The Simple Life

Tim Robinson

Test System Engineer in the Cincinnati, Ohio area

Using LabVIEW since 2005

forums.ni.com: crossrulz

Linkedin: http://www.linkedin.com/in/crossrulz

ARCHITECT

LabVIEW CHAMPION

Knight of NI

What is Simple?

Dictionary.com:

1. easy to understand, deal with, use, etc.

2. not elaborate or artificial; plain

5. not complicated

14. lacking mental acuteness or sense

Synonyms
Simple: clean, plain, straightforward
Simplicity: clarity, directness, obviousness, easiness

Goals of Simple

Make Things Easier

- Easier to develop
- Easier for others to help
- Easier to use
- Easier to support

Reduce Risk

- Simpler design has inheritably less risk
- Less things that can break

More Time for More Important Things

- Opportunity Cost

Goals of Simple

Promote SMoRES

- Scalable
- Modular
- Reusable
- Extensible
- Simple

Simple for Whom?

Developer

- You (for the programmers in audience)

Customer

- User of the Solution

Maintainers

- Those who will have to debug and/or extend your solution
- Likely includes Future You

Balance

As with everything, simple must be balanced with many other factors

The Simple Life

Practical Methods

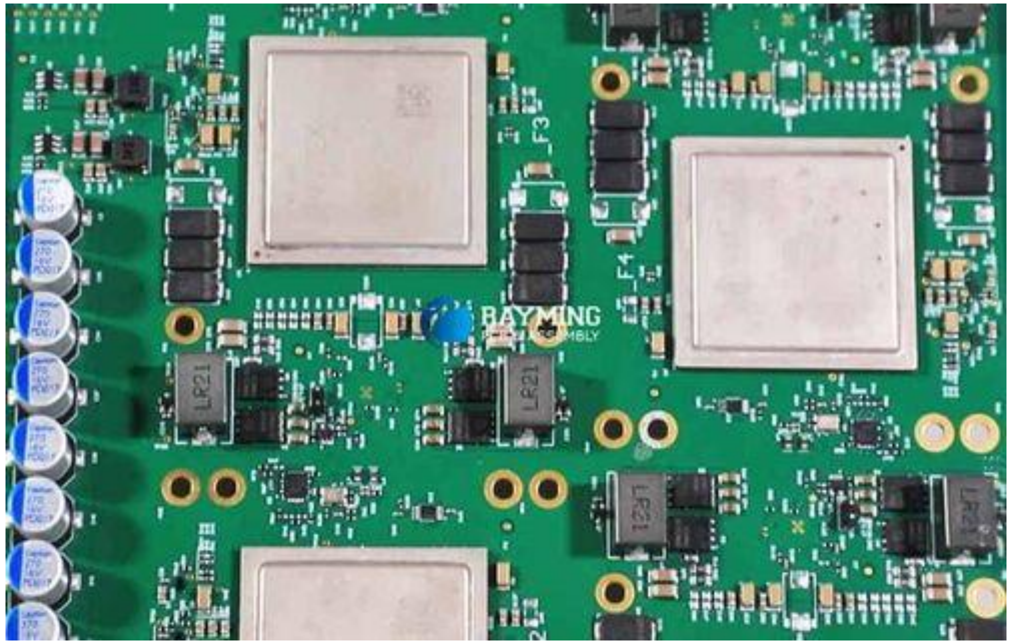
Requirements Negotiation

- Between Developer and Customer
- Removing requirements makes them simpler
  - Are all the requirements necessary?
  - What requirements are just “bells and whistles”?
- Set priorities

Hardware Test

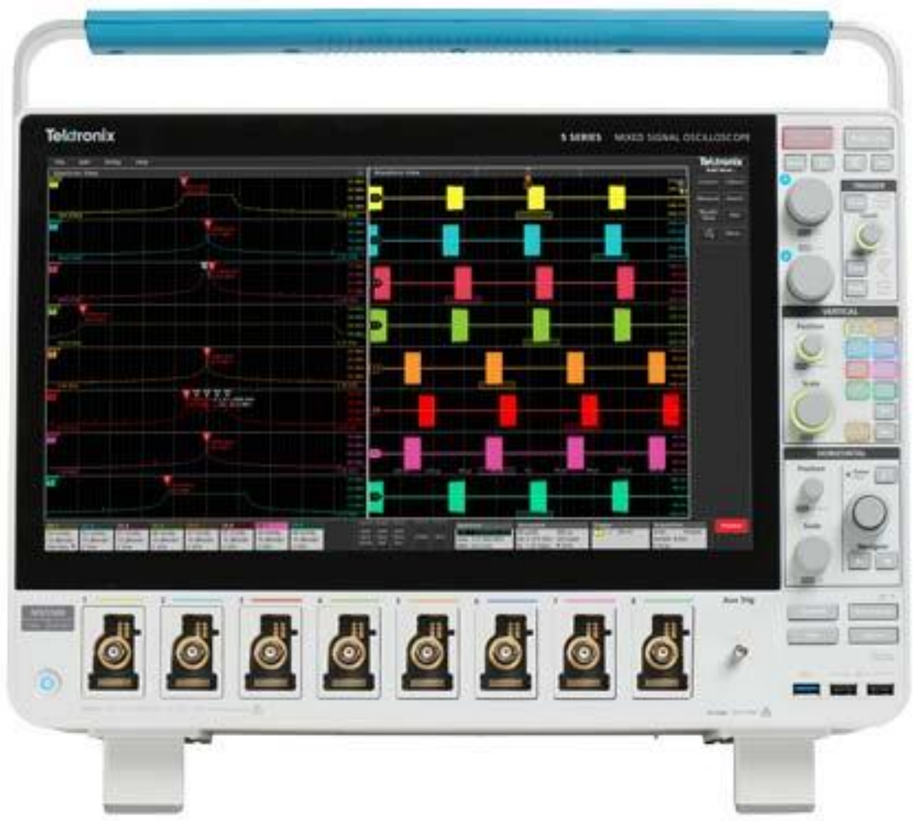
- What are we testing for?

