Analog Simulation and Testing via FPAA

Jeffrey Teng, Fall 2016

Introduction

This document discusses the process of performing real measurements on analog circuits using an FPAA remote system. The software required for utilizing the FPAA has been built into an Ubuntu virtual machine based on Oracle's VirtualBox.

Upon successful installation of required software, users of the FPAA system will be able to

1. Simulate analog circuits in Xcos.
2. Compile circuits to be programmed on the FPAA.
3. Take real measurements via remote system.

This guide is intended to be a quick-start manual, and focuses on installation, running simulations, and basic data collection. Details on the underlying software or hardware are not covered.

Initial Setup

1. Install VirtualBox

The entire system centers on Oracle's VirtualBox as a basis for a virtual machine. VirtualBox can be found here:

VirtualBox: https://www.virtualbox.org/wiki/Downloads

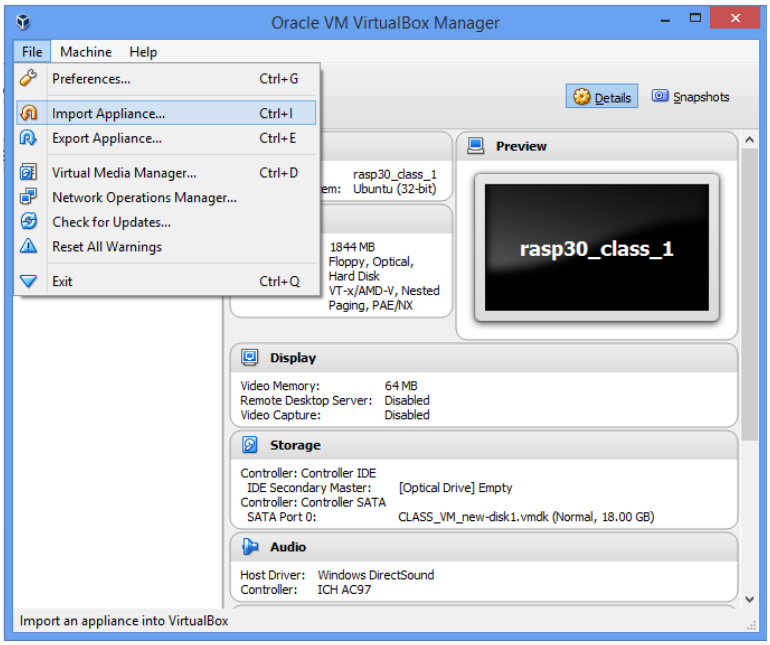
1. From the above link, download the binary file which corresponds to your operating system
2. From the above link, download the "VM VirtualBox Expansion Pack."
3. Install VirtualBox.
    a. When prompted to customize features, do not change anything.
    b. Creating a desktop shortcut and registering file extensions are recommended, but not required.
4. Once VirtualBox is successfully installed, run the expansion pack. This expansion pack allows compatibility with USB 2.0 and 3.0.

2. Configure Virtual Machine

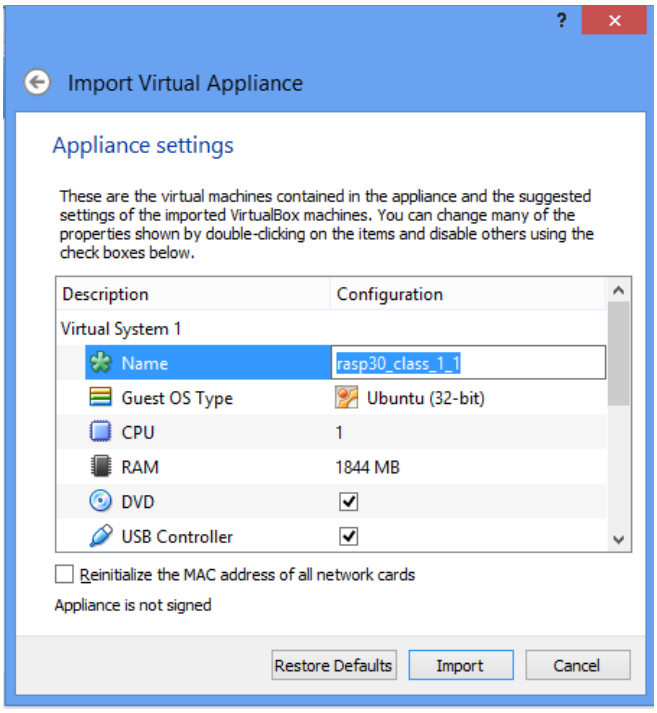
Once VirtualBox has been installed, the next step is to import the virtual machine which contains necessary software for interfacing with the FPAA system. The .ova file is found here:

Virtual Machine: http://users.ece.gatech.edu/~phasler/FPAAtool/class.ova

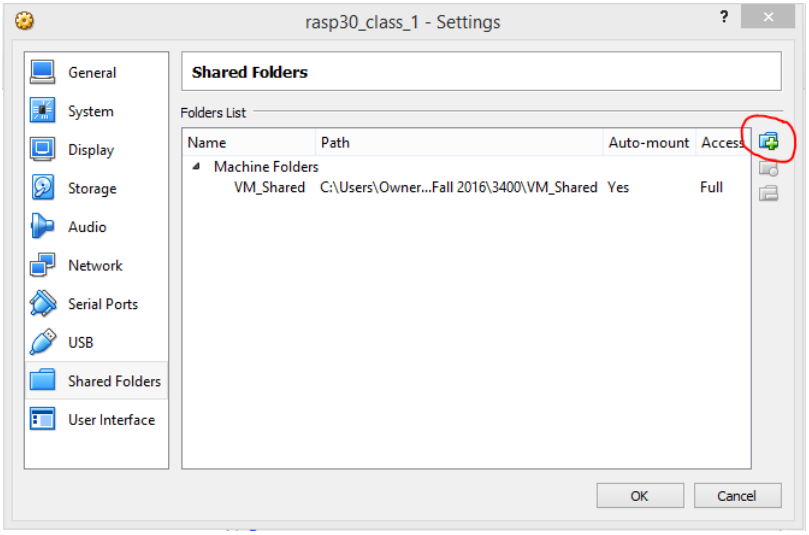
1. Download the .ova file from the above link.
2. Run VirtualBox. From the file menu, select "Import appliance…"



