

Developing software for controlling an ultracold atomic apparatus

KERRY WANG

AUBIN LAB
5/17/2022

Overview

Background and Motivation

Project 1: Improving the Control Software

Project 2: Improving Timing Performance

Summary

Background

Bose-Einstein Condensates (BEC) first experimentally created in 1995

- Cool bosons until they condense into ground state

BECs can manifest quantum effects on macroscopic scale

- Wave nature of atoms

Applications in interferometry

- Precision inertial navigation
- Gravitational mapping
- Sub-mm gravity

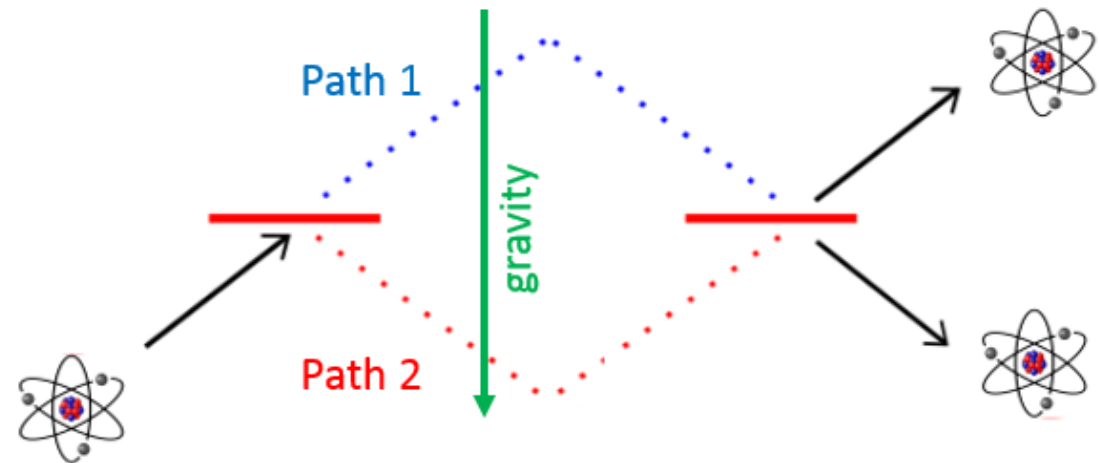


Image credit: S. Aubin

Motivation

Ultracold atomic apparatus comprised of many components

- 32 analog and 64 digital output channels
- Can generate BEC from ^{87}Rb every 40 seconds

Capabilities limited by software

- Only 32 of each type of channel

Future experiments need more precise timing

- Microsecond-range precision (uncharacterized)
- Want microsecond-precision and accuracy

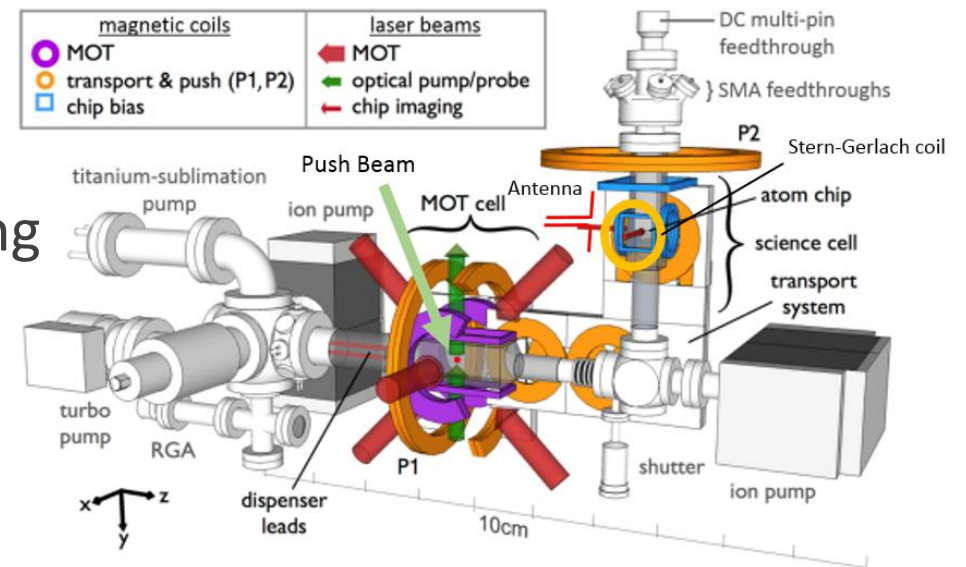


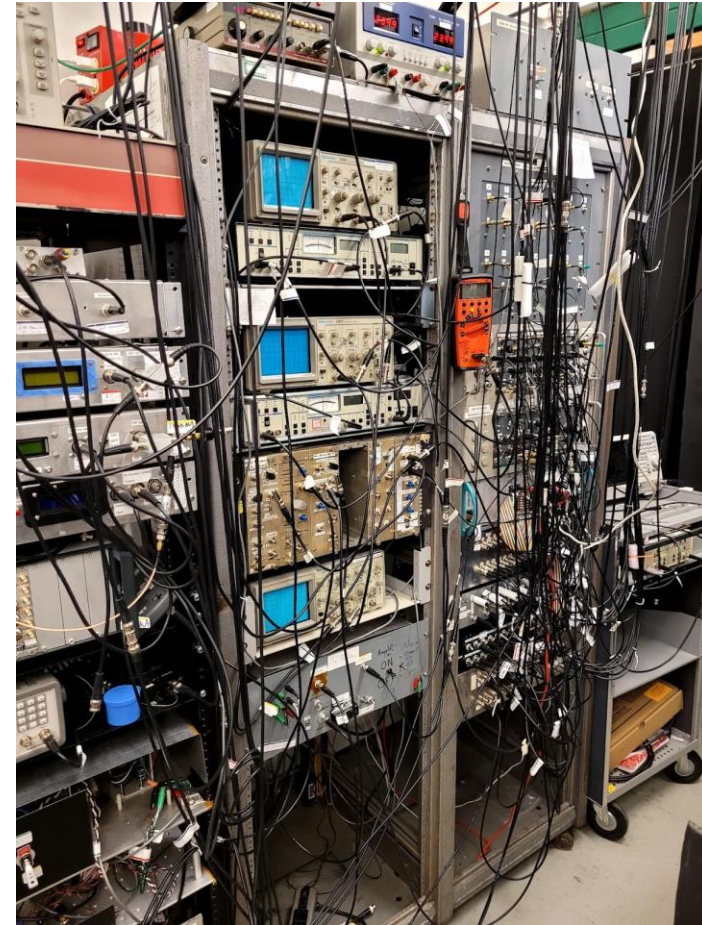
Image credit: S. Aubin

ADwin Sequencer

“Orchestra maestro”

Sends signals to the instruments

Each experiment is a “symphony”



Improving the Control Software

Expand software to control more channels

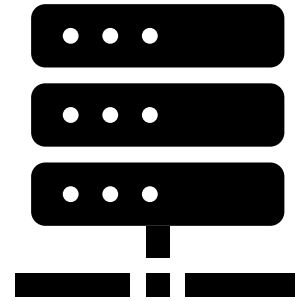
Organize and refactor codebase for easier maintenance

Document code for future work

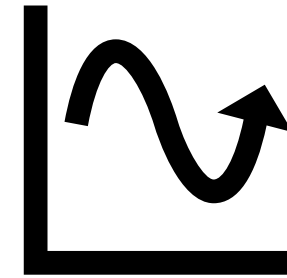
System Diagram



Computer running
control software
(LabWindows/CVI)



ADwin-Pro II
sequencer running
sequencer software
(ADbasic)



Analog outputs

1010
1010

Digital outputs

10 pages, 17 columns per page

Old Software

32 analog channels

3 DDS channels (obsolete)

32 digital channels

The screenshot shows the ADwin Sequencer software interface. At the top, there are menu options (File, Settings, Edit, Update, Preferences) and control buttons (Run Cycle, STOP, MOT, B-Trap, XFer, Evap, Hidden, etc.). A table displays data for 32 analog channels (e.g., Purple Haze, Chip ODT AOM 80 MHz, Trap Power) and 32 digital channels (e.g., Axial Camera, Radial Camera, Fast On). The table has 17 columns representing different time points or parameters. A 'Looping' section on the right includes Start, End, Page, Col, and Repeats controls. A 'Scan Values' section at the bottom right shows a vertical scale for 'Trap Bottom (MHz)' ranging from 8.250 to 8.350.

