Applying Model-Based Design to a GPS Receiver

Mike Donovan
Senior Applications Engineer
The MathWorks
June 15, 2006

Outline

- Goals and Objectives
- GPS History and Basics
- A Simple Receiver Model
- Detailed Model with Tracking Loops
- Acquiring Actual GPS Signals
- Searching for Satellites
- Receiver Using Actual Data
- Transition to Fixed Point
- FPGA Partition
- DSP Partition
- Hardware Demo

Boeing Delta II lofting a new GPS satellite, June 2004

Goals and Objectives

- Model-Based Design example – from concept to operating hardware
- Contemporary communications concepts
  - Fixed-rate receiver sampling
    - Fractional delay timing adjustment
    - Amenable to FPGA and ASIC
  - Code division multiple access
    - Signal acquisition
    - Signal tracking
- Hardware implementation
  - FPGA for heavy lifting
  - DSP for acquisition and control laws

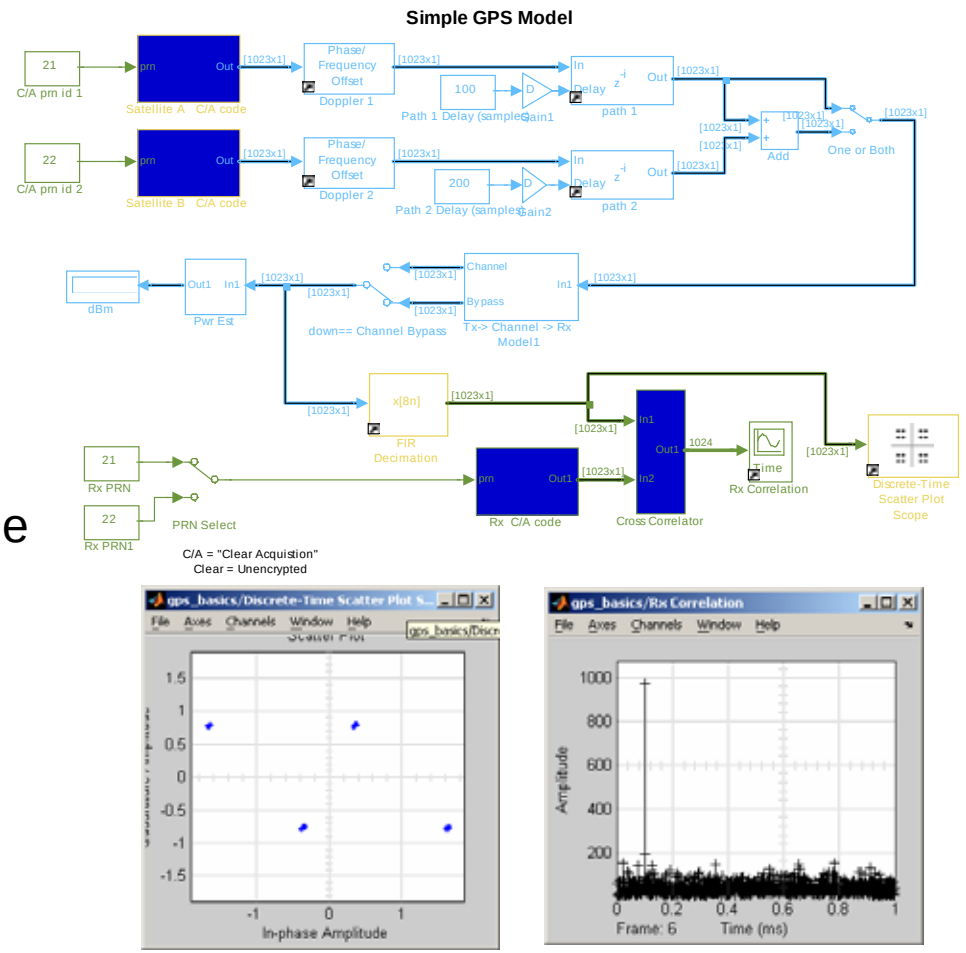
Global Positioning System Facts

- There are 24 satellites in the constellation.
- They orbit the earth about 12,255 miles (20,200 Km) above us.
- Traveling at 7000 miles an hour, they complete two orbits in less than 24 hours.
- Transmitter power is <=50 watts in the “L” band (1-2GHz), 10 dB antenna gain.
- Civilian GPS uses the frequency of 1575.42 MHz in the UHF band.
- All GPS timing is a multiple of 1.023 MHz (e.g., 1540*1.023MHz=1.57542 GHz).

(Garmin: http://www.garmin.com/aboutGPS/)

A Simple GPS Model

- Two clear acquisition (C/A) code (base band) transmitters
- Band limiting filters
- Simple channel model
- Cross correlating receiver
  - Q: How does one tell time?
  - A: Cross correlation of local code with received code
- Model purpose
  - Investigation of basic GPS concepts
  - CDMA example