3. When prompted, navigate to the "class.ova" file.
4. Appliance settings should not be changed except for the name.



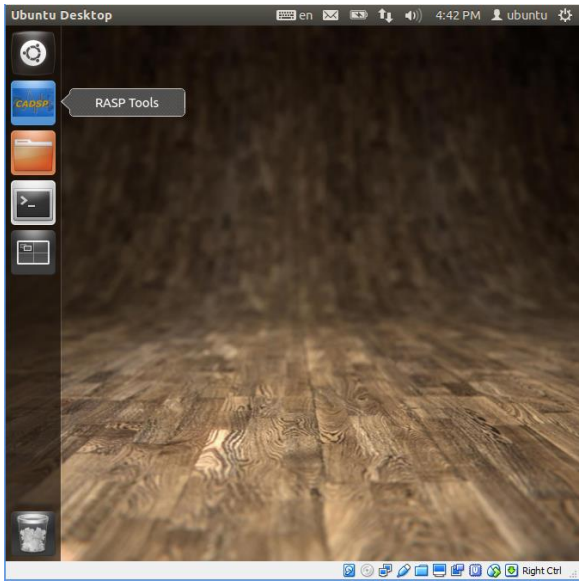
5. After the virtual machine has been imported, right-click on it and select “Settings…”
6. Navigate to “Shared Folders” in the left-hand menu. Shared folders are accessible from both your main operating system and the virtual machine. This allows quick file transfer between the two.
    a. Delete any existing folders.
    b. Add a new shared folder in a convenient location.
    c. When choosing a new folder, “auto-mount” should be selected.

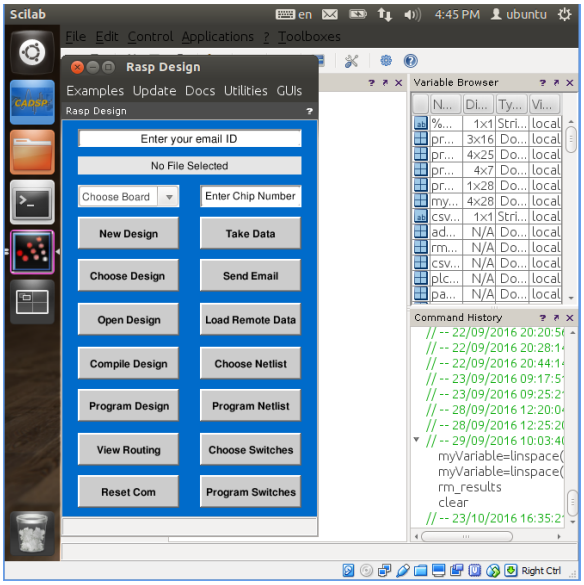


3. Setup RASP Tools

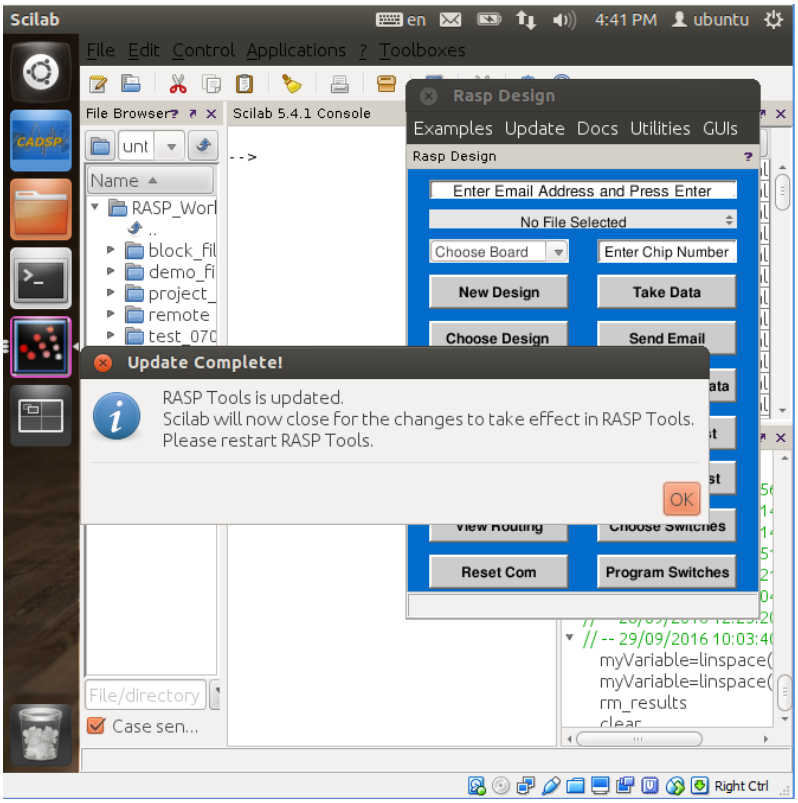
The virtual machine has now been configured and is ready to run.

1. Double-click on the virtual machine in VirtualBox to spawn a session.
2. When asked, the password is “**reverse**.”
3. After logging in click the blue icon for “CADSP” on the left-hand menu. **Wait until the blue GUI loads**.
    a. An icon for updates will appear. This can be ignored.





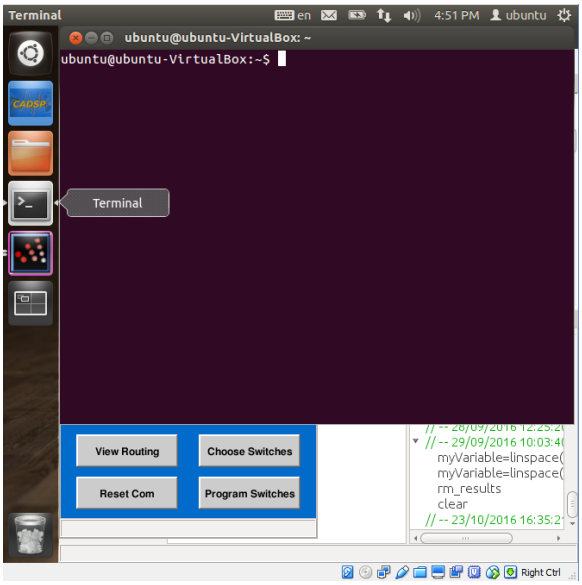
4. From the drop-down menu in the GUI, select “Updates” and choose “Update RASP Tools.” **Note that an internet connection is required**.
5. Once the update begins, a message box will appear which says to wait for an update confirmation to appear. **It may seem like nothing is happening, but DO NOT close Scilab during this time**.



6. When the update is finished, a second dialog box will appear. Click OK and Scilab will close to finalize the updates.

Navigating the Virtual Machine

The virtual machine uses a Unix/Linux style file system. From the left-hand menu, click on the "Terminal" icon to open the command-line interface.