14 pages, 20 columns per page

New Software

40 analog channels

64 digital channels

The screenshot displays a control software interface with a menu bar (File, Settings, Edit, Preferences) and a toolbar with buttons for Run Cycle, STOP, Scan, and Repeat. Below the toolbar is a table of 40 analog channels. The table has columns for Desc, MOT, MOT2, JPhuttt, Jbdshut, lightoff, fast off, wait, OM, OP, and time. The channels listed include Purple Haze, Chip ODT AOM 80 MHz, Trap Power, Trap Frequency, Horiz SG Coil, Repump Power, 40 MHz AOM Wref, Optical Pumping Power, MOT ATE Current, MOT coils V Set, MOT coils I Set, MOT coils HV Set (fast), Microwave VWA, Levithan, RF VVA-2, High Finesse RED (Chip), K trap power, K repump power, K trap frequency, K repump frequency, High Finesse BLUE (Chip), RF frequency, RF amp (VVA), PS A V Set, PS A I Set, PS B V Set, PS B I Set, PS C V Set, PS C I Set, Chip Hold (Keppco), Axial Camera, Radial Camera, Fast On, IGBTs Off, FET Switch (B trap suppl), High Finesse RED (Chip), Rb Trap Laser, Rb Repump Laser, Optical Pump Laser, Rb Fiber C Ax, Rb TA shutter, Rb OP Shutter #1, Rb Fiber AB RigOP, Rb TA Pre Fiber Shutter, K Trap Power, K repump power, K TA shutter, ODT 40 MHz pre fiber AO, RF Switch, DO NOT USE LINE, Microwave Mute, Chip ODT AOM 80 MHz, Do Not Use, Transport A1, Transport A2, Transport B1, Transport B2, Transport C1, Transport C2, Transport Blank, 6.8 GHz TTL Switch, Chip ODT pre fiber shuttle, K TA prefiber, Scope Trigger, and High Finesse BLUE (Chip). The right side of the interface shows a grid of 64 digital channels with a red pattern indicating their status. The grid has columns for time (0.000 to 1.000) and rows for various digital channels.

Improvements

	Original	Improved
Analog channels	32	40
Digital channels	32	64
Pages	10	14
Events per page	17	20
Total events per channel	170	280

```
10 #define NUMBERANALOGCHANNELS 40
11 #define NUMBERDIGITALCHANNELS 64
```

Values can now be adjusted via variables
(new cards will require new ADbasic code)

Organization

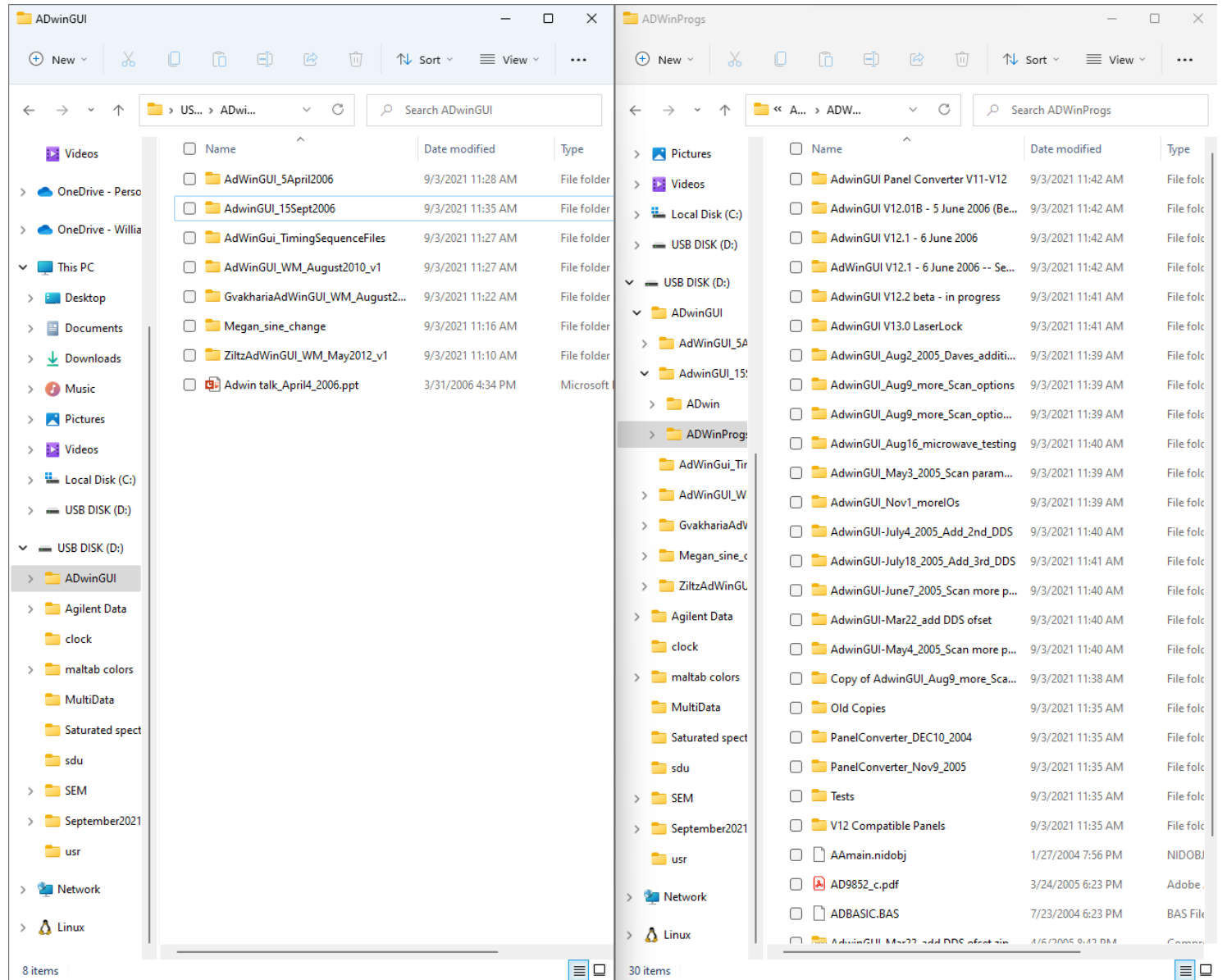
Versions as folders

Unclear history

- What changed?
- When?
- Which one is newest?

Large size

No redundancy



Version Control with Git

Versions as commits

- Incremental changes

Clear history

Can return to any previous commit

Lightweight and resilient

Change tracking

Branching

The screenshot displays a list of Git commits. Each commit entry includes a message, the author's name and profile picture, the time since the commit was made, a green checkmark, a copy icon, a commit hash, and a double arrow icon. The commits are grouped by date.

fixed delay off by 1 issue	tecnd committed 5 days ago ✓	f06a829	<>
removed LED code, added CPU_EVENT_CONFIG	tecnd committed 5 days ago ✓	bd92e2b	<>
Changed delay mechanism to work with external events	tecnd committed 5 days ago ✓	ce608b0	<>
corrected sync code for Pro I modules	tecnd committed 5 days ago	7422da2	<>
started work on external clock ADbasic code	tecnd committed 5 days ago	913ef8c	<>
Commits on Nov 30, 2021			
removed unneeded processdelay line	tecnd committed 8 days ago ✓	0d864cc	<>
added external event test	tecnd committed 8 days ago ✓	9860ce1	<>
Commits on Nov 19, 2021			
Update CITATION.cff ...	tecnd committed 19 days ago ✓	Verified 7fe5bdc	<>
Create CITATION.cff	tecnd committed 19 days ago ✓	Verified b5f036c	<>

Change Tracking

✓ fixed delay off by 1 issue [Browse files](#)

main

tecnd committed 5 days ago 1 parent bd92e2b commit f06a82924885dc5a3afaf058536e2157e0d9e694

Showing 2 changed files with 1 addition and 1 deletion. [Split](#) [Unified](#)

BIN +0 Bytes (100%) ADbasic/TransferDataExternalClock.TB1 [Copy](#) ...
Binary file not shown.

ADbasic/TransferDataExternalClock.bas [Copy](#) ...

↑	@@ -97,7 +97,7 @@ EVENT:	
97		97
98	' If we see a negative number, interpret it as a multi-event delay	98 ' If we see a negative number, interpret it as a multi-event delay
99	if(DATA_1[counts]<0) then	99 if(DATA_1[counts]<0) then
100	- delaymultinuse--1*DATA_1[counts]	100 + delaymultinuse--1 * (DATA_1[counts] + 1)
101	endif	101 endif
102		102
103	Inc counts ' Get number of updates in next event	103 Inc counts ' Get number of updates in next event

Example commit summary

base: 8c313b5 ← compare: main

Commits 136 Files changed 137

Showing 137 changed files with 5,345 additions and 18,606 deletions.