Functional	UUT meets all requirements
Fabrication	UUT was built correctly



Hardware Test

- Test Equipment
  - Use test hardware that is made for the desired measurement
- Test Measurements
  - Simple algorithm
  - Let the instrument handle the measurement



Solution Development

- Do not fall for “It’s the way we’ve always done it”
- But Balance with Chesterton’s Fence
- Use Industry Standards
  - UART
  - NI using gRPC for drivers

Solution Development

Complexity of Solution should match the complexity of Problem

Solution Development

Ways of breaking down problem:

- SubVIs and/or TestStand sequences
- Libraries/Classes
- Features
  - Agile sprints
- Separate Loops/Actors for a single task
  - Use a common framework

$\}$ Be aware that complexity will be shifted to communications and/or maintaining state

- Good framework handles this for you

Frameworks

- Avoid “Framework Dogma”
  - Use a framework appropriate for the solution
  - Ex: Don’t use Actors when a While Loop with Event Structure will do the job
- Use the rules of the framework
- Be aware of coupling between your code and the framework

Code

- Goal #1: Meets the requirement
- Goal #2: Understandable by whoever must read it

- Clarity is as important as functionality
  - Think about the Developer Experience (DX)
  - Do not get fancy

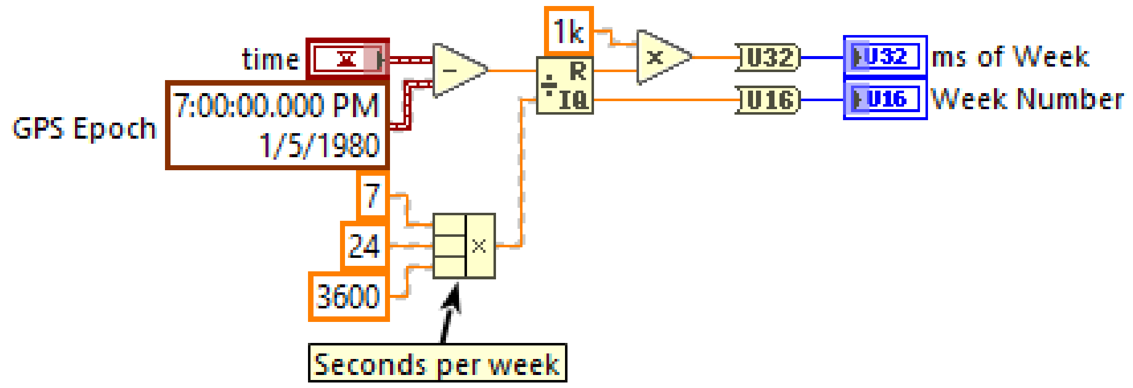
Code Clarity

- Avoid Global Data
  - Where and when is it being changed?
  - Where and when is it being read?
  - Race Conditions are hard to debug
- Functional Programming
  - Avoid state in functions
- Use tools on VIPM
  - JSONtext by JDP Science
  - OpenG

Code Documentation

- Free Labels
  - Describe non-obvious bits of code
  - Label wires to explain algorithm or formula

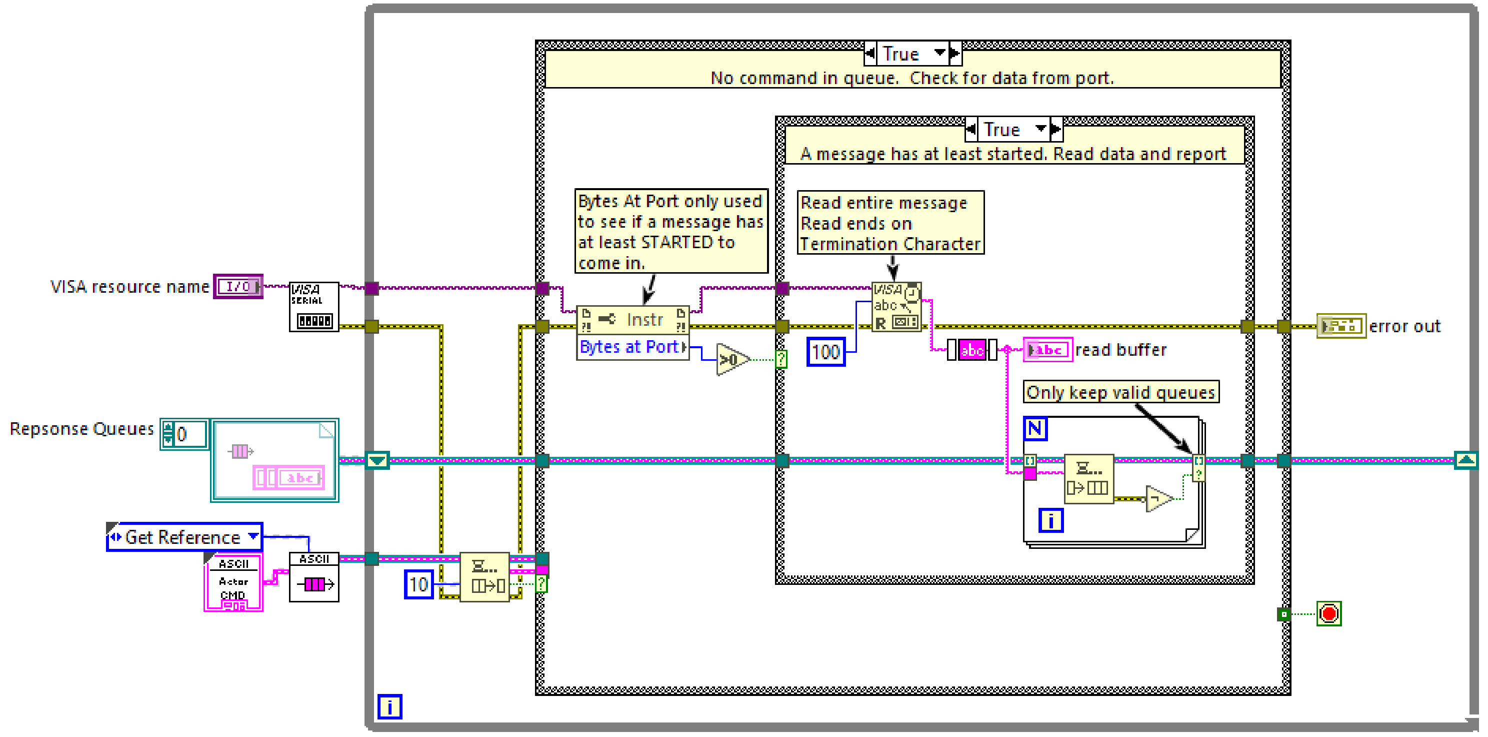
- Label Constants



Code Documentation

Subdiagram Labels

- Loops – Describe what the loop is doing
- Case Structures – Why is this case being called? What is happening in this case?



