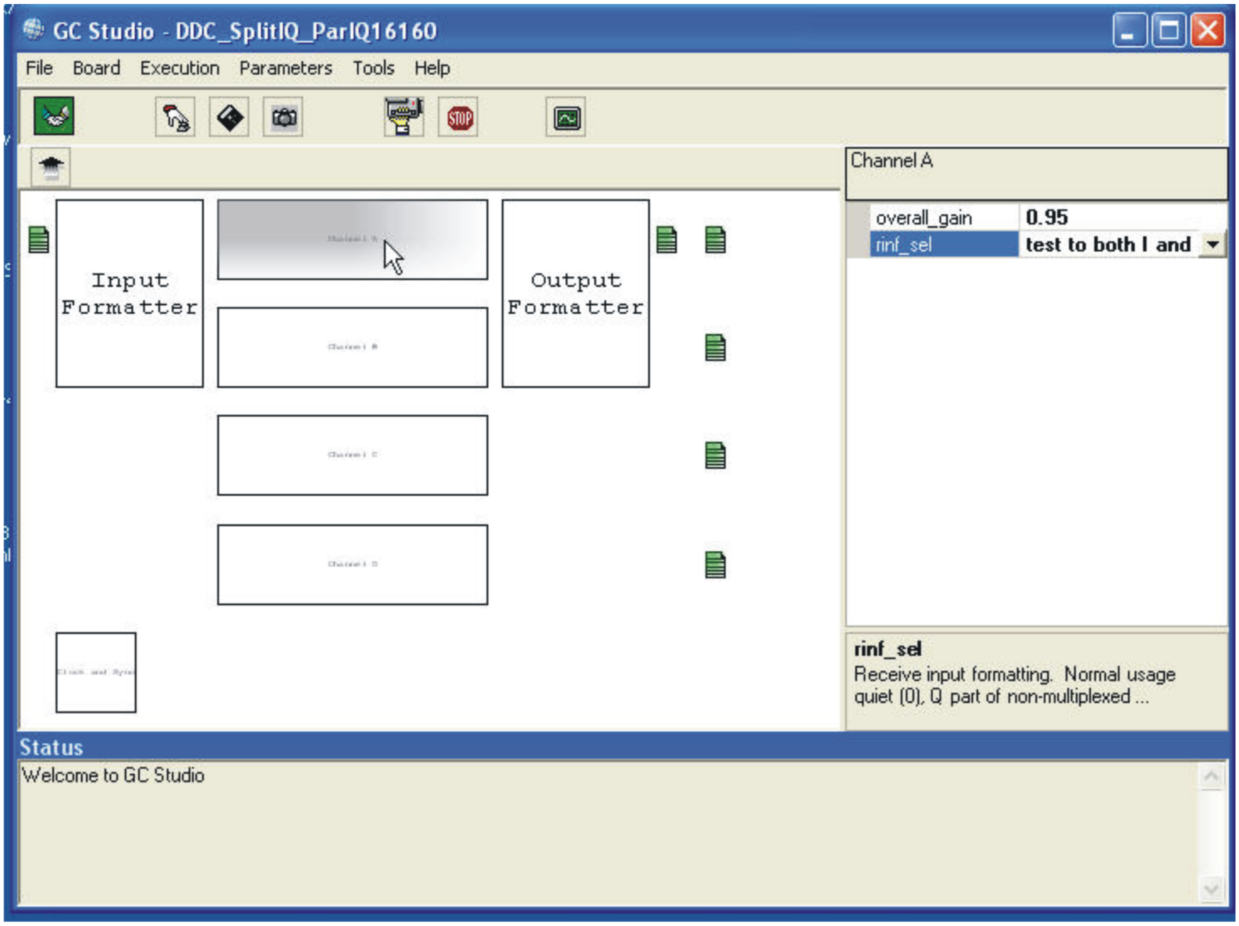


Figure 35. GC Studio D.C. — Channel Select

2.2.6.8 Output Formatter

There is one Output Block depicted on Figure 36 for all of the D.C. channel outputs. The GC5016 Datasheet and IO Mode Application Note describes all of the options. There are several D.C. output modes:

- Interleaved IQ – each D.C. output port works with the same D.C. channel.
- Parallel IQ – only used in splitIQ mode, to output both I and Q data on a pair of output ports
- TDM IQ – all channels at a common decimation are output from the D Output.
- Bits – the number of bits output from 8 to 20 in blocks of 4.
- Pins – the number of output port pins used to transfer data, can be 4, 8, or 16.
- Routf_iqmux – ‘1’ for interleaved IQ output mode
- Routf_tdm – ‘1’ for TDM output on port D
- Sck_div – the number of clock cycles – 1 that each output is presented with Frame Strobe
- Sck_sync – used to synchronize the output block. This should match the pfir_sync source value.

The combination of bits, pins, interface mode (routf_iqmux, routf_tdm), and sck_div identify the D.C. output interface rate and minimum decimation. The Interleaved IQ mode has a minimum decimation of 4. The parallel IQ output mode is used in splitIQ mode only. The TDM mode can be used in 4 channel or splitiq mode.

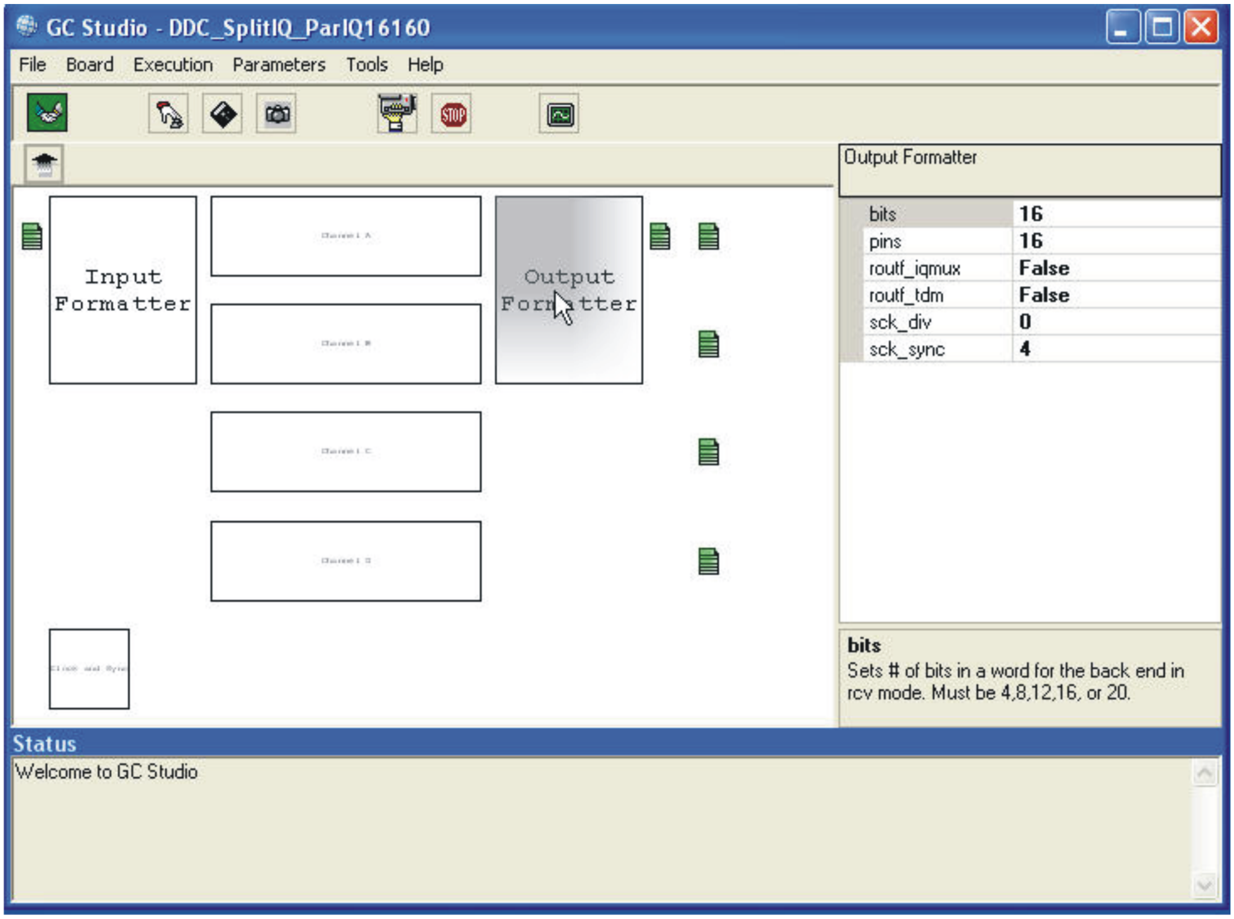


Figure 36. GC Studio D.C. — Output Formatter

2.2.6.9 Save D.C. Output Channel data to File

To the right of the Output Formatter are two columns of green file indicators as shown in Figure 37. The column closest to the Output Formatter is used to store a specific D.C. channel or splitIQ D.C. channels' IQ data to a decimal output file. Depending on the D.C. Output mode, there may be from 1 to 4 outputs in the left column. The right column is for individual sync, strobe, and Iflag bits and is not used. The D.C. channel output file is saved in the two column decimal complex format as I and Q (See Figure 37).

Note: The Status Log displays if a file is to be logged.

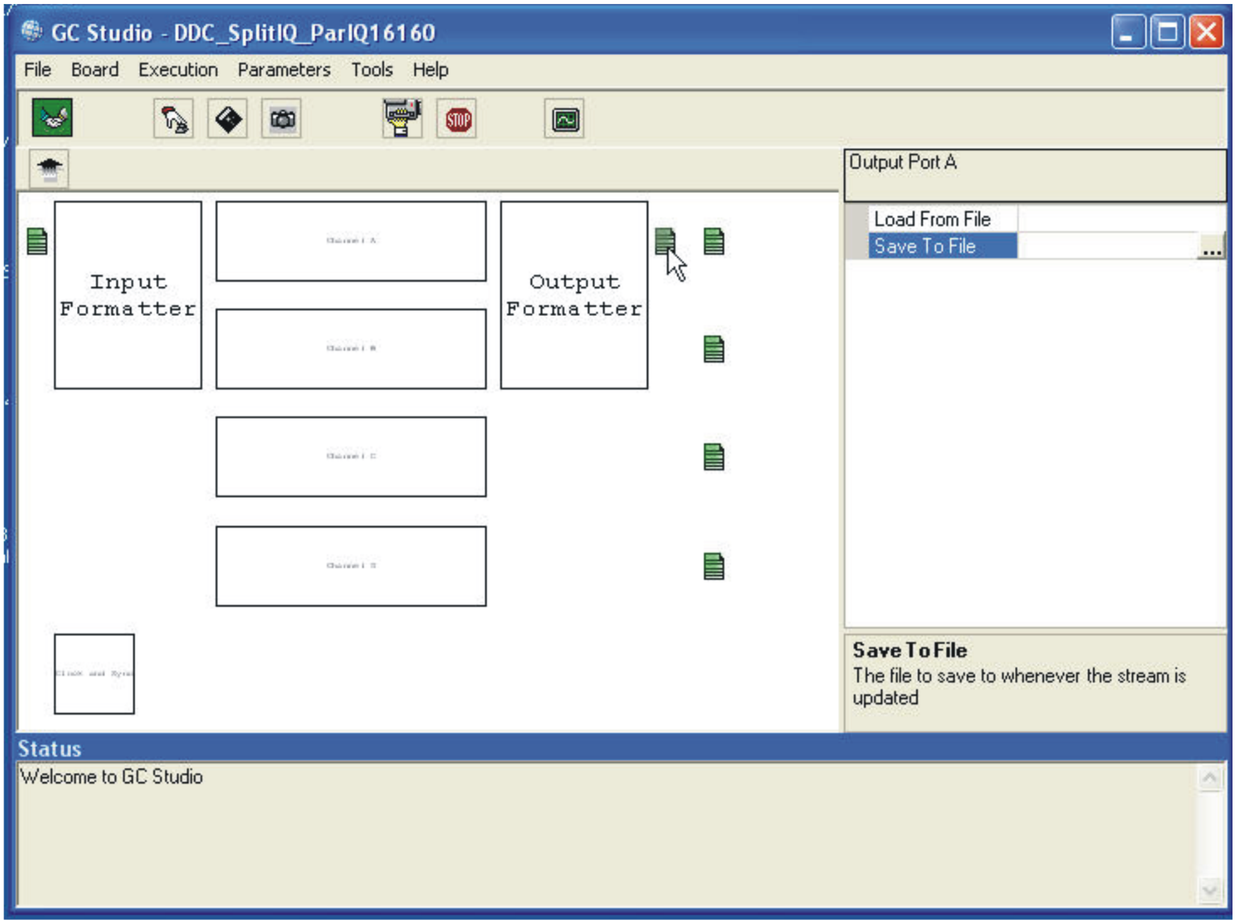


Figure 37. GC Studio D.C. — Saving the DDC Output

2.2.6.10 D.C. Channel Display Screen

Double click on a channel block. The D.C. Channel Display should be shown in Figure 38.

There are no user controls at this screen. Programming the detailed sections involves selecting the Box, and clicking. This accesses the channel specific Complex Mixer, CIC, PFIR, or Gain functions. Use the Up Arrow in the upper left corner to go back to the GC5016 view.

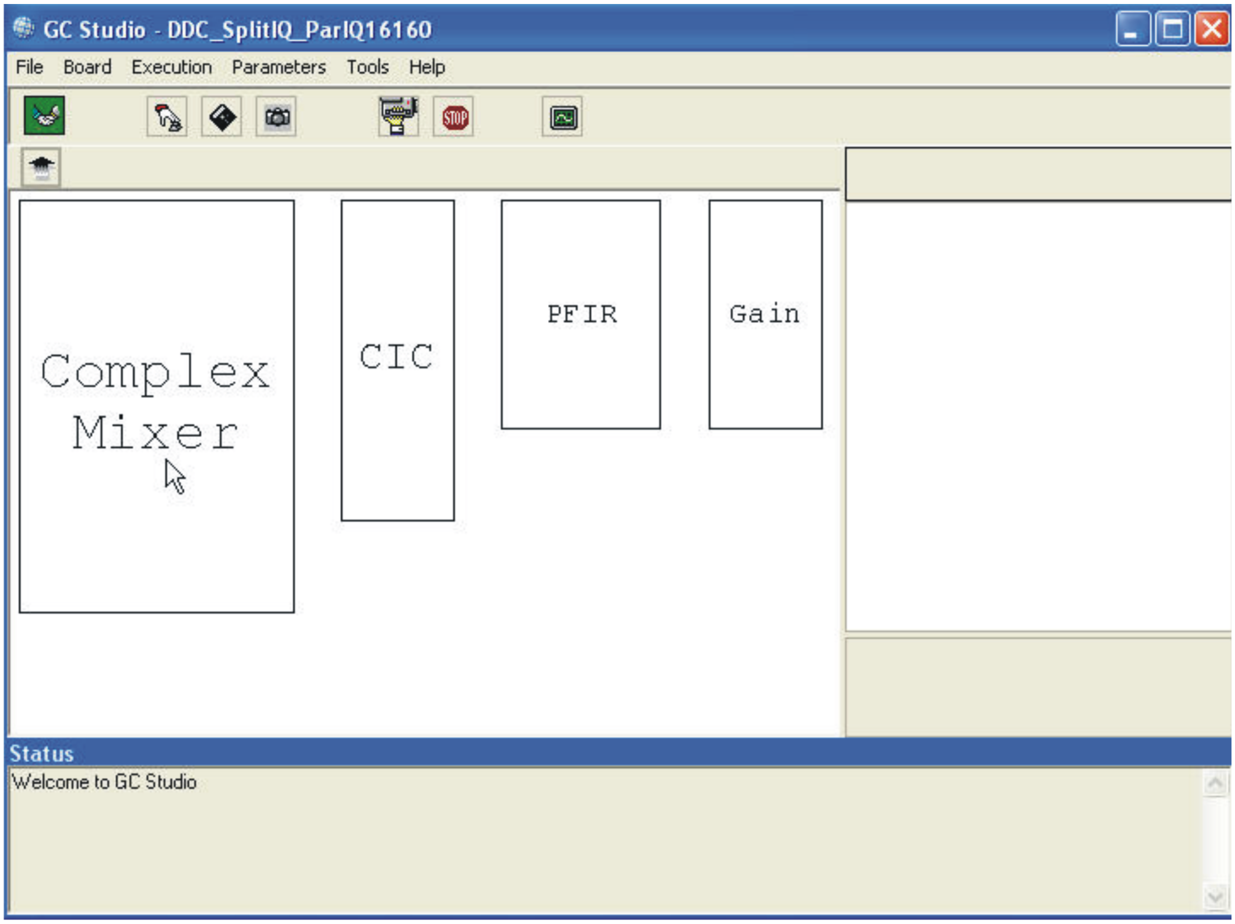


Figure 38. GC Studio D.C. — Detailed Channel Block View

2.2.6.11 Complex Mixer

The D.C. Complex Mixer is selected in Figure 39. This section is used to select the Complex Input Bus from the D.C. input, bypass the mixer, and control the down-conversion frequency. The sync selections can be set to 0, 4, or an external synchronization value.

- Bypass_Mix – normally set to false, is used for D.C. input testing.
- Dith_sync – controls the dither phase value; it is set to the startup sync source, or external sync. If no dithering is desired, set this to 7.
- Freq – The mixer frequency in MHz. It is set to a negative frequency for odd Nyquist zones to translate a positive IF frequency to 0. It is set to a positive frequency for even Nyquist zones to translate a negative IF frequency and flip the spectrum.
- Mix_rcv_sel – This selects one of 4 D.C. Complex input busses to be used for this D.C. channel. The selector has path a,b,c,d. These paths correspond to the D.C. Input Port or diagnostic input, selected in the GC5016 Channel screen.
- Nco_sync – This control is used to reset the NCO phase accumulator. This causes a phase disturbance, and is normally done with the startup sync source.
- Phase – This sets the phase offset of the tuning frequency.

When this section's variables have been updated, you can click on another section, or use the Up arrow to go back to the GC5016 view screen.

Note: Fout = Fin + Ftuned; this is the equation to translate frequency on the D.C. Mixer.

Note: For Diagnostic source – this example, the NCO frequency must be set within the complex D.C. output filter bandwidth (Fck × k(<1))/(cic_dec × fir_dec × 2)

Note: nco_sync and dith_sync are set to external synchronization if repeatable and deterministic phase outputs are required. The value of 4 for One shot, can be used to properly initialize, but the relationship of the absolute phase of the accumulator is not known.

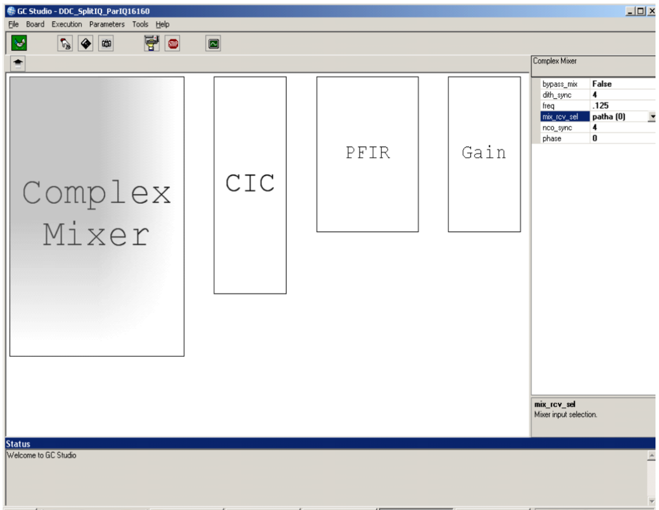


Figure 39. GC Studio D.C. — Detailed Complex Mixer

2.2.6.12 D.C. CIC Section

The CIC (Cascade Integrator Comb) is the first stage decimation filter in the GC5016. Click on the CIC block to view the parameters that are available to set as shown in Figure 40. The CIC is a 5 stage CIC in the D.C. mode. In the four channel configuration, the CIC must decimate by at least 2. In the splitIQ configuration, setting cic_dec = 1 is used to bypass the CIC effectively. The cic_sync controls the sync value to initialize the decimation counter (cic_decimation phase). The flush_sync is used to zero the CIC accumulator.

- Cic_dec – cic_decimation value, normally 2 – 222. The value can be 1 in splitIQ mode.
- Cic_sync – sync value source for the cic decimation counter. Normally set to the startup sync or external sync source. For One Shot, set to 4, for SIA, set to 2, for SIB, set to 3.
- Flush_sync – sync value source for the cic_accumulator zero. Normally set to the startup sync or external sync source. For One Shot set to 4, for SIA, set to 2, for SIB, set to 3.

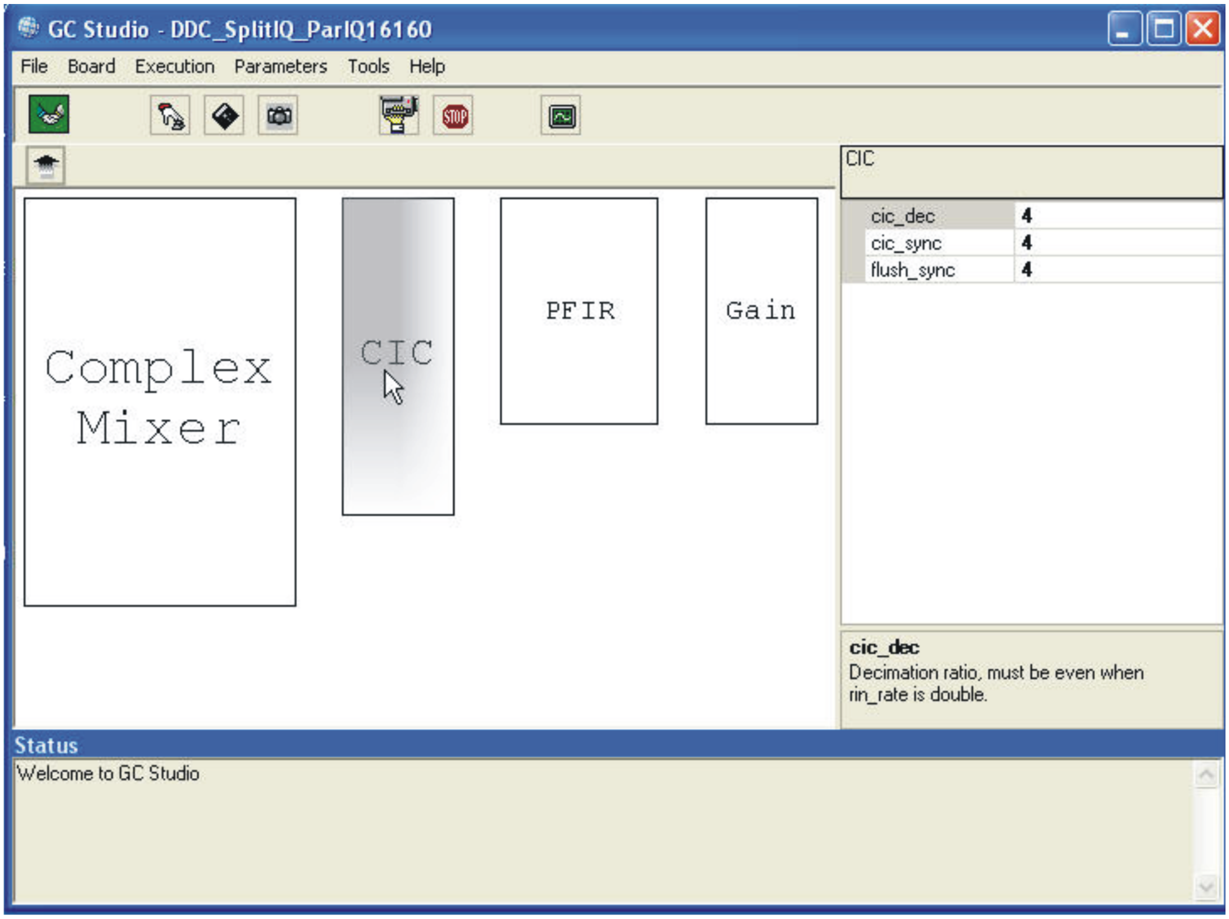


Figure 40. GC Studio D.C. — Detailed CIC

2.2.6.13 D.C. PFIR (Programmable Finite Impulse Response Decimating Filter)

The GC5016 PFIR has several internal sections, Memory, 16 multiply accumulate sections, a sum chain, forward and reverse delay lines, Address Generators, and State Machine Controls. These are programmed through cmd5016. The higher level commands that are used to program the PFIR are shown in Figure 41. There is a general equation for the number of PFIR taps in the GC5016 datasheet. The experiment folder has a .ANL file that can be viewed with a text editor. Near the bottom is the information about the number of PFIR taps allowed.