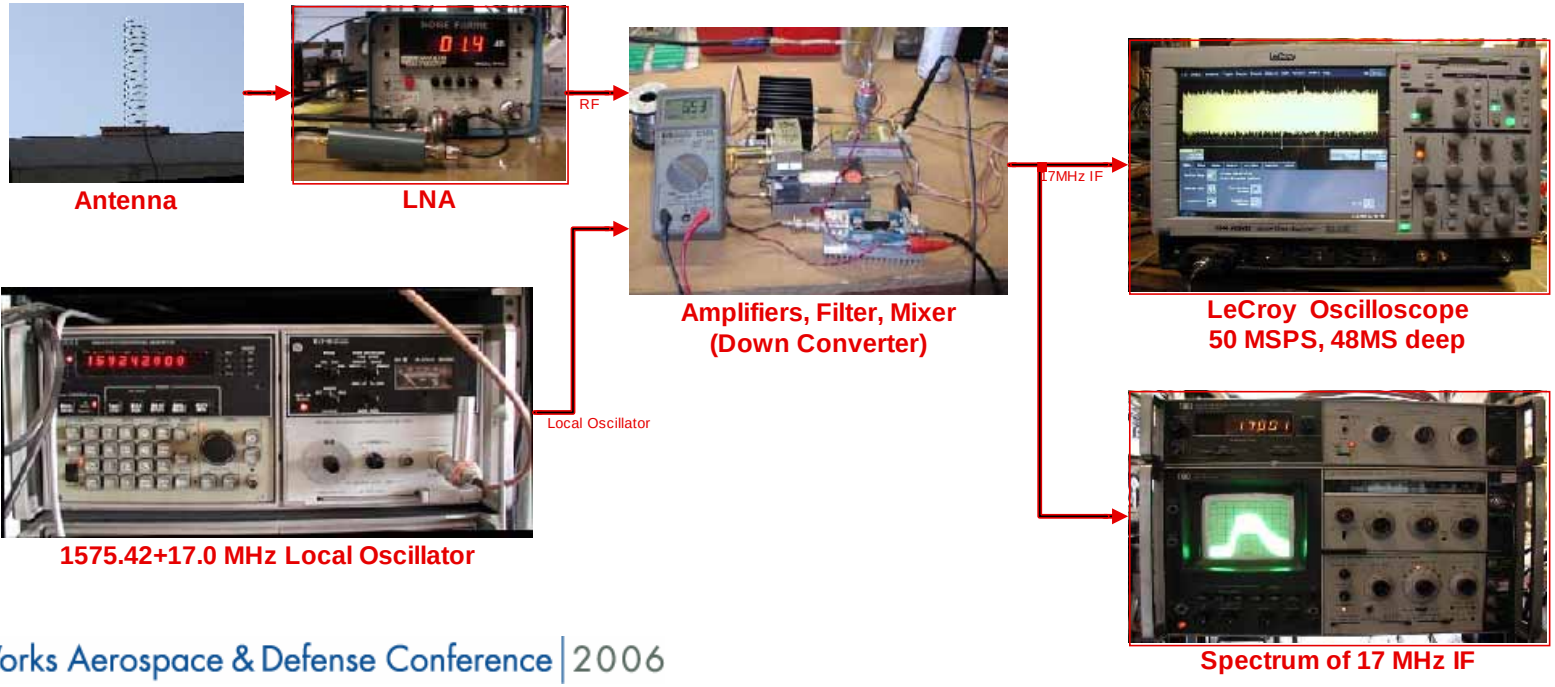
Limitations of Simple Model

- Doesn't use real GPS satellite source data
- No acquisition mode (search through possible Doppler shift and C/A code phase)
- Assumes perfect clock synchronization between Rx and Tx

Capturing Satellite Signals (LeCroy)

- Need down conversion hardware for rational data set size
- Instrument Control Toolbox *or* LeCroy interface

GPS Down Converter and Signal Capture Hardware



Simple Script to Capture 17 MHz IF

- Scope parameters
  - Fs= 50 MSPS
  - 48e6 MS deep memory
- Script: Eight lines of MATLAB®

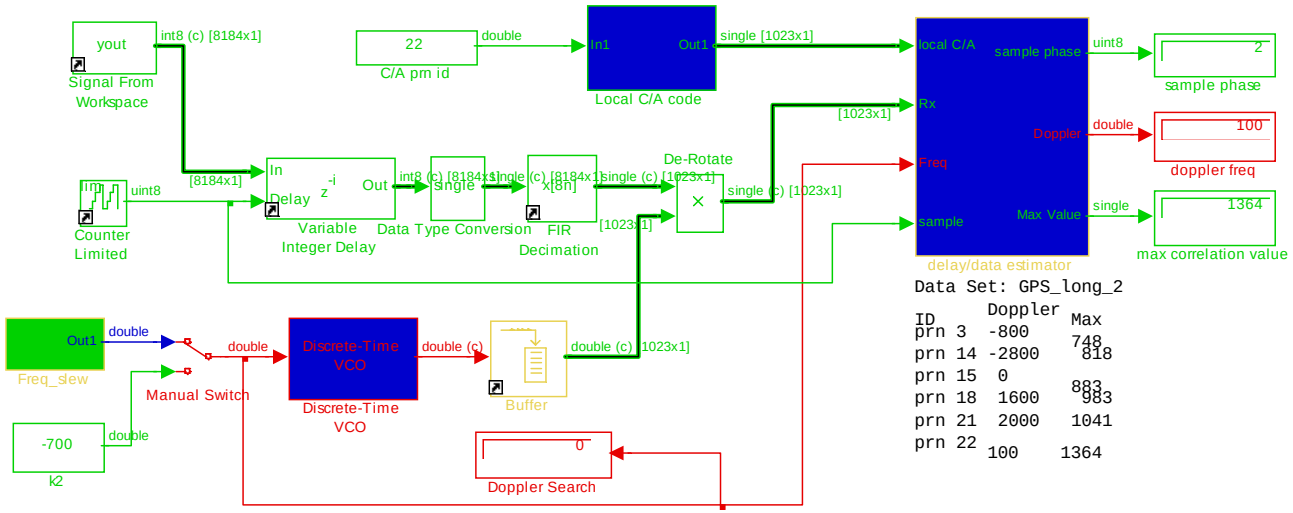
Editor - g:\tmw_r14\simulink\comms_models\GPS\GPS_Capture.m