True

No command in queue. Check for data from port.

True

A message has at least started. Read data and report

Bytes At Port only used to see if a message has at least STARTED to come in.

Read entire message
Read ends on
Termination Character

Only keep valid queues

Code Style

- Follow a coding standard
  - The LabVIEW Style Book by Peter Blume
  - Extreme LabVIEW Style Showdown – Hunter Smith and Tom McQuillan GDevCon 2023
- Use some type of diagram cleanup
  - Ctrl+U
  - Nattify VI (Darren Nattinger)
  - Blue Formatter (Sam Taggart)
- Use VI Analyzer

Code Bloat

Code just gets carried along, despite not being used

- Do not be a hoarder
- Dig through the Dependencies to check for cross-links and other undesirables
- Do not be afraid to refactor to eliminate bloat
- Find Non-Project Files (Darren Nattinger)
- Delete File From Project (BasvE)

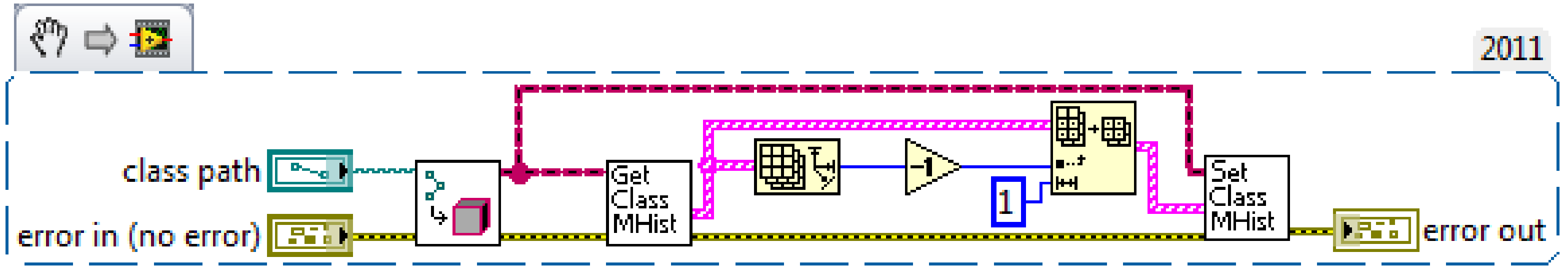
Mutation History

Slow Editor Performance with Large LabVIEW Projects Containing Many Classes

Also been known to cause build errors

To Clear Mutation History:

- Rename the class
- Use hidden VIs



Please go give this idea a kudo: Make Class Mutation History Optional

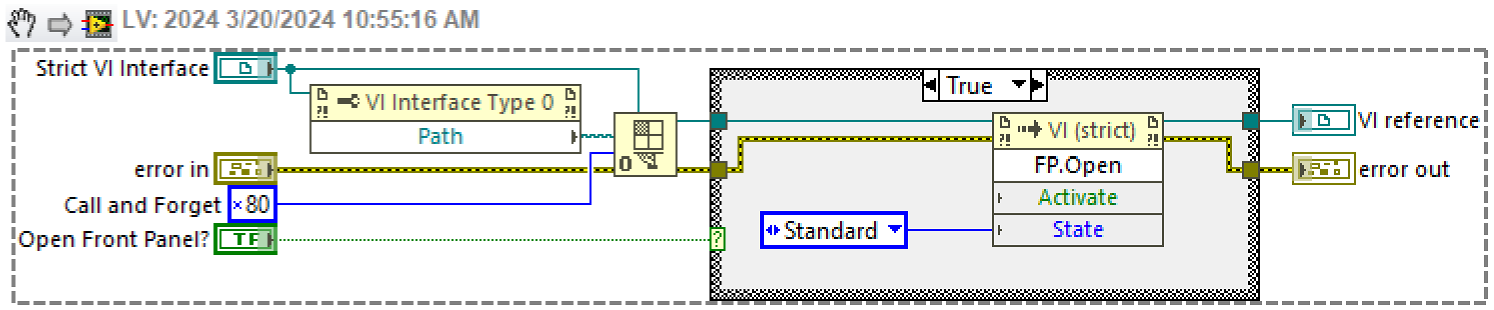
API

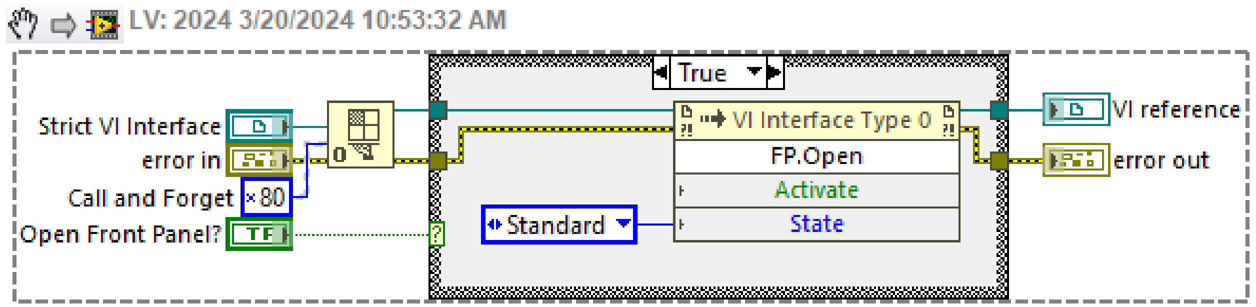
How somebody will use your library

- Consistency
  - Naming
  - Connector Pane Layout
- Use Objects
  - Encapsulation