The number of PFIR taps can be initially programmed as a 31 tap, center tap 32767, all other taps 0. The .ANL file can be checked and then the final PFIR taps can be generated. There is a matlab tool (available from your Field Application Engineer) for generating the PFIR filter coefficients. Manually generating a PFIR filter, there are 3 taps for the CIC correction [-.15,1,-.15] that are convolved with the user filter. If the cic_dec is 1, the CIC correction is not needed.

- Bypass_Fir – this variable is normally set to false.
- Fir_cmplx – this is set to 'TRUE' in 4 channel mode, it is set to 'FALSE' in splitIQ mode.
- Fir_coef – this variable allows a view of the PFIR coefficients, or a method to load the coefficients from a single column of 16-bit 2's complement coefficients. (See Figure 42 and Figure 43)
- Fir_dec – this is the decimation counter value + 1. This determines the PFIR_dec.
- Fir_diff – set to false.
- Fir_nchan – set to 1.
- Fir_Sync – Sync source for PFIR. Normally set to 4.

Note: The PFIR coefficient design can be a Low pass filter, real coefficients, or a pulse-shaping filter. The fir_dec is based on the desired output over-sampling rate from the DDC. Typically if the pfir_output rate is 2 samples per symbol or more, the fir_dec can be 2 or greater. If the pfir_output rate is 1 sample per symbol, the fir_dec is 3 or higher.

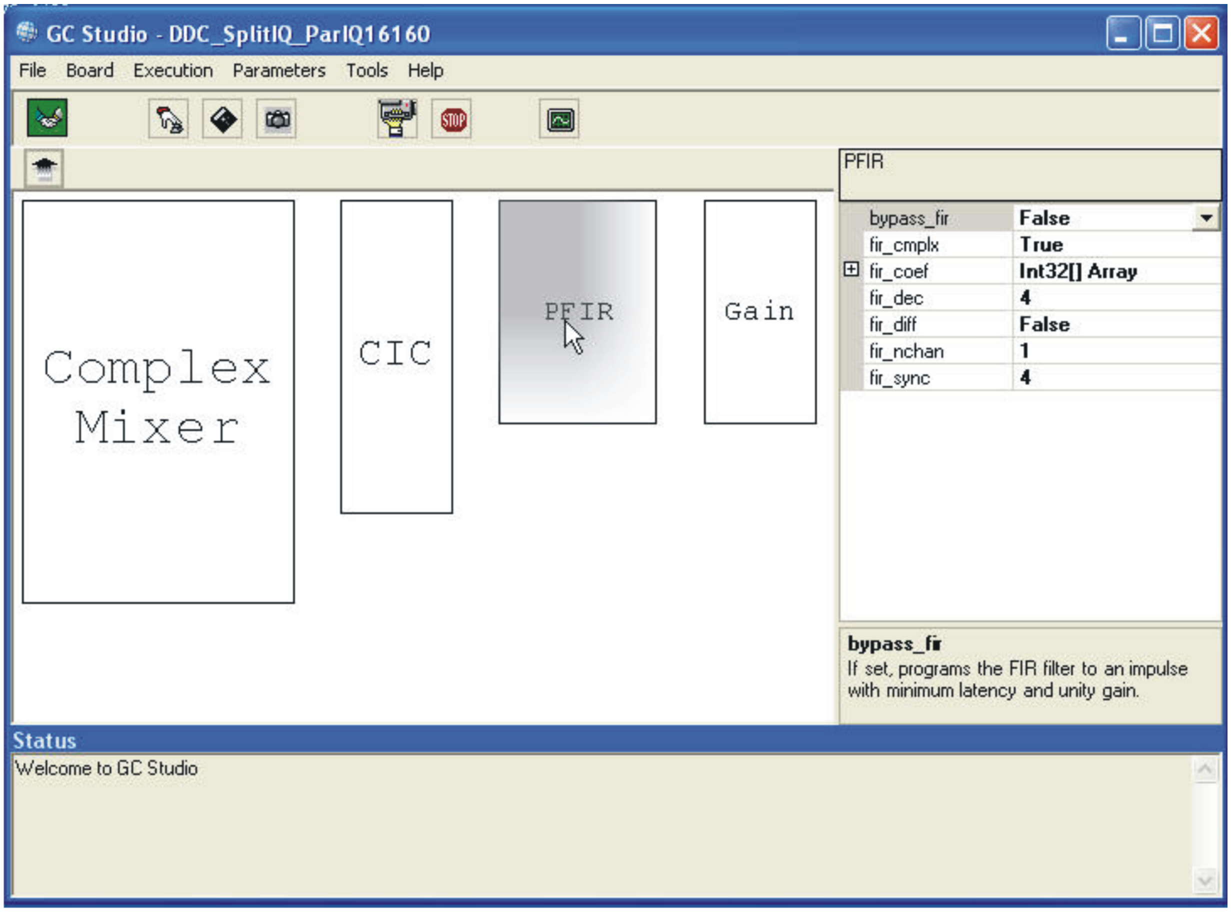


Figure 41. GC Studio DDC — Detailed PFIR

2.2.6.14 DDC PFIR Selecting Coefficients

This screen is selected from clicking on the fir_coef variable on the right side as shown in Figure 42.

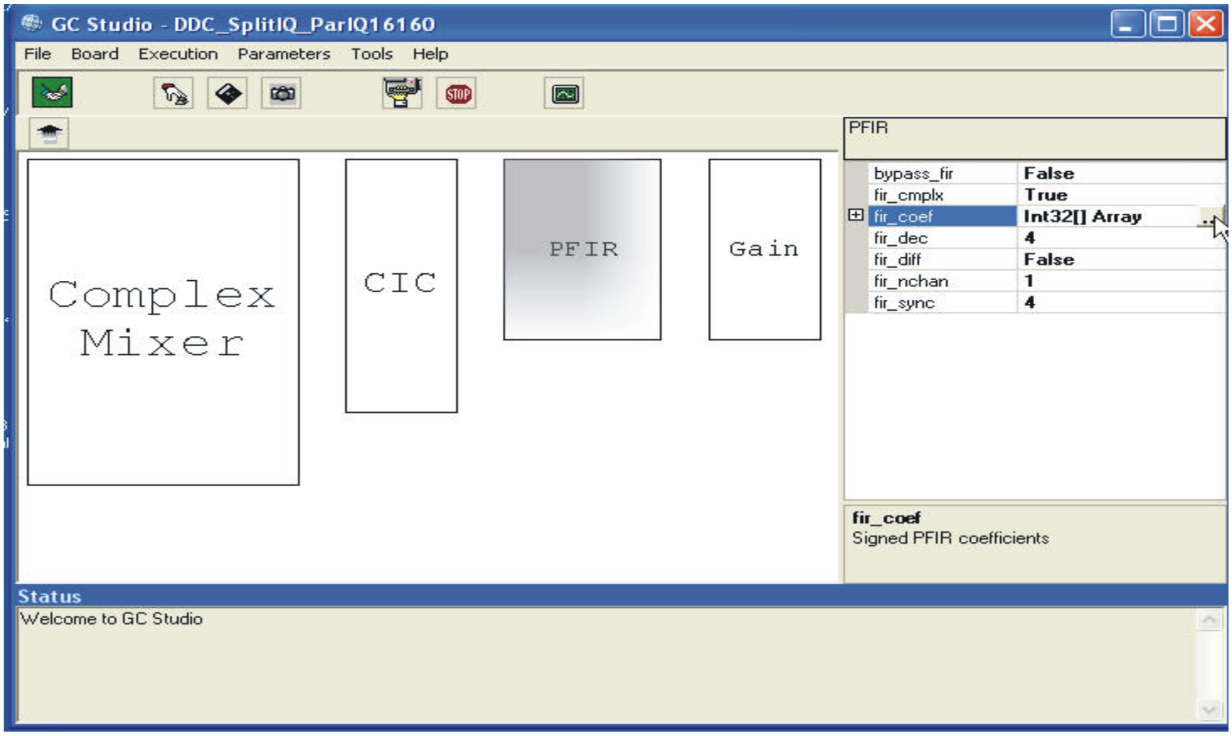


Figure 42. GC Studio DDC — Selecting PFIR Coefficients

2.2.6.15 DDC PFIR Coefficient Entry

The PFIR coefficients are viewed as an Impulse Response (i.e. Time Series) as shown in Figure 43. It is important to check that the scaling fits the 16 bit range -32768 to 32767. The number of taps needs to match the desired PFIR configuration mode (i.e. check the .ANL file).

There are several methods to enter coefficients:

a. Paste the values into the value list on the left of the coefficient screen.

b. Use the File button above the Values title, and open the File Browse box (see Figure 44).

Figure 45 shows the File Browse Box. The filter tap file can be selected, and then the user clicks on OK to load the coefficients. Check the Integer plot to verify the proper amplitude and number of taps.

The coefficients are then Applied (click Apply), and then click OK to return to the DDC PFIR Configuration screen.

The fir_shift applies a gain scaling at the PFIR output. The .ANL and .TBL files list the value. In this version it isn't set manually.

Note: It is important to depress Apply and OK to ensure that the loaded coefficients are used to program the GC5016. It is important that tap amplitudes are within 16bit signed decimal range.

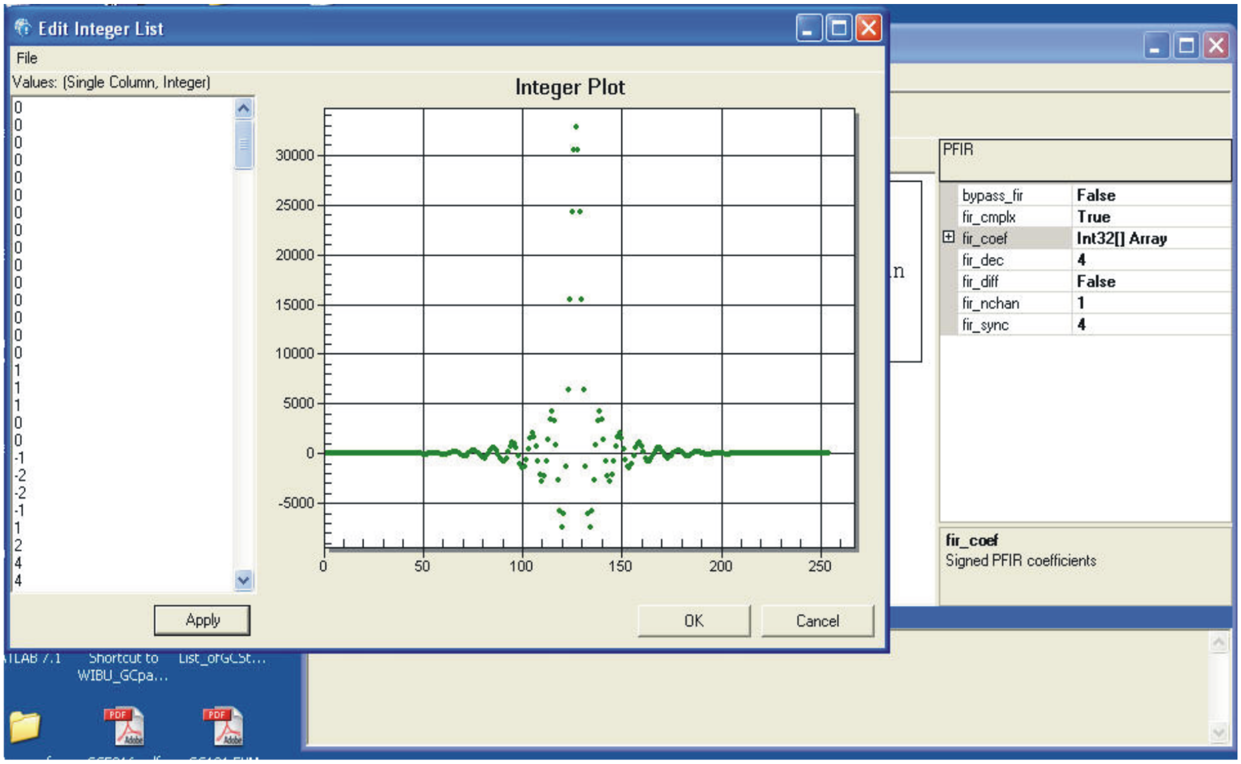


Figure 43. GC Studio DDC — PFIR Coefficient Entry

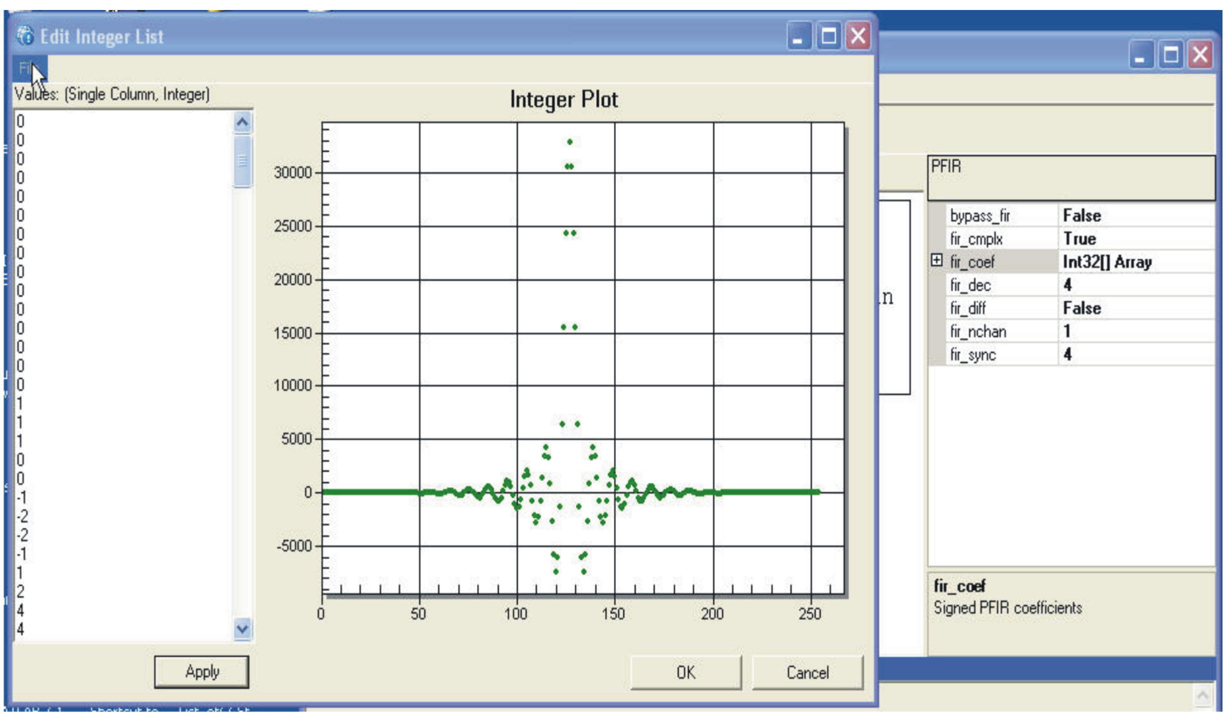


Figure 44. GC Studio DDC — PFIR Select Coefficient File

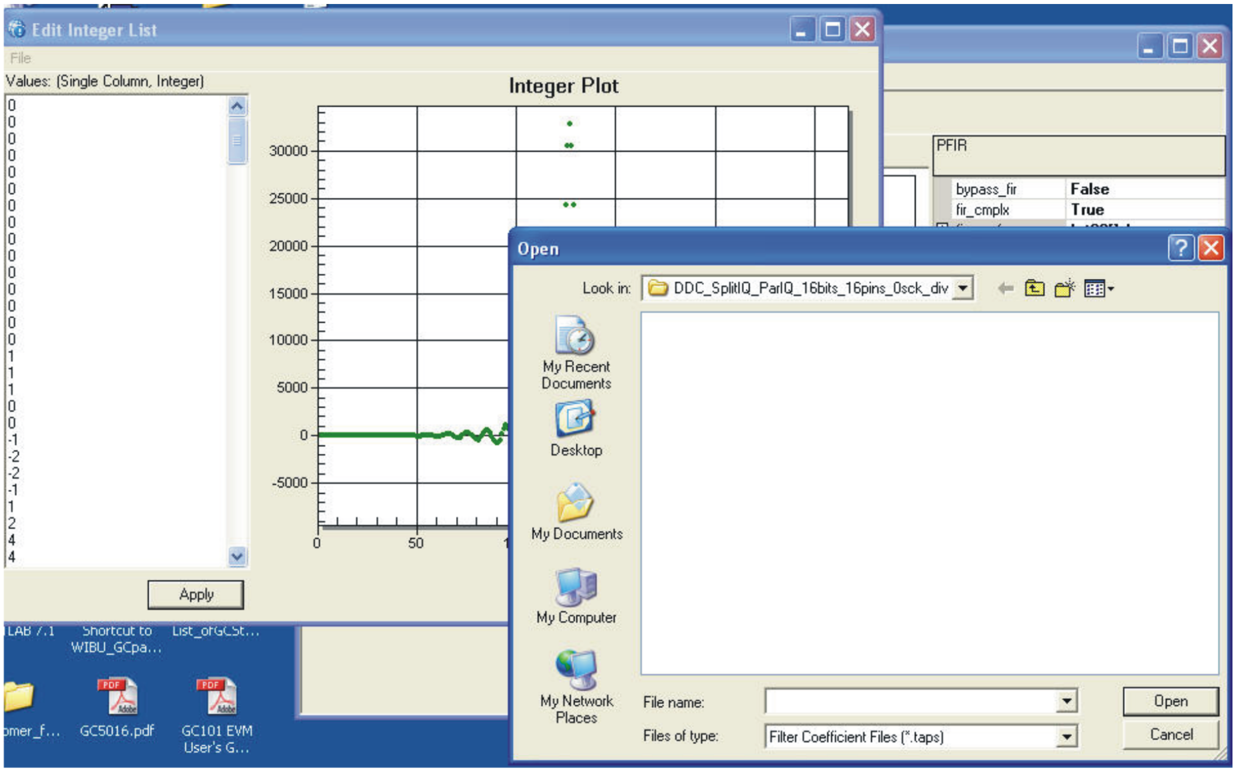


Figure 45. GC Studio PFIR Coefficients Select from Files

2.2.6.16 DDC Gain and AGC Commands

The GC5016 can be operated in manual gain, manual gain with synchronization, or AGC modes. The manual gain applies the gain value calculated from the overall_gain (gain_lsb and gain_msb) with the PFIR output.

- Manual Gain – gain_sync = 7.
- Manual Gain with synchronization – gain_sync set to sync source, agc_freeze = true.
- AGC – agc_cf set to signal crest factor (10), agc_freeze = false, agc_mode = 2, agc_tc = time constant desired in µsec.

The AGC has a threshold value. The variables agc_thresh is the set point for the upper 8 bits of I or Q (exclusively I or Q for splitIQ) comparison. The current gain value is incremented by a gain fraction if the input data is below threshold up to an upper limit. The current gain value is decremented by a gain fraction if the input data is above threshold down to a lower limit. The time to correct a value with increasing or decreasing gain are the agc_Dblw or agc_Dabv values.

Engineering values agc_mode, agc_cf, and agc_tc, which cmd5016 uses to set the lower level values, are shown in Table 7 of the Data Sheet. Click the Gain block. The following parameters appear as shown in Figure 46.

- Agc_cf – signal crest factor, also used to adjust final signal level.
- Agc_freeze – signal is true to stop agc gain adaption.
- Agc_hold – used to hold the current agc value to allow a microprocessor to read agc value.
- Agc_mode – different settings for lower level agc commands. See "AGC Circuit Functional Description and Application Note" for more information.
- Agc_tc – attack and decay time for settings gain fraction for accumulator update.
- Gain_rnd – used to select DDC output bits = 20 – gain_rnd.
- Gain_sync – set to 7 for manual gain, set to sync source for manual gain with sync. or AGC.
- Pwr_mtr_on – set to 1 for power meter. Currently the other power meter variables are not set.

Note: The DDC Gain and AGC application note are needed to set the AGC parameters. The manual gain (gain_lsb and gain_msb) can be read from the ,TBL or .ANL files after the experiment is run.

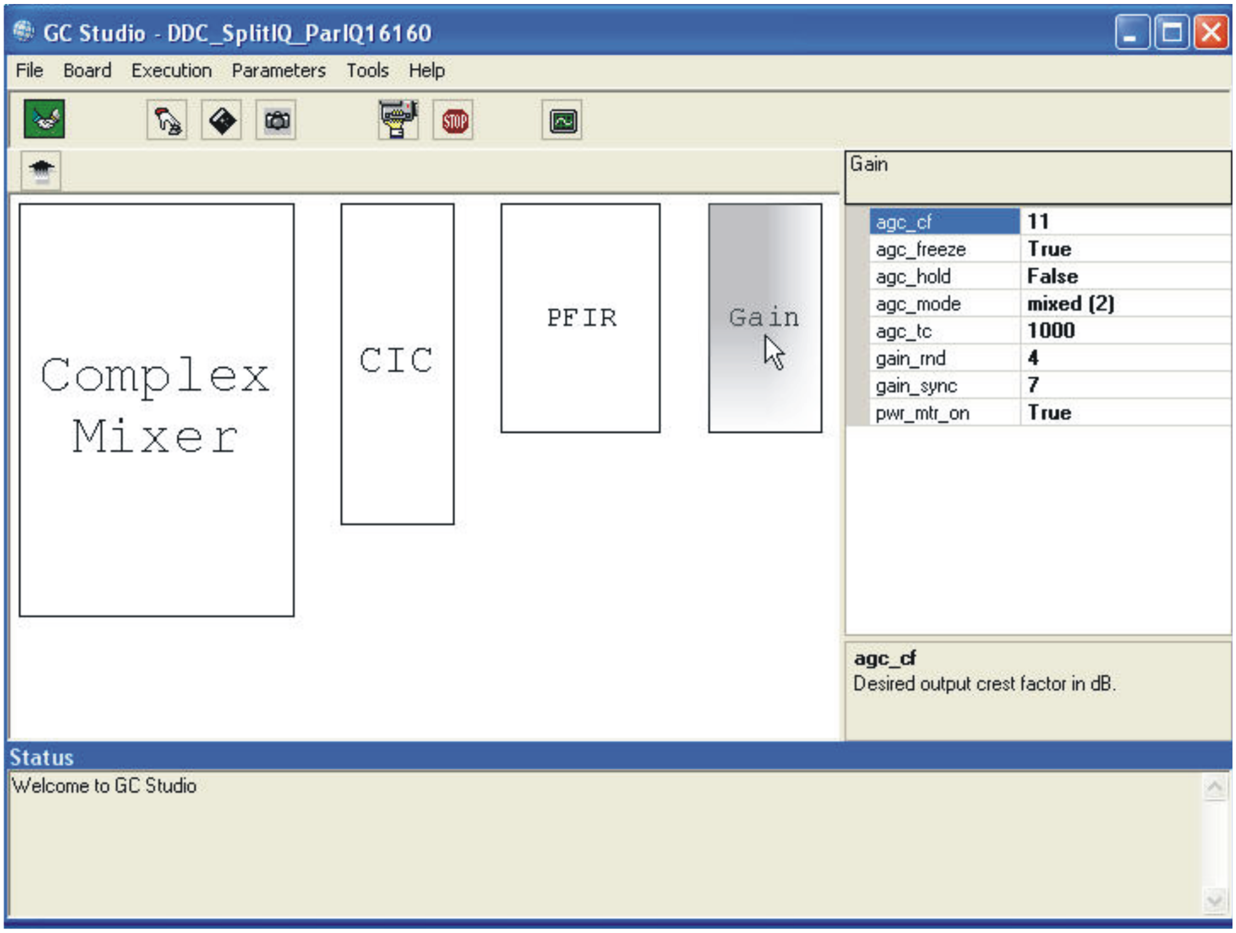


Figure 46. GC Studio DDC Gain Display

On the top menu, click on the parameter "Channel Copy". This can be used to copy the designed channel A, to B, C, and D as appropriate.