Changes since initial organization

base: fe56ca9 ← compare: main

Commits 138 Files changed 2,720

Showing 2,720 changed files with 5,354 additions and 767,563 deletions.

Changes since very beginning

Documentation

“The only thing worse than no documentation is wrong documentation.”

```
ReadMe Or Else.txt x
1 Mar 10, 2006
2
3 Abandon hope, all ye who enter here.
4
```

```
47 GuiDesign.c The 'engine'
48 {
49 ***** important subroutines *****
50 RunOnce Takes the 10 'pages' and creates a single page, a 'meta' list.
51 BuildUpdateList The 'real' engine. This takes the Meta-arrays and generates the 3 arrays that are transferred to the ADwin.
52 Also runs routines to generate the DDS commands (only the 1st is currently enabled)
53 Boots ADwin and sends the program to it.
54
55
56 CalcFcnValue pretty straightforward. Calculates next value of the analog channel.
57 Optimize TimeLoop compresses strings of zero's in the updatenum[] array. Replaces with a negative number
58 UpdateScanValue Determine next value in a scan.
59 Timer_Callback Controls what happens when repeated runs occur...i.e. do we scan a value?
60 SCANSETTING_CALLBACK
61
62 DrawNewTable
63 DigTableCallback
64 TimeTableCallback
65 Cmd_Run_Callback
66 Cmd_Scan_Callback
67 Cmd_Stop_Callback
68
69 BootAdwin
70 BootLoad
```

Documentation in Code

Comments exist in code, but...

- Scattered through multiple files
- Difficult to cross-reference
- “Wall of text”

```
45 2005
46 July 29 Added option to output a history of the Scan values when running a scan.
47         Adding option to stream the panel to text files;
48 JULY 20 commands for DDS2 are now generated, also sent to ADwin now (see new
49 ADBasic code TransferData_Jul20.BAS) Fixed ADBasic software bug. Turns out
50 that DIG_WRITELATCH1 and DIG_WRITELATCH2 (for lower and upper 16 bits
51 respectively) are incompatible. Use DIG_WRITELATCH32 instead. July 18 Added 3rd
52 DDS interface, simplified DDS control by reusing the existing DDS control panel
53         DDS 2 and 3 clock settings displayed in DDSSettings.uir Not
54 modifiable. Set using a #define in vars.h Save DDS 2,3 info to file. July4
55 Adding 2nd DDS (interface only, creating dual dds command structure still needs
56 to be done.) June7 Finalized scan programming. Now scans in amplitude, time
57 or DDS frequency. (Only DDS1 so far) April 20 Changed the way that the table
58 cells are coloured. Now all cells are coloured based on the information in the
59 cell. No longer based on the history of that row. Sine wave output now relabels
60 amplitude and frequency on analog control panel. Colours Cyan on table. April 7
61 Fixed a bug where we didn't reach the final value on a ramp, but reached the
62 value before. Cause: in calculating ramps, i.e determine slope=amplitude/number
63 of steps but should be amplitude/number_steps-1 Mar 23 Added A Frequency Offset
64 to the DDS...so the same ramp can be continually used while changing the trap
65 bottom. Fixed duration of ramps etc at column duration. Mar 10 Fixed a problem
66 where the frequency ramps generated by the DDS finished in half the expected
67 time Can't find a reason for this, except that the DDS manual might be wrong.
68         Added a Sinewave option to the list of possible functions. Only
69 accepts amplitude and frequency..no bias
70         - bias could be 'worked around' using bias setting under
71 Analog Channel Setup Jan 18 Add menu option to turn off the DDS for all
72 cells. Avoids a string of warnings created by the DDS command routines if a DDS
73 command is written before the previous DDS command was done; Jan 5 Fix bug in
74 code where the timing isn't always copied into the DDS commands Fixed the last
75 panel mobile ability
```

Documentation from Code with Doxygen

Generates searchable, formatted, hyperlinked documentation from comments

Can be accessed online

Regenerates on every commit

```
238 // Send the new arrays to BuildUpdateList()
239 BuildUpdateList(MetaTimeArray, MetaAnalogArray, MetaDigitalArray, mindex);
240 }
241
242 /**
243 @param TMatrix[] Stores the interval time of each column
244 @param AMat[] Stores info located in the analog table
245 @param DMat[] Stores info located in the digital table
246
247 all the above have 500 update period elements note that valid elements are base1
248
249 @param numtimes = the actual number of valid update period elements.
250
251 Generate the data that is sent to the ADwin and sends the data.
252 From the meta-lists, we generate 3 arrays.
253 - UpdateNum - each entry is the number of channel updates that we perform during the ADwin EVENT, where an
254 ADwin event is an update cycle, i.e. 10 microseconds, 100 microseconds... etc. We advance through this
255 array once per ADwin Event. UpdateNum controls how fast we scan through ChNum and ChVal
256 - ChNum - An array that contains the channel number to be updated. Synchronous with ChVal. Channels listed below
257 - ChVal - An array that contains the value to be written to a channel. Synchronous with ChVal.
258
259 - ChNum
260 - Value 1-32: Analog lines, 4 cards with 8 lines each. ChVal is -10V to 10V
261 - Value 51: DDS1 line. ChVal is either a 2-bit value (0-3) to write, or (4-7) a reset signal
262 - Value 52: DDS2 line. ChVal is either a 2-bit value (0-3) to write, or (4-7) a reset signal
263 - Value 101, 102 First 16 and last 16 lines on the first DIO card. ChVal is a 16 bit integer
264 - Value 103, 104 First 16 and last 16 lines on the second DIO card. ChVal is a 16 bit integer
265 - ChNum 201, 202 These are codes to enable/disable looping. Corresponding ChVal is the number of loops.
```

Function Documentation

◆ BuildUpdateList()

```
void BuildUpdateList ( double          TMatrix[500],
                      struct AnalogTableValues AMat[NUMBERANALOGCHANNELS+1][500],
                      int              DMat[NUMBERDIGITALCHANNELS+1][500],
                      int              numtimes
                      )
```

Parameters

- TMatrix[]** Stores the interval time of each column
- AMat[]** Stores info located in the analog table
- DMat[]** Stores info located in the digital table

all the above have 500 update period elements note that valid elements are base1

Parameters

- numtimes** = the actual number of valid update period elements.

Generate the data that is sent to the ADwin and sends the data. From the meta-lists, we generate 3 arrays.

- UpdateNum - each entry is the number of channel updates that we perform during the ADwin EVENT, where an ADwin event is an update cycle, i.e. 10 microseconds, 100 microseconds... etc. We advance through this array once per ADwin Event. UpdateNum controls how

























Some things never change.

ddstranlator.c

This file contains functions necessary for interpreting the data entered into the control panels and outputting two-bit commands for the serial programming of the DDSes. This file remains a mystery to me, but I'm sure whomever is reading this more than has the intellectual capacity to understand it. Go you.

File List

Here is a list of all documented files with brief descriptions:

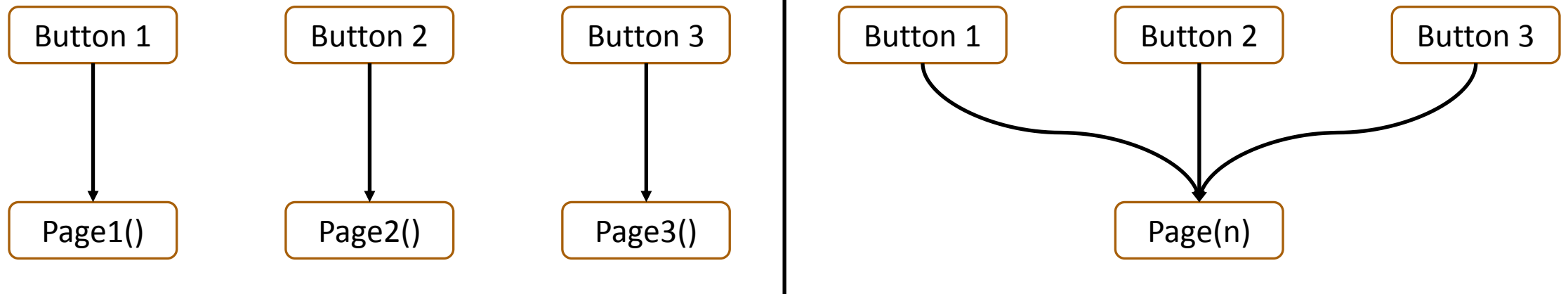
 Adwin.h	
 AnalogControl.c	This panel is used to set the value of an AnalogTableValues in AnalogTable
 AnalogControl.h	
 AnalogControl2.h	
 AnalogSettings.c	Sets the AnalogChannelProperties for each analog line
 AnalogSettings.h	
 AnalogSettings2.h	
 Comments.c	Contains code for the comments window
 Comments.h	
 DigitalSettings.c	Sets the DigitalChannelProperties for each digital line
 DigitalSettings.h	
 DigitalSettings2.h	
 GUIDesign.c	Contains functions and callbacks for primary GUI. Handles data processing a
 GUIDesign.h	
 GUIDesign2.h	
 main.c	Contains code to build the GUI layout
 main.h	Defines arrays for dynamically-generated labels, buttons, and checkboxes
 scan.c	The inner workings of this is a mystery, wrapped in an enigma
 Scan.h	
 scan2.h	
 ScanTableLoader.c	Generates values for the scan table using either linear or exponential rampin
 ScanTableLoader.h	
 ScanTableLoader2.h	
 vars.h	Contains global macros, variables, typedefs, and struct definitions

