A Software-Defined GPS and Galileo Receiver: Single-Frequency Approach

15-Sept-2005

ION-GNSS-2005
Session C4: GNSS Software Receiver Systems 2

K. Borre; *Aalborg University*
D. Akos; *University of Colorado*

Presentation Overview

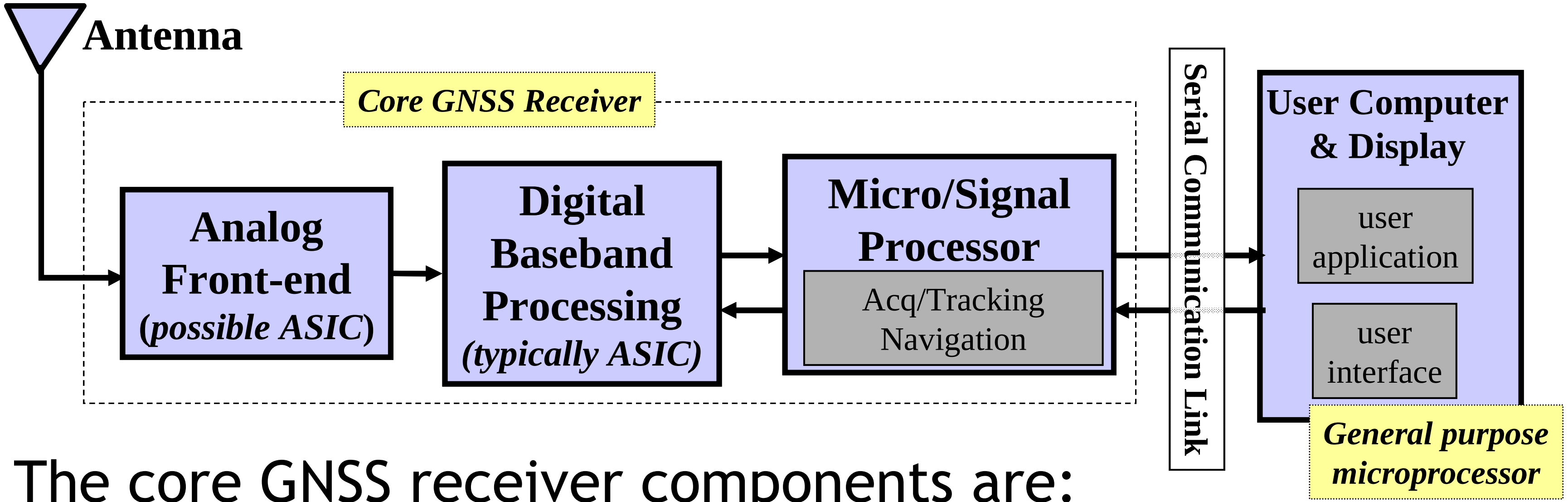
- Motivation
- Software GNSS Receiver Architectures
- Front End Design & Signal Conditioning
  - Sample GPS Data Set
- Signal Acquisition
- Code & Carrier Tracking
- Navigation Data Decoding & Position Solution
- Future Work

Motivation

- Develop a software GNSS receiver to process both GPS and Galileo narrowband L1 components
- Develop accompanying textbook for teaching/educational aspects of GNSS software receivers
- Provide an open source (GPL) fully functional GNSS software receiver basis for further development and refinement by the research community

Traditional GNSS Receiver Architecture

- A generic GNSS receiver block diagram is depicted below:

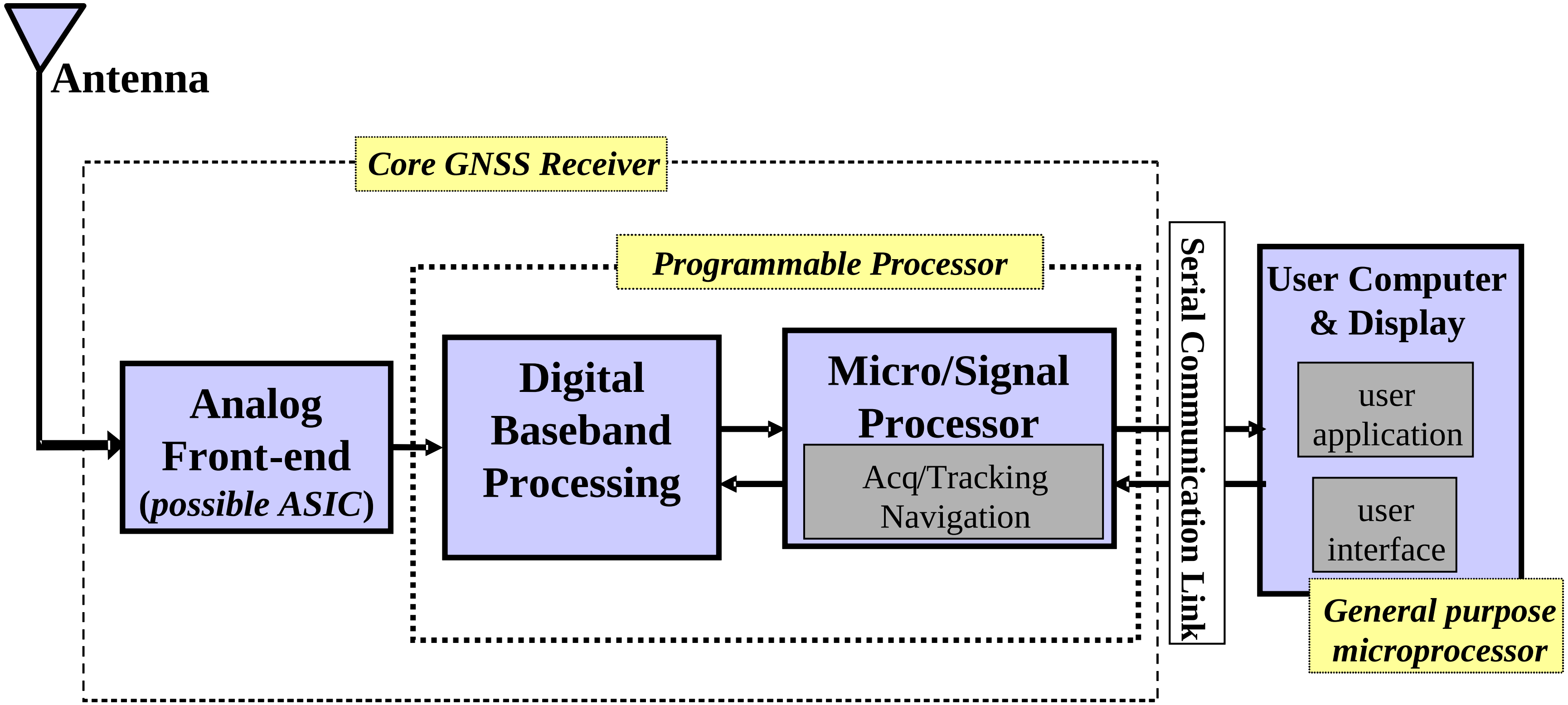


- The core GNSS receiver components are:
  - Antenna
  - Front end for analog signal conditioning, filtering, and digitization
  - High speed correlation ASIC (application specific integrated circuit)
  - Embedded programmable micro/signal processor

- Hardware (ASIC-based) receivers provide minimal flexibility and little support for GNSS additions and/or research