2.2.7 GC5016 DUC Parameters in GC Studio Experiments

2.2.7.1 General Variable Information

When selecting a variable on the Right portion of the GC Studio display, there is additional information at the bottom right of the display. The other references and application notes has additional information. In general there are top level variables, ‘pseudo-variables’ listed in table 7 of the GC5016 datasheet. These are not register settings but engineering variables for configuration of the GC5016. The Guide for CMD5016, GC5016 IO Mode Application Note, GC5016 Gain and AGC application note and reference experiments should be consulted for additional information. The next level of variables are the fields in each of the GC5016 microprocessor registers. These are further described in the GC5016 datasheet tables 8-62, and can be referenced in the cmd5016 experiments, or block diagrams with variables assigned to their functional blocks:

- TABLE 7 – Engineering Programming Values for GC5016 (use Programming guide)
- TABLES 8-62 – register fields within the GC5016 Microprocessor registers

This release of the GC5016 software uses mostly the table 7 values, and sets the register values for the user. There are two choices for providing more register controls:

a. Using the Tools->Options->Debug you can Save the Device Programming to a file. You can modify the values in the file. Using the same screen, uncheck the Save Device Programming, and check the Use alternate Device Programming Script. This includes the low level register changes desired.
b. The user can override registers using the GC Scripter after running the experiment first. In this case, the user needs to manually capture the Output file after running the script commands. Consult the GC Scripter documentation for writing scripts using the Tools -> Script Editor menu options.

2.2.7.2 Top Panel

Depending on the User controls at the top level, there are different elements displayed. Select the GC Studio panel, and click on the box labeled GC5016, as shown in Figure 47.

The DUC experiments are identified by the mode_ab and mode_cd have the transmit selection. Splitiq can also be used, when two DUC channels are combined to utilize more PFIR taps.

A four channel configuration is when the splitiq mode is false.

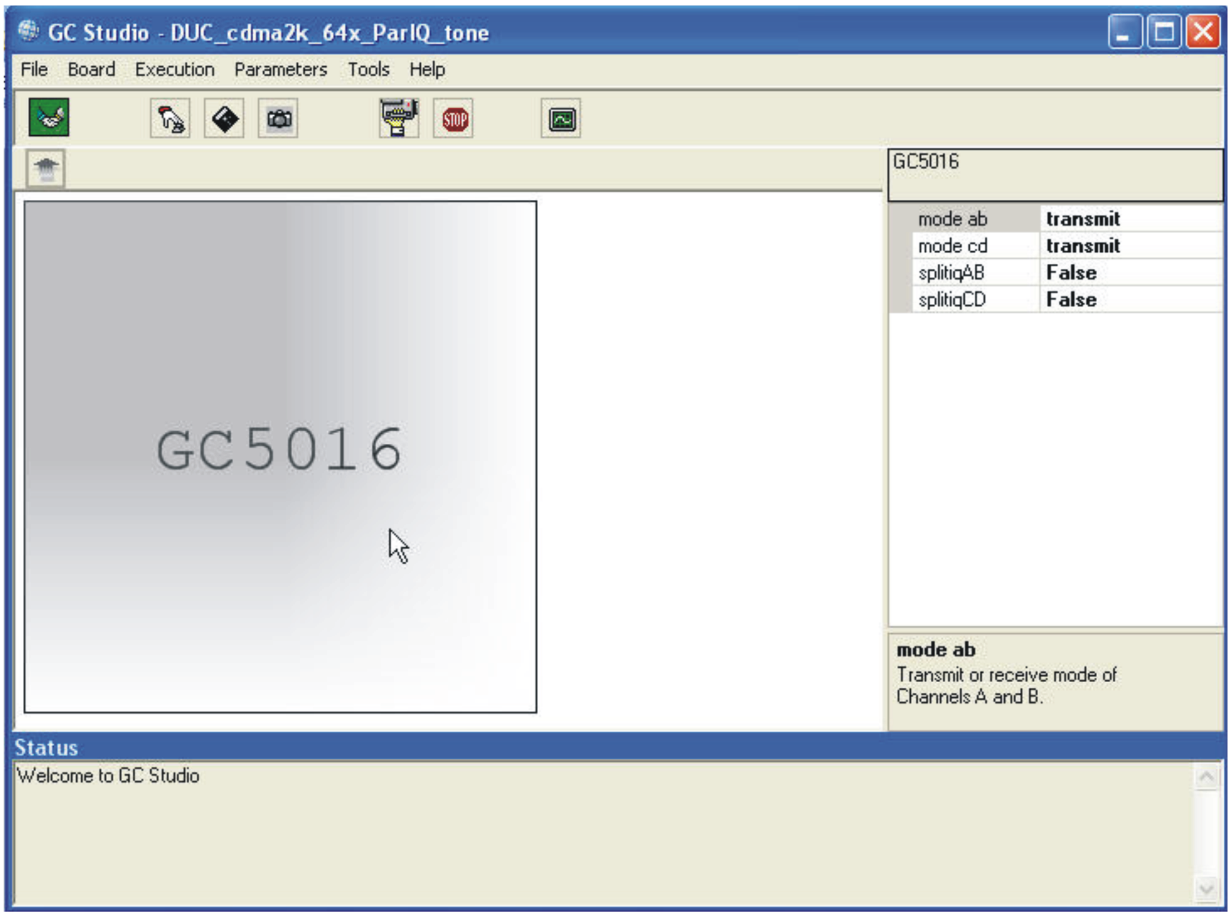


Figure 47. GC Studio GC5016 Top Panel

2.2.7.3 GC5016 Input, Channels, and Output

Selecting the GC5016 box, and double clicking should open the DUC detail view. (See Figure 48) . This shows the experiment view of the Input Stimulus, GC5016 programmed for experiment, and Output Monitor memory.

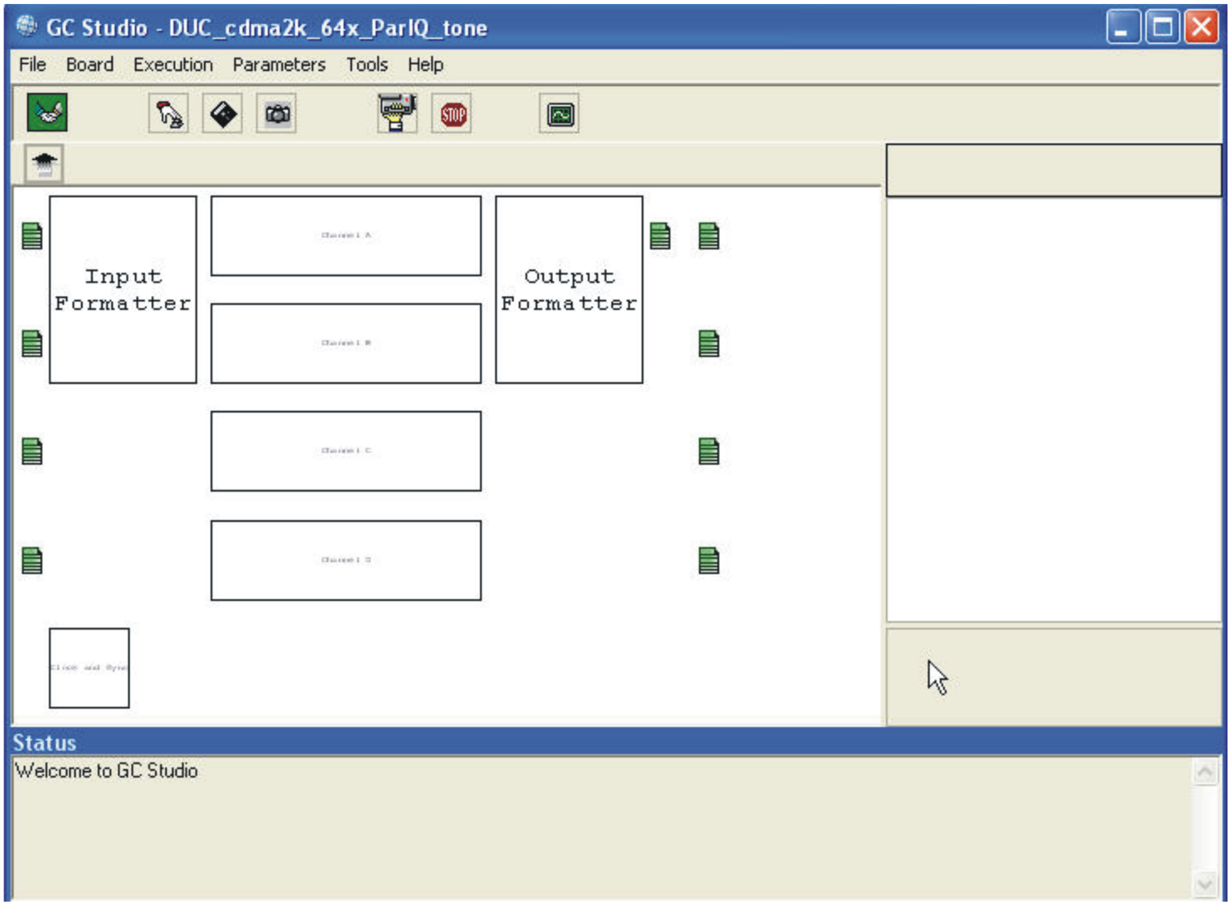


Figure 48. GC Studio DUC Input, Channels, and Output

2.2.7.4 *Input Files*

On the left edge, the green boxes are the input files to be loaded into the input memory. There can be from 1 to 4 Green boxes, one per channel. Click on a green box, and on the upper right, the Load From File can be used to enter, or with the ….box you can browse for an input file as shown in Figure 49 and Figure 50.

The input file is a decimal column file. A Real input has a single column. A Complex input has two (I and Q) columns.

Once you have a entered a value into a variable, it is tagged as user modified.

It is good practice to put experiments' data files in with the project folder.

Complex File
In-phase-Decimal <sp> Quadrature-Decimal

Note: The user would select the LoadFrom File browse box (…) to navigate the file. See Figure 49 for the Navigation button. See Figure 50 for the File Browse Box displayed.

Note: In the lower right corner of the screen, when selecting an object or variable name, context sensitive help is printed in this section.

Note: At the bottom of the display is a Status log. This displays the current status. Errors need to be identified and fixed. If there are warnings, you can check example experiments' JPEG files to see if they are expected warnings.

Note: The Tools->Options->Debug->Use Alternate Stimulus Memory Image – can input a hexadecimal file with no formatting. This is useful if you suspect that the input file conversion is in error. The Dump Contents of Stimulus Memory is the post Input File Conversion hexadecimal file.

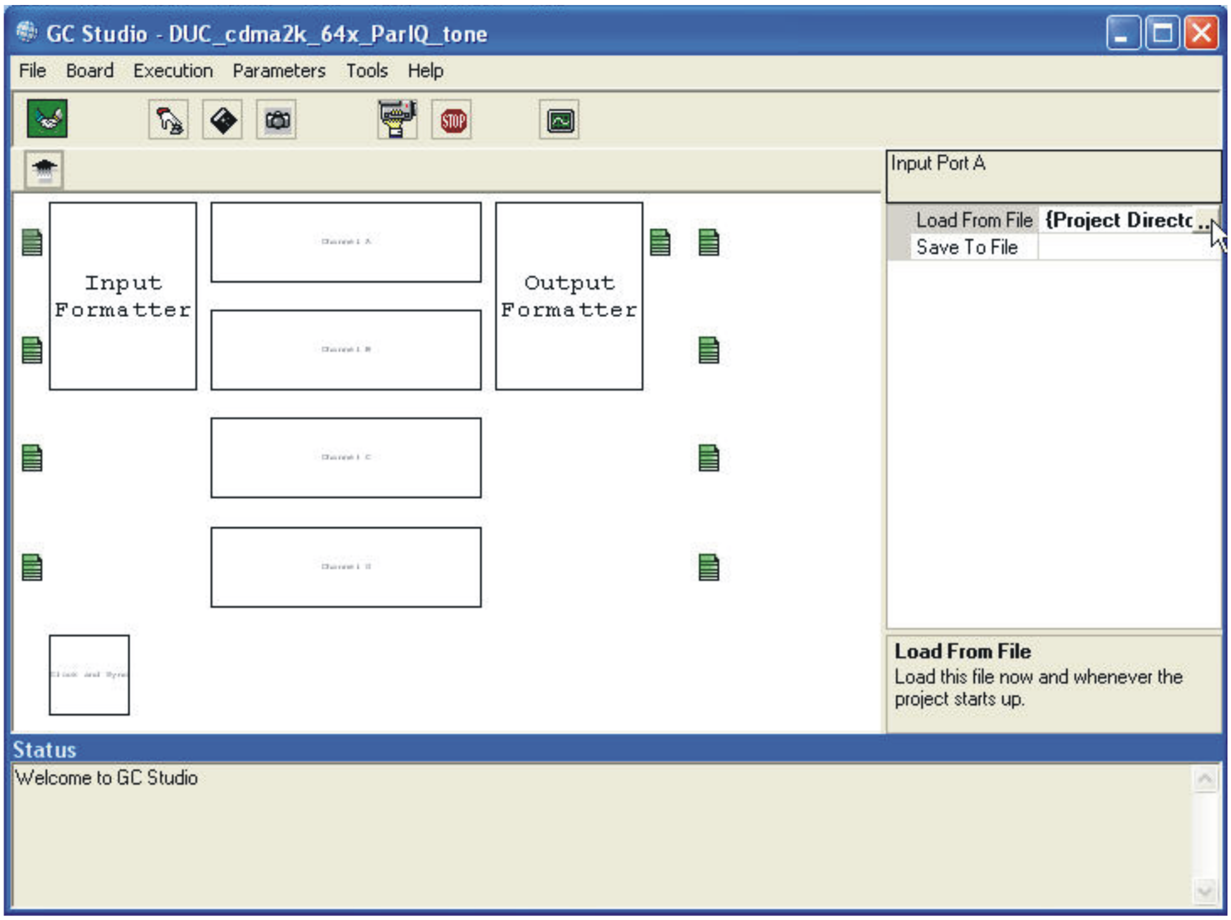


Figure 49. GC Studio Detail Input, GC5016, and Output Panel

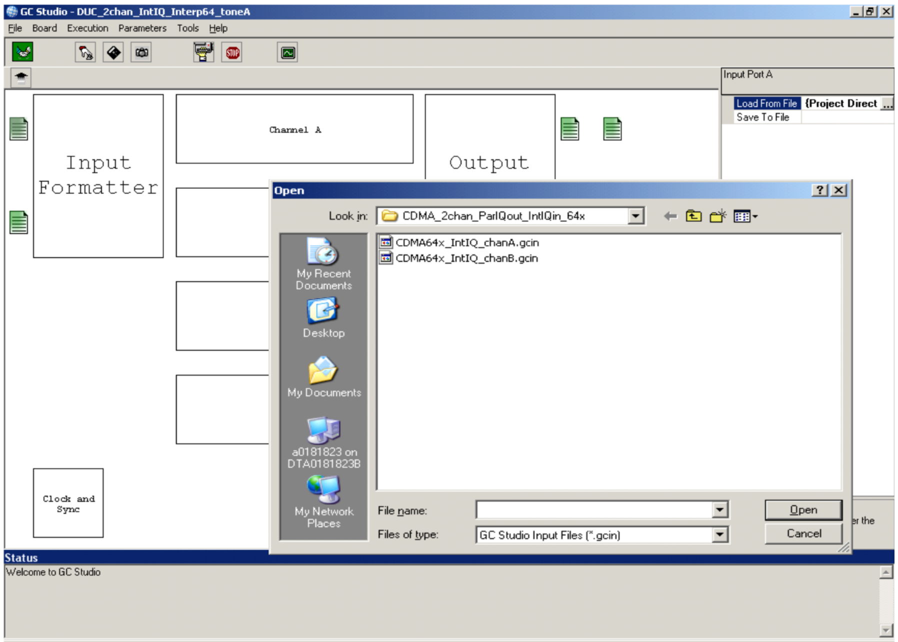


Figure 50. GC Studio File Input, Selecting Load from File

2.2.7.5 *DUC Input Formatter*

The DUC Input Formatter block selects the complex data input format for the GC5016. Click on the Input Formatter block to view the parameters that are available to set, as shown in Figure 51. There are several modes, with illustrations for each of these modes shown in the GC5016 IO Mode Applications Note.

Table 3. DUC Input Formatter Block — Data Input Format Modes

MODE	BITS	PINS	Tinf_cmplx	Tinf_Pariq	Tinf_tdm
4 channel 8 bit I and 8 bit Q parallel	8	16	1	1	0
4 channel interleaved IQ 16 bits	16	X	1	0	0
4 channel common interpolation TDM mode - portA	16	X	1	0	1
2 channel parallel IQ – 16 bit	16	X	0	0	0
2 channel common interpolation TDM mode – portA	16	X	0	0	1

The DUC Inputs have complex inputs. The Green Input File display matches the number of channels.

- Bits – the DUC has a 20 bit input bus internally. The number of input bits is scaled to align the MSB (most significant bit). Normally 8 or 16 bit inputs are used.
- Pins – the DUC input has a 16 bit input for each DUC port. If less than 16 bits are used, the channel can multiplex the Upper and Lower sections. Normally 8 or 16 pins are specified.
- Sck_div – The Ck clock is used to sample the DUC input. The sck_div is used to stretch the input a number of Ck clocks. In parallel IQ modes, the sck_div can be set to the pfir_int. In interleaved IQ or TDM modes, the Sck_div and bits/pins ratio must have enough data cycles to input the DUC data within the interpolation number of clocks.
- Tinf_cmplx – indicates that the input is complex on the DUC input port. This is only set to 0 for split IQ modes.
- Tinf_fs_dly – this control provides a delay on sampling the DUC input data, in divided input clocks. This is adjustable, but is mostly set to 1.
- Tinf_pariq – this variable is used for the special 8 bits I, 8 bits Q special input mode

- Tinf_tdm – this variable is used for the common interpolation TDM mode, where the input data is multiplexed onto port A.

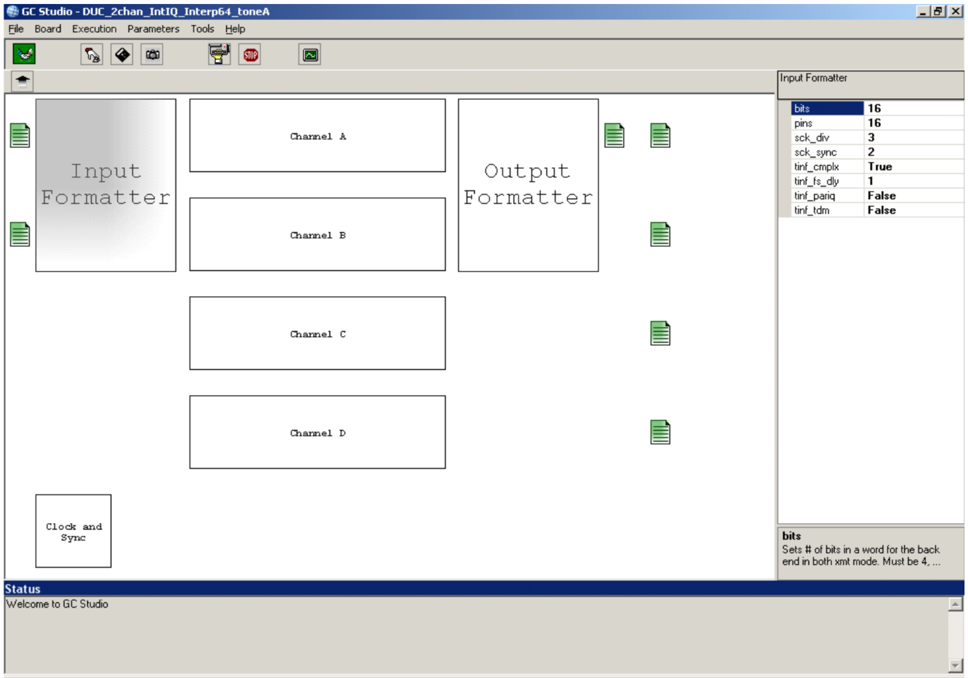


Figure 51. GC Studio DUC Input Formatter Variables

2.2.7.5.1 DUC Special Parameter Control for Input Memory

The DUC Input customer logic is required to provide data based on synchronization with the GC5016 Frame Strobe (FS) signal. The GC101 does not synchronize in this method, but offsets the input data. In DUC experiments the user must determine the proper offset value (starting with 0 and being as large as (cic_int * fir_int) – 1 delays.

An Fs/8 (8 complex values for a cosine wave) tone is used to calibrate the offset. See Figure 52 to select the Fs Delay, and Figure 53 for the data entry.

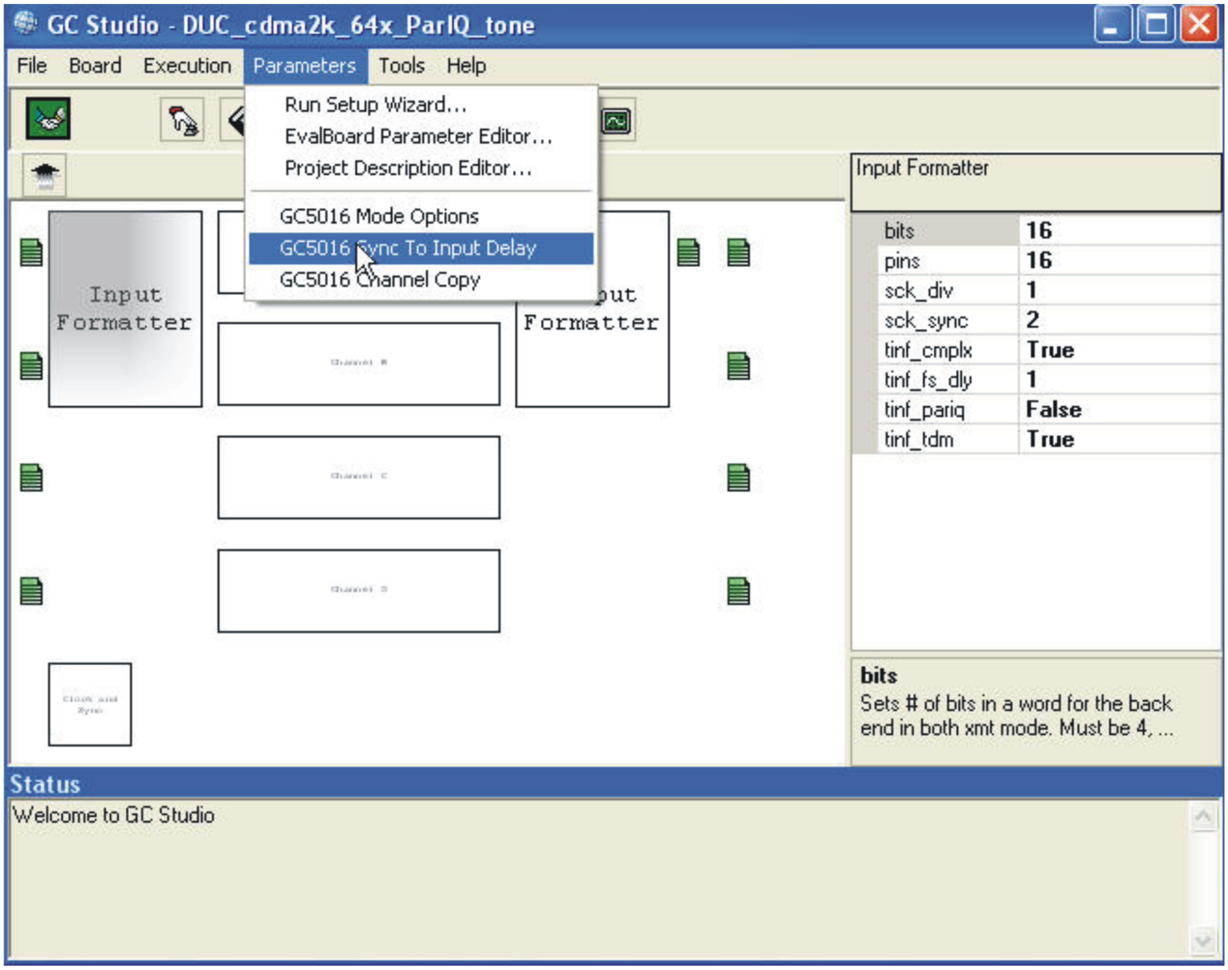


Figure 52. GC Studio

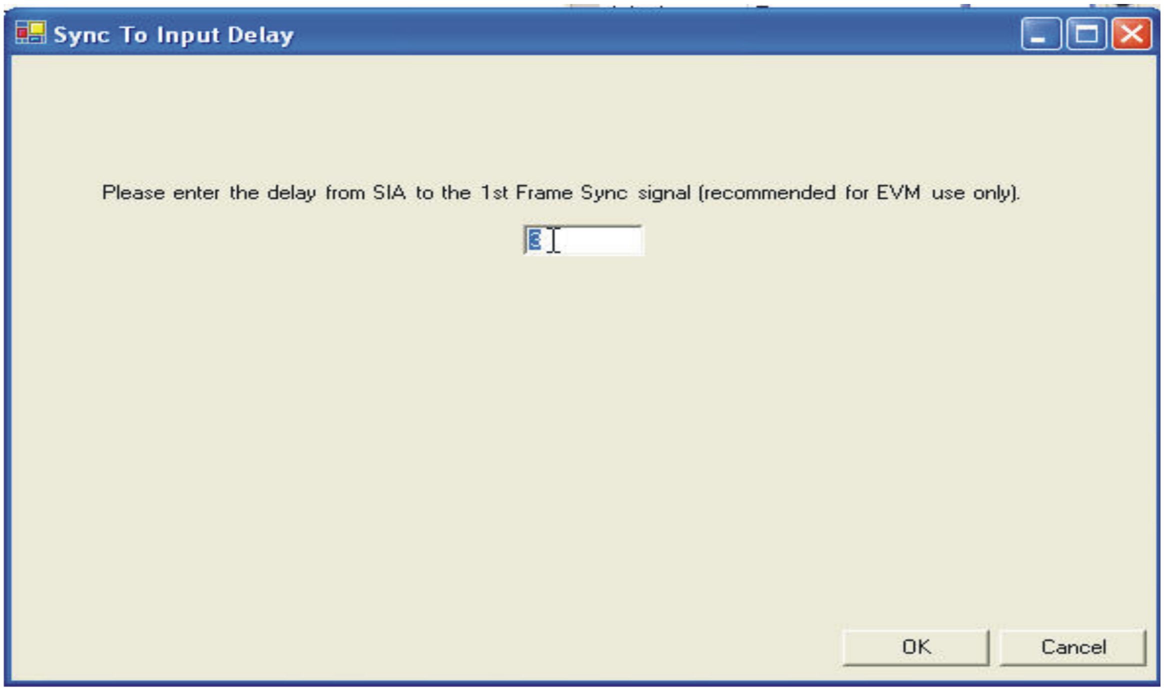


Figure 53. GC Studio Clock and Sync Variables

The DUC delay is an important adjustment for DUC input experiments. The input data can have spectrum reversals, and improper I and Q data to specific channels if the adjustment is not correct. The delay adjustment is affected by the DUC 4 channel or splitIQ mode, the pfir_int, cic_int, number of PFIR taps, DUC input mode, sck_div, and tinf_fs_dly.