File Edit Text Cell Tools Debug Desktop Window Help

```
1   % GPS Signal Capture Script
2   % This requires LeCroy's ActiveDSO Interface, available
3   % No additional MathWorks products are required.
4   % The scope state has been setup via front panel contro
5   % 50 MSPS rate, 48 M words record length, 0.1V/div, BWL
6   % and a single shot capture has been made.
7
8   %% Create ActiveX Control to establish communication wi
9 - hDSO = actxcontrol('lecroy.activedsoctrl.1',[130 80 800
10 - MakeConnection(hDSO,'TCPIP: 192.168.0.13'); % static IP
11
12  %  inspect(hDSO)      %What do we know about it?
13  %  methodsview(hDSO)  %What can we do with it, and how?
14
15  % returns unsigned int8, need to remove offset, and rec
16 - y = int8(int16((GetByteWaveform(hDSO,'C1',48e6,0)))-128
17 - figure; plot(y(1:1e5)); % sanity check
18 - Disconnect(hDSO);
19 - clear hDSO
20 - s=[6 10 30 21 2 29 5]; % satellites that should be in t
21  % save the capture
22 - save GPS_long_3_X y s
```

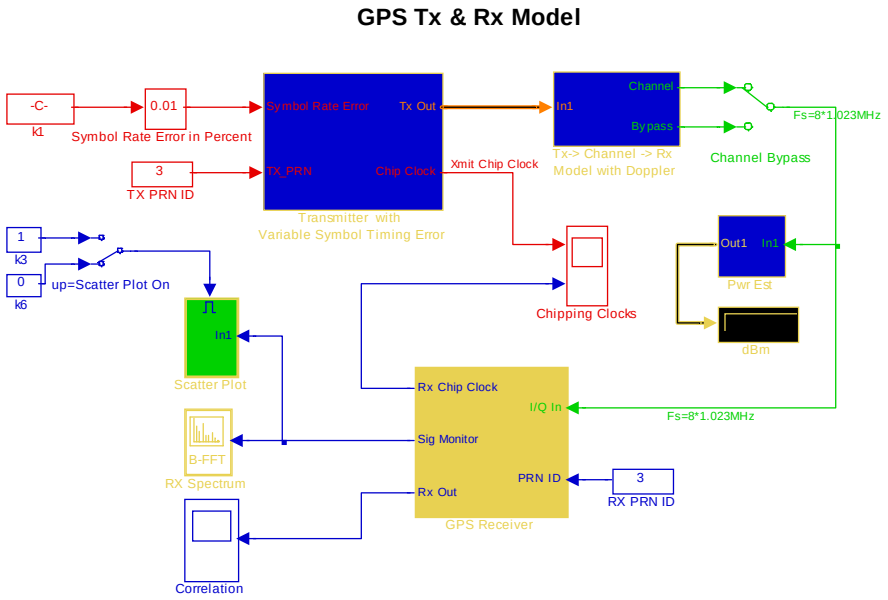
Searching for Satellites in the Data Set

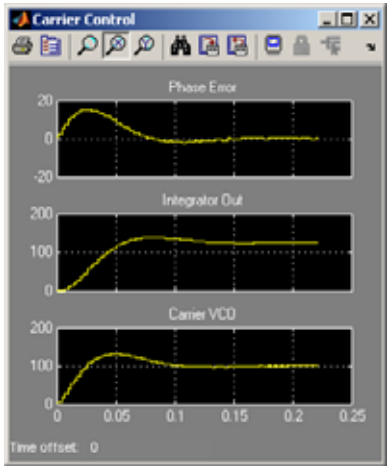
- Typically three parameters to search
  - Which birds are visible (PRN ID)
  - C/A code phase
  - Doppler shift
- Commercial GPS used to identify visible birds (simplify search)
- Key model outputs
  - Verification of signal level
  - Doppler shift value

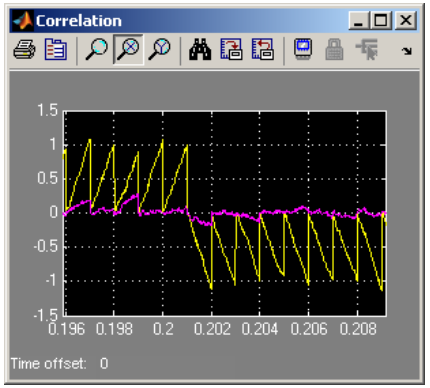


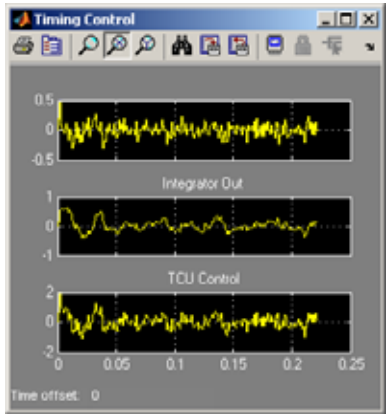
GPS Model with Timing Recovery

- Variable clock rate at Tx (Fs_Tx = 8*1.023e6*(1 +/- delta))
- Essentially the same channel model including Doppler
- Fixed sampling rate receiver (Fs_Rx = 8*1.023e6)
- Fractional delay scheme for C/A Rx code loop
- Doppler NCO loop