API

New feature in LabVIEW 2024Q1





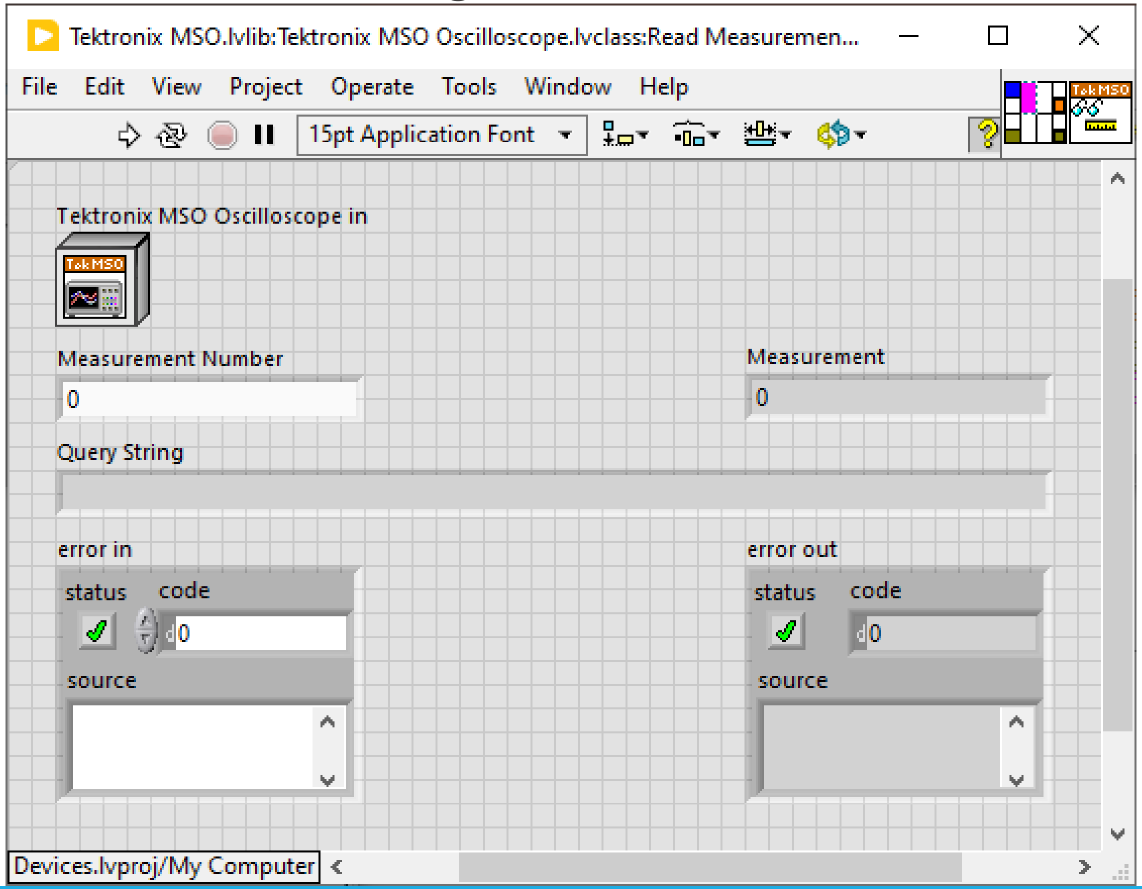
Debugging

You, or somebody you don’t want to curse your name, will have to debug your code.

Make debugging simple.

Debugging

Have “Probing” Front Panel Indicators


Tektronix MSO.lvlib:Tektronix MSO Oscilloscope.lvclass:Read Measuremen...

Tektronix MSO Oscilloscope in

Measurement Number: 0

Measurement: 0

Query String:

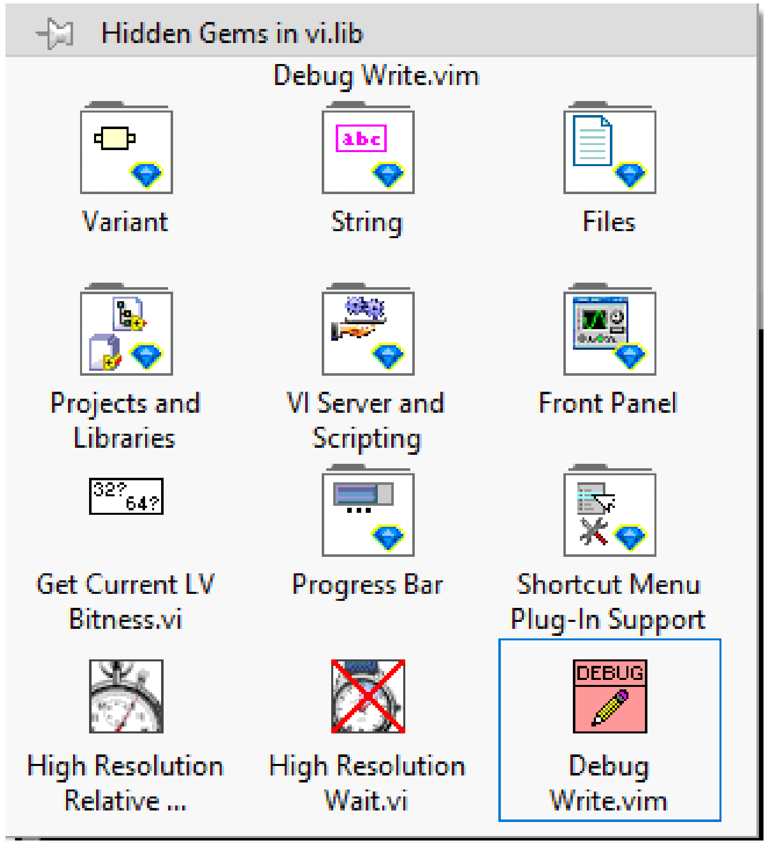
error in: status ✓ code 0 source

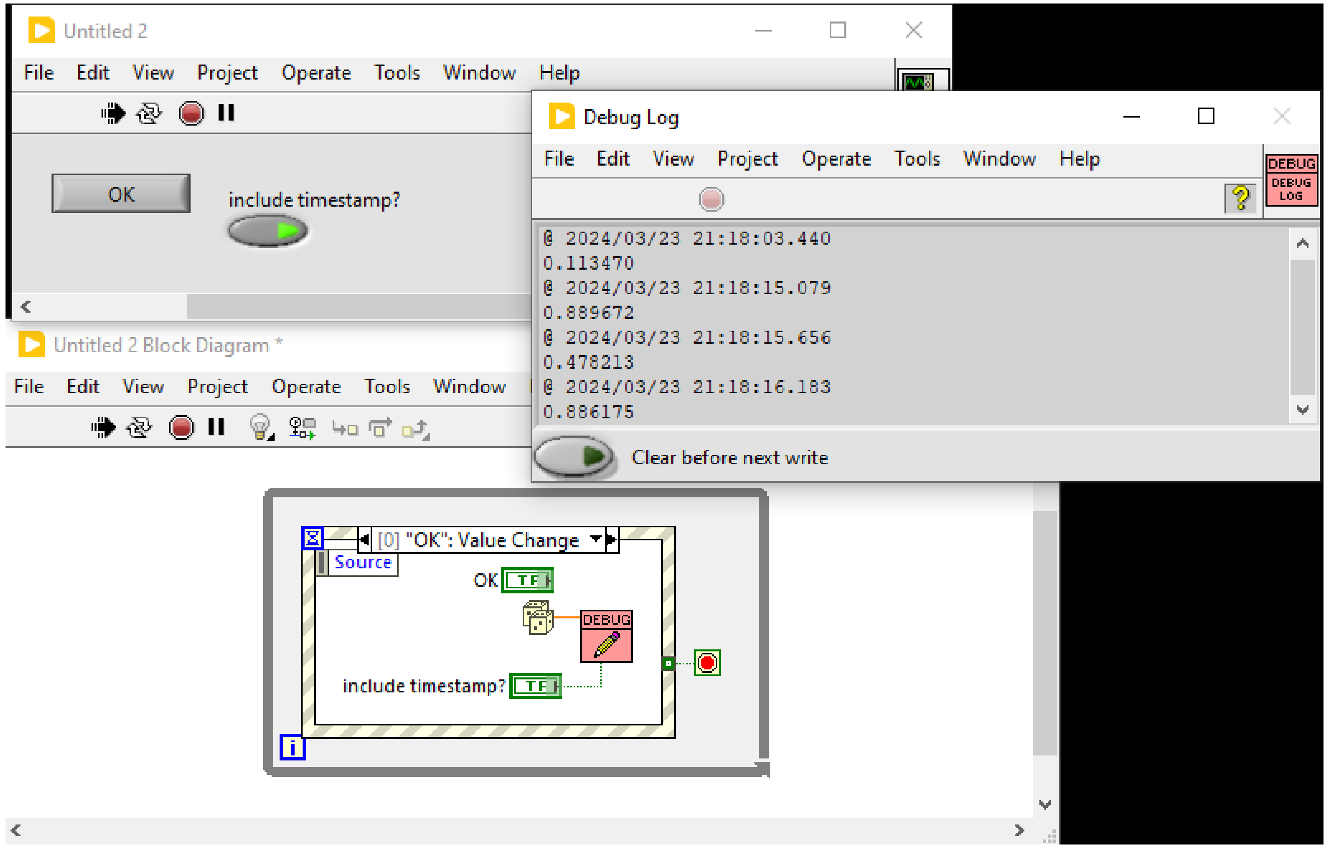
error out: status ✓ code 0 source

Devices.lvproj/My Computer

Debugging

Hidden Gem: Debug Write.vim

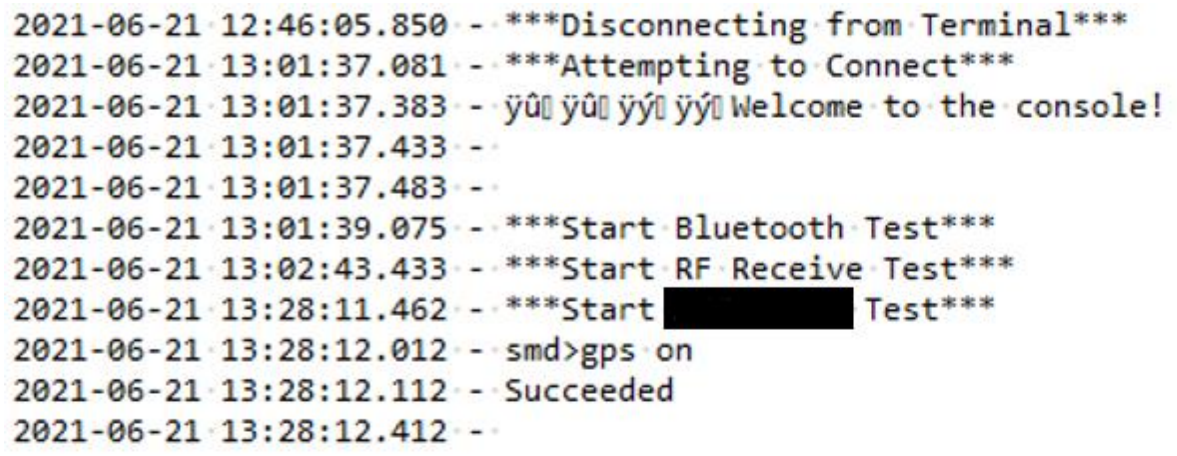




```
@ 2024/03/23 21:18:03.440
0.113470
@ 2024/03/23 21:18:15.079
0.889672
@ 2024/03/23 21:18:15.656
0.478213
@ 2024/03/23 21:18:16.183
0.886175
```