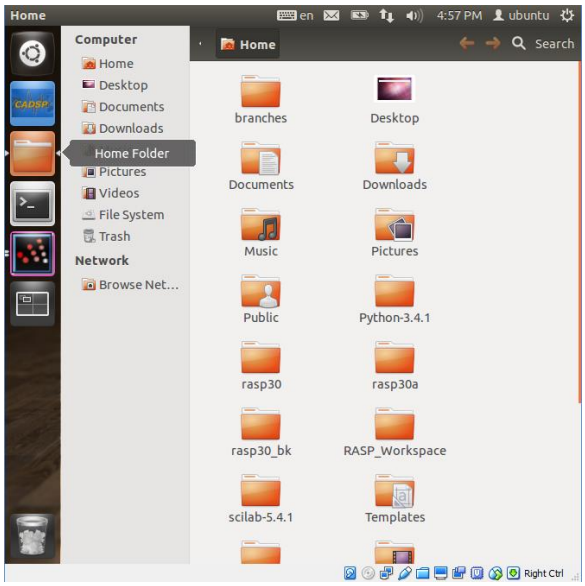


If you are unfamiliar with the command-line interface, appendix B details some basic command line operations. The following short course is also recommended to get a quick start.

Command Line Intro: https://www.codecademy.com/learn/learn-the-command-line

Please note that file and folder names CANNOT contain spaces.

Alternatively, file manipulation can be done via the folder browser. It can be opened from the left-hand menu.



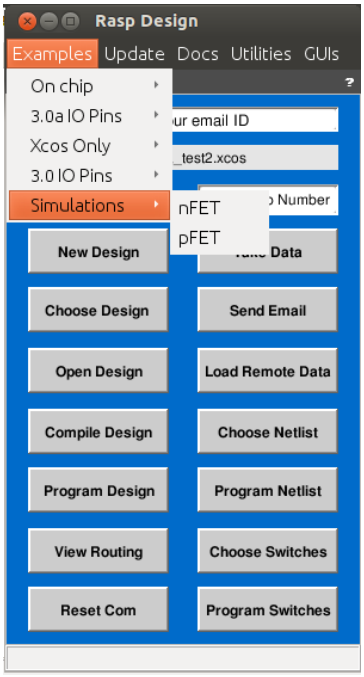
The shared folder that you created earlier is found in the "media" directory under the main file system.

Xcos

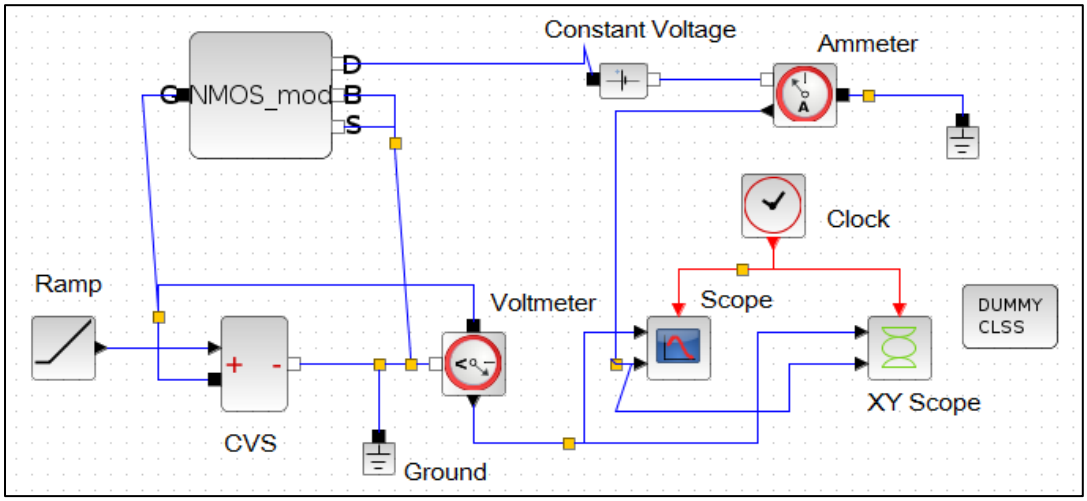
The FPAA system utilizes Scilab and Xcos. Scilab is an open-source clone of MATLAB, and most MATLAB commands can be executed in Scilab. Xcos is a system analysis tool similar to Simulink that is used to perform circuit level simulation.

First Example: nFET I-V Curve

The best way to illustrate using Xcos is with an example. From the blue GUI, select "Examples" from the drop-down menu, and navigate to "Simulations" and then "nFET."



This example shows how to run a drain current vs. gate voltage sweep.



First look at the **two different kinds of ports.** The triangular ports are explicit input/output ports and the square ports are implicit. **These two different ports CANNOT be connected together**. Implicit ports can be considered electrical nodes, where every connection to a node has the same voltage. The explicit ports denote signals flowing one direction.

Most blocks have parameters that can be set. The parameter menu is opened by double-clicking a block.

In terms of electrical blocks, this example uses

- Constant Voltage
- Ground
- Controlled Voltage Source (CVS)

The controlled voltage source takes a voltage value from its explicit input port and generates that voltage on its implicit port.

To generate an input signal, the **RAMP** block. Its parameters are the initial voltage and the slope of the ramp in volts per second. Its generated values are passed into the controlled voltage source, which actuates those values.

The **NMOS_mod** block contains the MOSFET model that will be used in ECE 3400. It uses the EKV model, which is special for being accurate in subthreshold operations, whereas most models consider subthreshold FETs to carry no current.

Two different kinds of meters are used. The **voltmeter** measures the voltage at the gate of the FET and outputs that on its explicit output port. The **ammeter** measures the current through the drain of the FET.

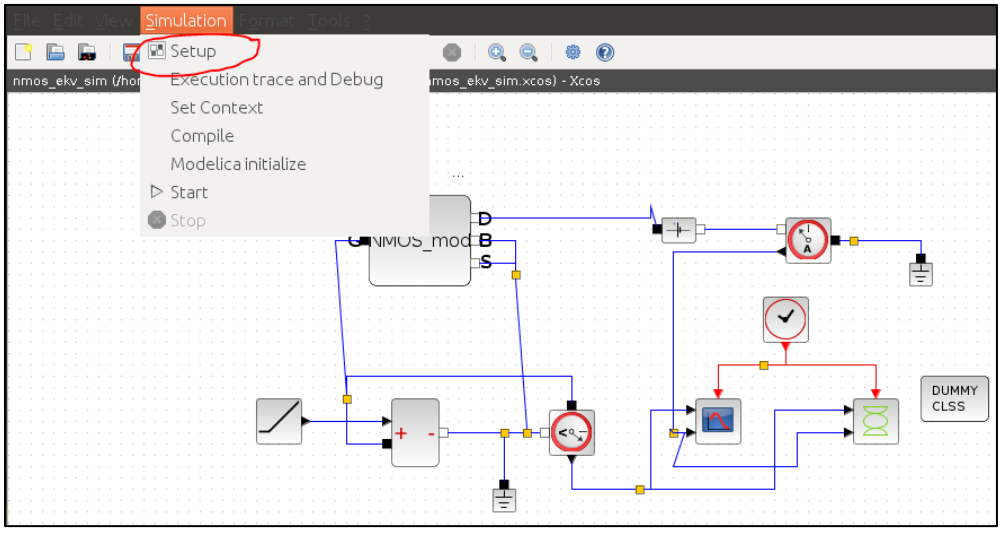
Two types of scopes are used to record the outputs of the voltmeter and ammeter. The **simple scope** creates plots of its two inputs vs. time. The **XY scope** plots its second input vs. its first input, giving the I-V curve.