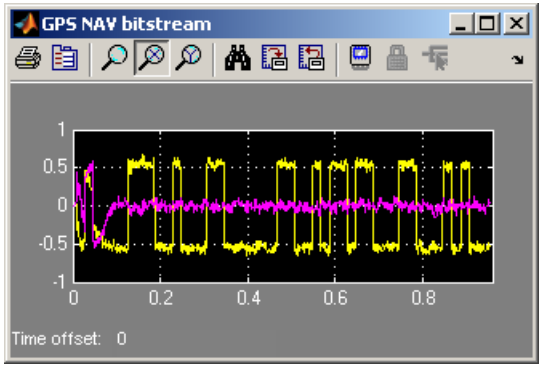


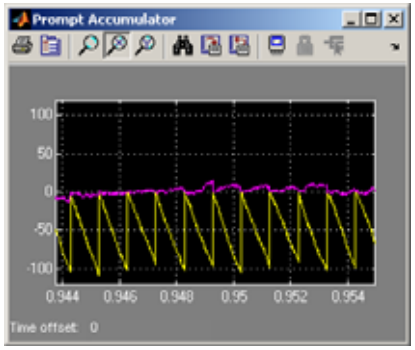




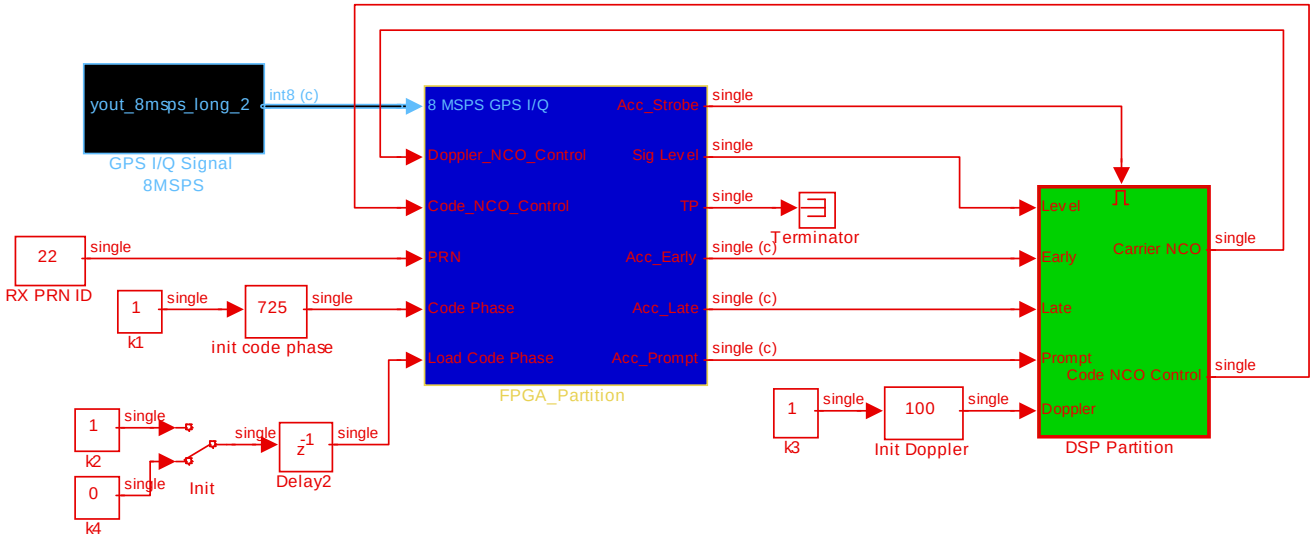
Receiver Processing Satellite Data

- Fixed 8.000 MSPS I/Q data rate
- Partitioning of tasks to FPGA and DSP
- Inputs: PRN, Doppler, and initial C/A code phase
- Simple code acquisition control
- Model *verifies* that the processing scheme is sound