Debugging

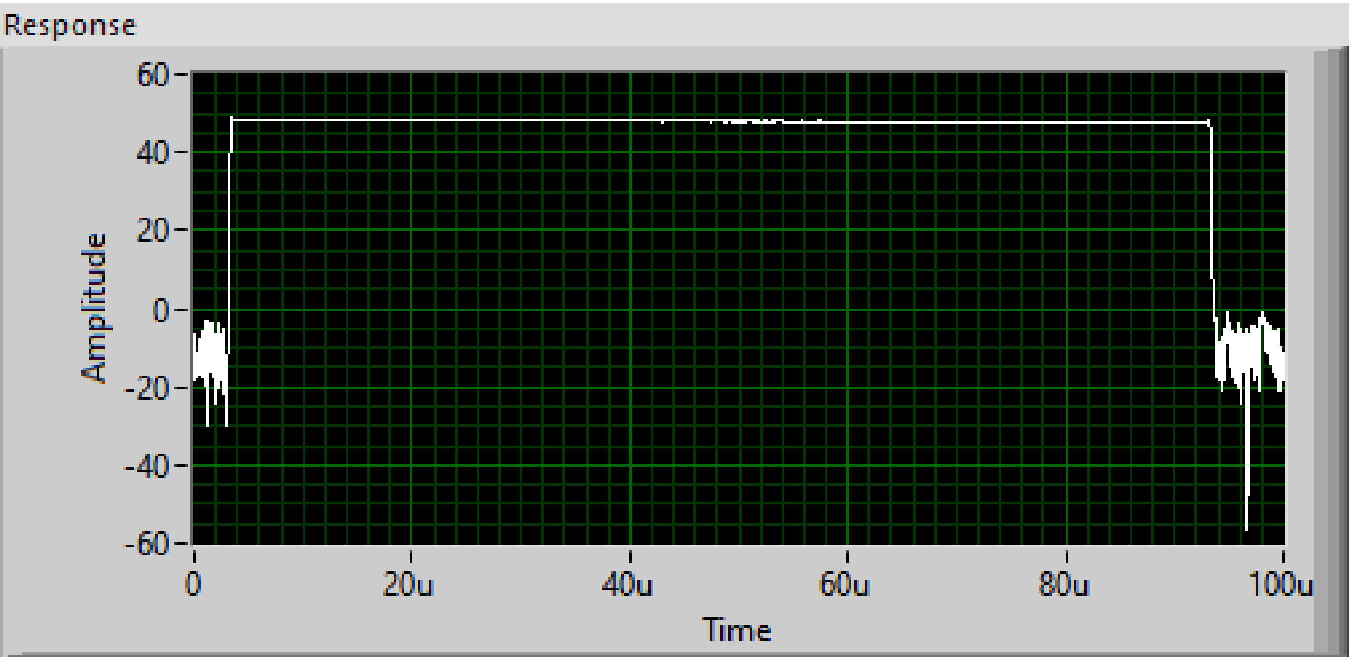
- Use Log File(s)
  - Errors
  - General Trace Log
  - Terminal Communications

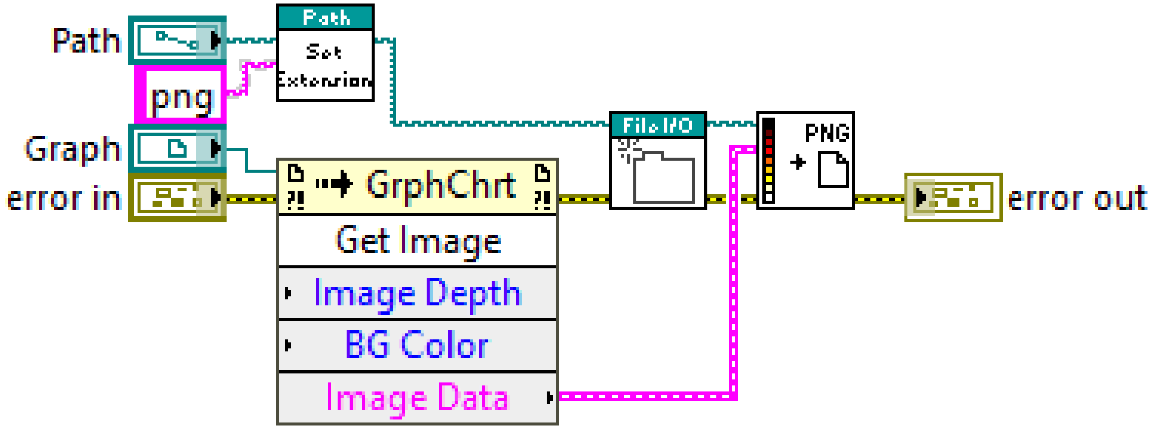


```
2021-06-21 12:46:05.850 - ***Disconnecting from Terminal***
2021-06-21 13:01:37.081 - ***Attempting to Connect***
2021-06-21 13:01:37.383 - ÿû▯ ÿû▯ ÿý▯ ÿý▯ Welcome to the console!
2021-06-21 13:01:37.433 -
2021-06-21 13:01:37.483 -
2021-06-21 13:01:39.075 - ***Start Bluetooth Test***
2021-06-21 13:02:43.433 - ***Start RF Receive Test***
2021-06-21 13:28:11.462 - ***Start [redacted] Test***
2021-06-21 13:28:12.012 - smd>gps on
2021-06-21 13:28:12.112 - Succeeded
2021-06-21 13:28:12.412 -
```