Finally, a **clock signal** is necessary to synchronize the two scopes. **Timing connections are denoted by the red ports and connectors**.

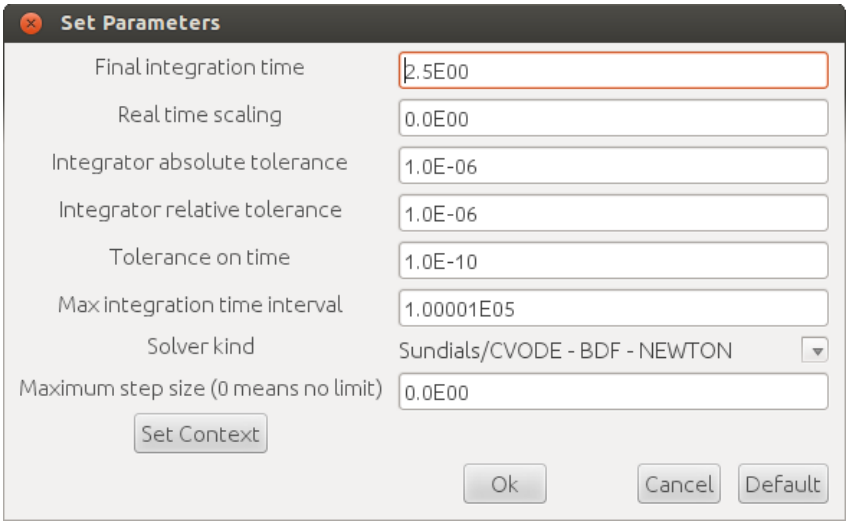
The "dummy" block serves no real purpose. It is a placeholder that the simulator uses in the solving of differential equations.

Simulation

To set up a simulation, from Xcos, choose "Simulation" on the drop-down menu and select "Setup."



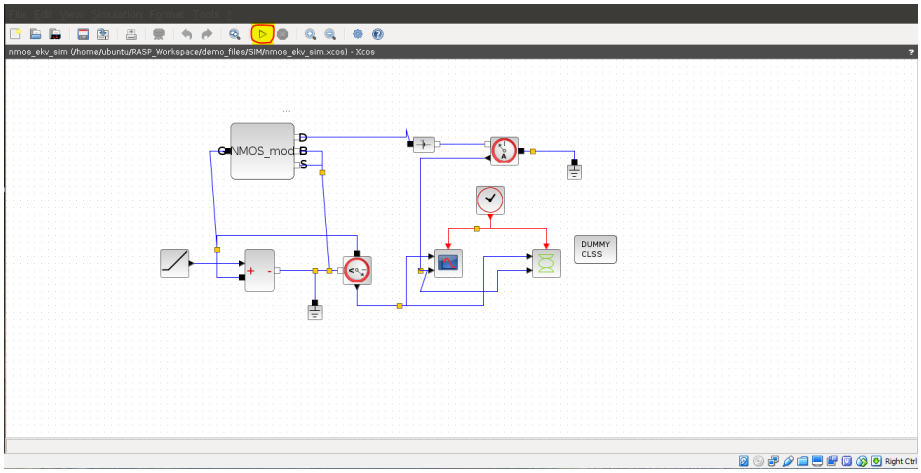
This opens the simulation configuration menu.



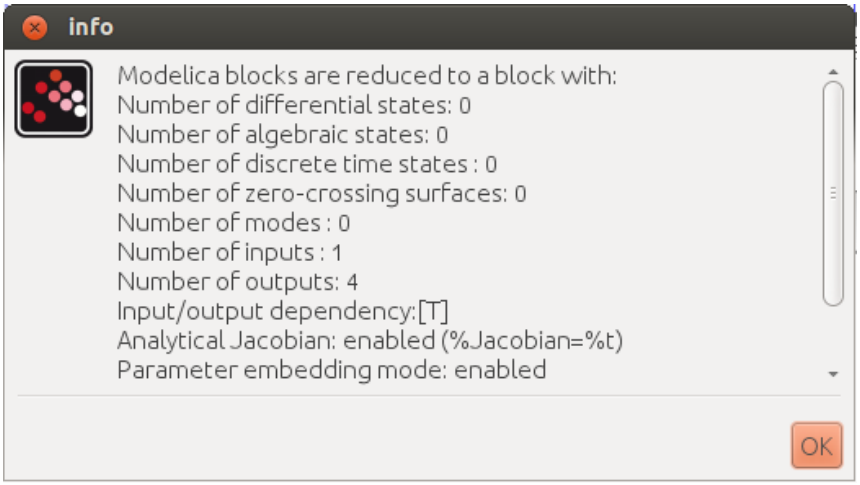
Simulation is all time based. The ramp only knows its slope and initial value, so the final input value depends on the time. For a slope of one and initial value of 0, this simulation sweeps input voltage from 0 to 2.5 volts.

For our purposes, the main setting to change is the **final integration time**. Most of the other parameters in this menu will not drastically impact results.

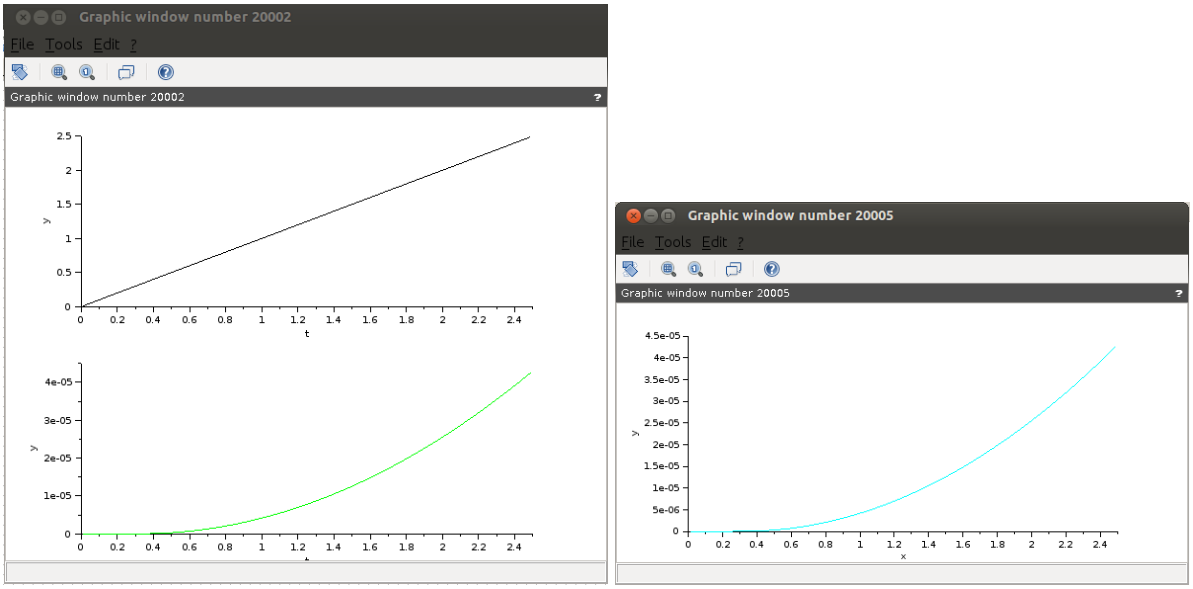
To run the simulation, press the play button on the top menu in Xcos.