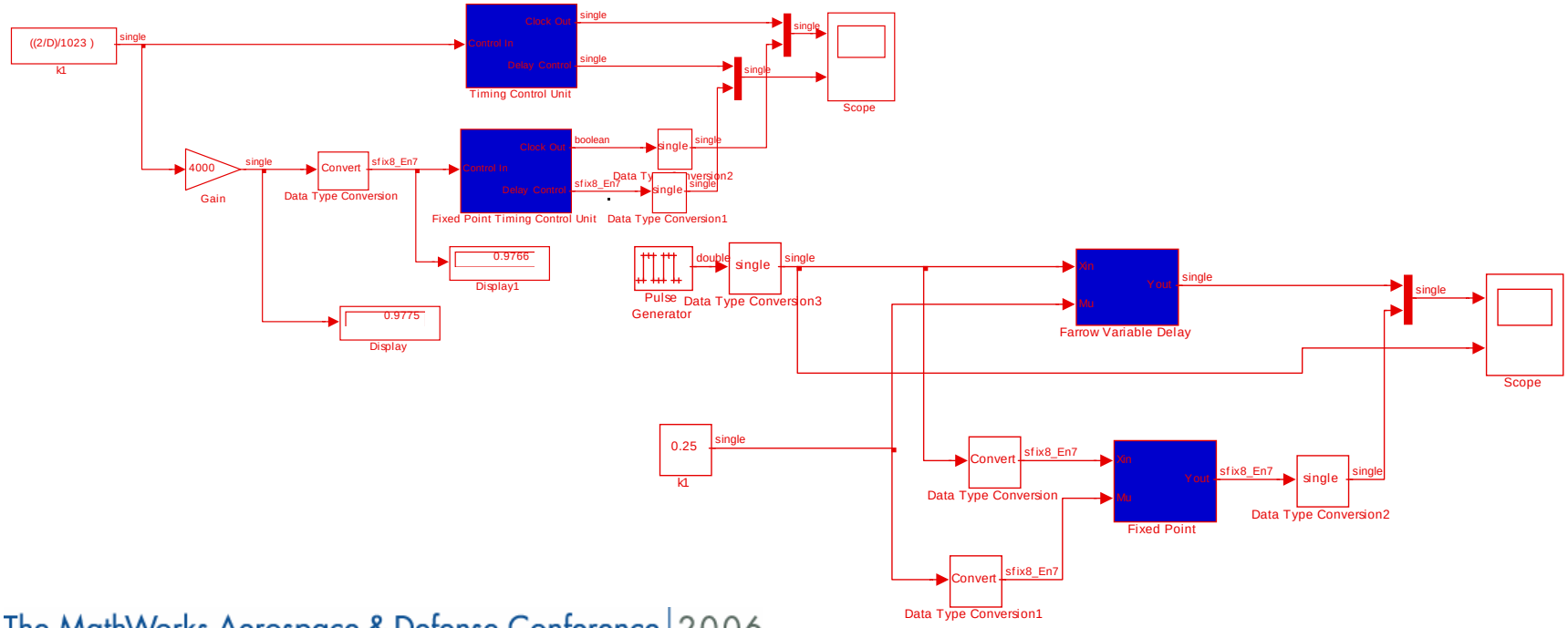


GPS Receiver Processing Captured Satellite Data.



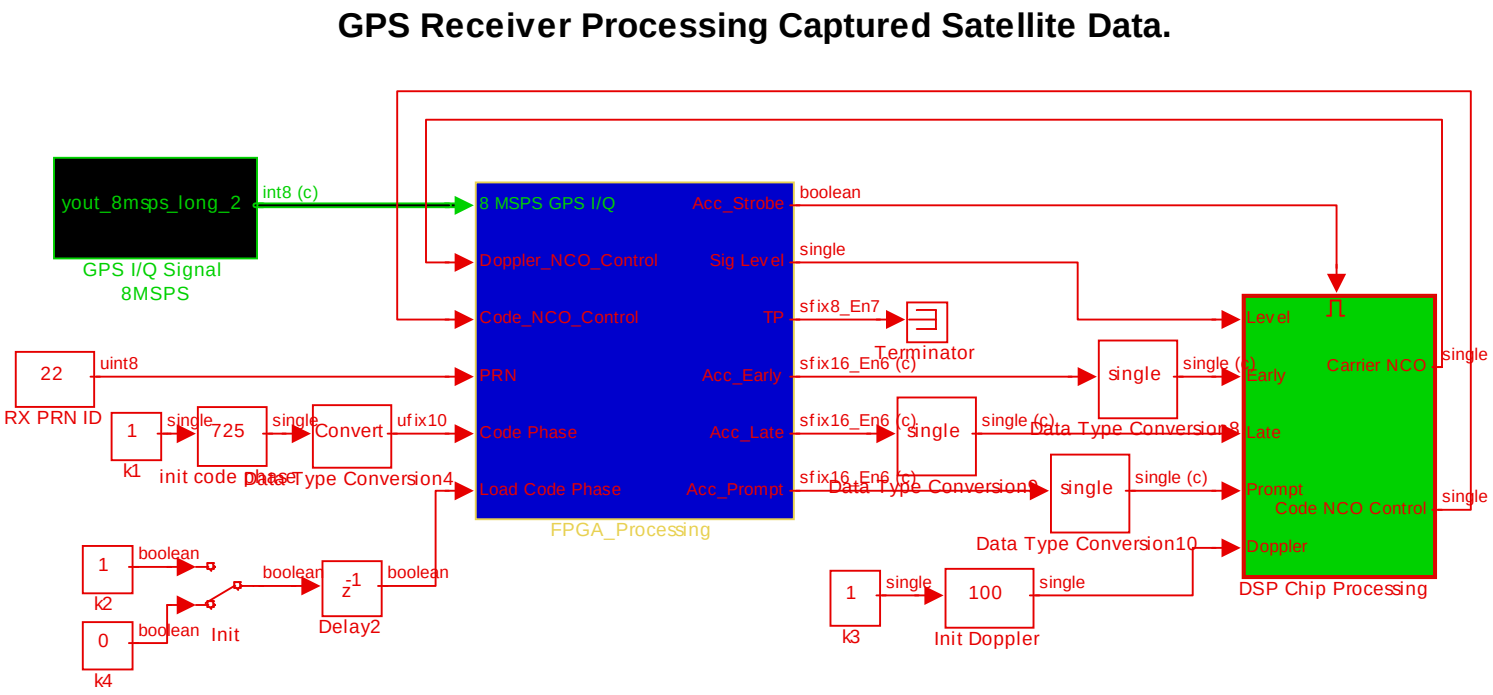
Transition to Fixed Point – First, Take it Apart