Debugging

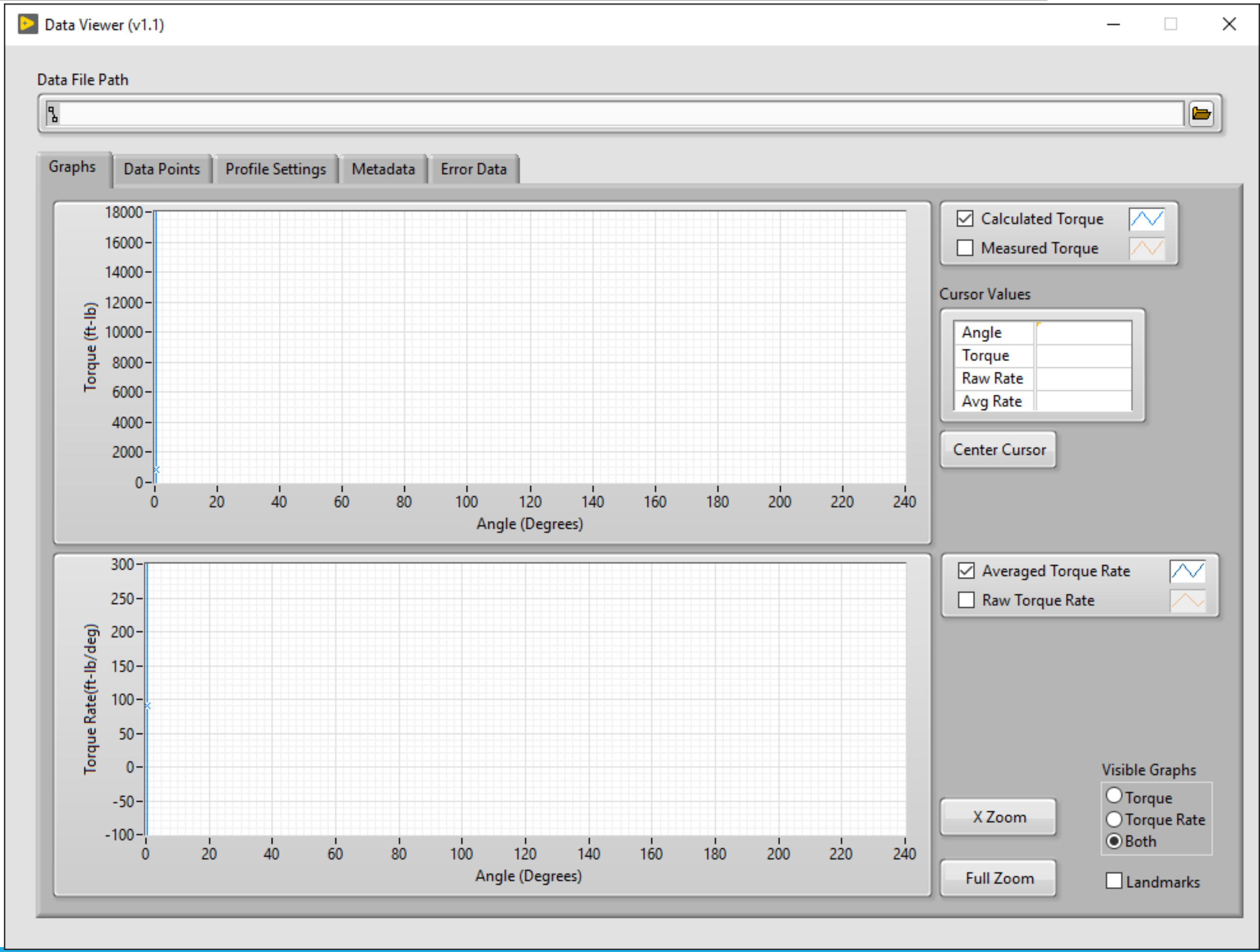
Save plots and instrument screen shots





GUI

- Avoid information overload
- Use Graphs and Charts to show historical data
- Group related data together
- Include the software version

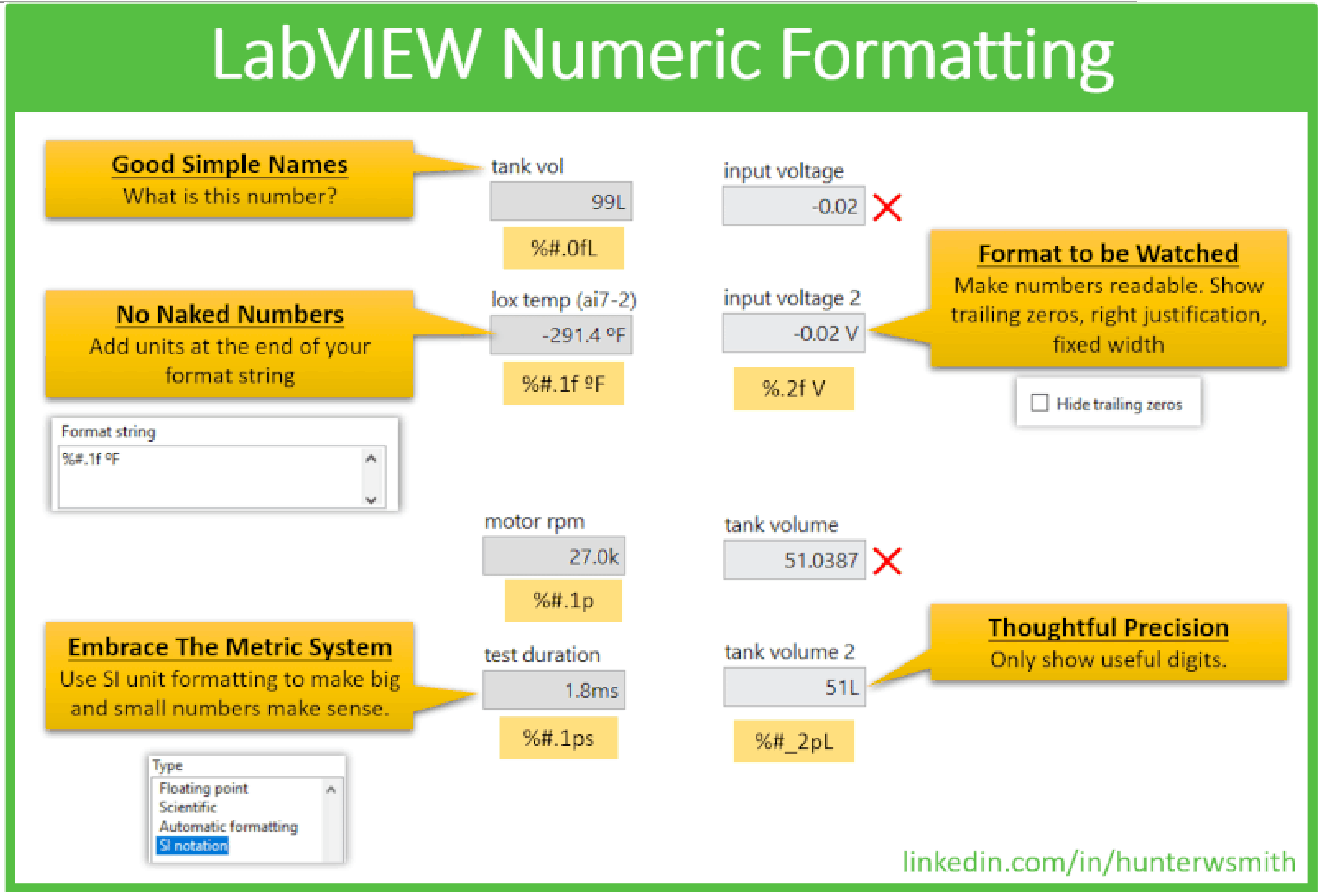


Settings shown:

- [✓] Calculated Torque
- [ ] Measured Torque
- [✓] Averaged Torque Rate
- [ ] Raw Torque Rate
- Visible Graphs: Torque, Torque Rate, [✓ Both]
- [ ] Landmarks

GUI

Format your numerics



LabVIEW Numeric Formatting

Good Simple Names
What is this number?

No Naked Numbers
Add units at the end of your format string

Format string
%#.1f ºF

Embrace The Metric System
Use SI unit formatting to make big and small numbers make sense.

Type
Floating point
Scientific
Automatic formatting
SI notation

tank vol	99L	%#.0fL
lox temp (ai7-2)	-291.4 ºF	%#.1f ºF
motor rpm	27.0k	%#.1p
test duration	1.8ms	%#.1ps
input voltage	-0.02 ✗	
input voltage 2	-0.02 V	%.2f V
tank volume	51.0387 ✗	
tank volume 2	51L	%#_2pL

Format to be Watched
Make numbers readable. Show trailing zeros, right justification, fixed width

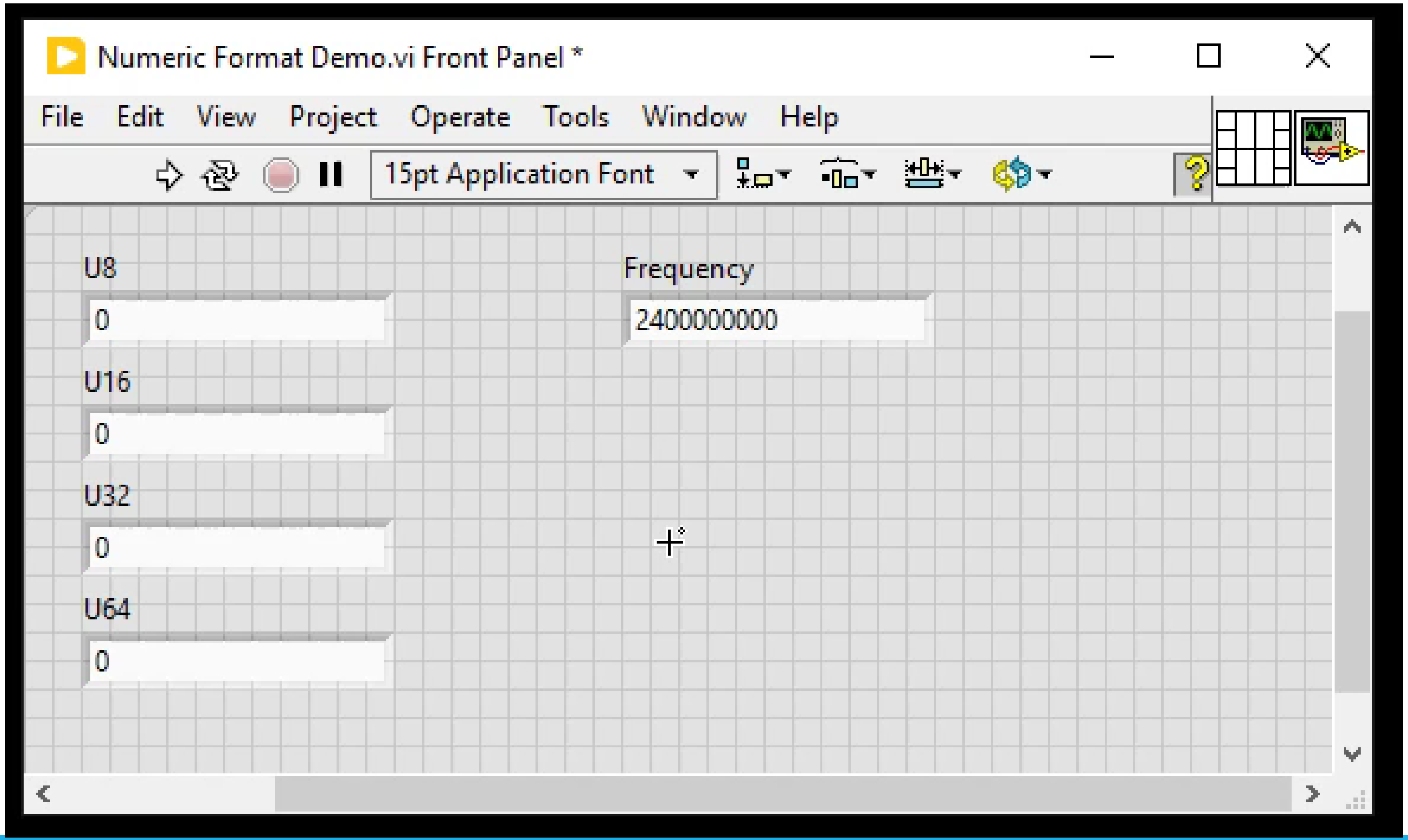
☐ Hide trailing zeros

Thoughtful Precision
Only show useful digits.

linkedin.com/in/hunterwsmith

GUI

Format Numeric QuickDrop


Numeric Format Demo.vi Front Panel *

File Edit View Project Operate Tools Window Help

15pt Application Font

U8
0

U16
0

U32
0

U64
0

Frequency
2400000000

In Your Life

- Be aware of what is consuming your CPU time
  - Multitasking is hard
  - Refactor out things in your life that are less important
  - Sacrifice – Giving up something you love for something you love even more

The Simple Life

Questions?