A dialog box will appear showing a summary of the simulation settings.



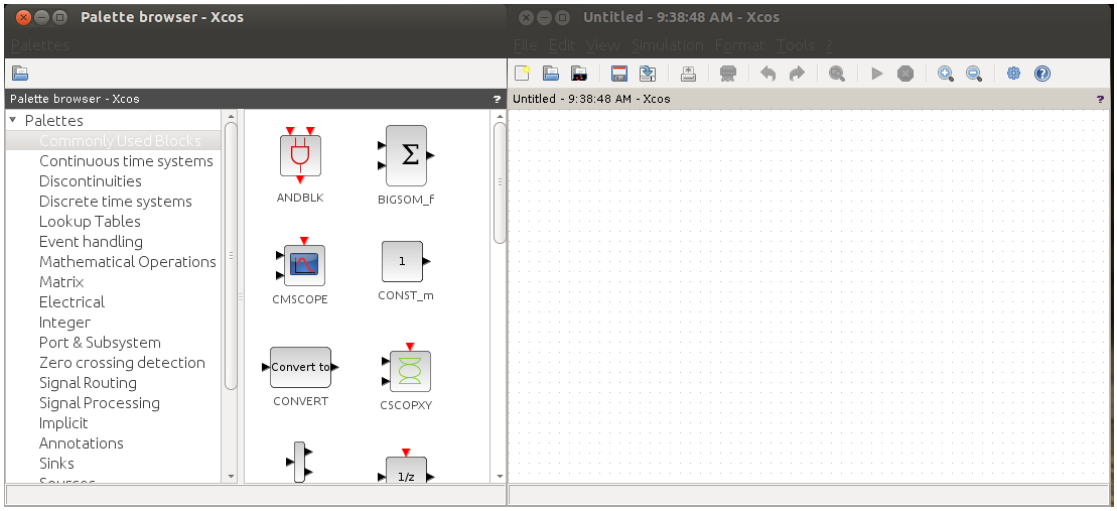
Press OK. After the simulation is finished, the plots created by the two scopes should appear.



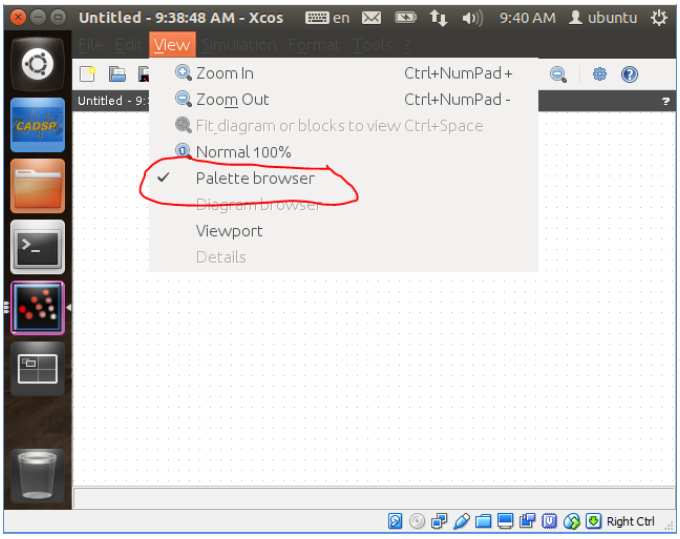
These plots might seem redundant, but this is only because the ramp had a slope of one, so the plots of output voltage vs. input voltage and output voltage vs. time look identical.

Creating a New Design

To create a new design, select “New Design” from the blue GUI. This opens a blank Xcos file.



The second window is the “Palette Browser.” This menu contains all the blocks that can be used in Xcos, and is organized by block type in the left-hand bar. To open the palette browser manually, select “View” from the drop-down menu and select “Palette Browser.”



The main sections of interest in the palette browser are

- Palettes
  - Electrical
  - Sinks
  - Sources
- FPAA

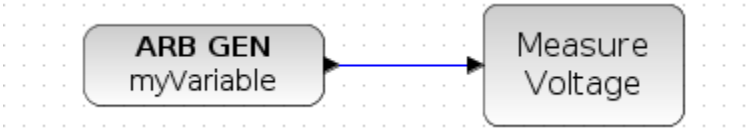
Documentation of blocks required for ECE 3400 can be found in appendix A. It is important to determine if your design is for simulation or programming to the FPAA board. Depending on the end goal, different blocks are required.

All blocks found under “Palettes” are not compatible with the board, but can be used for simulation. Those blocks under “Modelica Blocks” in the FPAA section are models to use for simulation. They use implicit ports and can be connected to electrical blocks.

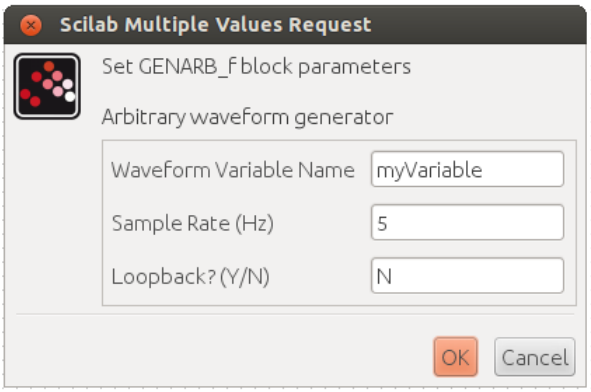
On the other hand, blocks under “Level 1 Analog Blocks”, “Level 2 Analog Blocks”, “Utility Blocks”, and “Mixed Signal Blocks” in the FPAA section of the palette browser can be compiled and programmed onto the board.