- Multiple subsystems with local and global feedback are a challenge
- My preferred method: Transition one subsystem at a time



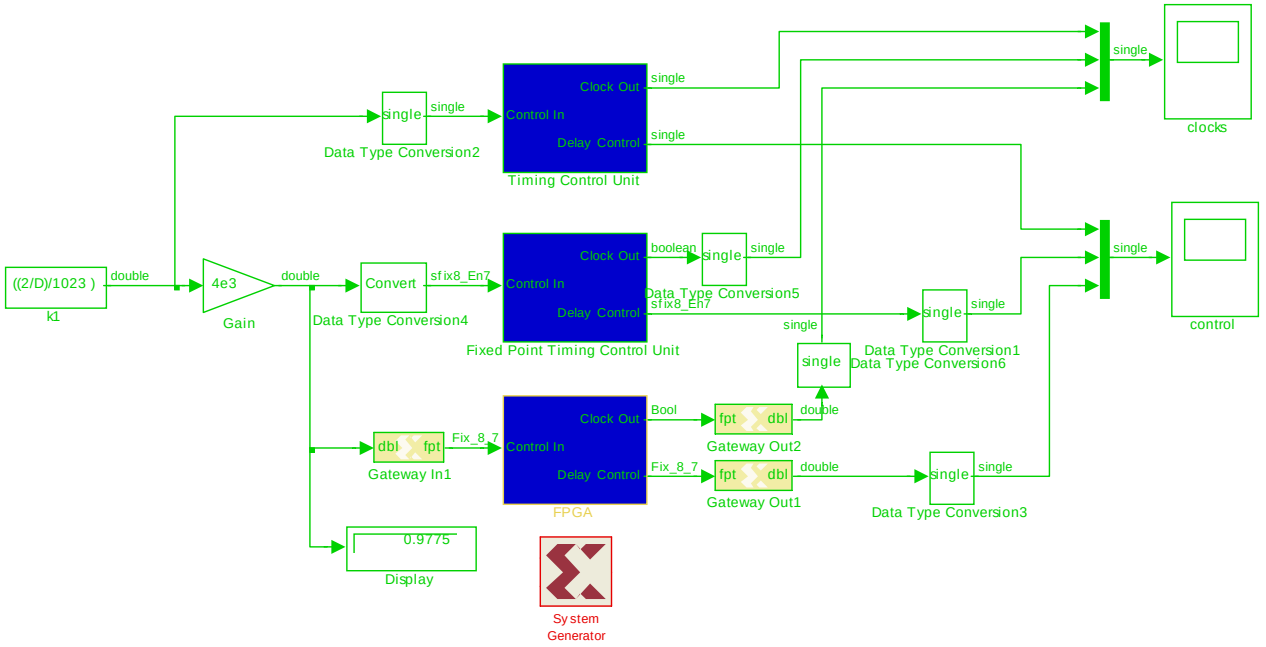
Then, Put it Back Together

- High probability of success
- Re-test with actual GPS signals



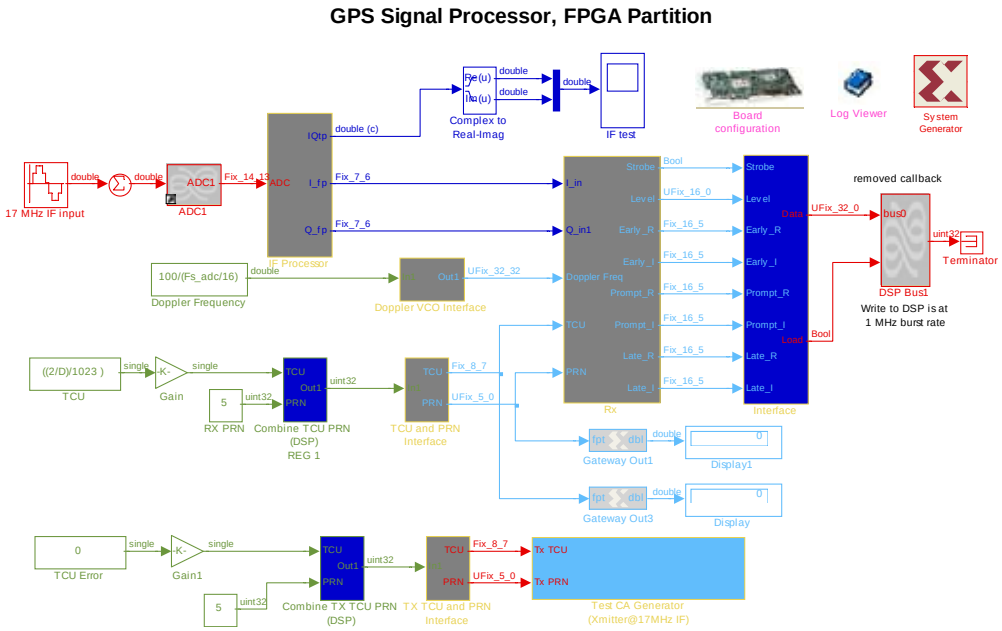
Transition to FPGA

- Plan Ahead
  - Use blocks that will easily map to the target hardware or IP
  - Avoid divides, square roots, logs, trig functions if possible
  - Use techniques amenable to hardware
    - Enabled, not triggered, subsystems
- Same strategy: Transition one subsystem at a time