Note: The data pattern can be found in several of the DUC experiments, as an Fs8_2048.dat file.

Figure 54 shows the results of the channel A tone calibration project, under the DUC_2chan_IntIQ_Interp64 folder. It is important to have a tone output at Fmixer + (Fck/(cic_int × fir_int × 8)). Having a single tone output indicates that the delay parameter is aligned for this experiment.

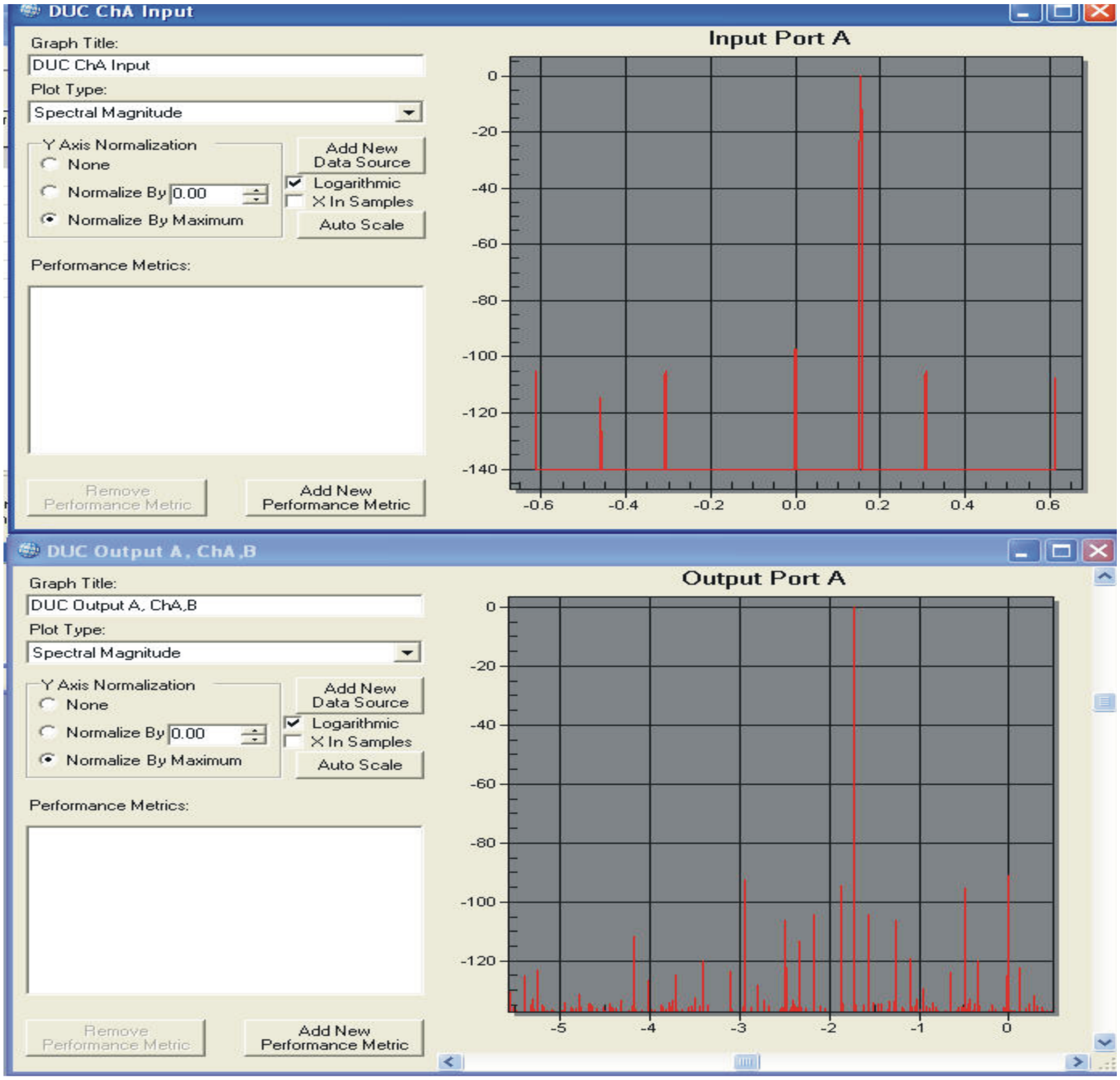


Figure 54. GC Studio DUC Fs/8 Tone Input for Delay Calibration

2.2.7.6 Clock and Sync

The Clock and Sync section has two variables, shown in Figure 55. The fck is the main Clock rate for the GC5016. The sync_mode is initially used to set the sync_index values for the different sections of the GC5016 synced sections.

The Fck or Ck is the desired clock rate of the GC5016. It does not need to be the same as the GC101 clocking, unless external data input or output is needed.

The common sync_mode values are:

- 0 – used for SIA external or memory synchronization.
- 9 – used for One Shot Pulse start.

The GC101 memory has a special delay memory control which is used to offset the Input data.

If the interleaved IQ or TDM IQ input modes are used, the 0 sync_mode is required. Parallel IQ input modes can use the sync_mode = 9.

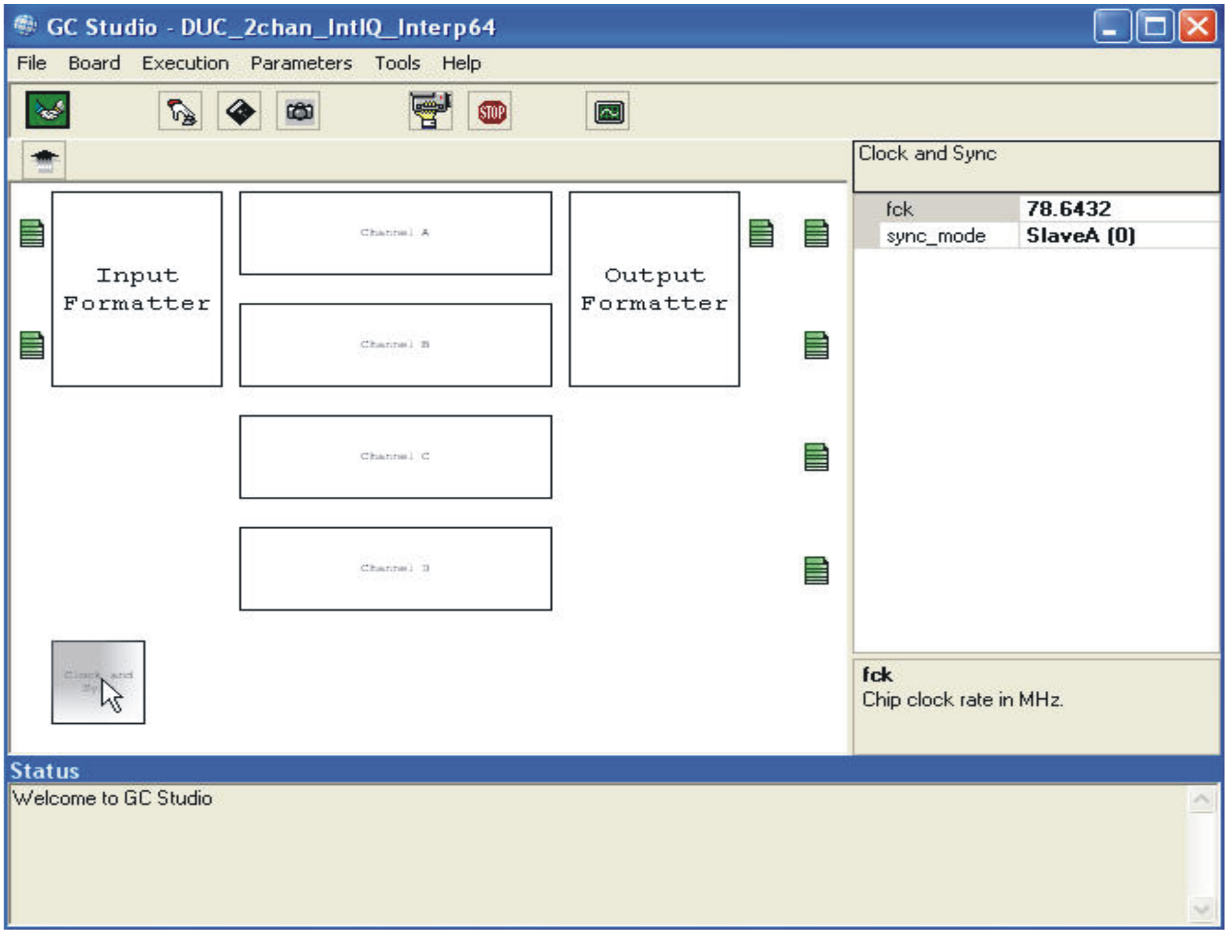


Figure 55. GC Studio Clock and Sync Variables

2.2.7.7 DUC Channel Display Screen

Figure 56, shows the DUC channel display. Select the proper channel, and double click on the box. The detailed DUC Channel Display is shown in Figure 57.

There are several programming variables displayed:

- Overall_gain – used for the engineering gain calculation. These include gain, PFIR shift, cic_shift, and complex mixer related to the output formatter (sum shift). There is a DUC Overall gain application note. The overall gain is adjusted for a combination of the number of channels, desired peak level, and desired stop band rejection. See application note *GC5016 DUC Mode Gain by CMD5016* (TI literature numbers SLWA032 and SLWA033) for more information.
- Tinf_src – this is typically a 2 for modulated inputs. It can be set to 0 for diagnostic inputs.
- Tx_pat_gen – the Tx pattern generator is used for testing when the DUC input is not used. Normally this has a value of 0, but can be set to a constant for tone output testing, to check the DUC -> DAC connections.

Programming the detailed sections involves selecting the Box, and clicking. This accesses the channel specific Complex Mixer, CIC, PFIR, or Gain functions. Use the Up Arrow in the upper left corner to go back to the GC5016 view.

Note: Overall gain, has a manually adjusted sum_shift in the output formatter. The output formatter default value should be set to 4. The gain value can be =~ 1/numchannels summed together. The A and B output ports can use the sum_chainA and sum_chainB in the GC5016. The sum_shift can be adjusted over the range of 0-7. DUC output ports C and D have a fixed scale, equivalent to a sum_shift of 4, for a single channel output.

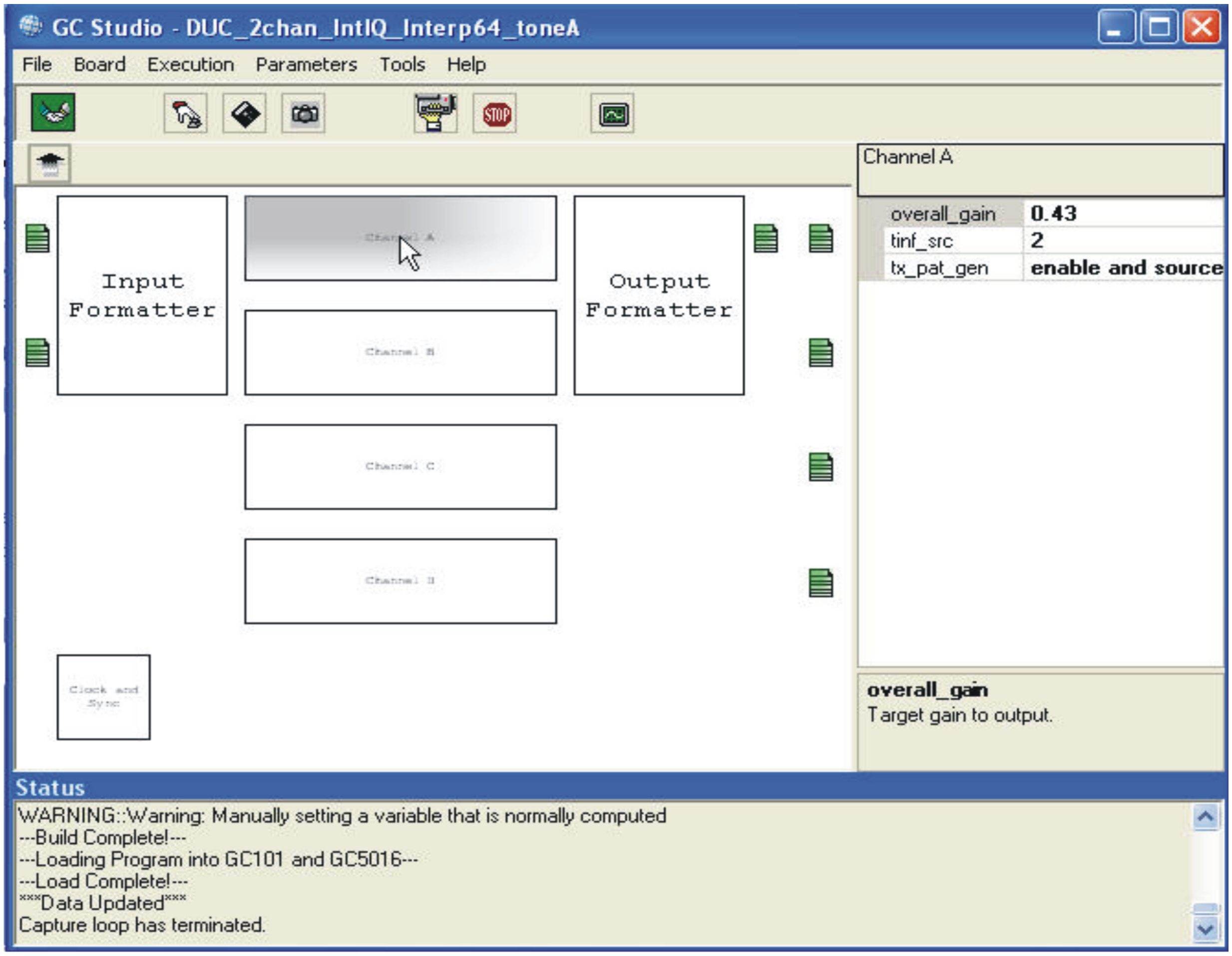


Figure 56. GC Studio DUC — Selecting a DUC Channel

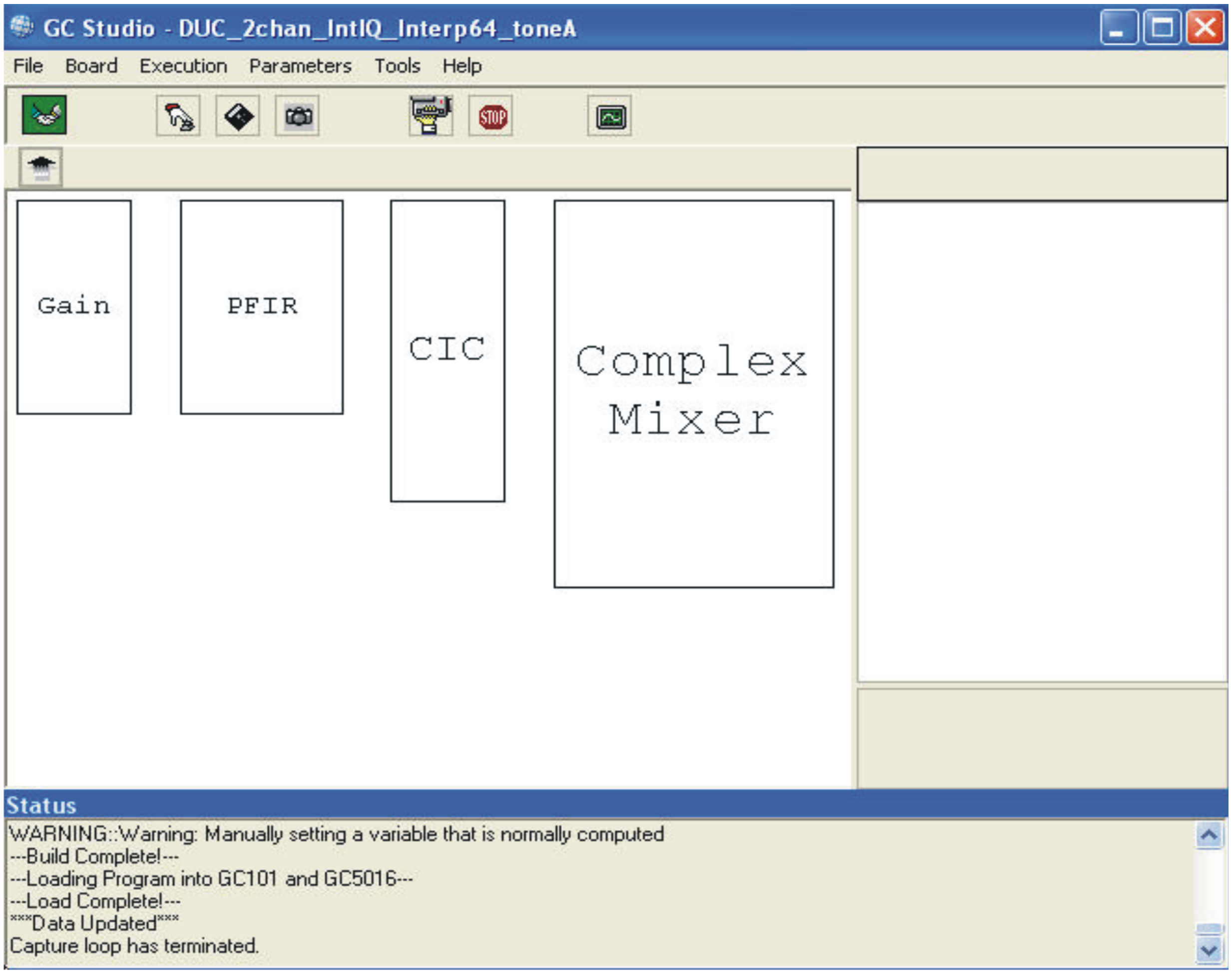


Figure 57. GC Studio DUC – Detailed DUC Channel View

2.2.7.8 DUC Gain Section

The DUC Gain section was selected, by double clicking the Gain Box, on the DUC detailed channel display (See Figure 58). The variables shown in the right section are controlled in this section.

- Gain_rnd – indicates the number of bits to round off the bottom of the 20 bit input. This is typically left as 2, because the PFIR has an 18 bit input.
- Gain_sync – this control can be set to 7, always update, or for synchronous updates for multiple channels. The 4-OS, 3-SIB, or 2-SIA are the other usable choices, and controls the synchronization between updating the internal gain block from the microprocessor gain registers.

When this section's variables have been updated, you can click on another section, or use the Up arrow to go back to the GC5016 view screen.

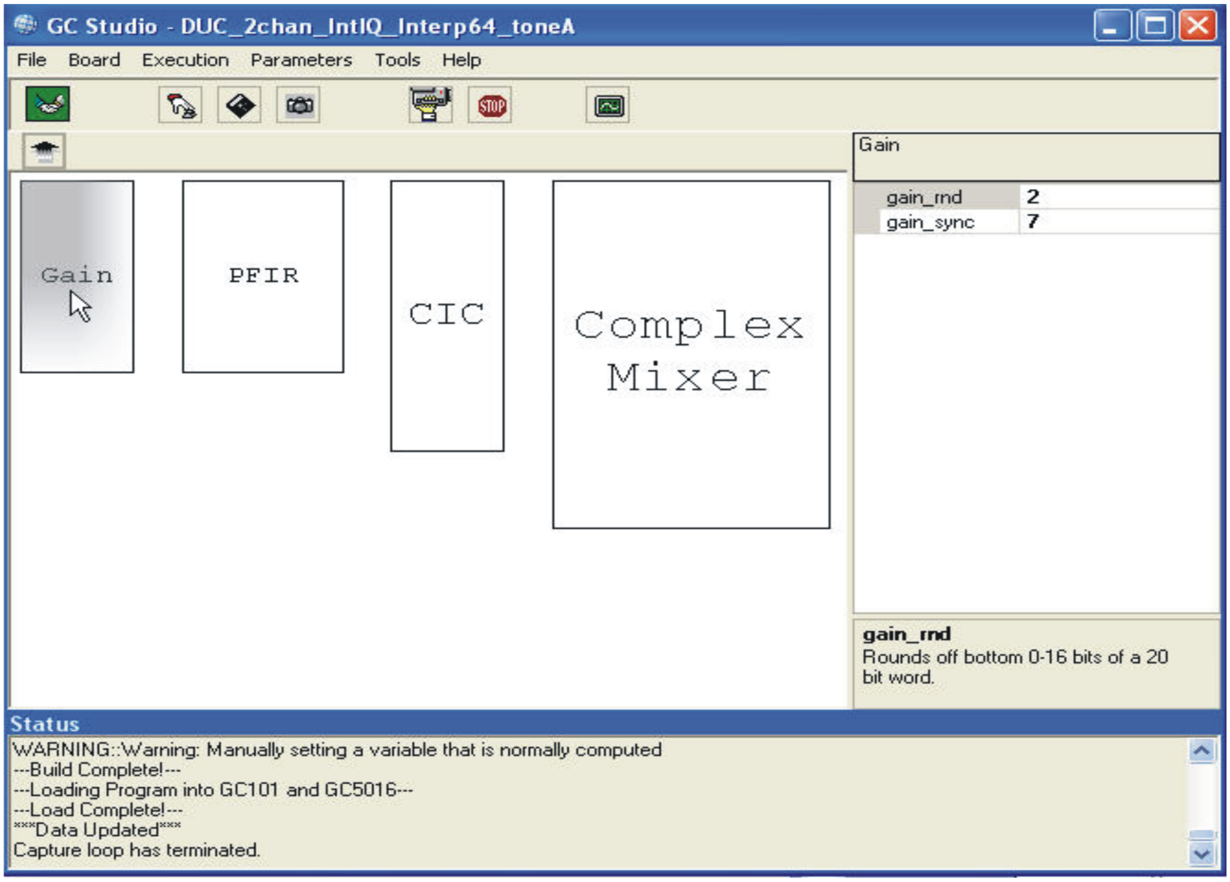


Figure 58. GC Studio DUC — Detailed DUC Gain

2.2.7.9 DUC PFIR (Programmable Finite Impulse Response Interpolation Filter)

The GC5016 PFIR has several internal sections, memory, 16 multiply accumulate sections, a sum chain, forward and reverse delay lines, Address Generators, and State Machine Controls. These are programmed through cmd5016. The higher level commands that are used to program the PFIR are shown in Figure 59. There is a general equation for the number of PFIR taps in the GC5016 datasheet. The experiment folder has a .ANL file that can be viewed with a text editor. Near the bottom is the information about the number of PFIR taps allowed.