Remote System Operation

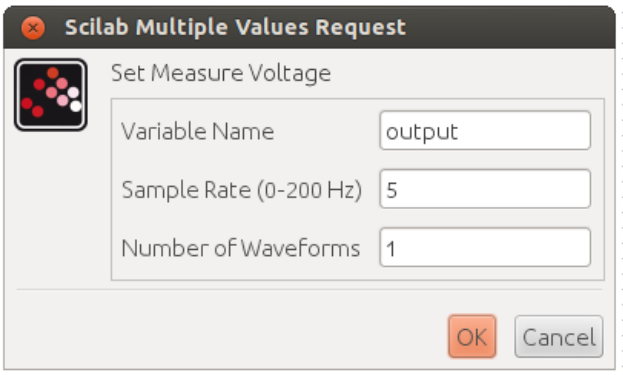
A simple example can be created to illustrate operation of the remote system. Open a new design. From the palette browser, drag and drop the **ARB GEN** from Input / Output Blocks section under FPAA. Then, add the **Measure Voltage** block from the same group. Finally, connect the two blocks.



Both blocks require configuration. Double click on ARB GEN to open its parameter window. Set “Loopback” to “N”. **If loopback is not properly configured, the FPAA system may crash**. Sample rate may be set to any value less than 200 Hz in this example. The default of 5 Hz is suitable.



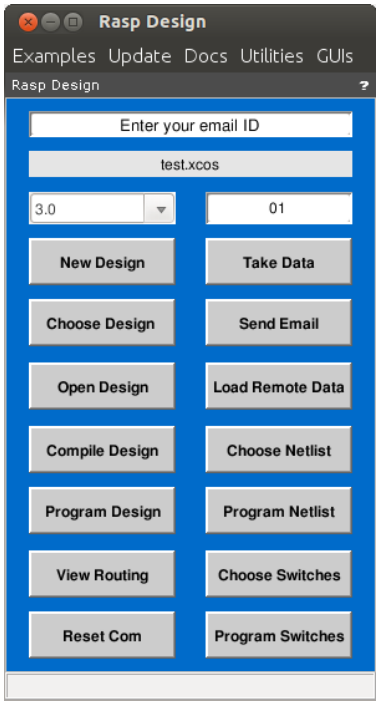
For Measure Voltage, set the sample rate to match ARB GEN. Number of waveforms will be one. This parameter accounts for the possibility of bussing signals together.



The Xcos file is now prepared for use with the remote system. Save the file. **Make sure to manually add the ".xcos" file extension**.

Return to the blue GUI. Select "Choose Design". Navigate to the file you just created and select it. The name of the design should now appear on the GUI. Select 3.0 under "Choose Board" and "01" for chip number. **When using any text entry box, make sure to press enter after making any changes to finalize**.

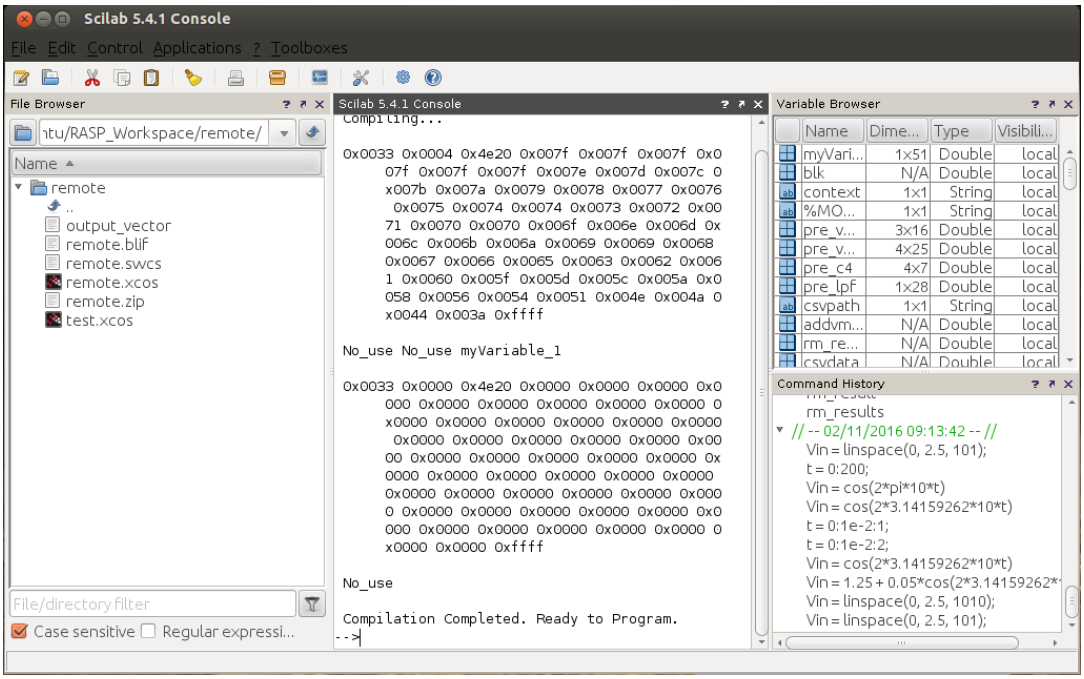
When using the remote system, all boards are version 3.0. For ECE 3400, the available chip numbers are 01 and 13.



One step remains before compiling the design. The ARB GEN block requires an input vector from the workspace name "myVariable". This name may be changed in the ARB GEN settings. In the Scilab command window, enter

```
myVariable = linspace(0, 2.5, 51);
```

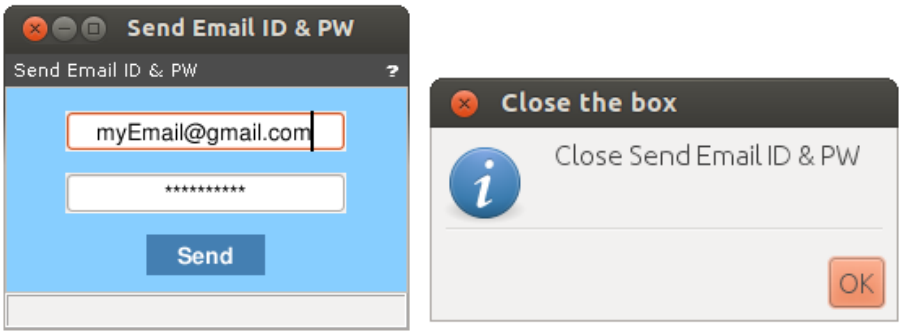
Now everything is set up to compile. From the GUI, select "Compile Design". Upon successful compilation, a success message should appear in the command window.



There are two ways to collect data. One is to select “Program Design” if a USB connection to a board is available. Alternatively, an email containing relevant information is can be sent to an email server to take data remotely.