Expanding the Software

Remove broken/obsolete code

Dynamic instantiation of elements

Code deduplication



Save File Format

Changing the number of saved elements breaks save compatibility

Reverse-engineer save format to keep old experiments usable

Template injection

```
C:\Users\labadmin\Desktop\adwinGUI\NewTemplate.pan
0000 00A0: 44 45 53 43 30 31 30 31 00 00 00 04 00 00 00 00 DESC0101 .....
0000 00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 00D0: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 00E0: 44 45 53 43 30 31 30 32 00 00 00 04 00 00 00 00 DESC0102 .....
0000 00F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0110: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 0120: 44 45 53 43 30 31 30 33 00 00 00 04 00 00 00 00 DESC0103 .....
0000 0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0150: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 0160: 44 45 53 43 30 31 30 34 00 00 00 04 00 00 00 00 DESC0104 .....
0000 0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0190: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 01A0: 44 45 53 43 30 31 30 35 00 00 00 04 00 00 00 00 DESC0105 .....
0000 01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 01D0: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 01E0: 44 45 53 43 30 31 30 36 00 00 00 04 00 00 00 00 DESC0106 .....
0000 01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

\\SNAPDRAGON\Common\Undergrads\Kerry\test2.pan
0000 00A0: 4D 4F 54 00 00 00 00 00 00 00 00 04 00 00 00 00 MOT.....
0000 00B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 00C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 00D0: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 00E0: 4D 4F 54 32 00 00 00 00 00 00 00 04 00 00 00 00 MOT2.....
0000 00F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0110: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 0120: 4F 50 73 68 75 74 74 65 00 00 00 04 00 00 00 00 OPshutte
0000 0130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0140: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0150: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 0160: 52 61 64 73 68 75 74 00 00 00 00 04 00 00 00 00 Radshut.
0000 0170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 0190: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 01A0: 00 00 00 00 00 00 00 00 00 00 00 00 04 00 00 00 .....
0000 01B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 01C0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0000 01D0: 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 08 .....
0000 01E0: 00 00 00 00 00 00 00 00 00 00 00 00 04 00 00 00 .....
0000 01F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

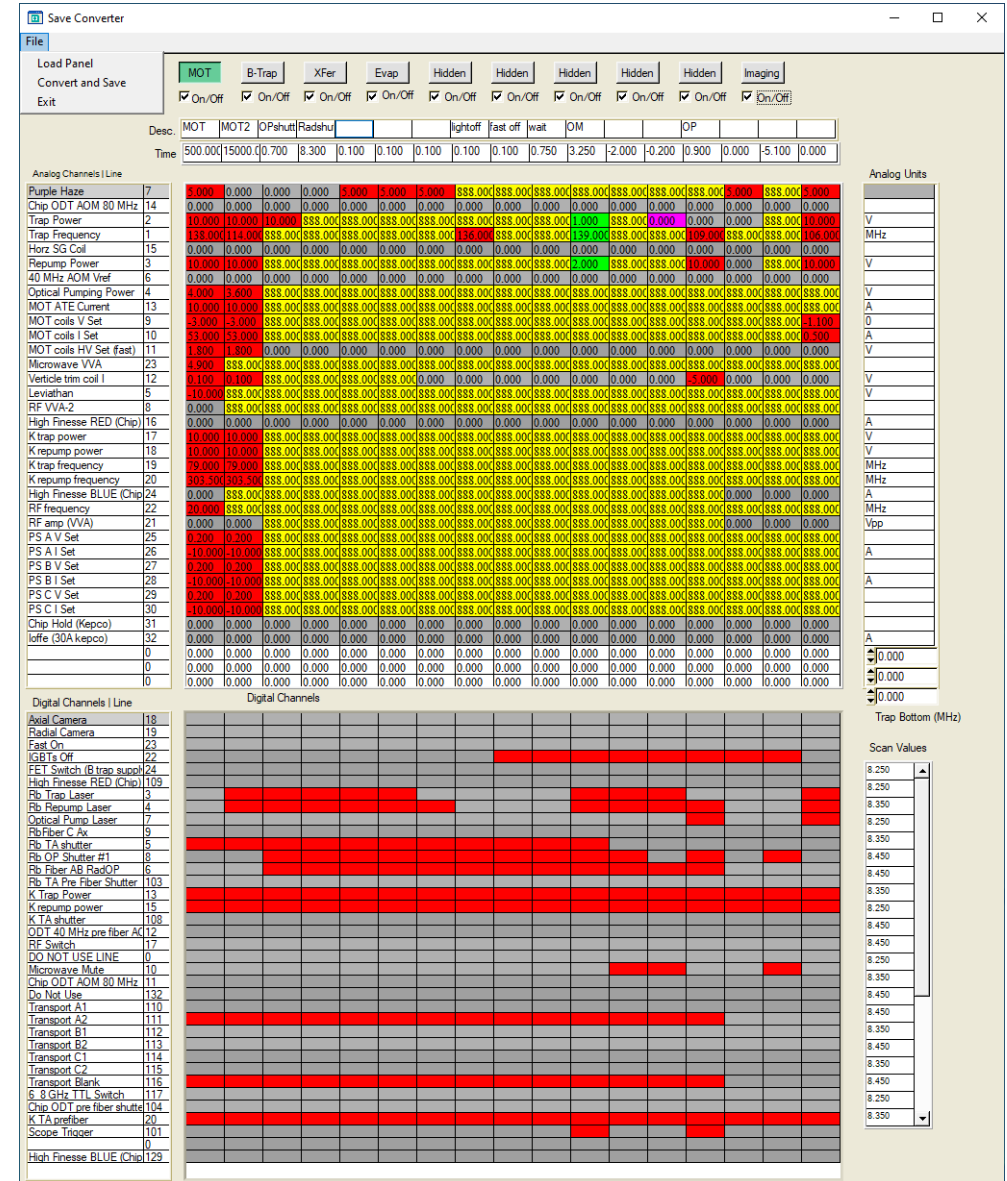
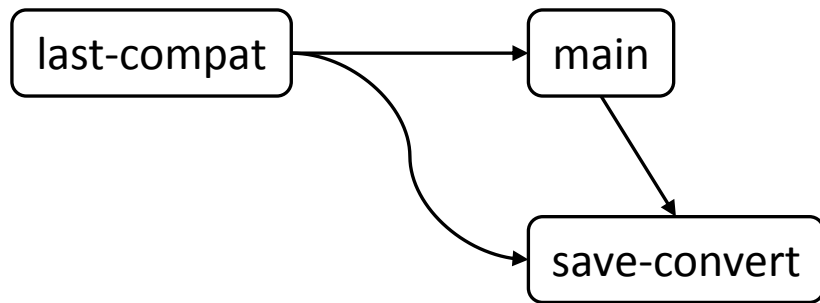
Arrow keys move  F find      RET next difference  ESC quit  ALT freeze top
C ASCII/EBCDIC  E edit file   G goto position    Q quit   CTRL freeze bottom
```