The number of PFIR taps can be initially programmed as a 31 tap, center tap 32767, all other taps 0. The .ANL file can be checked and then the final PFIR taps can be generated. There is a matlab tool (available from your Field Application Engineer) for generating the PFIR filter coefficients. Manually generating a PFIR filter, there are 3 taps for the CIC correction [-.17,1,-.17] that are convolved with the user filter. If the cic_int is 1, the CIC correction is not needed.

- Bypass_Fir – variable that is normally set to false.
- Fir_cmplx – set to 1 in 4-channel mode, it is set to 0 in splitIQ mode.
- Fir_coef – variable that allows a view of the PFIR coefficients, or a method to load the coefficients from a single column of 16 bit 2s complement coefficients (See Figure 42 and Figure 43).
- Fir_int – interpolation counter value + 1 that determines the PFIR_interp.
- Fir_diff – set to false.
- Fir_nchan – set to 1.
- Fir_sync – sync source for PFIR which is normally set to 2.

The PFIR has odd symmetry only for fir_int of 1 or 2. In some cases the splitIQ mode is needed in the DUC mode, due to needing more taps to meet a spectrum mask and the desired stop band rejection.

Section 2.2.5.13 has more details on selecting the variable fir_coef, to enter or view the PFIR coefficients. This section for the DDC or DUC mode works identically.

When this section's variables have been updated, you can click on another section, or use the Up arrow to go back to the GC5016 view screen.

Note: If the DUC input sample rate is 2 samples per symbol or higher, the pfir_int of 1 or 2 can be used. If the DUC input sample rate is 1 sample per symbol, the pfir_interp is usually set to 3.

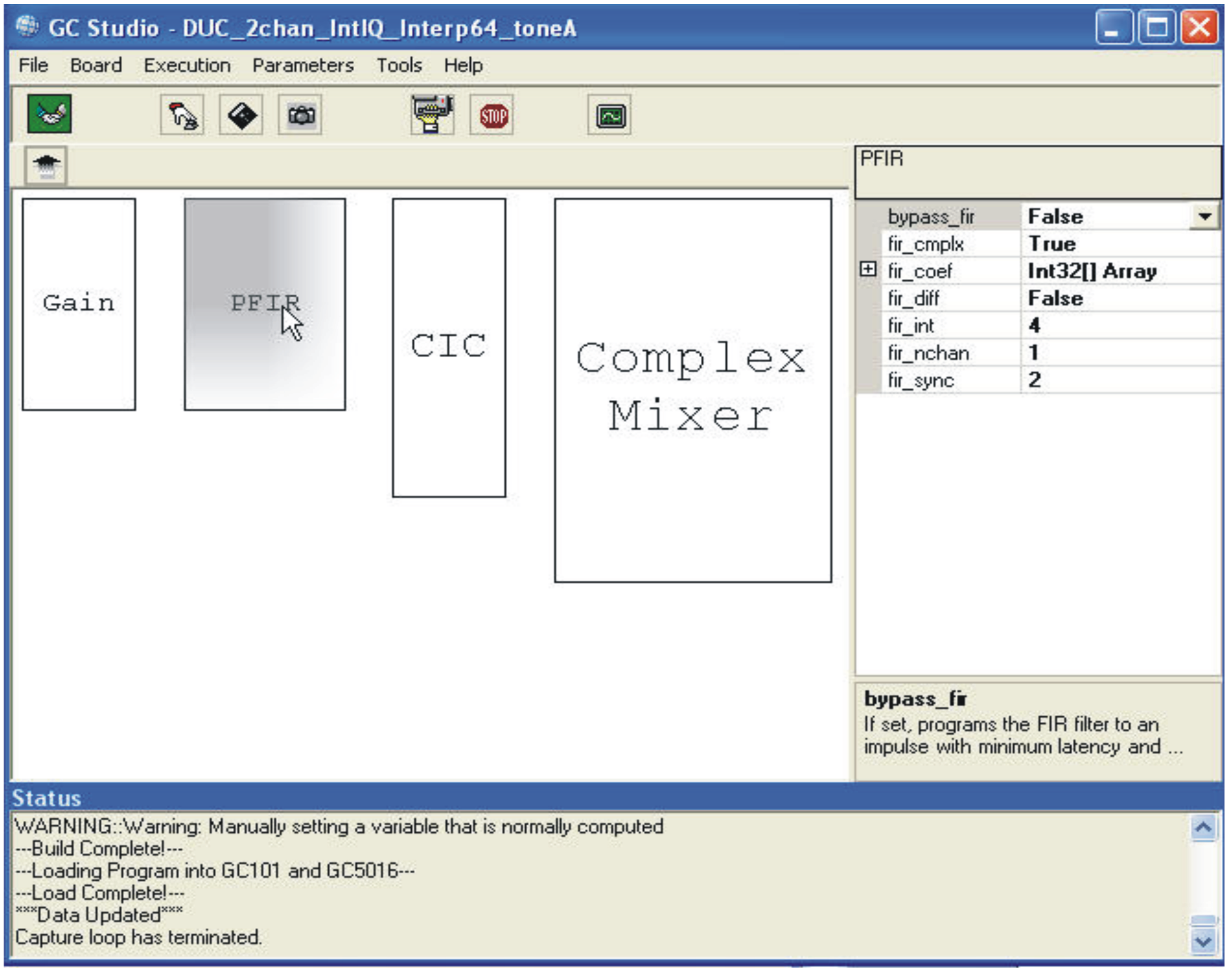


Figure 59. GC Studio DUC – Detailed DUC PFIR Section

2.2.7.10 DUC CIC Section

The CIC (Cascade Integrator Comb) is the second stage interpolation filter in the GC5016. Click on the CIC block to view the parameters that are available to set, as shown in Figure 60. The CIC is a 5 or 6 stage CIC in the DUC mode. The current implementation is 6 stage only in the GC Studio software. The 6 stage CIC is normally used, the 5 stage is only used for high DUC interpolations (>= 600).

In the four channel configuration, the CIC must interpolate by at least 2. In the splitIQ configuration, setting cic_int = 1 is used to bypass the CIC effectively. In the 4 channel mode, the CIC operates on I and Q data on alternate cycles. In the splitIQ mode, the CIC in each DUC channel, only operates on I or Q exclusively.

The cic_sync controls the sync value to initialize the interpolation counter (cic_interpolation phase). The flush_sync is used to zero the CIC accumulator.

- Cic_bypass – not normally used. Set to false.

- Cic_int – cic_interpolation value, normally 2 – 222. The value can be 1 in splitIQ mode.
- Cic_sync – sync value source for the cic interpolation counter. Normally set to the startup sync or external sync source. Set to 4 for One Shot. Set to 2 for SIA 2. Set to 3 for SIB.
- Flush_sync – sync value source for the cic_accumulator zero. Normally set to the startup sync or external sync source. Set to 4 for One Shot. Set to 2 for SIA 2. Set to 3 for SIB.

When this section's variables have been updated, you can click on another section, or use the Up arrow to go back to the GC5016 view screen.

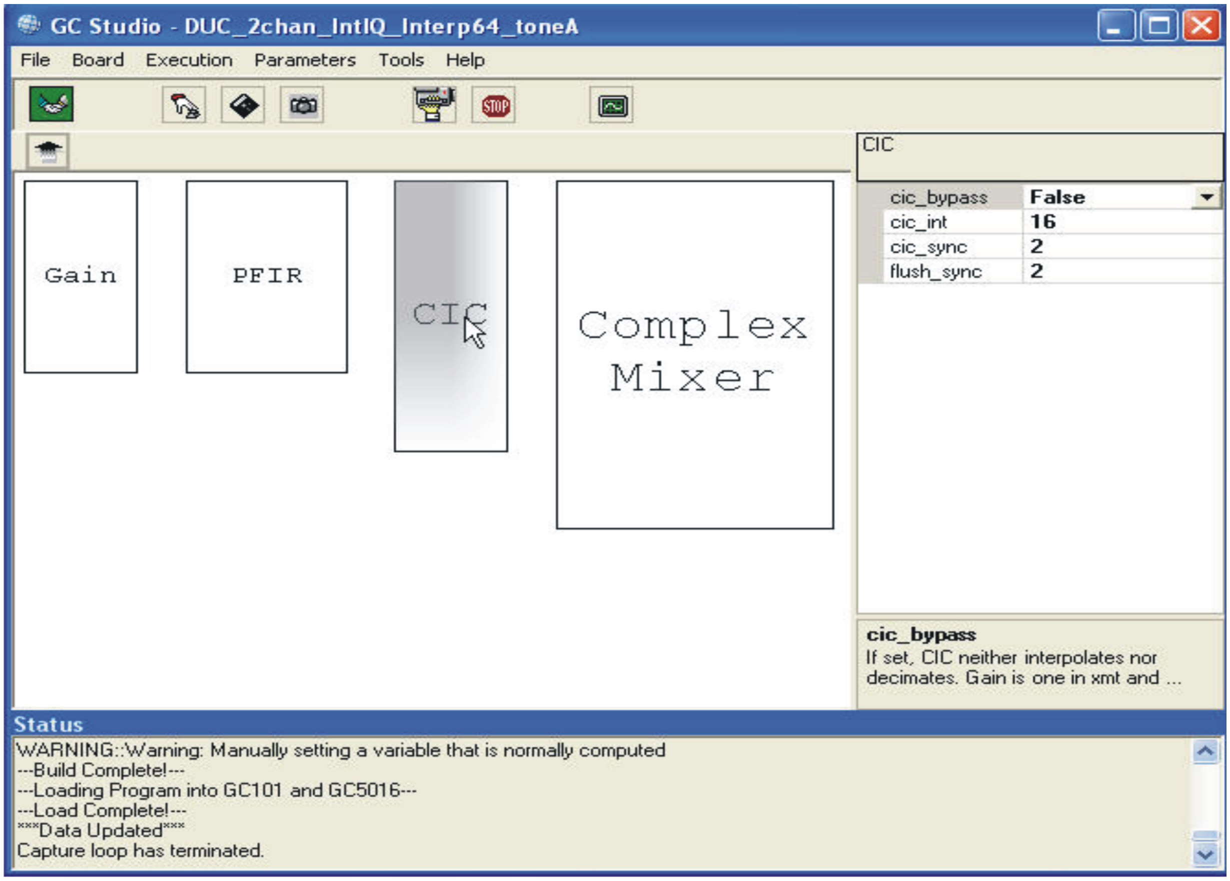


Figure 60. GC Studio DUC — Detailed DUC CIC Section

2.2.7.11 DUC Complex Mixer Section

The DUC Complex mixer is selected in Figure 61. The DUC selects a Complex signal from the corresponding CIC channel, or for splitIQ mode, can combine the I and Q from separate channels. The splitiqAB and splitiqCD engineering variables determine the mixer mode.

- Bypass_Mix – normally set to false that is used for DDC input testing.
- Dith_sync – controls the dither phase value. Dith_sync is set to the startup-sync source, or external sync. If no dithering is desired, Dith_synth is set to 7.
- Freq – The mixer frequency in MHz. It is set to a negative frequency for odd Nyquist zones to translate a positive IF frequency to 0. It is set to a positive frequency for even Nyquist zones to translate a negative IF frequency and flip the spectrum.
- Nco_sync – This control is used to reset the NCO phase accumulator. This causes a phase disturbance, and is normally done with the startup sync source.

When this section's variables have been updated, you can click on another section, or use the Up arrow to go back to the GC5016 view screen.

Note: nco_sync and dith_sync are set to external synchronization if repeatable and deterministic phase outputs are required. The value of 4 for One shot, can be used to properly initialize, but the relationship of the absolute phase of the accumulator is not known.

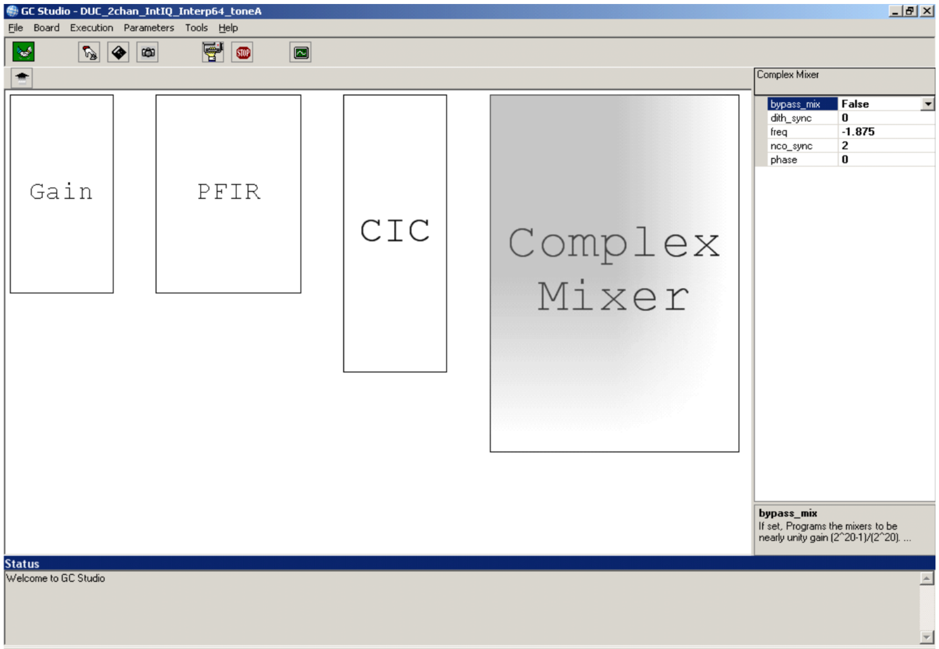


Figure 61. Studio DUC — Detailed DUC Complex Mixer Section

2.2.7.12 DUC Output Formatter Section

The GC5016 has several DUC output modes. These are described both in the GC5016 datasheet and in the GC5016 IO Mode Application note. There are output modes that are not usable for the GC101 and GC5016 daughtercard, due to the number of IO pins needed. The GC5016 daughtercard switches 3 and 4 are used to select which pair of output ports are monitored by the GC101 Output memory. 3(ON) and 4(OFF) is for channels A and B. 3(OFF) and 4(ON) is for channels C and D.

The DUC Output modes supported are:

Table 4. DUC Output Modes

DUC OUTPUT MODE	tout_cmplex	tout_rate	tout_res	tout_nsig	tout_sumin
2 channel 16 bit real output	0	1	0	4, 2	0
2 channel 16 bit interleaved complex output	1	0	0	4, 2	0
2 channel 16 bit parallel complex output	1	1	0	4, 2	0
1 channel 16 bit real output	0	1	0	1	0
1 channel 16 bit interleaved complex output	1	0	0	1	0
1 channel 16 bit parallel complex output	1	1	0	1	0

The Output Formatter variables include the Sum_shift gain variable for the sum_tree A and B logic, as shown in Figure 62.

- Sum_shift – power of 2 shift for the sum tree A and B DUC output.
- Tout_cmplx – DUC output is complex.
- Tout_nsig – number of DUC outputs that are summed onto an output port.
- Tout_rate – indicates interleaved (0), full rate(1), or double rate(2).
- Tout_res – indicates 16(0) or 22(1) bit output. 16-bit mode is supported in software.

- Tout_sumin – converts the C and D outputs to sum input ports. Not supported in software.
- Toutf_offsetbin – changes the numeric output for some DACs, normally 0.

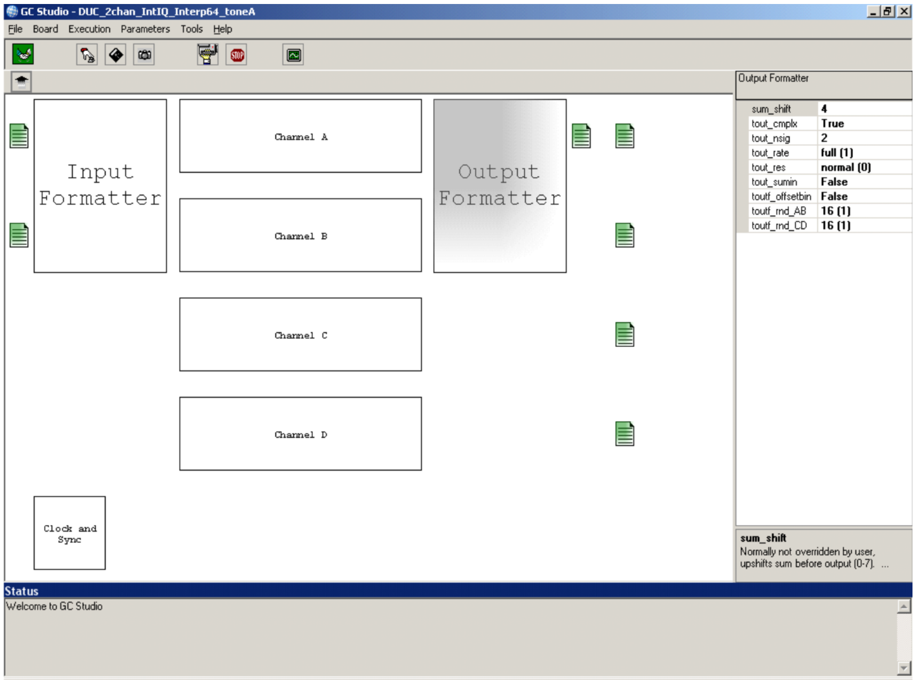


Figure 62. GC Studio DUC — Output Formatter section

2.2.7.13 DUC Output Save to File

The DUC output can be saved to a decimal column (real) or two column (complex) file. Selecting the green box next the desired channel (the example shows the A output only), displays a parameter on the right hand parameter box. Using the browse box, allows the user to select or browse for an output file. The data is saved for each data run as shown in Figure 63.

Note: Under the Tools -> Option -> Debug -> Dump Contents of Response Memory, a hexadecimal output file can be generated for checking output file conversion features.

Parameter Channel Copy can be used to copy the designed channel A, to B, C, and D as appropriate. Parameters → GC5016 Channel Copy

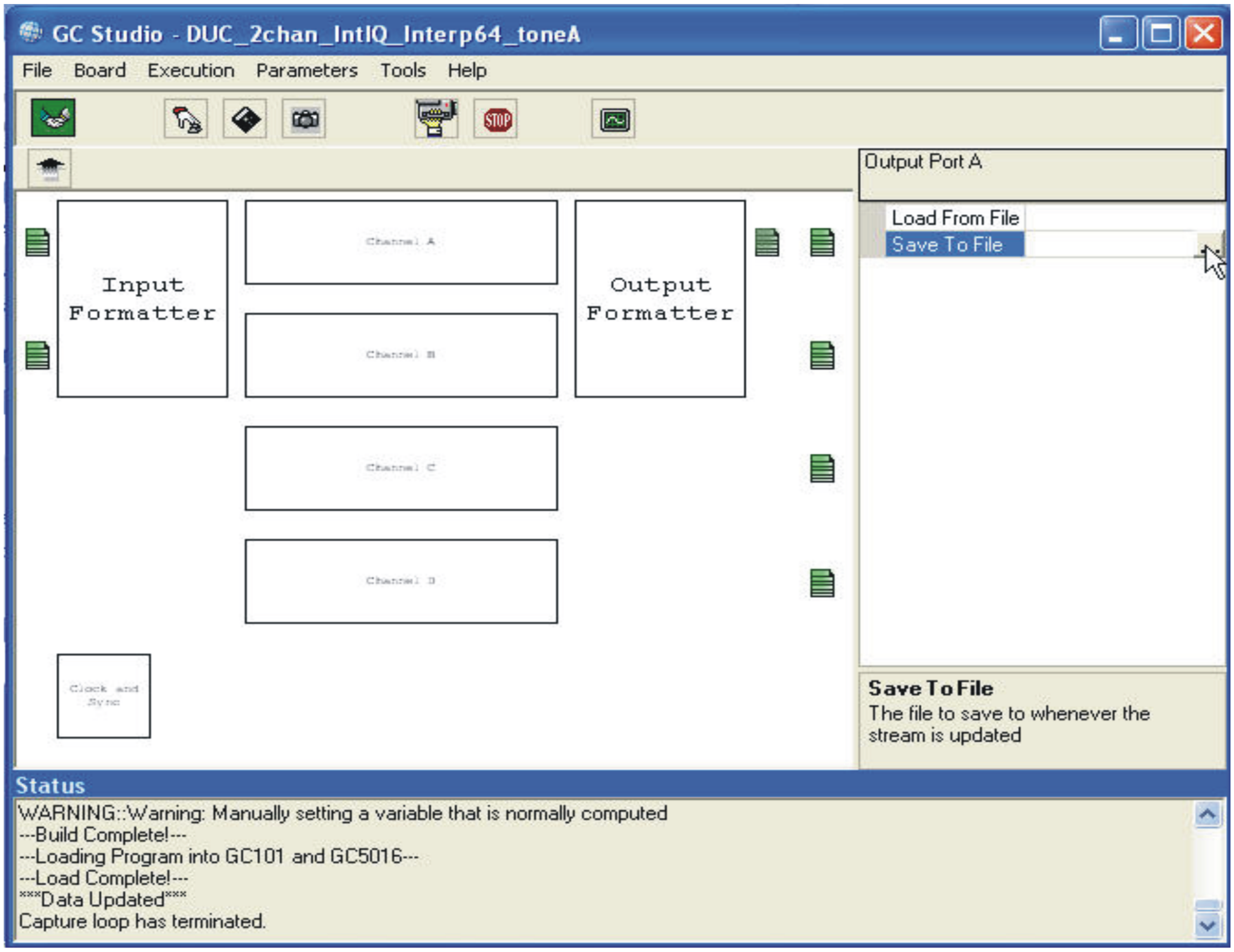


Figure 63. GC Studio DUC — Write Output Memory to File

2.2.8 GC5016 Top Panel for Transceiver GC Studio Experiments

GC5016 configurations that are used for the Transceiver mode use both the DDC and DUC modes. Channels A and B must be in the Transmit mode. Channels C and D must be in the Receive mode.

Each experiment can only perform DDC or DUC stimulus and monitoring one at a time. You can have the two sets of GC5016 menu entries. These are identical to those described in sections 2.2.5 and 2.2.6.

2.2.9 GC5016 Programming Cross-reference in GC Studio Experiments

This section has a list of all of the GC5016 programming variables used in the DDC and DUC programming. The variable description is repeated here as a summary of the text descriptions.