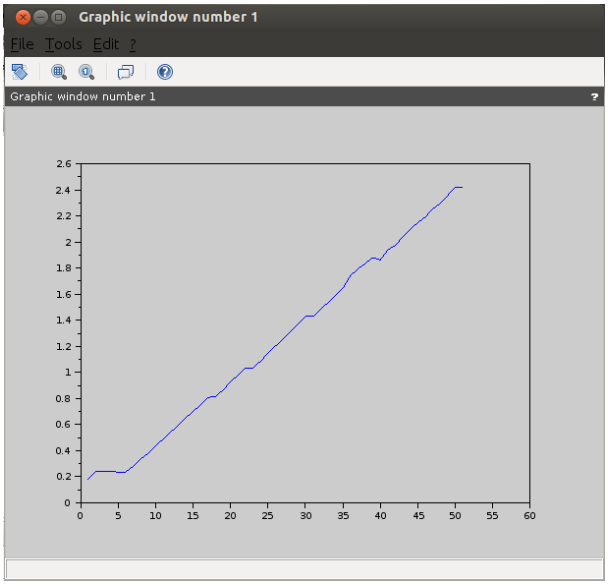
In the blue GUI, enter an email address to serve as the recipient of the data. **Remember to press enter**. Then, select “Send Email”.

A new window will open asking for an email address and password to use as the sender. A Gmail account is recommended. Make sure that the Google account associated with your Gmail is set to allow less secure apps. Enter your email address and password, **remembering again to press enter**, and press “Send”. Once the email is sent, a success message will appear.



Soon after, you should receive an email containing the resulting data in a “**results.zip**” file. Download this file and move it into the directory containing your design. **Do not rename the file**. The program which reads the data is looking for this specific file name in the active directory.

From the GUI, press “Load Remote Data” to extract the data from the zip file. A plot should automatically be generated of output vs. sample number. The values of the data will be stored in a variable called rm_results.



Further Reading

This guide neglects the physics and operation of the FPAA board. Additionally, many advanced features of the Xcos system, such as creating macro-blocks or manually setting routing options, are not discussed. For more information, see

- The paper on T-Square
  - FPAA architecture and infrastructure
  - Experimental results for basic elements
- Papers on Dr. Hasler’s webpage
  - http://users.ece.gatech.edu/~phasler/
  - Design and testing of FPAA hardware
  - Discussion on FPAA layout and routing algorithms
  - Typical hardware specifications
- Information provided on the project webpages
  - http://users.ece.gatech.edu/phasler/ECE3400/exp_schedule.html
  - Project specific details regarding FPAA behavior

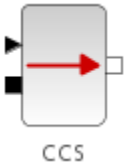
Appendix A: Xcos Blocks

Electrical

Ammeter

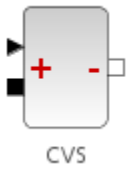
Senses current between its black input port and its white output port. Measurements are converted to an explicit output.

Controlled Current Source



Converts the value on its explicit input port to a current entering the black implicit port and leaving the white implicit port.

Constant Voltage Source



Converts the value on its explicit input port to a voltage between its black implicit port and white implicit port.

Capacitor

An ideal capacitor. Capacitance can be changed in parameters.

Constant Voltage

An ideal voltage source with the white implicit port as the low side. Voltage value can be set in parameters.

Ground

Electrical ground. All connections to grounds are zero volts.

Resistor

An ideal resistor. Resistance can be changed in parameters.

Voltmeter

Senses voltage between its black implicit port and white implicit port. Measurements are converted to an explicit output.

Sinks

Scope

Plots a single waveform versus time. The red input port is for a clock signal to control sample rate. Parameters include the plot window limits and the size of the data buffer.

XY Scope

Plots one input versus the other. The upper port is on the horizontal axis and the lower port is on the vertical axis. The red input port is for a clock signal. Parameters include the plot window limits and the size of the data buffer.

To Workspace

Packages data into a vector and stores it in the Scilab workspace. The name of the output variable and the length of the vector can be set in parameters. The red input port is for a clock signal. The output variable is a structure with fields “time” and “values”.

Sources

Clock

Generates a clock signal. Clock period can be set in parameters.

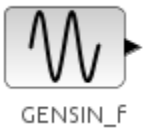
Constant

Generates a constant value. The value can be set in parameters.

From Workspace

Generates a waveform using a vector from the workspace. The name of the vector can be set in parameters. The variable created in the workspace must be a structure with fields “time” and “values”.

Sinusoid



Creates a sinusoid. Amplitude, frequency, and phase can be set in parameters.

Ramp

Generates a ramp signal. Ramp start time and slope can be set in parameters.

Step

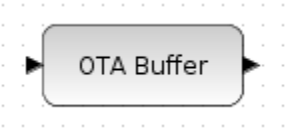
Creates a step signal. Initial and final values, as well as step time, can be set in parameters.

Level 1 Analog Blocks

Capacitor

A capacitor. Capacitors are typically added as loads to amplifiers. Typically, capacitance from lines and switches is sufficient. Capacitance is set as a parameter.

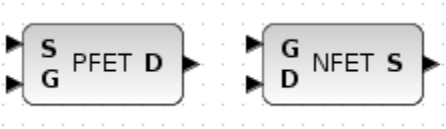
Buffer



The standard voltage buffer using an amplifier with negative feedback. Useful as a stage between DUT and measurement blocks.

Level 2 Analog Blocks

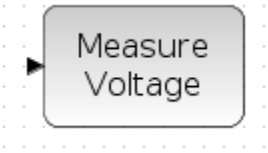
nFET and pFET



FETs in the FPAA. These are available in large numbers, and are the building blocks for macro-blocks such as the OTA or common source.

Input / Output Blocks

Measure Voltage

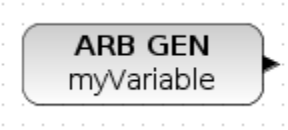


The standard ADC for measuring voltages. A slow but accurate ADC. Sampling rate is limited to 200 Hz.

Ramp ADC

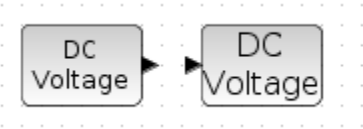
A fast but less accurate ADC. Useful for measuring transient responses, when voltages change very quickly.

Arbitrary Generator



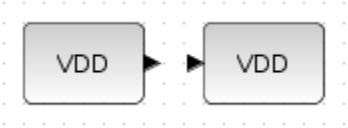
Generates an input signal using a DAC. A variable whose name is configurable as a parameters is required when compiling. The sample rate can be set as a parameter as well. Loopback should be set to "N".

DC Voltage



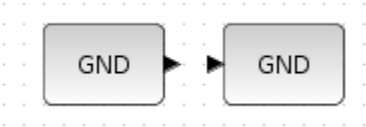
Constant voltage blocks. Two are available to be connected to input or output ports. DC voltages are generated by programming floating-gate devices.