Save File Conversion

Branch split

Stripped down version of old software

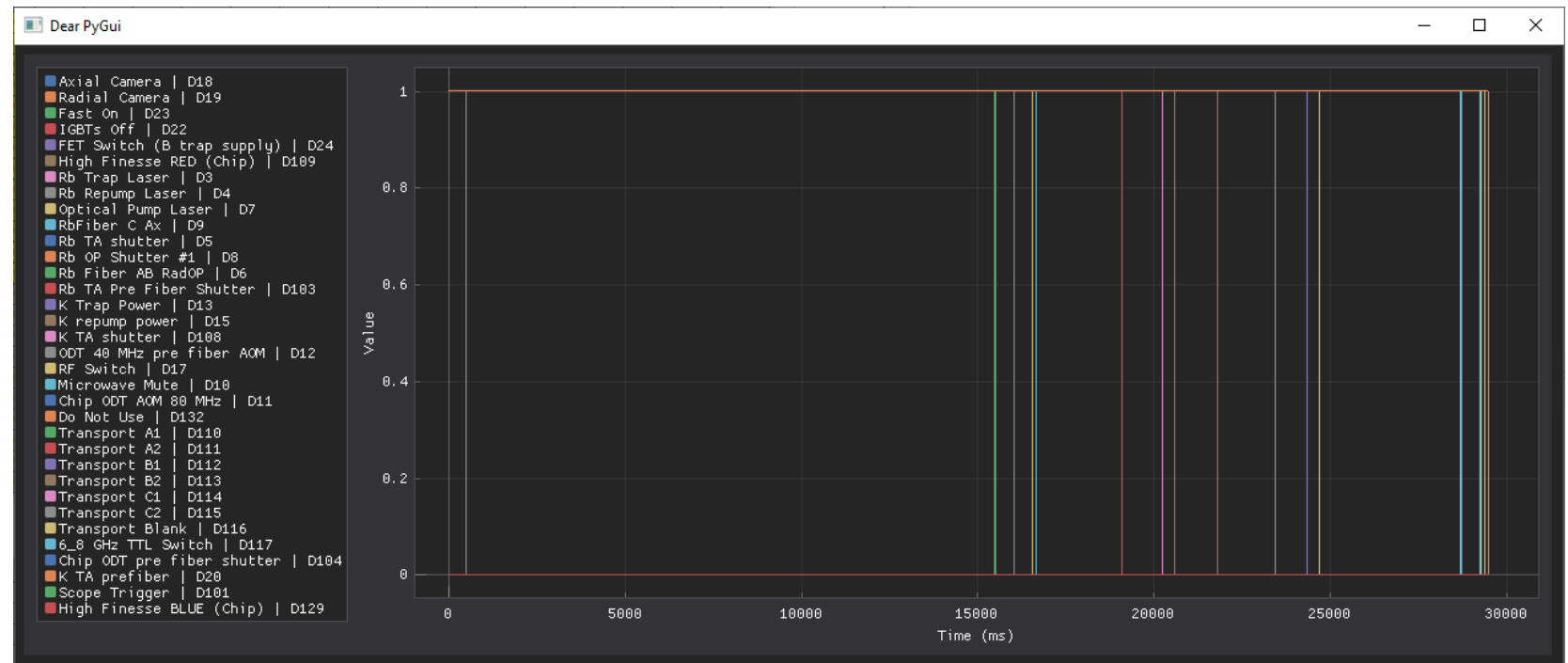
Saves in new format



Add-ons

Export data in text format for external programs

- Can use a different language
- Not dependent on LabWindows codebase



Overview

Background and Motivation

Project 1: Improving the Control Software

Project 2: Improving Timing Performance

Summary

Improving Timing Performance

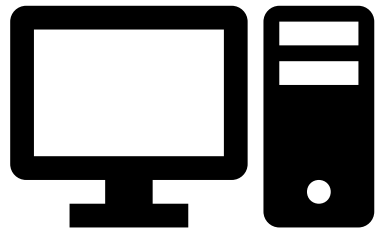
Integrate with atomic clock

- Stanford Research Systems PRS10 Rubidium Standard
- Drifts <0.05 nanoseconds per second

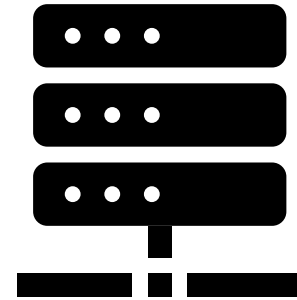
Measure and reduce jitter

- Characterize precision and accuracy
- Did atomic clock integration help?

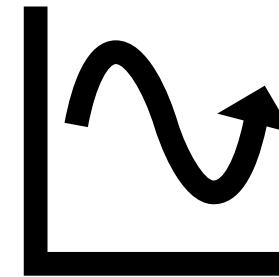
System Diagram



Computer running
control software
(LabWindows/CVI)



ADwin-Pro II
sequencer running
sequencer software
(ADbasic)