Table 5. GC Studio GC5016 Figure and Variable Programming

SCREEN #	HIERARCHY	DESCRIPTION	VARIABLES
Fig 9 Fig 28	1	GC5016 Box display, general variables for mode	Mode_ab – receive or transmit Mode_cd – receive or transmit splitiqAB – splitIQ mode for AB channels splitiqCD – splitIQ mode for CD channels
Fig 10 Fig 30	2, 5	Shows GC5016 channels, Input source for stimulus memory, and output source for response memory	Oversampled real or complex data for DDC input. Baseband complex data for DUC input.
Fig 11 Fig 31	2-A,6A - User Input	Shows GC5016 channels, Input source for stimulus memory	Load from File – inputs channel input data, for each active input channel
Fig 14	2-B1 DDC Input Formatter	Shows DDC input formatting for real or complex inputs, diagnostic selection zpad, and zpad sync for all channels.	Rin_cmplx – indicates DDC input 1 complex, 0 real Rin_rate – normally 1 for full rate Rinf_diag – selects receive input diagnostic Rinf_zpad – zero padding the DDC input from a lower-integer rate digital source Rinf_zpad_sync – set to main sync source
Fig 15 Fig 36	2-B2 , 6-B2 Clock And Sync	Shows sync_mode and Fck settings for all channels.	Fck – GC5016 Ck rate Sync_mode – selects startup sync source, and if one shot is selected include the startup sync
Fig 16	2-B3.4.5.6 DDC Channel Select	User selects from one of 4 channels. Also used to set channel gain, and mixer complex source.	overall_gain – sets channel gain rinf_sel – selects this channel mixer input 0, test,
Fig 17	2-B7 DDC Output Formatter	Shows the Output DDC configuration and synchronization	Bits – the number of bits output from 8 to 20 in blocks of 4. Pins – the number of output port pins used to transfer data, can be 4, 8, or 16. Routf_iqmux – logic 1 for interleaved IQ output mode Routf_tdm – 1 for TDM output on port D Sck_div – the number of clock cycles – 1 that each output is presented with Frame Strobe Sck_sync – used to synchronize the output block. This should match the pfir_sync source value.
Fig 18	2-B8 - User Output	Shows Save DDC Output to Disk File	Save To File – outputs DDC complex channel data to a disk file
Fig 20	2-C1 DDC Complex Mixer	Shows Complex Mixer Box highlighted, displays variables for mixer, and complex input bus selection	Bypass_Mix – normally set to false. It is used for DDC input testing Dith_sync – controls the dither phase value, it is set to the startup sync source, or external sync. If no dithering is desired, set this to 7. Freq – The mixer frequency in MHz. Freq is set to a negative frequency for odd Nyquist zones to translate a positive IF frequency to 0. It is set to a positive frequency for even Nyquist zones to translate a negative IF frequency and flip the spectrum. Mix_rcv_sel – Selects one of 4 DDC Complex input busses to be used for this DDC channel. Nco_sync – This control resets the NCO phase accumulator. This causes a phase disturbance, and is normally done with the startup sync source.
Fig 21	2-C2 DDC	Shows CIC decimation filter controls	Cic_dec – cic_decimation value, normally 2 – 222. The value can be 1 in splitIQ mode. Cic_sync – sync value source for the cic decimation counter. Normally set to the startup sync or external sync source. One Shot – 4, SIA 2, SIB 3. Some customers use 0. Flush_sync – sync value source for the cic_accumulator zero. Normally set to the startup sync or external sync source. One Shot – 4, SIA 2, SIB 3. Some customers use 0.

Table 5. GC Studio GC5016 Figure and Variable Programming (continued)

SCREEN #	HIERARCHY	DESCRIPTION	VARIABLES
Fig 22	2-C3 DDC	Shows PFIR decimation filter controls	Bypass_Fir – variable normally set to false. Fir_cmplx – set to 1 in 4-channel mode, set to 0 in splitIQ mode Fir_coef – variable that allows a view of the PFIR coefficients, or a method to load the coefficients from a single column of 16 bit 2s complement coefficients. (See Figure 42 and Figure 43) Fir_dec – decimation counter value + 1, that determines the PFIR_dec Fir_diff – set to false Fir_nchan – set to 1
Fig 24	3 - 1	PFIR Coefficient Integer Display	The user pastes decimal coefficients from Excel into the left array of coefficients.
Fig 26	4- 1	PFIR Coefficients Load from file	The user selects a file that contains the decimal coefficients.
Fig 27	2-C4 DDC	Shows manual gain, AGC, and power meter control	Agc_cf – signal crest factor, also used to adjust final signal level Agc_freeze – signal is true to stop agc gain adaption Agc_hold – holds the current agc value, to allow microprocessor bus to read agc value Agc_mode – different settings for lower level agc commands (see application note) Agc_tc – attack and decay time for settings gain fraction for accumulator update Gain_rnd – selects DDC output bits = 20 – gain_rnd Gain_sync – set to 7 for manual gain, set to sync source for manual gain with sync. or AGC Pwr_mtr_on – set to 1 for power meter, currently the other power meter variables are not set
Fig 32	6-B1	DUC Input Formatter – controls for the DUC complex input data	Bits – the DUC has a 20-bit input bus internally. The number of input bits is scaled to align the MSB (most significant bit). Normally 8- or 16-bit inputs are used. Pins – the DUC input has a 16-bit input for each DUC port. If less than 16 bits are used, the channel can multiplex the upper and lower sections. Normally 8 or 16 pins are specified. Sck_div – The Ck clock samples the DUC input. The sck_div stretches the input a number of Ck clocks. In parallel IQ modes, the sck_div can be set to the pfir_int. In interleaved IQ or TDM modes, the Sck_div and bits/pins ratio must have enough data cycles to input the DUC data within the interpolation number of clocks. Tinf_cmplx – indicates the input is complex on the DUC input port. This is only set to 0 for split IQ modes. Tinf_fs_dly – this control provides a delay on sampling the DUC input data, in divided input clocks. This is adjustable, but is mostly set to 1. Tinf_pariq – this variable is used for the special 8bits I, 8bits Q special input mode Tinf_tdm – this variable is used for the common interpolation TDM mode, where the input data is multiplexed onto port A.
Fig 37	6-B3,4,5,6	Allows DUC channel selection. Also used to set channel gain, transmit input test source	Overall_gain – engineering gain calculation. These include gain, PFIR shift, cic_shift, and related to the output formatter (sum shift). There is a DUC Overall gain application note. The overall gain is adjusted for a combination of the number of channels, desired peak level, and desired stop band rejection. Tinf_src – typically a 2 for modulated inputs. It can be set to 0 for diagnostic inputs. Tx_pat_gen – the Tx pattern generator is used for testing, when the DUC input is not used. Normally this has a value of 0, but can be set to a constant for tone output testing, to check the DUC -> DAC connections.

Table 5. GC Studio GC5016 Figure and Variable Programming (continued)

SCREEN #	HIERARCHY	DESCRIPTION	VARIABLES
Fig 39	6-C1	Detailed DUC-Gain input block	Gain_rnd – indicates the number of bits to round off the bottom of the 20 bit input. This is typically left as 2, because the PFIR has an 18 bit input. Gain_sync – this control can be set to 7, always update, or for synchronous updates for multiple channels. The 4-OS, 3-SIB, or 2-SIA are the other usable choices, and controls the synchronization between updating the internal gain block from the microprocessor gain registers.
Fig 40	6-C2	DUC Interpolating PFIR configuration	Bypass_Fir – this variable is normally set to false. Fir_cmplx – set to 1 in 4-channel mode, it is set to 0 in splitIQ mode Fir_coef – this variable allows a view of the PFIR coefficients, or a method to load the coefficients from a single column of 16 bit 2's complement coefficients. (See Figure 42 and Figure 43) Fir_int – interpolation counter value + 1. Fir_int determines the PFIR_interp. Fir_diff – set to false Fir_nchan – set to 1
Fig 41	6 – C3	DUC CIC configuration	Cic_bypass – not normally used. Set to false Cic_int – cic_interpolation value, normally 2 – 222. The value can be 1 in splitIQ mode. Cic_sync – sync value source for the cic interpolation counter. Normally set to the startup sync or external sync source. One Shot – 4, SIA 2, SIB 3. Some customers use 0. Flush_sync – sync value source for the cic_accumulator zero. Normally set to the startup sync or external sync source. One Shot – 4, SIA 2, SIB 3. Some customers use 0.
Fig 42	6 – C4	DUC Complex Mixer configuration	Bypass_Mix – normally set to false, is used for DDC input testing Dith_sync – controls the dither phase value, it is set to the startup sync source, or external sync. If no dithering is desired, set this to 7. Freq – The mixer frequency in MHz. It is set to a negative frequency for odd Nyquist zones to translate a positive IF frequency to 0. It is set to a positive frequency for even Nyquist zones to translate a negative IF frequency and flip the spectrum. Nco_sync – This control is used to reset the NCO phase accumulator. This causes a phase disturbance, and is normally done with the startup sync source.
Fig 43	6 – B7	DUC Output Formatter	Sum_shift – power of 2 shift for the sum tree A and B DUC output Tout_cmplx – DUC output is complex Tout_nsig – number of DUC outputs that are summed onto an output port Tout_rate – indicates interleaved (0), full rate(1), or double rate(2) Tout_res – indicates 16(0) or 22(1) bit output. 16-bit mode is supported in software Tout_sumin – used to convert the C and D outputs to sum input ports Not supported in software Toutf_offsetbin – changes the numeric output for some DACs, normally 0 Toutf_rnd_AB(CD) – selects 22, 16, 14, or 12 bit DUC output rounding, normally 16(1) The AB or CD outputs are the pairs of DUC channels.

3 Creating a GC5016 Experiment with GC Studio

Creating a GC Studio GC5016 project requires some pre-planning, follow the desired PFIR and CIC ratios, then performing final filter generation and gain adjustment.

There are two suggested methods for creating a project:

- Open an existing project that is close to the desired configuration, and save the project to a new folder location. Begin the project editing between this section, and the DDC and DUC sections.
- Create a new project, and then continue developing the project with the DDC or DUC sections as appropriate.

3.1 Creating a new GC Studio Project

3.1.1 Start GC Studio, from the File menu, select a new project.

3.1.2 Under the Choose a Project Location, select a Project Folder using the Choose Box

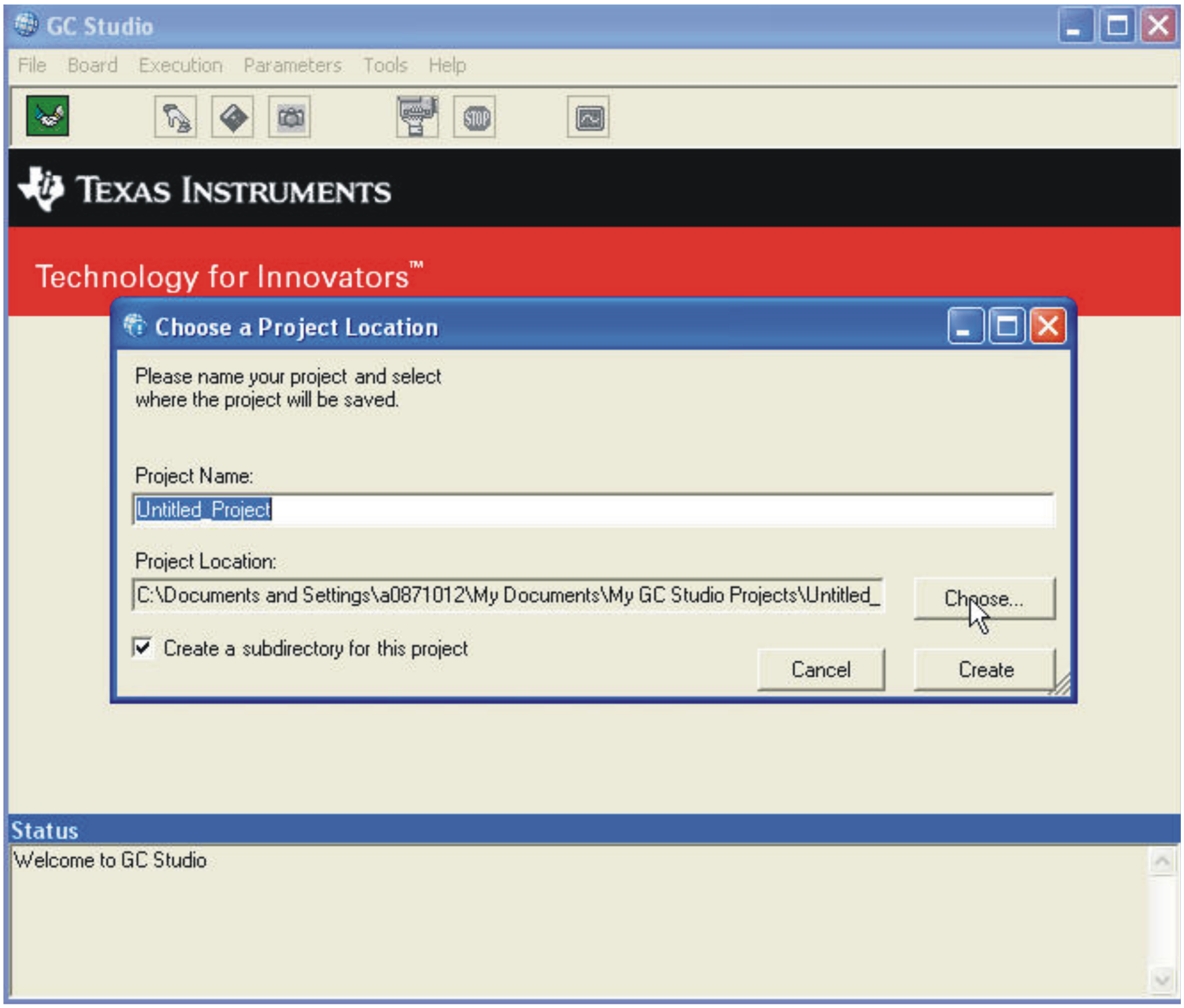


Figure 64. GC Studio GC5016 Choose New Project Location

3.1.3 Navigate to the Desired Folder Location or Create Your Own Location

Note: Place the new experiments in different folder / different file names from the supplied experiments. In this manner, the GC Studio software experiments that are re-installed will not overwrite the new projects.

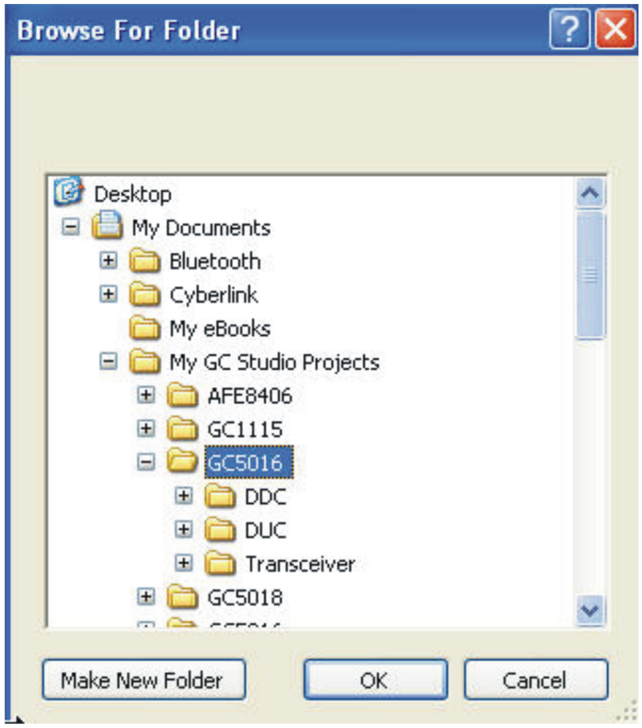


Figure 65. GC Studio GC5016 Folder Locations

Once the folder has been created (Make new Folder) or selected, the project name is entered. If the ‘Create a subdirectory for this project’ is selected, the project is saved to a folder with the project name.

Then select Create.

3.1.4 Select the GC5016 Plug-in and click OK

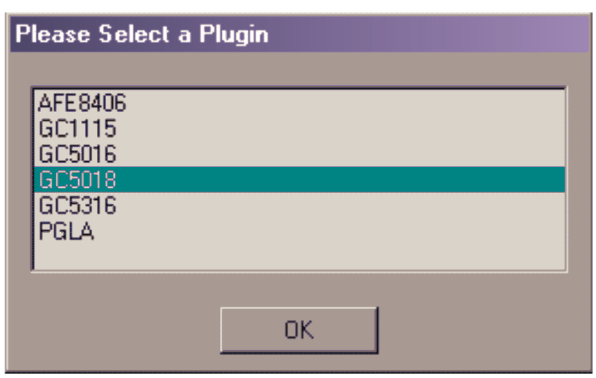


Figure 66. GC Studio GC5016 Plug in Selection

Using the wizard style interface, setup the GC101 evaluation board, then click Next.

3.1.5 the GC101 Stimulus Memory, Response Memory, and Clock Parameters

3.1.5.1 Stimulus Memory

The stimulus memory provides the Input data and sync for the GC5016 plug in board. If the experiment data comes from a(an) input file(s), set the Mode to Source, as shown in Figure 67.

The data path should be selected to "Memory to Daughtercard".

The memory length needs to be selected for proper time framing, and based on the input data provided. The Memory Length must be modulo 4, and be <= 2^{20}-4 values.

The Bank Skew controls are described in the GC101 User's Guide, and can be adjusted to allow for higher speed or special external data clock skew. In most cases, at 60 MHz, the values of 4 are the default.

The different banks are used to control the timing for:

1. Digital Input, InputFPGA
2. Input SRAM
3. DIMM (daughterboard) clock
4. DIMM Input Register, and clock to Output Clk PLL

- Sync Source – this control is used to start the Stimulus memory. Force is the default value.
- External Sync Source – this applies to the Sync source, On-Board indicates that the sync comes from the stimulus memory. SMA and Digital Input connector are different hardware connections if an external sync source is used.

Note: If the input data selected has less samples than the memory size, the input is repeated.

Note: If an ADC input or other digital input is used, the Data Path is changed to "External In to Memory to Daughtercard". The Mode can also be changed to "Capture".

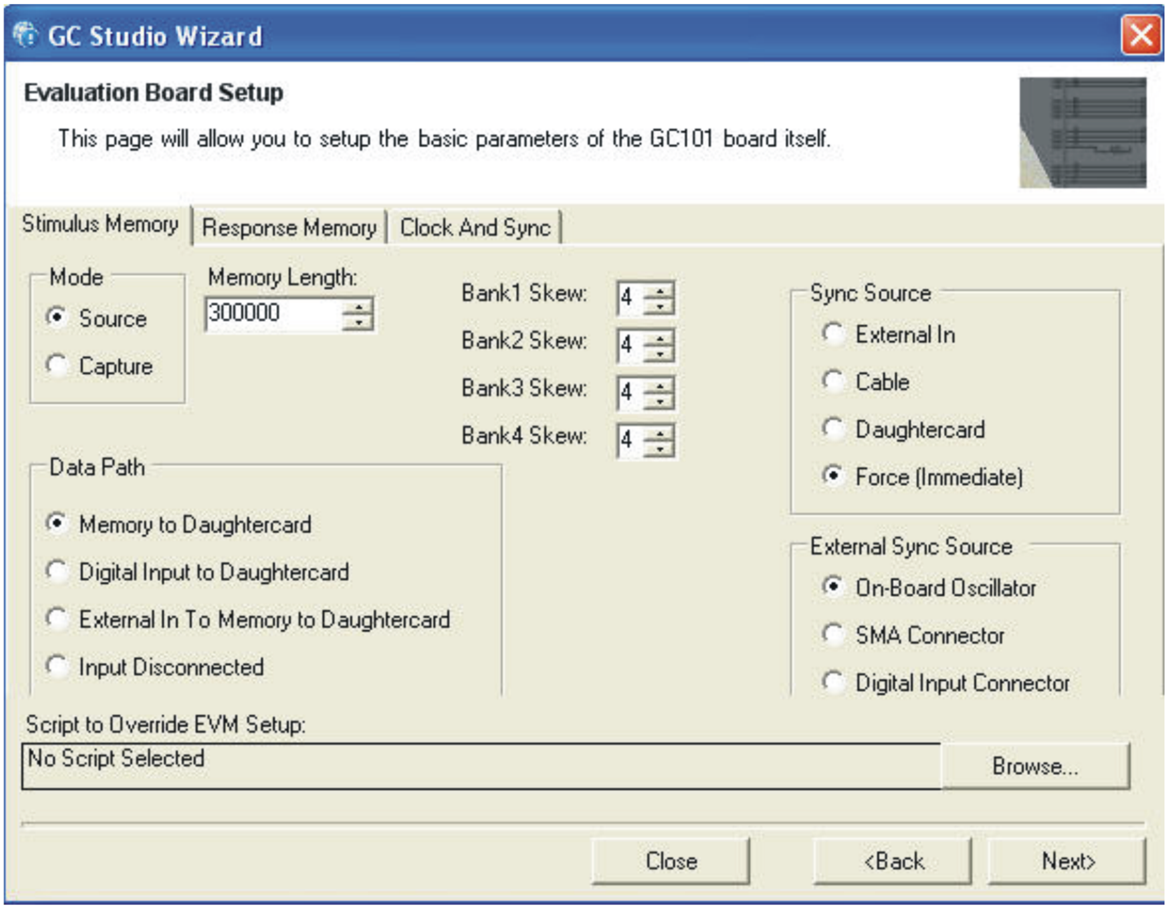


Figure 67. Stimulus Memory Selections

3.1.5.2 Response Memory

The GC5016 DDC or DUC output is available for capture from the selected ports (based on DIP switches 1,2,3,4). The Response Memory is typically capturing this output data once. The settings used by the Response Memory is shown in Figure 68.

The Response Memory for the GC5016 uses the Capture Mode only. The data path for the GC5016 experiments uses the default daughtercard to Memory to Digital Out. The Sync Source for all of the standard experiments uses the Sync Source set to Stimulus memory.

The Memory Length needs to be a modulo 4 number, that captures the desired output frame and startup transient.

The Bank Skew controls are described in the GC101 Hardware User's Guide and can be adjusted to allow for higher speed or special external clock skew. In most cases, at 60MHz, the values of 4 are the default.

The different banks are used to control the timing for:

1. Output FPGA, registers after DIMM
2. Digital Output registers
3. Digital Output clock
4. Capture SRAM

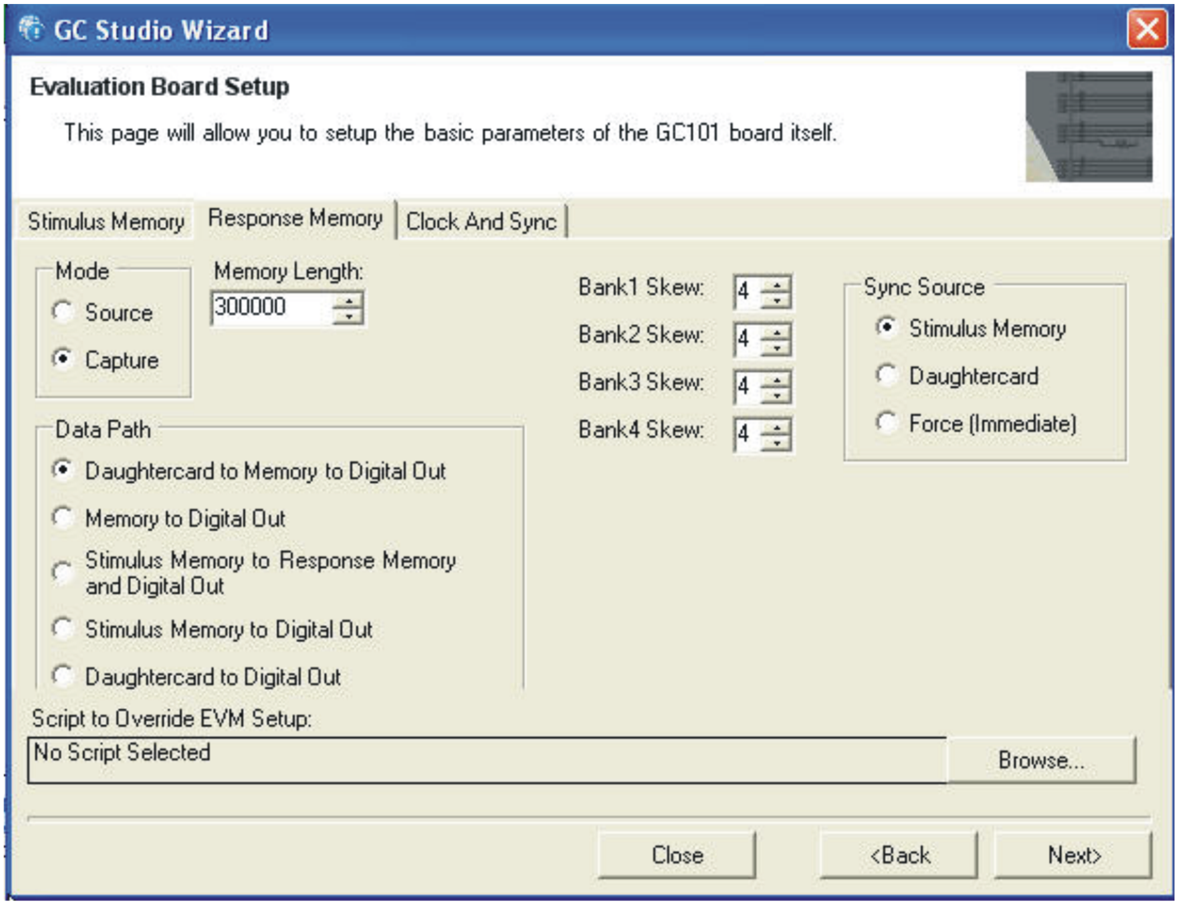


Figure 68. Response Memory Selections

3.1.5.3 Clock and Sync

The clock and sync section can be set for a variety of configurations (See Figure 69):

- Internal Clock
- External Clock from ADC, external data
- External Clock, output data to DAC

The Internal Clock mode operates the stimulus memory, daughtercard, and response memory at a clock rate that doesn't need to match the customer target clock rate.