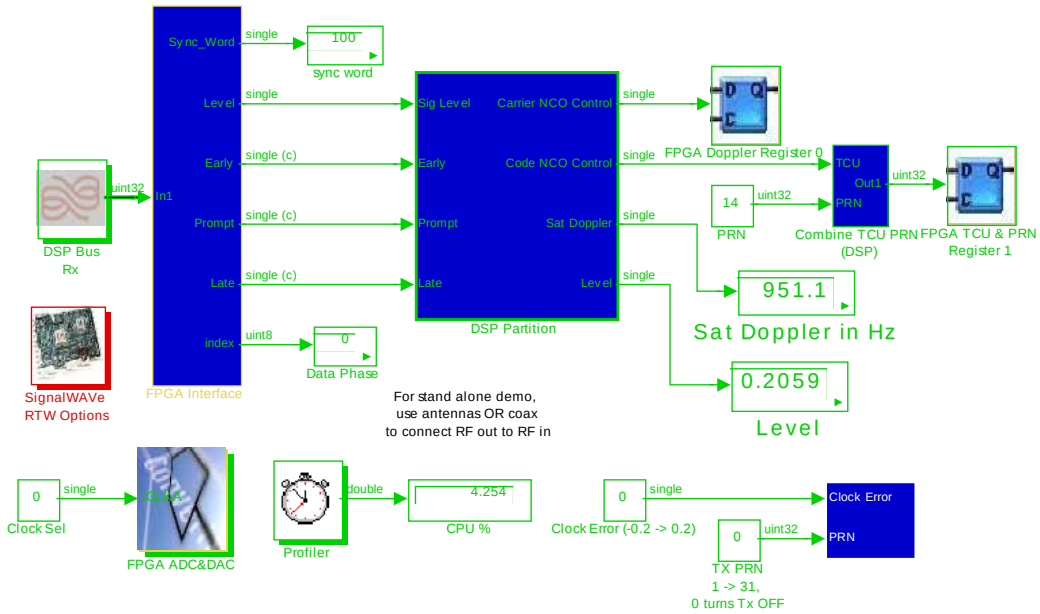
FPGA Partition

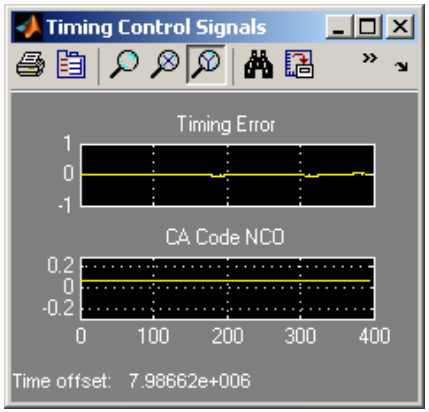
- Reassemble the “transitioned” subsystems
- Add interface to DSP
  - Static gateways to control
    - PRN selection
    - Doppler NCO
    - Code NCO
  - FIFO buffered interface for correlation data and signal level
- FPGA design includes GPS signal generator for stand-alone operation

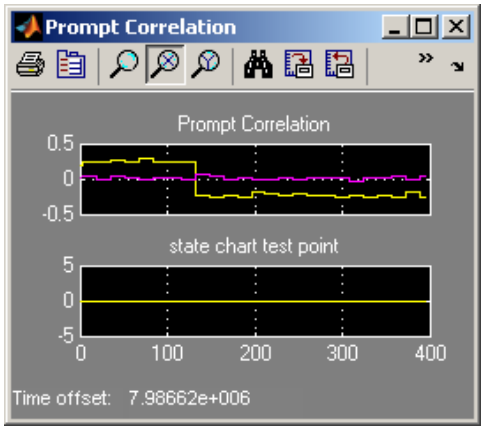


DSP Partition

- Includes
  - C/A code timing loop filter
  - Doppler loop filter
  - Automatic gain control
  - Stateflow® signal acquisition controller
  - Control of built-in GPS test signal
- Two modes of operation
  - Lab needs RF hardware
  - Stand alone uses test signal
- Extensive use of *external mode* for (20) controls and displays
- 1 KHz C/A update rate consumes < 5% of CPU