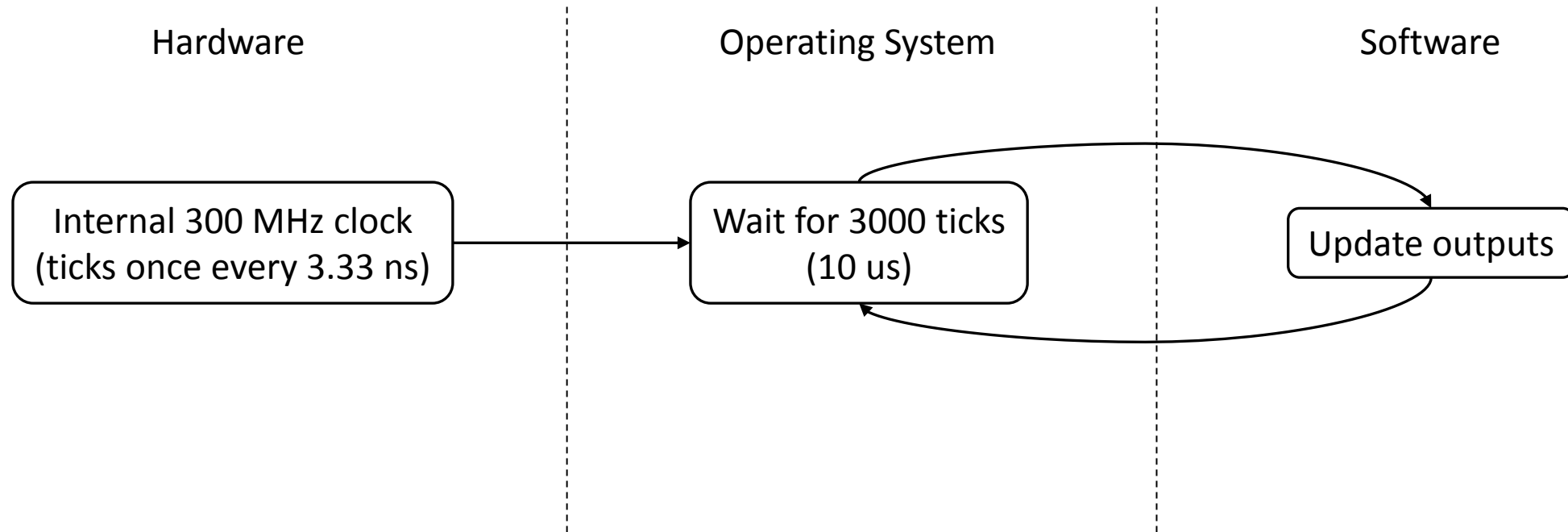


Analog outputs

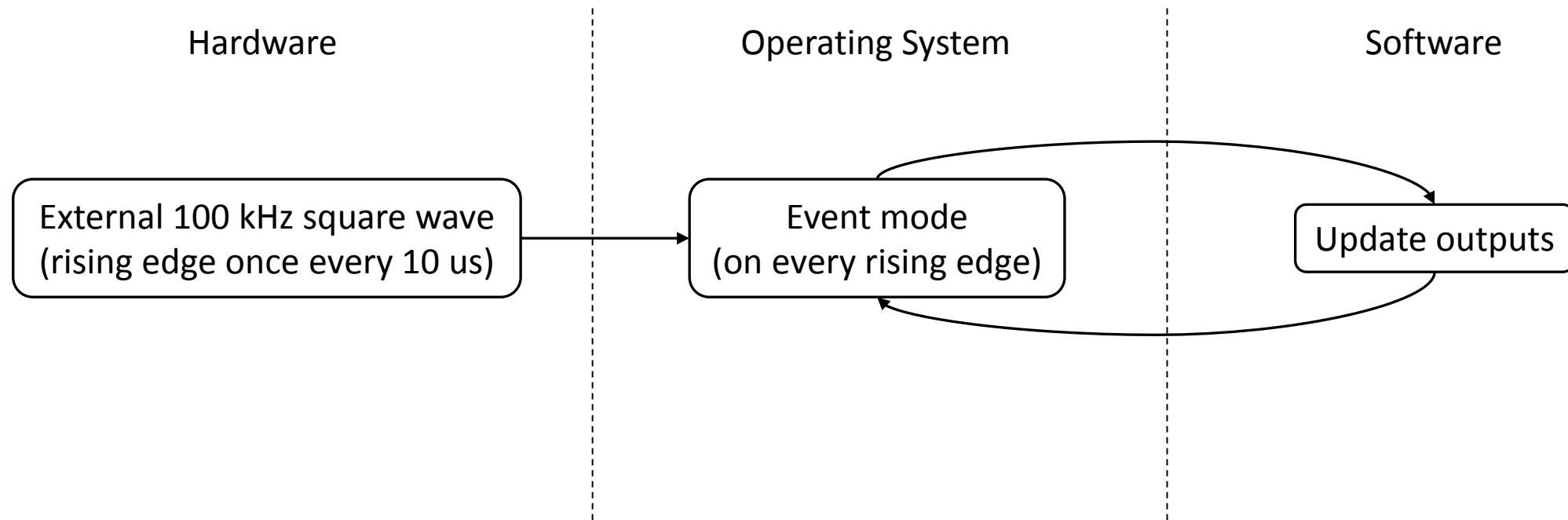
1010
1010

Digital outputs

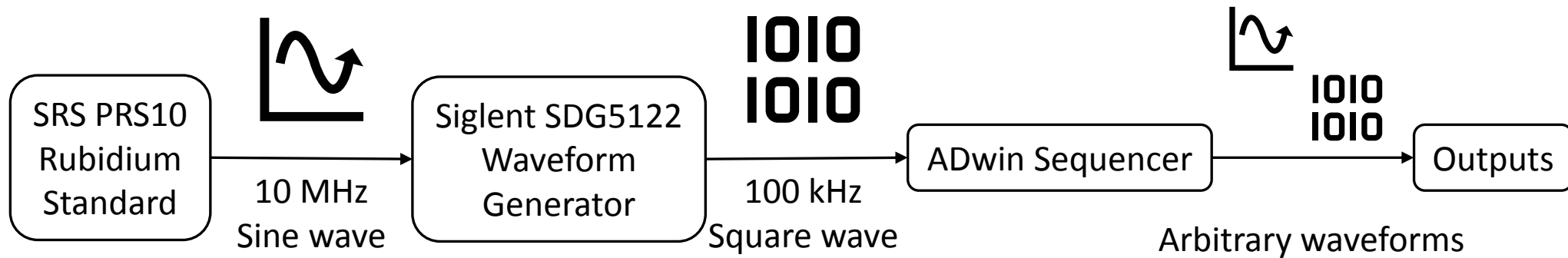
Timekeeping with Internal Clock



Timekeeping with Event Mode



Integrating the Atomic Clock

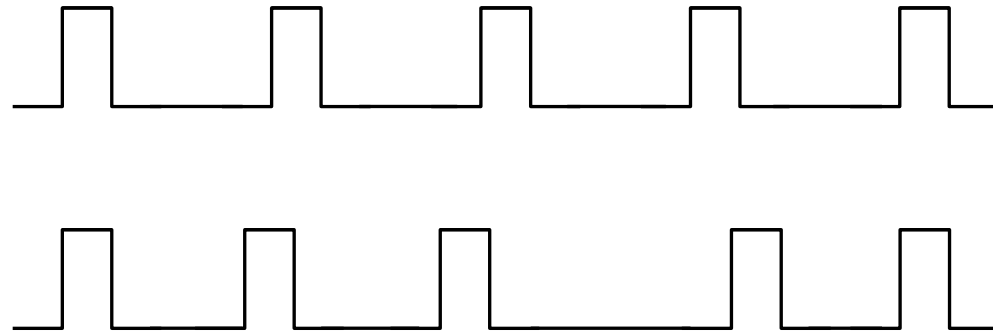


Jitter

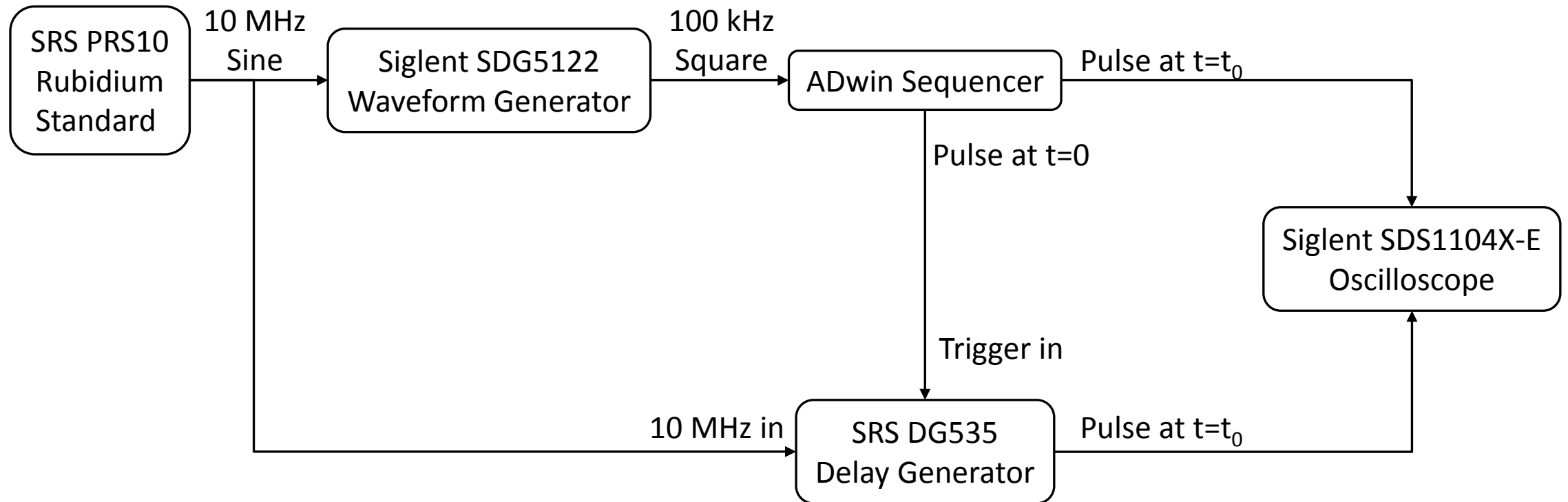
Clocks are not perfect

Clock “ticks” may not be spaced evenly

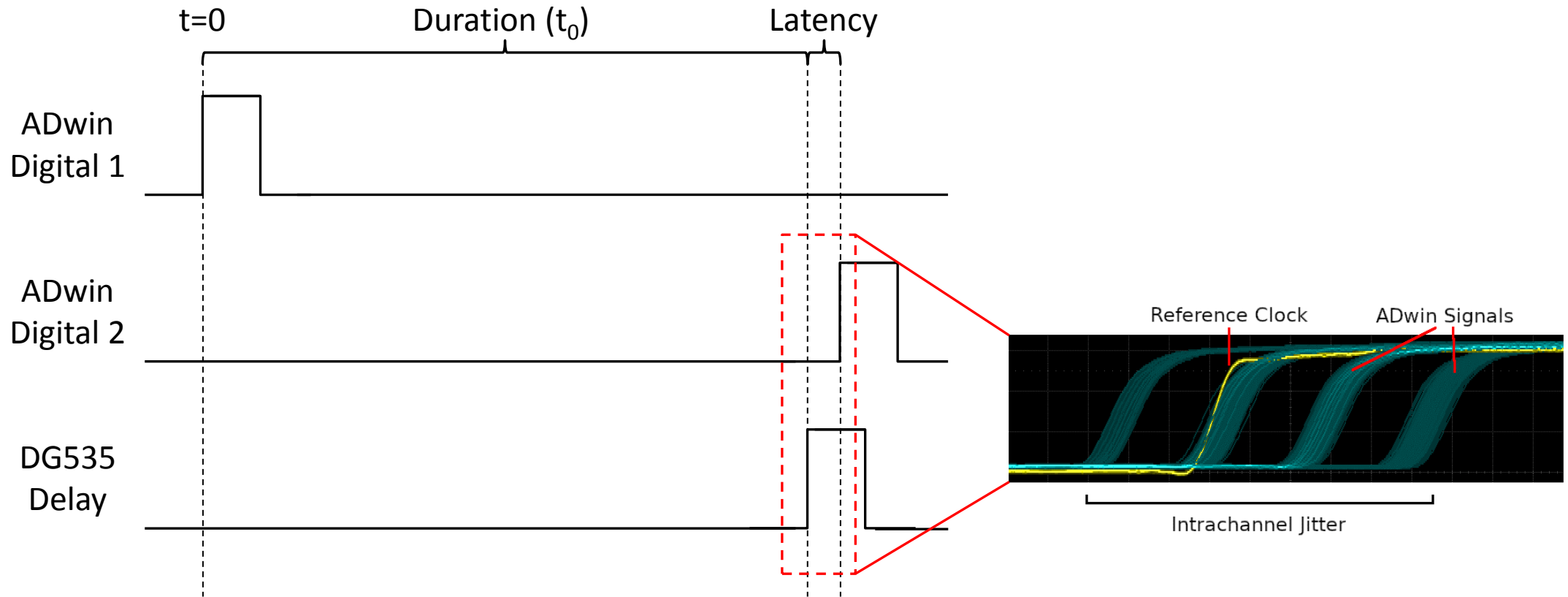
Causes variations across many trials



Measuring Jitter



Measuring Jitter



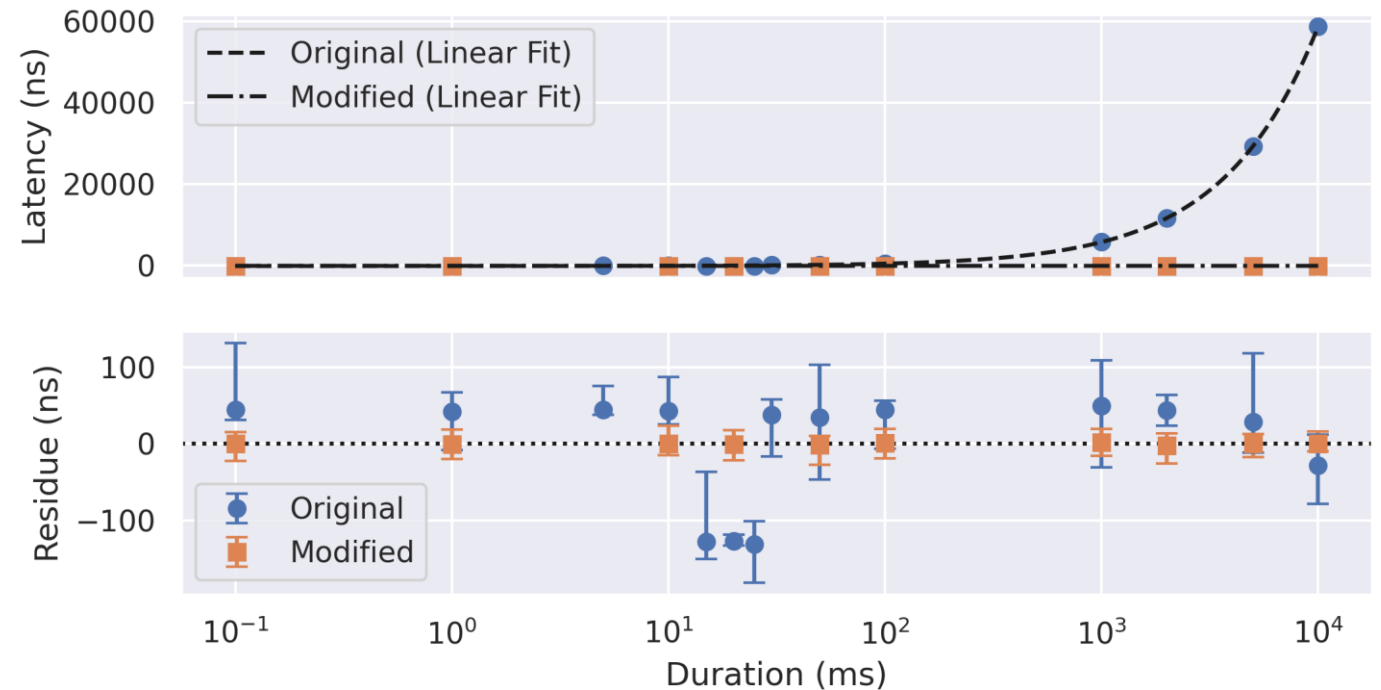