In the standard internal clock mode, the Clock and Sync values are set to:

- Onboard Oscillator
- Input Clock Frequency (matches oscillator in socket) – 10MHz
- Use Analog PLL (if Input Clock Frequency is 10-27MHz)
- Multiplier selected to 6
- Response Memory set to Stimulus Clock

In the External Clock from ADC, external data mode, the Clock and Sync values are set to:

- ENC clock from Digital Input Connector or SMA clock from external clock source
- Input Clock Frequency matches input clock rate
- Use Analog PLL is unchecked, and Multiplier is set to 1

3.1.5.4 Response Memory set to Stimulus Clock

Note: The Stimulus Clock Skew needs to be adjusted to meet the external data setup and hold time relative to the bank 1 timing and external clock.

In the External Clock, output data to DAC mode, the Clock and Sync values are set to:

- SMA clock from external clock source
- Input Clock Frequency matches input clock rate
- Use Analog PLL is unchecked, and Multiplier is set to 1
- Response Memory set to Stimulus Clock

Note: The Stimulus Clock and Response Clock need to be adjusted to meet the external data Digital output timing relative to the DAC clock signal.

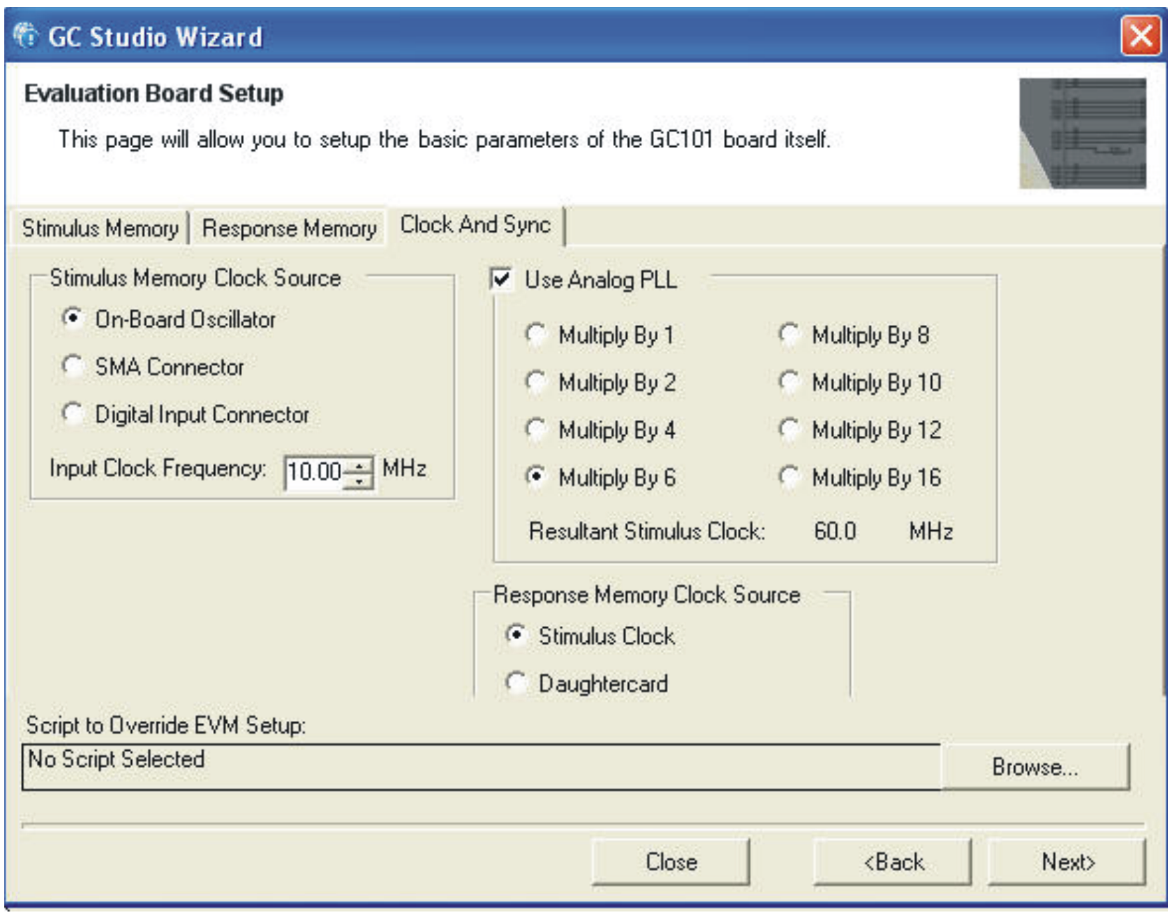


Figure 69. GC101 Clock and Sync Selection

After selecting the Stimulus, Response, and GC101 Clock and Sync selections, click Next.

3.1.6 GC5016 Channel Mode Options

The continuing Wizard setup of the channel or the Parameters GC5016 Device Mode is used to set The top level GC5016 settings, and configures the GC Studio top level software.

The channel pairs A & B, and C & D can be separately configured for Receive or Transmit. There are three normal options:

- 4 channel Receiver → AB RECEIVE, CD RECEIVE
- 4 channel Transmitter → AB TRANSMIT, CD TRANSMIT
- 2 channel Transceiver → AB TRANSMIT, CD RECEIVE

The SplitIQ configuration combines the PFIRs of the two channels AB or CD.

These configuration items must be done before configuring the individual channel options for the GC5016.

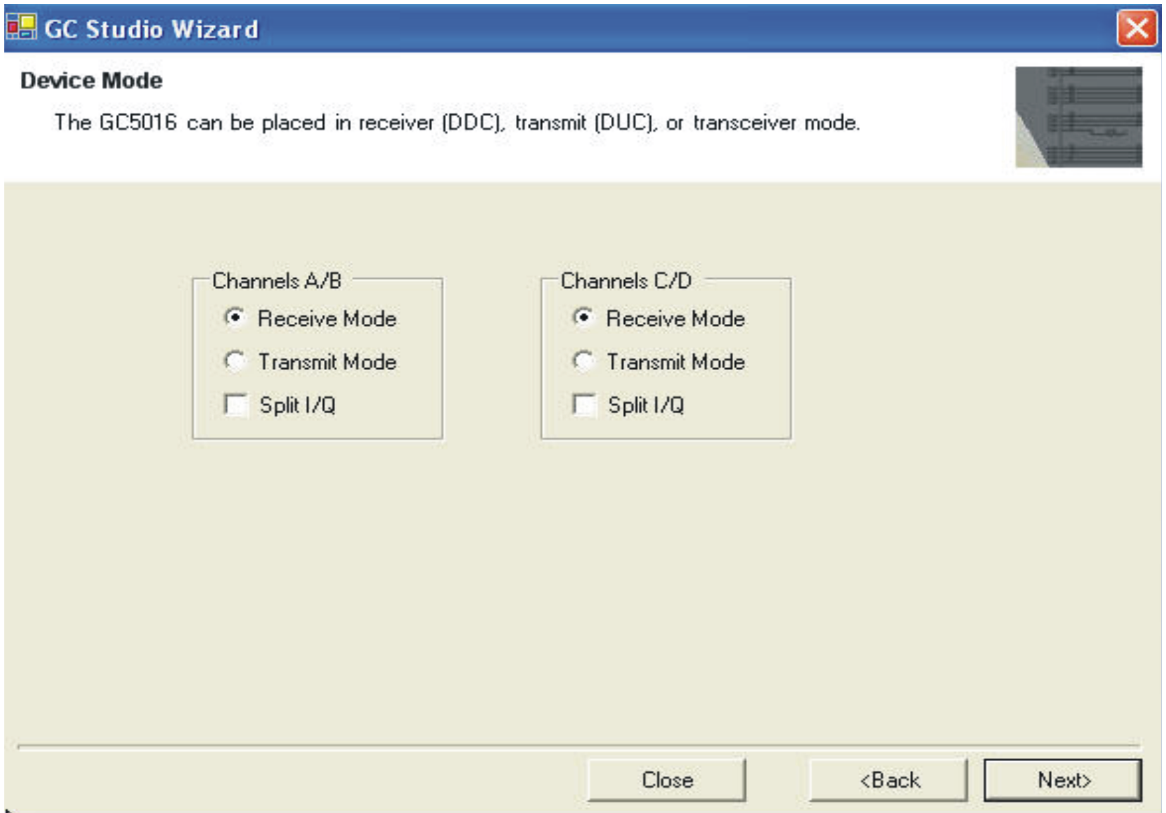


Figure 70. GC Studio and GC5016 Device Mode

3.1.7 GC Studio Sync to Input Delay

The Sync to Input Delay can be left as the default setting until the other channel items are specified. This delay is only used with Interleaved IQ or TDM IQ DUC inputs, to calibrate the data delay. This is further described in the DUC Input section.

This control delays the data by a listed number of Ck clocks after the synchronization. This control must be configured after the user channel and data configuration, to get the correct data at a specific time.

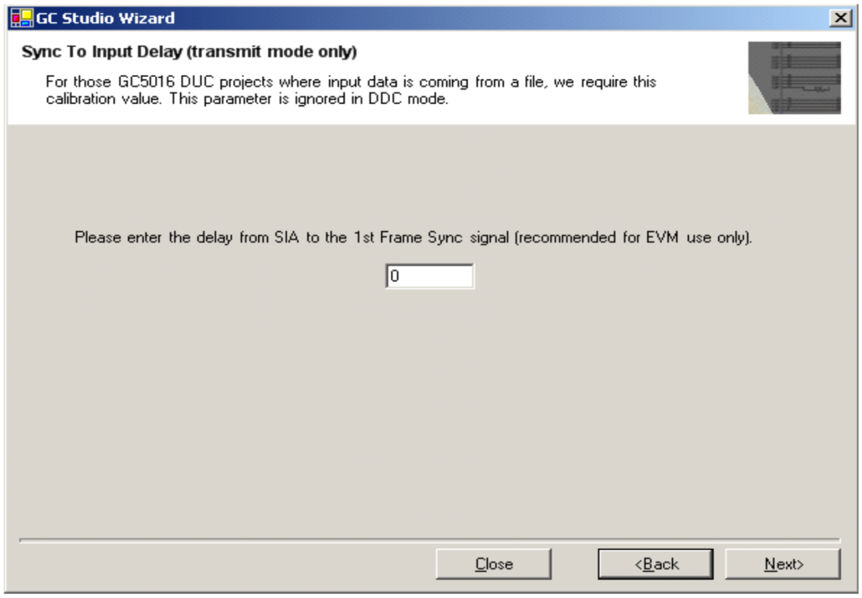


Figure 71. 53 GC Studio and GC5016 Device Mode

3.1.8 GC Studio Channel Copy

The GC Studio software can copy the settings of one channel to another. This minimizes errors when making the DDC or DUC channels the same. There are limitations for Transceiver configurations. This screen is used after the first channel is configured, in creating a project.

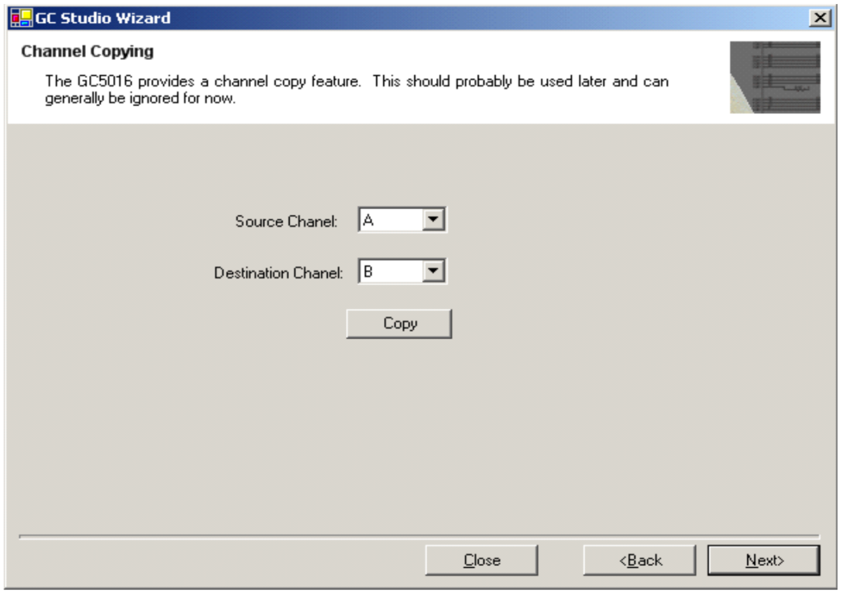


Figure 72. GC Studio and GC5016 Device Mode

3.1.9 GC5016 Input File, GC5016 configuration, GC5016 Output configurations

Section 2.2.5 can be followed to setup a DDC configuration. Section 2.2.6 can be followed to setup a DUC configuration. The Guide for using CMD5016, its examples, and other GC Studio projects are good references to assist the user with testing the GC5016.

3.2 GC5016 Projects included in the GC Studio Release

The following example projects are included in the GC Studio Release:

3.2.1 DDC Experiments

- DDC_4chan_sinein_tdmAGC_out – This experiment uses a fixed gain and AGC gain for a slowly changing sine wave input, to show the changes in output amplitude.
- DDC_cdma2k_64x – example CDMA 64x input 2x output multi-carrier 4-channel DDC example, TDM output mode, includes fixed gain and AGC
- DDC_dec60_step – used for determining the latency through the DDC channel, measured From the input offset zero crossing, to the DDC output zero crossing.
- DDC_diagconst_in\DDC_intIQ_20bits_8pins_7sck_div – DDC internal tone, tests IQ output mode. 8 data pins, Frame Strobe and channel clock are used to output 20bitsI and 20 bit Q data, minimum decimation of 6.
- DDC_diagconst_in\DDC_SplitIQ_ParIQ_16bits_16pins_0sck_div – DDC internal tone, test parallel IQ output, 32 data pins, Frame Strobe, and channel clock are used to output 16bits I and 16bits Q data, minimum decimation of 1.
- DDC_diagconst_in\DDC_SplitIQ_TDMIQ_16Bits_16Pins_0sck_div - DDC internal tone, test TDM IQ output, 16 data pins, Frame Strobe, and channel clock are used to output two channels of 16bits I and 16bits Q data, minimum decimation of 4.
- DDC_diagconst_in\DDC_SplitIQ_TDMIQ_20Bits_8Pins_0sck_div – DDC internal tone, test TDM IQ output, 8 data pins, Frame Strobe, and channel clock are used to output two channels of 20bits I and 20bits Q data, minimum decimation of 12.
- DDC_diagconst_in\TDMIQ_16Bits_16Pins_0sck_div – DDC internal tone, test TDM IQ output, 16 Data pins, Frame Strobe, and channel clock are used to output 4 channels of 16bits I and 16 bits of Q, minimum decimation of 8.
- DDC_diagconst_in\DDC_TDMIQ_20Bits_8Pins_0sck_div - DDC internal tone, test TDM IQ output, 8 Data pins, Frame Strobe, and channel clock are used to output 4 channels of 20bits I and 20bits of Q, minimum decimation of 24.
- DDC_realin_intiq20 – Real A input, two tone, 4 DDC channels using interleaved IQ outputs.
- DDC_umtsrcv_32x – Real A, 4 carrier UMTS input, TDM DDC output, fixed and AGC gain modes, 2x output.
- gc5016_ddc_splitiq_ofdm10x – 2 DDC splitIQ channels with parallel IQ output. 11.52MHz BW, 10MHz Wimax channels.
- RxSplitIQ_100_dec2 – Complex Input DDC, CIC is bypassed , PFIR decimates by 2. Parallel Complex Output.
- DDC_4carTDSCDMA – multi-carrier DDC 4-channel, 1 Real input, TDM output

3.2.2 DUC Experiments

- DUC_2chan_ParIQout_IntIQ_Interp64 – DUC experiment supports Interleaved IQ input for ChA,B; and Interleaved IQ Output Multi-carrier on Channel A, Interpolation of 64.
- DUC_diagconstin\ DUC_IntIQAB_16Bits_16Pins – 4 channel tone diagnostic input, each carrier has a separate output
- DUC_diagconstin\ DUC_ParIQAB_16Bits_16Pins – 4 channel tone diagnostic input, all 4 carriers are summed to one output. Parallel IQ output mode.
- DUC_diagconstin\ DUC_RealAB_16Bits_16Pins – 4 channel tone diagnostic input, carriers A and B output on portA, carriers C and D output on port B. Real output mode.
- DUC_QAM256_8I8Q_4ch2Out_RealOut – 4 channel DUC, 2channel A , 2 channel B real output. 5.36Mbaud IQ input, 8bit parallel I and Q.
- UMTS_4car_ParIQout_TDMIn_32x – 4 channel DUC, TDM input, 3 active carriers UMTS/WCDMA Parallel IQ output
- gc5016_duc_splitiq_ofdm10x_0IF – 2 splitIQ channel DUC, parallel IQ input, parallel IQ output 10MHz Wimax OFDM

- DUC_SplitIQ_Int4 – SplitIQ Interpolate by 4, Parallel IQ Output – illustrates 22MHz BW channel.
- DUC_4carTDSCDMA – 4 carrier TDSCDMA, TDM input, Complex Output
- CDMA_2chan_ParIQout_IntIQin_64x_latencytest - step response latency test from DUC input to output

3.3 GC Studio References

For more information on GC Studio, see the GC Studio User's Manual included with the software distribution. GC Studio also contains a powerful scripting language, the SCR GC101 Language Reference, also included in the software distribution, for expert users.

Selecting Help → User's Manuals in the GC Studio menu to access these documents.

Other References include:

- GC5016 Data Sheet (SLWS142)
- Using CMD5016 Programming guide and examples
- GC5016 IO Application Note (SWLA037)
- GC5016 DUC Gain Application Note (SLWA033)
- GC5016 AGC DDC Application Note (SLWA032)
- GC5016 DDC Block Diagram with programming variables
- GC5016 DUC Block Diagram with programming variables
- GC101 EVM Hardware Description (SLWU018)
- GC101 Rev C Schematic Diagram
- GC5016 Daughtercard Description (SLWU019)
- GC5016 Rev B Schematic Diagram

4 Physical Description

This section describes the physical characteristics and PCB layout of the EVM and lists the components used on the module.

4.1 PCB Layout

The EVM is constructed on a 6-layer, 2.6-inch x 5.25-inch, 0.05-inch thick PCB using FR-4 material. Figure 73 through Figure 78 show the PCB layout for the EVM.