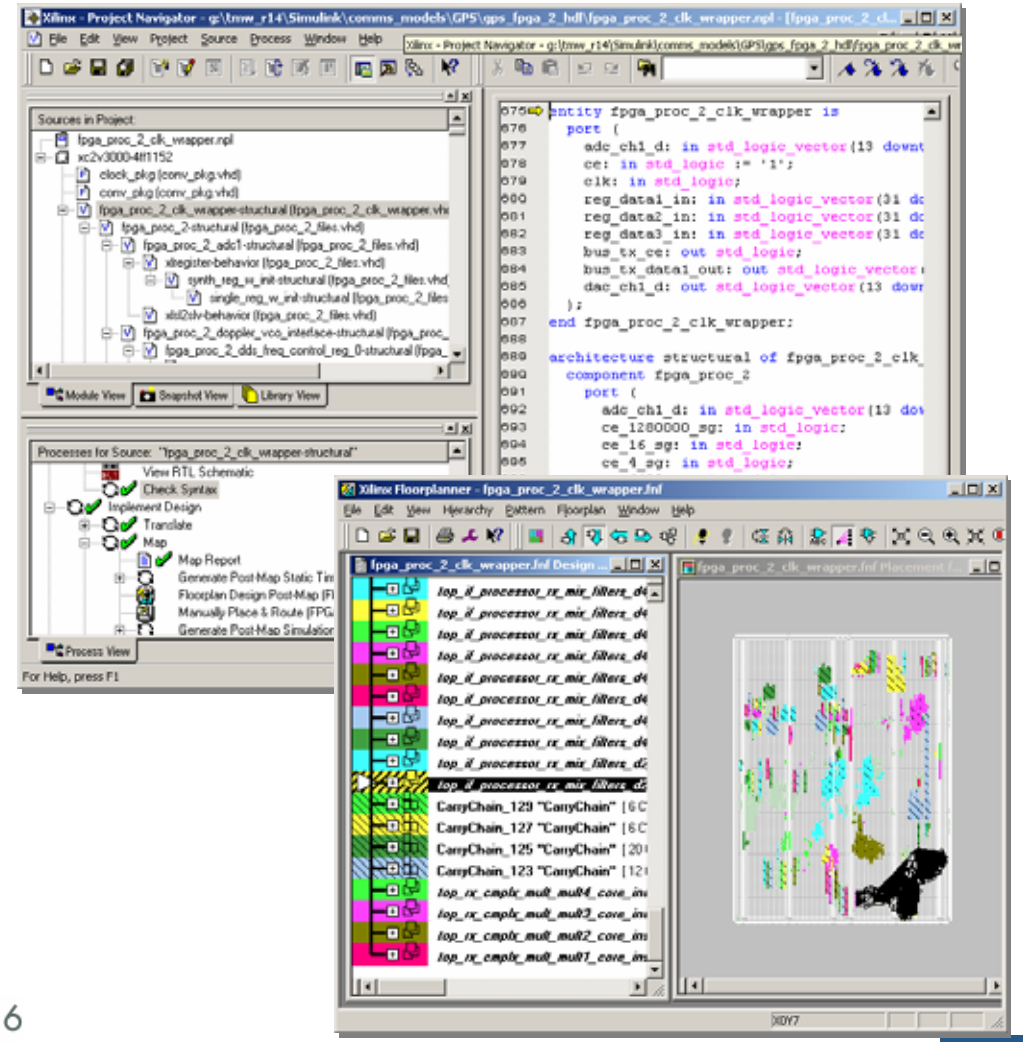


Development Hardware: Lyrtech SignalWAVe

- Texas Instruments C6713 DSP
- Xilinx Virtex-II XC2V3000 FPGA
- Memory
  - 32 MB SDRAM (DSP)
  - 32 MB SDRAM (FPGA)
- Input/Output
  - 65MSPS 14-bit ADC with programmable gain
  - 125MSPS 14-bit DAC
  - NTSC/PAL composite video Decoder
  - NTSC/PAL composite video encoder
  - Audio codec

Resources Consumed

- FPGA Resources
  - 364 *.vhd entries, 794 KB source
  - 22% of xc2v3000 fabric (RX+Tx)
  - 64 MSPS Rx in, 64 MSPS Tx out
- DSP Resources
  - 10 files, 155 KB of source code
  - <5% of CPU horsepower
  - 8 KSPS in, 4 KSPS out (16b)



Sandia implements high-performance radar receiver using MathWorks and Xilinx DSP design tools

The Challenge

- To design a digital radar receiver and implement it on the Virtex-II platform FPGA within strict time and budgetary limits

The Solution

- Use MathWorks and Xilinx DSP design software to create, simulate, test, and synthesize designs into hardware

The Results

- Accelerated design
- High-speed simulation
- Accurate representation of the actual system

The first prototype of the digital IF receiver module

"We are so impressed with the tools and the direction in which The MathWorks and Xilinx are going that we plan to make this our mainstream DSP design flow."

Dale Dubbert
Sandia National Laboratories

Rockwell Collins accelerates development of next-generation military GPS receivers with code generated by Embedded Target for TI C6000™ DSP

The Challenge

To study and validate the new requirements of next-generation advanced military GPS receivers

The Solution

Use MathWorks tools for Model-Based Design to streamline modeling, simulation, automatic code generation, and hardware-in-the-loop testing of GPS functionality

The Results

- Prototype iterations accelerated
- Customer expectations exceeded
- Target debugging minimized

Next-generation GPS device

"MathWorks tools for Model-Based Design provide us with an integrated environment and a push-button solution for automatically generating quality code that significantly cuts development time."

Kevin Neigum, Rockwell Collins

Conclusion

- Challenging problem
  ("needles in a haystack")
- Entire Model-Based Design flow
  - Signal Processing
  - T&M
  - Control Systems
  - Control Logic
- Contemporary comms concepts
  - CDMA
  - Fractional delay timing recovery
- Hardware and software design
  - FPGA using Xilinx System Generator
  - DSP using Real-Time Workshop® and Embedded Target for TI C6000™ DSP
  - Implemented on SignalWave
- SDR example, hardware can be
  - SSB Tx/Rx
  - GPS Tx/Rx

An F-16 drops a JDAM-equipped GBU-31 2,000-pound bomb.

More Information on this Example

- Google Search: “Dick Benson GPS”
- MathWorks recorded webinar (10/06/2005)
  www.mathworks.com/cmspro/req11242.html?eventid=31371
- DSP-FPGA.com article online (01/07/2006)
  www.dsp-fpga.com/articles/benson/