Intrachannel Jitter

ADwin internal clock drifts at 5.895 ppm against the reference

- Around 500 ms per day
- <150 ns jitter

Atomic clock integrated successfully

- <40 ns jitter

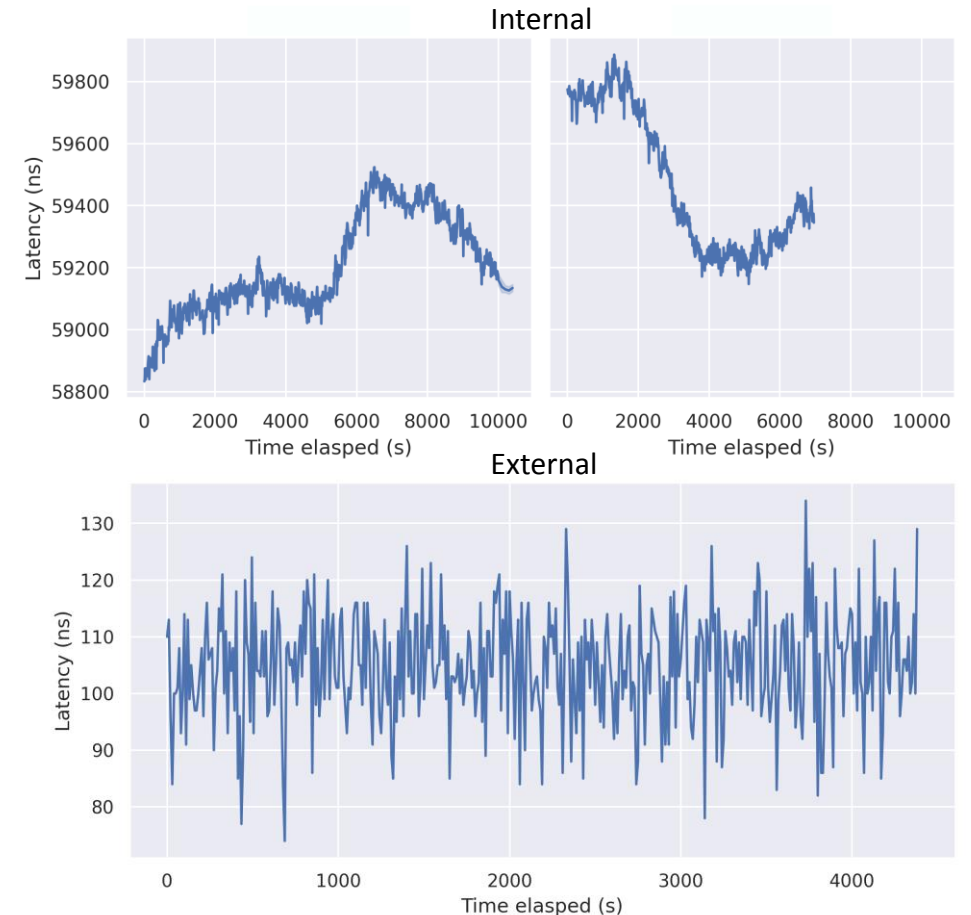


Long-term Intrachannel Jitter

Repeat experiment with $t_0=10$ seconds

- Internal clock wanders away from initial value
- More jitter than when measured over short periods
- External clock stays steady

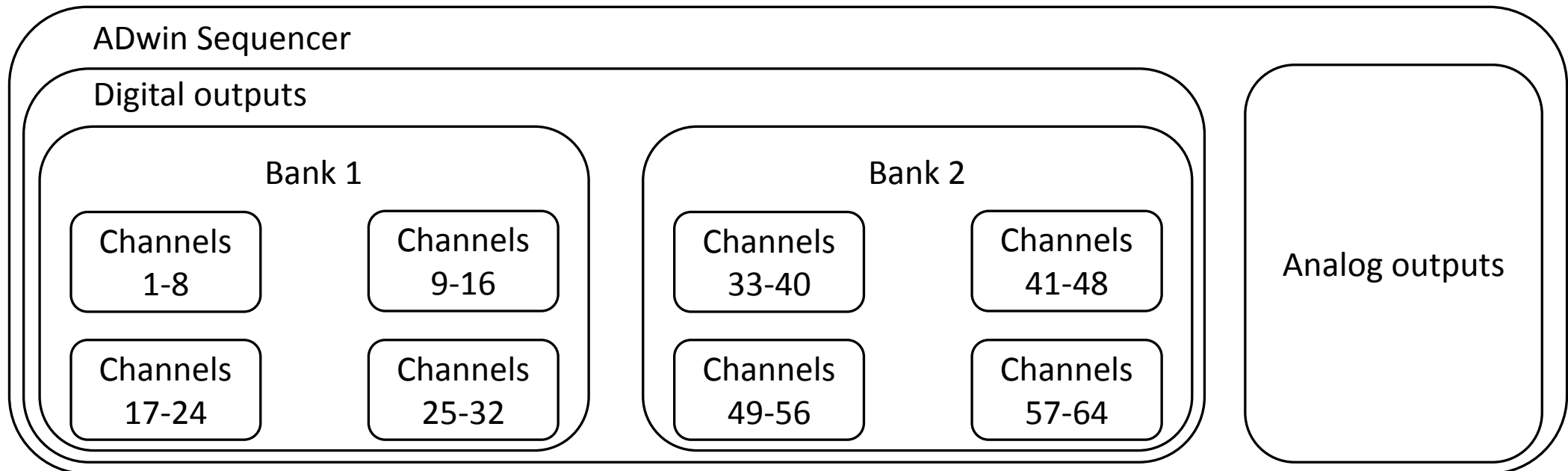
Clock	Samples	Range	Std. Deviation
Internal	1039	690 ns	158 ns
Internal	697	740 ns	225 ns
External	435	60 ns	6.5 ns