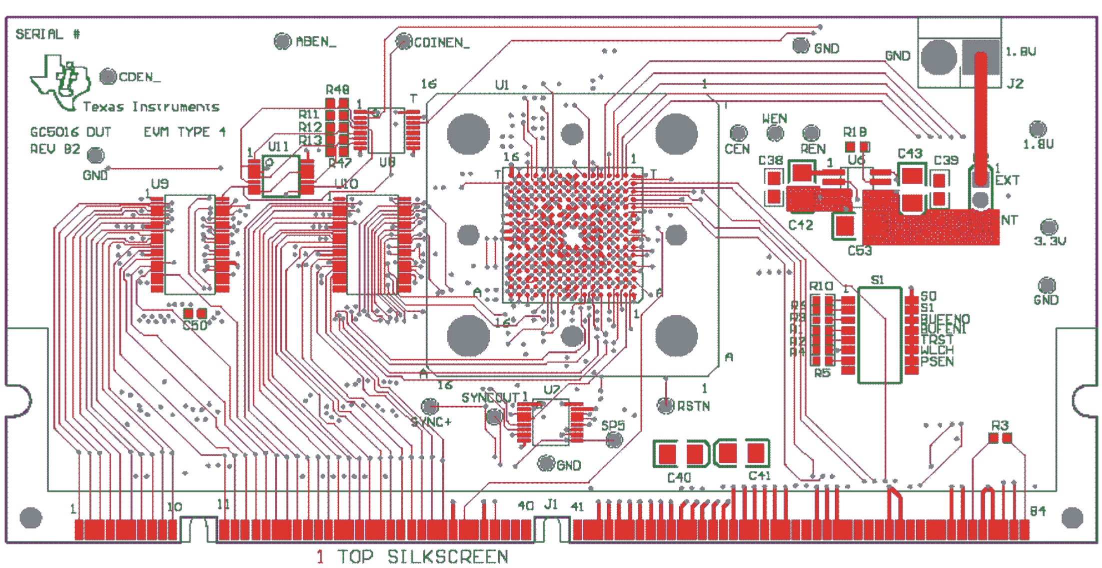


Figure 73. Top Layer 1

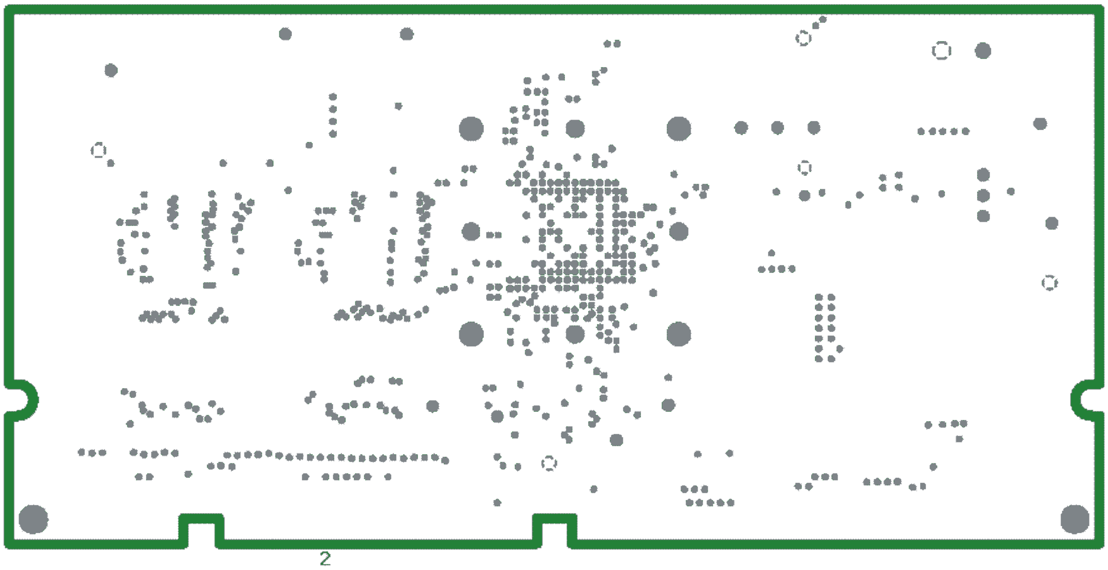


Figure 74. Ground Plane Layer 2

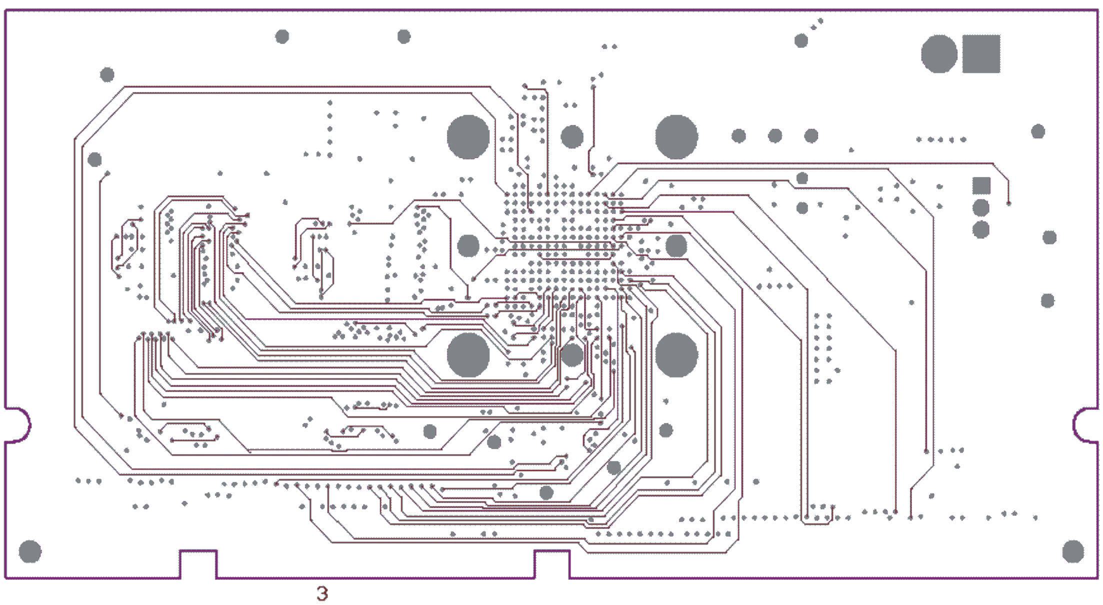


Figure 75. Signal Layer 3

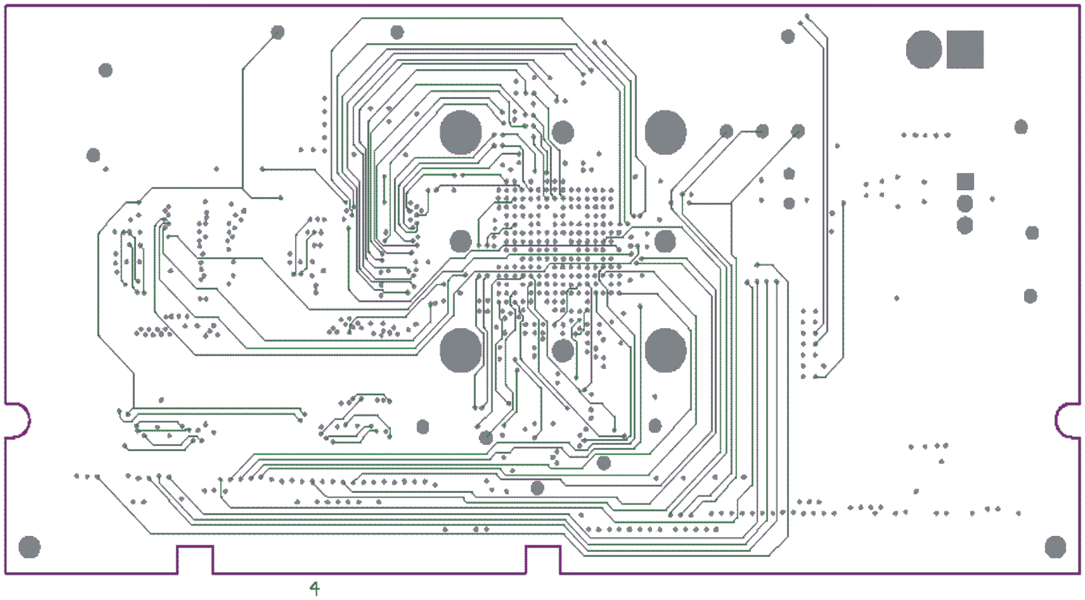


Figure 76. Signal Layer 4

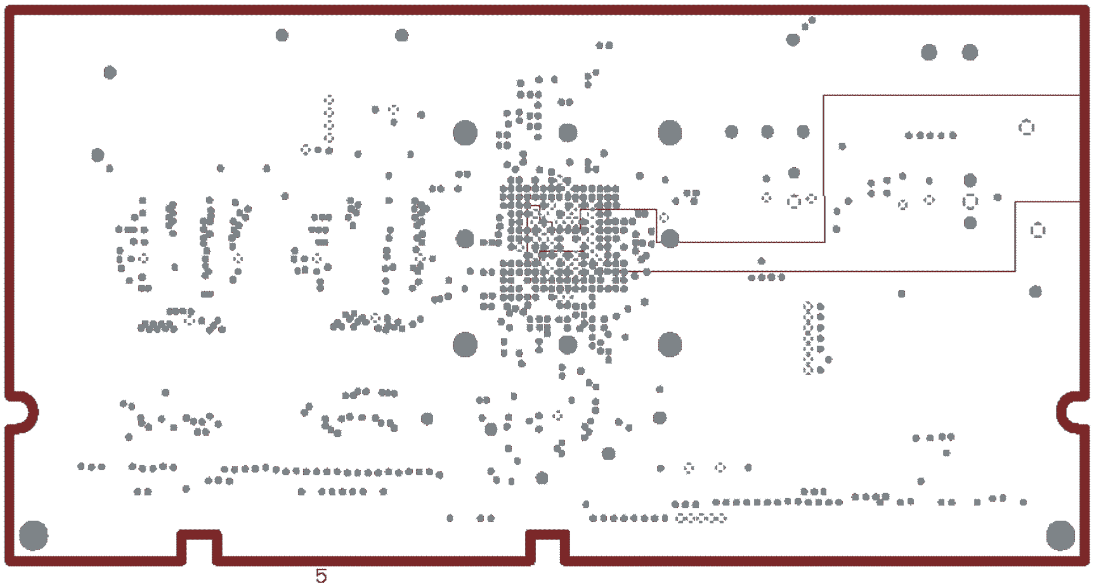


Figure 77. Power Plane Layer 5

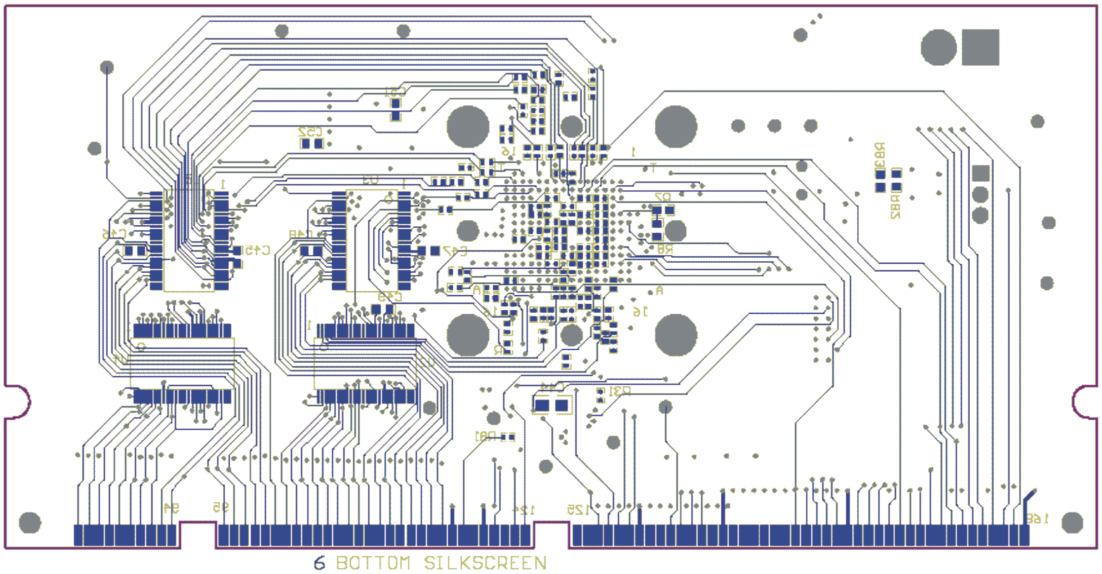


Figure 78. Bottom Layer 6

4.2 Parts List

Table 6 lists the parts used in constructing the EVM.

Table 6. Bill of Materials for GC5016 EVM

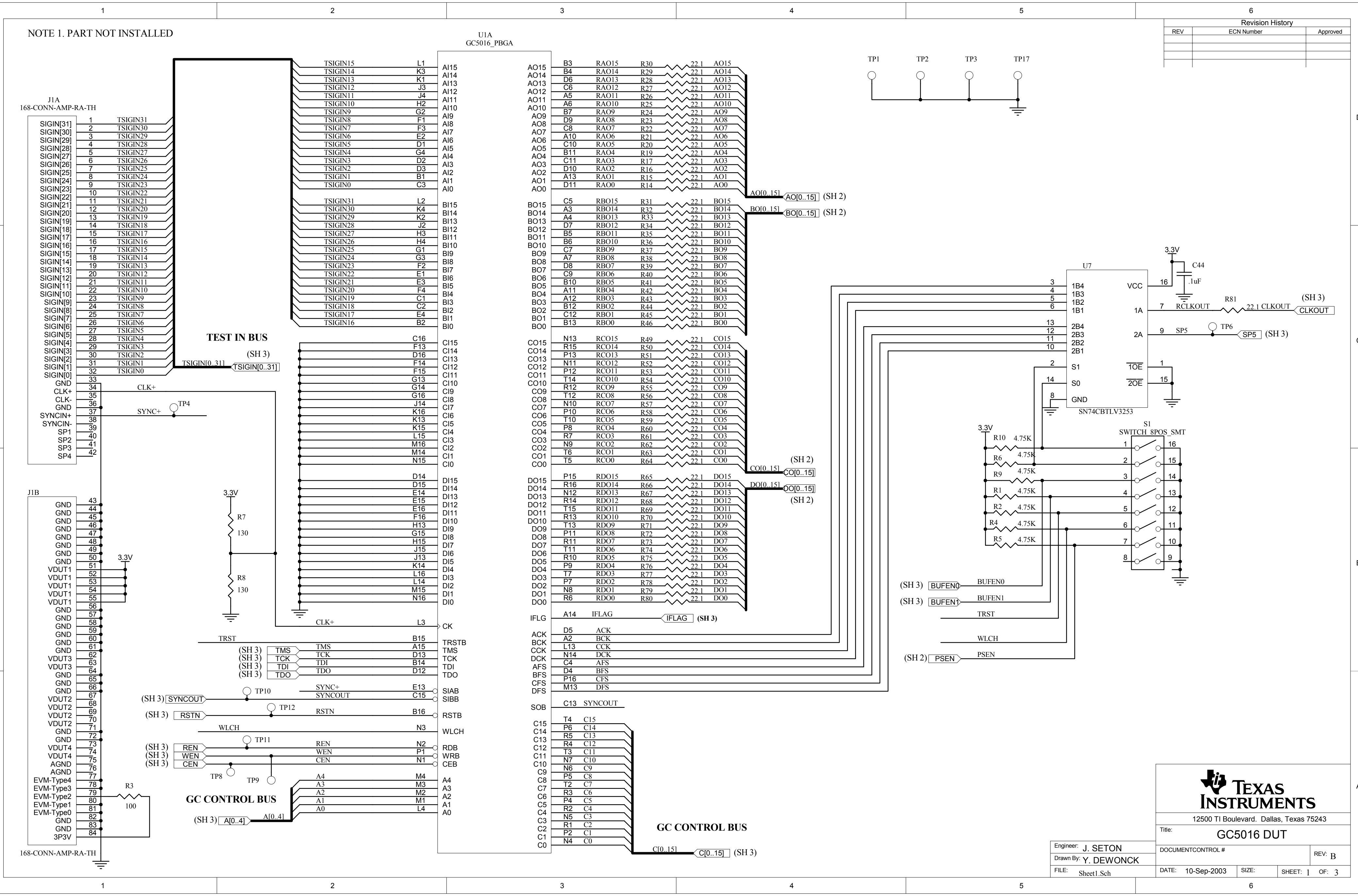
VALUE	QTY.	PART NUMBER	MANUFACTURER	REF DES	NOT INSTALLED
2.2 µF, 20 V, 10%, Capacitor	1	ECS-T1DX225R	PANASONIC	C40	
10 µF, 20 V, 10%, Capacitor	4	ECS-T1DX106R	PANASONIC	C41 C42 C43 C53	
0.1 µF, 16 V, 10%, Capacitor	3	ECJ-2VB1C104K	PANASONIC	C38 C39 C44	
0.1 µF, 16 V, 10%, Capacitor	8	ECJ-1VB1C104K	PANASONIC	C45-C52	
0.1 µF, 16 V, 10%, Capacitor	37	ECJ-0EF1C104Z	PANASONIC	C1-C37	
100 Ω resistor, 1/16 W, 1%	1	ERJ-3EKF1000V	PANASONIC	R3	
130 Ω resistor, 1/16 W, 1%	2	ERJ-3EKF1300V	PANASONIC	R7 R8	
0 Ω Resistor, 1/16 W, 1%	1	ERJ-3GEY0R00V	PANASONIC	R47	
4.75 kΩ Resistor, 1/16 W, 1%	11	ERJ-3EKF4752V	PANASONIC	R1R2 R4 R5 R6 R9-R13 R48	
30.1 kΩ Resistor, 1/16 W, 1%	1	ERJ-3EKF3012V	PANASONIC	R18	
15.8 kΩ Resistor, 1/16 W, 1%	1	ERJ-3EKF1582V	PANASONIC	R83	
249 kΩ Resistor, 1/16 W, 1%	1	ERJ-3EKF2491V	PANASONIC	R82	
22.1 Ω resistor, 1/16 W, 1%	65	ERJ-2RKF22R1V	PANASONIC	R14-R17 R19-R46 R49-R81	
Red test point	12	5000k	KEYSTONE	TP4-TP6 TP8-TP16	

Table 6. Bill of Materials for GC5016 EVM (continued)

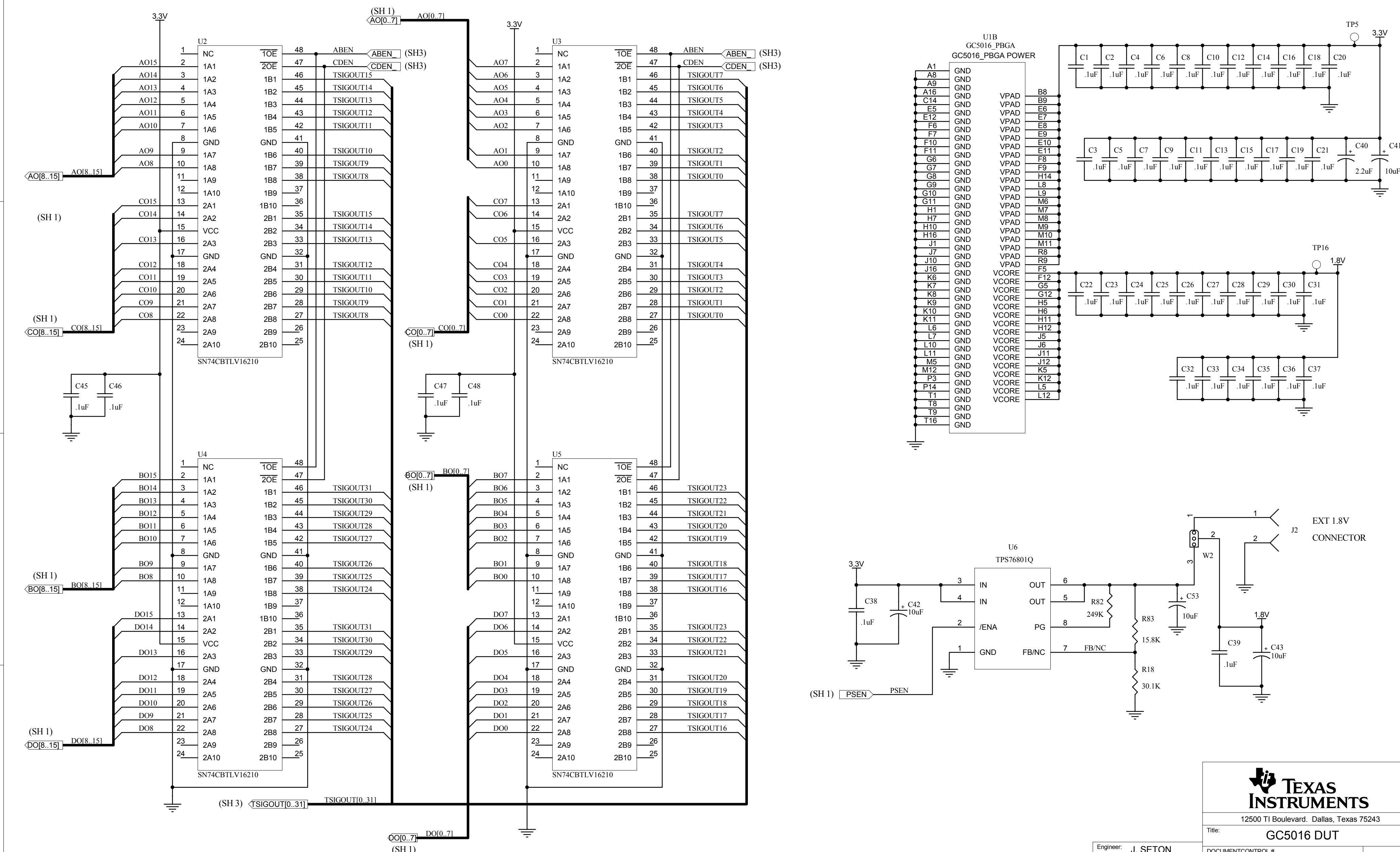
VALUE	QTY.	PART NUMBER	MANUFACTURER	REF DES	NOT INSTALLED
Black test point	4	5001k	KEYSTONE	TP1 TP2 TP3 TP17	
CON_2TERM_SCREW	0	KRMZ3	LUMBERG		J2
Switch_8pos_smt	1	DHS8S	APEM	S1	
GC5016_PBGA	1	GC5016-PB	Texas Instruments	U1	
SN74CBTLV16210	6	SN74CBTLV16210GR	Texas Instruments	U2-U5 U9 U10	
SN74CBTLV3253	2	SN74CBTLV3253PWR	Texas Instruments	U7 U8	
SN74LVC08	1	SN74LVC08APWR	Texas Instruments	U11	
TPS76801Q	1	TPS76801QD	Texas Instruments	U6	

4.3 Schematics

This section contains the schematics for the EVM.



NOTE 1. PART NOT INSTALLED
U1A
GC5016_PBGA
J1A
168-CONN-AMP-RA-TH
J1B
168-CONN-AMP-RA-TH
TEST IN BUS
TSIGIN[0..31]
(SH 3)
GC CONTROL BUS
A[0..4]
C[0..15]
AO[0..15] (SH 2)
BO[0..15] (SH 2)
CO[0..15] (SH 2)
DO[0..15] (SH 2)
IFLAG (SH 3)
SYNCOUT (SH 3)
RSTN (SH 3)
REN (SH 3)
WEN (SH 3)
CEN (SH 3)
TMS (SH 3)
TCK (SH 3)
TDI (SH 3)
TDO (SH 3)
BUFEN0 (SH 3)
BUFEN1 (SH 3)
PSEN (SH 2)
CLKOUT (SH 3)
SP5 (SH 3)
U7
SN74CBTLV3253
S1
SWITCH_8POS_SMT
C44 .1uF
R81 22.1
R7 130
R8 130
R3 100
3.3V
TP1 TP2 TP3 TP17 TP4 TP6 TP8 TP9 TP10 TP11 TP12
Revision History
REV
ECN Number
Approved
TEXAS INSTRUMENTS
12500 TI Boulevard. Dallas, Texas 75243
Title: GC5016 DUT
Engineer: J. SETON
Drawn By: Y. DEWONCK
FILE: Sheet1.Sch
DOCUMENTCONTROL #
REV: B
DATE: 10-Sep-2003
SIZE:
SHEET: 1 OF: 3

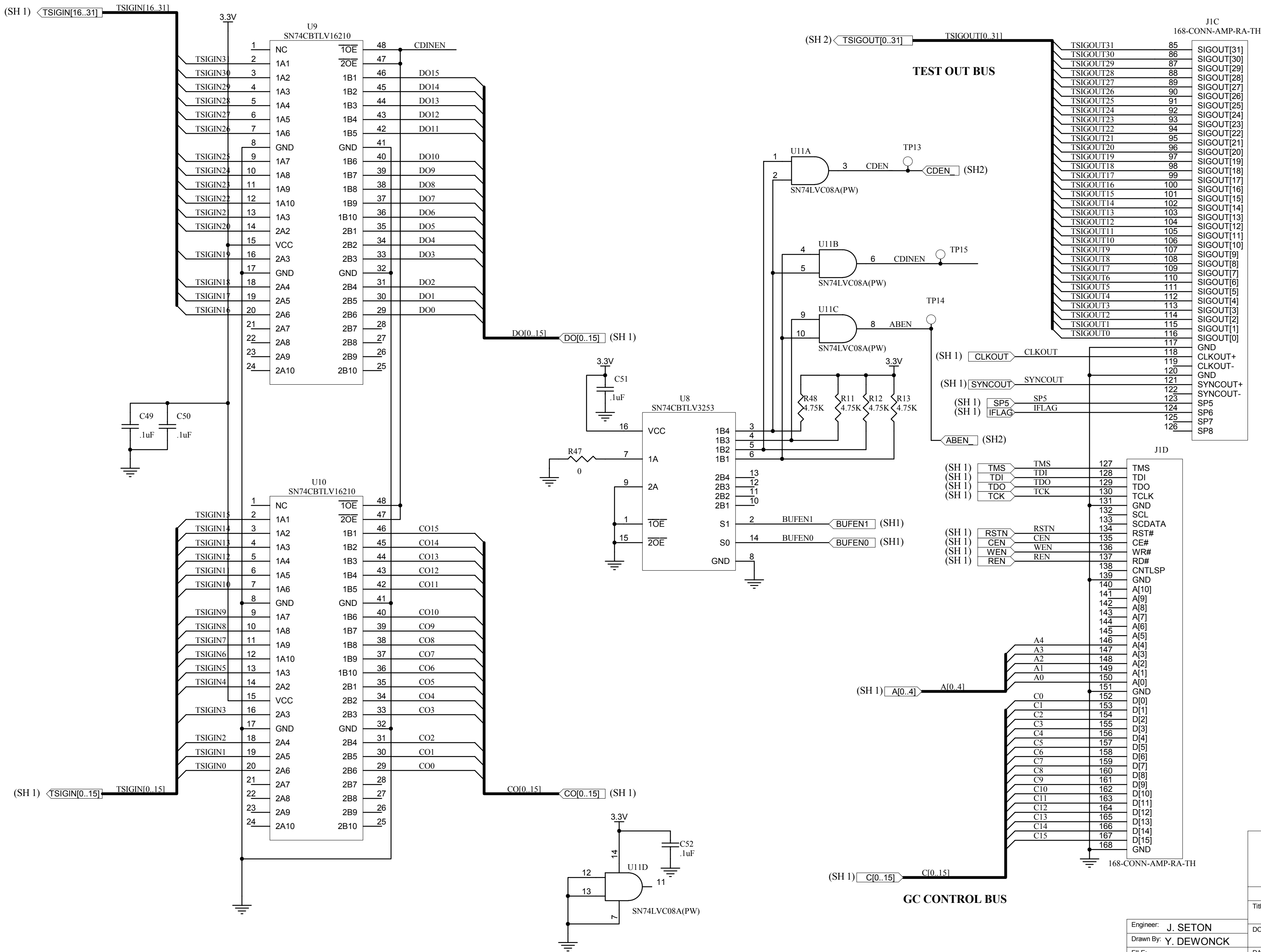


NOTE 1: PART NOT INSTALLED

TEXAS INSTRUMENTS

12500 TI Boulevard. Dallas, Texas 75243

Engineer: J. SETON	Title: GC5016 DUT		
Drawn By: Y. DEWONCK	DOCUMENTCONTROL #		REV: B
FILE:	DATE: 10-Sep-2003	SIZE:	SHEET: 2 OF: 3



TEXAS INSTRUMENTS

12500 TI Boulevard. Dallas, Texas 75243

Title: GC5016 DUT

Engineer: J. SETON

Drawn By: Y. DEWONCK

FILE:

DOCUMENTCONTROL #

REV: B

DATE: 10-Sep-2003

SIZE:

SHEET: 3 OF: 3

EVALUATION BOARD/KIT IMPORTANT NOTICE

Texas Instruments (TI) provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by TI to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and therefore may not meet the technical requirements of these directives or other related directives.

Should this evaluation board/kit not meet the specifications indicated in the User's Guide, the board/kit may be returned within 30 days from the date of delivery for a full refund. THE FOREGOING WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY SELLER TO BUYER AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE.

The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies TI from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

TI currently deals with a variety of customers for products, and therefore our arrangement with the user **is not exclusive.**

TI assumes **no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.**

Please read the User's Guide and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages. For additional information on TI's environmental and/or safety programs, please contact the TI application engineer or visit www.ti.com/esh.

No license is granted under any patent right or other intellectual property right of TI covering or relating to any machine, process, or combination in which such TI products or services might be or are used.

FCC Warning

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by TI to be a finished end-product fit for general consumer use. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

IMPORTANT NOTICE

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265