VDD



VDD blocks (2.5 V). Two are available for connection to input or output ports. Arbitrarily many of these may be used, they are internally bussed together on the FPAA.

Ground



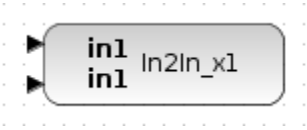
Ground blocks. These blocks represent system ground. Two are available for connection to input or output ports. Arbitrarily many of these may be used, they are internally bussed together on the FPAA.

Utility Blocks

Join

When routing the FPAA, many nodes cannot be joined together easily. The join block has a configurable number of outputs, and is used when many ports need to be connected together.

Input-to-Input



Sometimes it is necessary to connect two input ports together. The input-to-input block allows this to be done on the FPAA.

Mixed Signal Blocks

Common-Source

The classical common-source amplifier. The active device is a pFET. Bias current can be set in parameters, and is programmed using a floating-gate device as a current mirror.

Common-Drain

The classical common-drain amplifier. The active device is a pFET. The reference voltage is the voltage programmed onto the gate of the pFET current source.

OTA

The nine-transistor operational transconductance amplifier. Bias current is programmed using floating-gate devices as current mirrors. The two inputs represent the positive and negative input terminals.

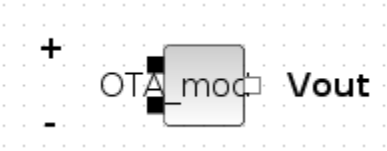
OTA Low-Pass Filter

This block contains the OTA configured as a voltage follower. The FPAA's internal capacitance creates a low-pass characteristic. The cut-off frequency can be set in parameters.

Modelica Blocks

Modelica blocks cannot be complied onto the FPAA. They contain circuit models and equations so simulations can be done.

OTA



The nine-transistor operational transconductance amplifier. Bias current can be set in parameters.

nFET and pFET

Models for the nFET and pFET using the EKV model, which is accurate in subthreshold operation. The parameters to be set are kappa, thermal voltage, threshold current, sigma, and threshold voltage. These parameters should represent the actual FETs which will be used in hardware.

Appendix B: The Command Line

When operating with a Linux/Unix system, it is useful to know the command-line interface. This section details a few simple file manipulation commands.

The File Tree

The command line uses common standards for referring to files and directories. In general, a file name is a relative path from the current directory. E.g. if a file is in a subdirectory, it can be referred to as

"/subdirectory/filename.txt"

When the forward slash is used, the system knows to search under the current directory and the root directory. From any directory, files and directories in the root directory can be accessed by using the forward slash.

Every directory contains special names for "that" directory, the parent directory, and the root directory. The single dot ('.') refers to the current active directory. The double dot ('..') refers to the directory one level above the current directory, called the "parent" directory. The tilde ('~') refers to the root directory.

In general, when using commands in the command line, file names and directory names can contain many layers of other directories, e.g.

"../../directory1/directory2/file.txt"

Basic File Commands

- New Directory
  - mkdir [*dirname*]
  - Creates a new directory with the name *dirname*
- File Delete
  - rm [*fname1*] [*fname2*] …
  - Removes files named *fname1*, *fname2*, etc.
- Copy File
  - cp [fromfile] [tofile]
  - Copies a the file named *fromfile* to a new file named *tofile*
- Move / Rename File
  - mv [fromfile] [tofile]
  - Moves the file named *fromfile* to a new file named *tofile*.
- List Files in Directory
  - ls [*dirname*]
  - Lists all viewable files in the directory named *dirname*. Default directory is the current directory.
- Change Active Directory
  - cd [*dirname*]
  - Changes the active directory to the directory named *dirname*.
- Show File Contents
  - cat [*fname*]
  - Displays the contents of a file in the command window as ASCII text.

Appendix C: Troubleshooting

Update and Reset

Often, updates and changes are made to the FPAA system. It is usually good to update the system as described in the "initial setup" part of this document.

Sometimes, resetting the system is helpful. This is done in the same menu as updating. As with updating, **do not close Scilab while resetting occurs**. Doing so may cause Scilab to crash, requiring the virtual machine to be reimported from scratch. Once the reset is done, Scilab should close itself.

Compilation Errors

Compilation errors occur when the "Compile Design" button is pressed on the GUI.

- Block Type
  - Any blocks from the "palettes" section of the palette browser cannot be compiled onto the FPAA. Additionally, any blocks containing "_mod" are only models, and can also not be compiled.
- Input Variable
  - If using the ARB GEN block, was an input vector properly created?
- Join Blocks
  - Are there nodes in the design with more than two connections? If so, make sure to use join blocks. The FPAA routing sometimes fails when multiple ports are connected together.
- Board and Chip
  - Make sure the board type and chip number are set in the GUI. Press enter after changing the chip number.

Email Errors

Email errors occur when selecting the "Send Email" button on the GUI or when pressing the "Send" button in the email address and password window.

- Receiver Email Address
  - Is the email address in the GUI a valid address? Make sure to press enter after changing this box.
- Sender Security Settings
  - If using a Gmail account as a sender, make sure Google settings are changed to "allow less secure apps". Georgia Tech email accounts will not work for sending emails, but may be used as recipients.
- Sender Email Address
  - Were the sender address and password correct? Make sure to press enter after changing these entry boxes.
  - If an error occurs when working with the address and password window, close the window and reinitiate the email process.

Data Errors

Data errors occur when loading data from the remote system. This may also include the problem of not getting a response.

- Loopback
  - When using the ARB GEN block, make sure loopback is set to "N". Otherwise, the system may loop an input vector indefinitely.
- Voltage Levels
  - Do not try to set voltages above 2.5 V or below 0 V. This will cause malfunctions in the programming of the board.
- Nonsensical Data
  - When using measure voltage blocks, make sure the sample rate matches that of the ARB GEN block.
  - After performing updates, some blocks may have changed. Unfortunately, Xcos does not replace blocks with updated versions automatically. Manually delete and replace and blocks from the FPAA section of the palette browser.
- Board and Chip
  - When using a USB connection on physical boards, the board type is 3.0a. When using the remote server, the board type is 3.0.
  - The only chips available for the remote system are 01 and 13.
- Biasing
  - Keep in mind that the circuits will not operate unless the proper DC biasing is achieved. E.g. when applying a step input, steps from 0 to 0.5 V do not work.
- Data Length
  - The system is not designed to handle enormous data sets. Keep input vectors less than 200 samples.
- File Names
  - Make sure the results.zip file was not renamed.
  - Make sure there are not spaces in file or directory names.

Restart

If all the above troubleshooting tips have failed, restart from only the Xcos file. During the process of compilation and loading data, many additional files are created in the project directory. Move your Xcos file to a new directory and start over without the extra files.