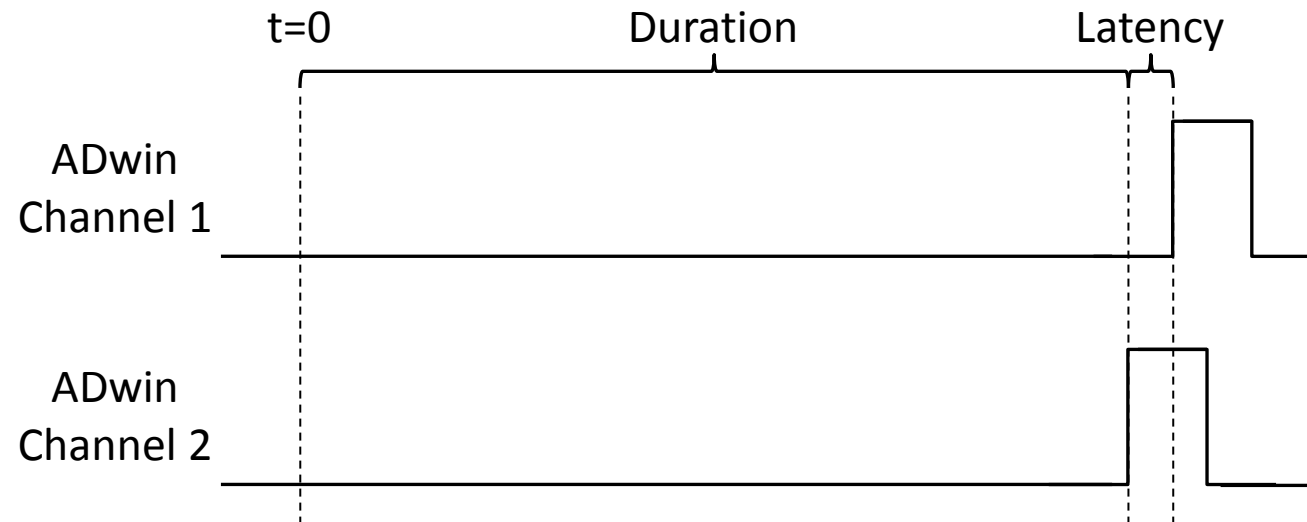
Interchannel Jitter

When multiple channels transition at the same time, they should do so

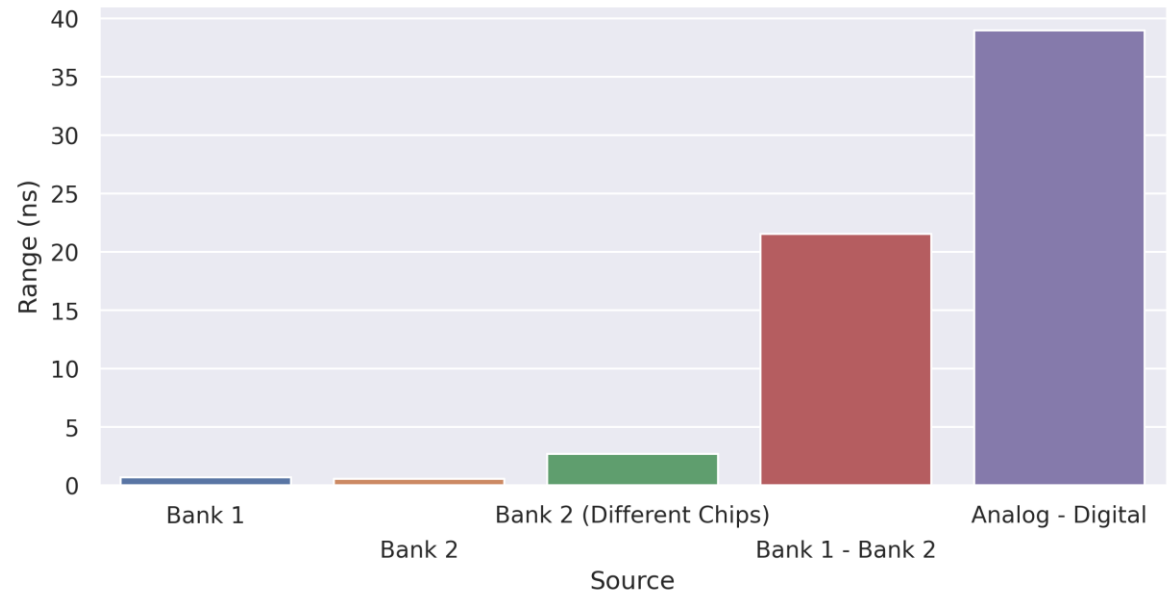
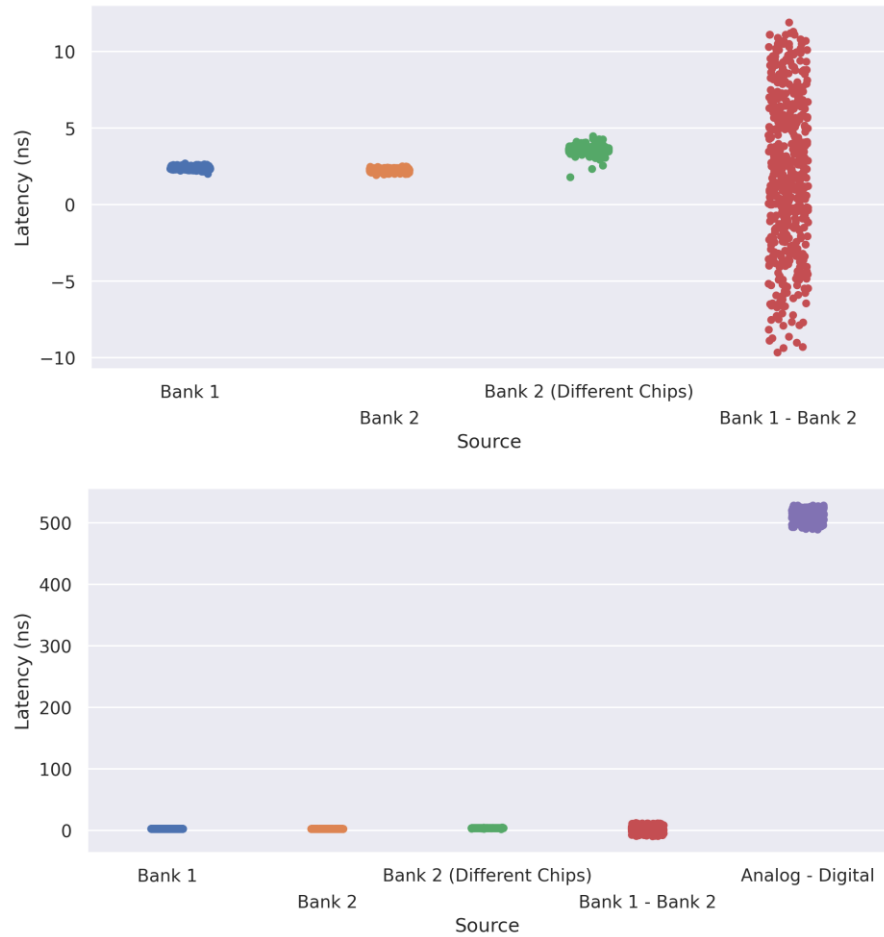
- How much delay?
- Do different channel sources behave differently?



Measuring Interchannel Jitter



Interchannel Jitter



Summary

Reworked control software

- Supports up to 40 analog and 64 digital devices, easily expandable in code
- Cleaned up and organized the codebase
- Wrote documentation
- Created framework for add-on programs

Integrated ADwin sequencer with atomic clock

- Decreased intrachannel jitter from 740 ns to 60 ns
- Characterized interchannel jitter between many channel sources

Acknowledgements

Dr. Seth Aubin, for his guidance and support

My fellow lab members: Will, Stephen, Morgan, and Sindu, for keeping me company

Work in support of NSF and DTRA projects

