

# Life Without Matlab

DSP System Design and Analysis on the Cheap

Grant Griffin

Iowegian International Corporation

<http://www.iowegian.com>

# Why Didn't I Use Matlab?

- Originally, it didn't exist
- **My employer thought it was too expensive to purchase**
- I got used to using other methods
- I got used to using other scripting languages - first Perl, then Python
- I never had much need for it

# My Methods

- GUI tools:
  - ScopeDSP – signal analysis
  - ScopeFIR – FIR filter design
  - ScopeIIR – IIR filter design (to be released in Q2 2010)
- C/C++ vector libraries
  - Zero-based
  - Minimal change from design to implementation
- Python
  - General use, built-in support for complex numbers
  - Extensions support vectors and matrices
  - Excellent for data munging
- Excel
  - Tedious, but interactive
  - Good numeric visualization
  - Optimizer works well

# GUI Tools for System Design and Analysis

- Tools
  - ScopeDSP: Time and frequency data plotting and analysis of data in files
  - ScopeFIR: FIR filter design and analysis
  - ScopeIIR: IIR filter design and analysis
- Benefits
  - Inexpensive compared to Matlab
  - Easy to learn
  - Easy to use for common DSP tasks
  - Visual
  - Interactive

# GUI Tool: ScopeDSP

- Plots time and frequency data from files
- Great for A/D converter performance analysis
- Transforms between domains via an arbitrary-N FFT
- Provides spectral statistics
- Provides basic DSP data manipulations like windowing, scaling, zero-stuffing, etc.
- Optimized for interactive use, e.g. zooming

# GUI Tool: ScopeFIR

- Designs FIR filters via the Parks-McClellan algorithm and many others
- Designed for easy interactive use:
  - Data visualization via four-panel layout: specifications, coefficients, frequency plot, impulse plot
  - Provides common FIR data manipulations: minimizing NTaps, quantizing, scaling, polyphase extraction, more

# C/C++ Vector Libraries for System Simulation and Implementation

- Allow you to write code as statements rather than loops: simpler to test and debug
- Natural for block-based processing, especially if target supports DMA
- Allow development on multiple platforms (PC and target) and allow easy porting to a new platform
- Can simulate your system using the actual implementation code
- May be more/less efficient on the target than explicit loop code depending on the compiler and processor

# System Analysis with Excel

- Benefits:
  - Commonly available
  - Provides numeric visualization
  - Interactive
  - Easy optimization
- Drawbacks
  - Setting up formulas is tedious
  - Plotting is primitive, e.g. no mouse zooming

# What is Matlab?

- Matlab is a *system* consisting of:
  - A scripting language which uses complex matrices as the primary data type
  - An extensive set of toyboxes
  - An Integrated Development Environment (IDE) for the above

# Advantages of Matlab

- Good IDE
- Large, comprehensive system
  - Lots of toyboxes
  - Related systems like Simulink
- “It’s the standard”
  - You were taught to use it in college
  - You’ve developed your own set of scripts to do common tasks, e.g. plotting a frequency spectrum
  - Your co-workers use it

# Problems With Matlab

- High cost
- Hard for beginners to get started with
- Dogmatically a “matrix laboratory”
  - Use of one-based matrices as the fundamental type
  - Poor support for other types, notably strings
  - Use of parens for de-referencing variables (*ick*)
- “Doesn’t do what you expect”

# More Problems With Matlab

- Poor design as a scripting language
  - Poor modularization:
    - The only form of modularization is that functions (must) live in their own file
    - No namespacing
  - Poor string handling
  - No hash/dictionary type
  - Object-orientated programming (classes) added only recently
  - One-based
- Emphasis is almost entirely on scripting: GUI elements like “fdatool” are clunky
- That darn schizophrenic semicolon!

# What Does Matlab Cost?

- The cost isn't listed on the The Mathworks' web site
- I sent an inquiry via the form on their website but didn't receive a response for weeks
- Finally received a response but repeated emails were needed before they would divulge the price
- Price is in the multi-thousands for Matlab and Signal Processing Toolbox
- Compare to Mathcad, which costs about \$1100
- Compare to lowegian tools in the low hundreds

# Solution to Matlab's High Cost

- Free/open alternatives
  - Matlab-like systems
    - Octave
      - Highly Matlab compatible
      - No IDE
    - Scilab
      - Partially Matlab compatible
      - Includes IDE
      - Provides SciCos as a Simulink replacement
  - Python with SciPy, e.g. “Python(x, y)”
- MathCAD
- Use a mix of free/available/cheap tools like I do

# Solution to Matlab's Design

- Use a better scripting language: Python
- Python:
  - Is easy to learn
  - Is general use
  - Supports numeric applications via SciPy (see [scipy.org](http://scipy.org))
  - Has a very clean and simple syntax
  - Has a strong system of modularization
  - Can be embedded and extended

# Summary

- Matlab has strengths and weaknesses
- It's perfectly possible to do DSP system design without Matlab
- There are plenty of free/available/cheap alternatives
- No single tool does it all – including Matlab
- Use the tool that best fits the job

# A Final Thought

**If all you have is Matlab, everything looks like a script.**