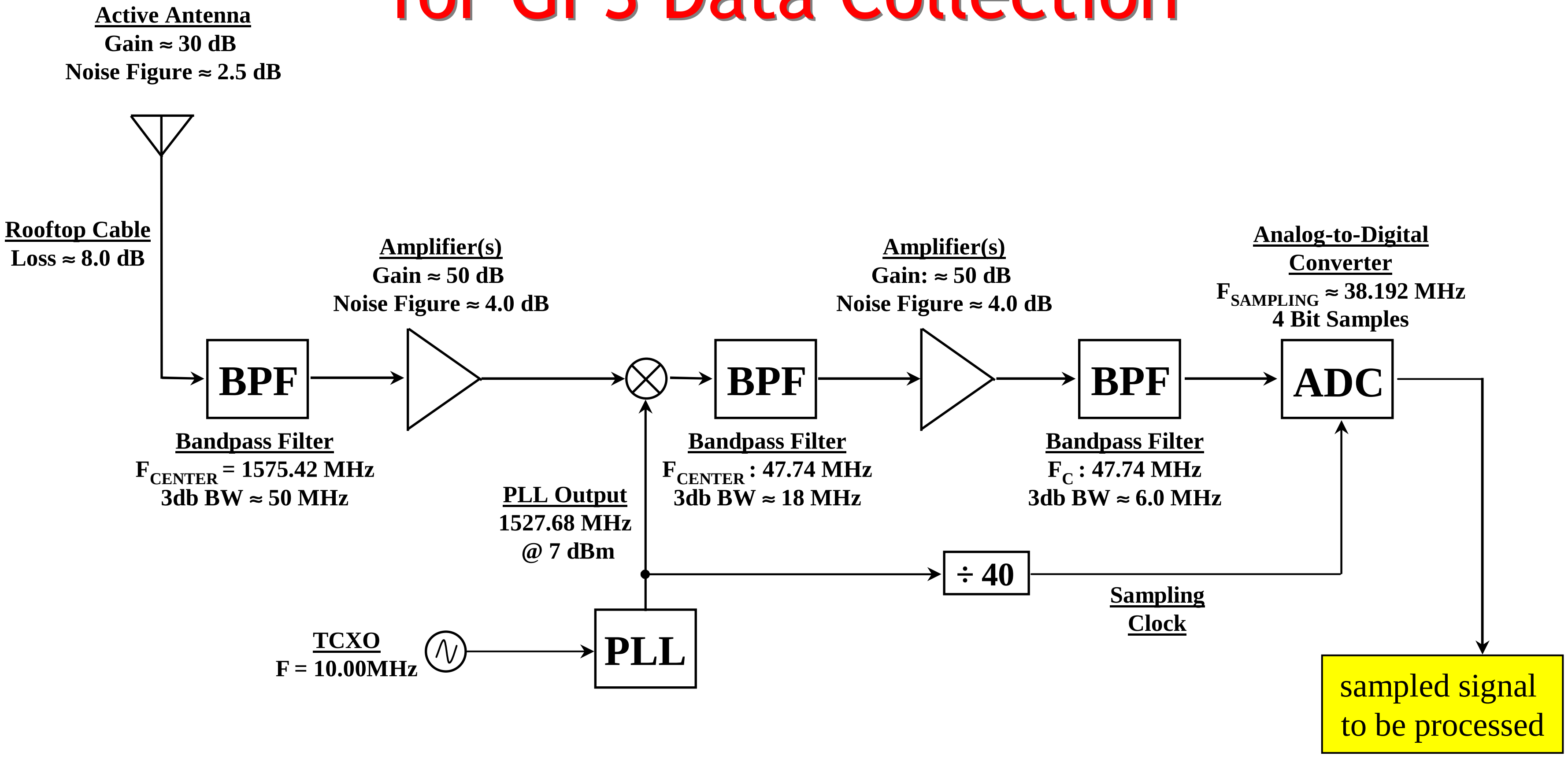
GNSS Software Receiver Architecture

- The modification to a "software" GNSS receiver architecture is subtle



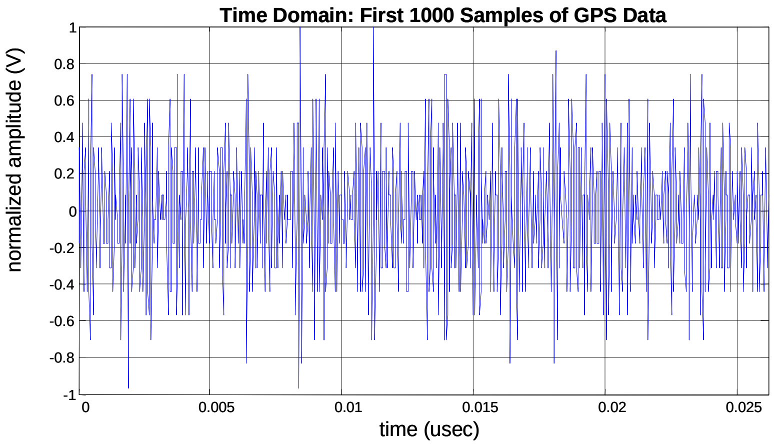
- Now all the signal processing (spread spectrum) after the analog-to-digital converter (ADC) is accomplished within a programmable processor

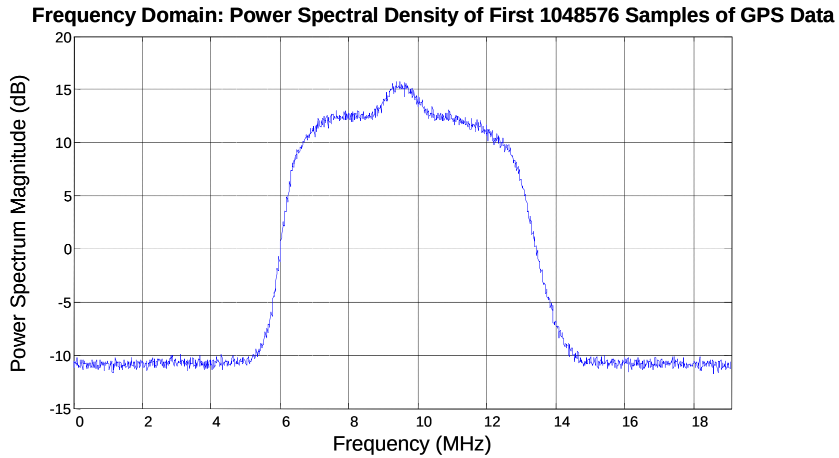
Signal Conditioning or Front End Design for GPS Data Collection



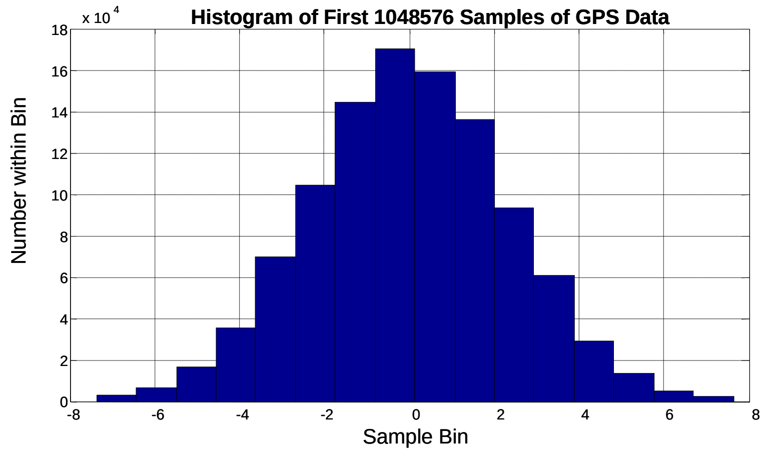
- Above front end design provided a raw digitized sampled signal for algorithm development & processing
- Data set is included with the software algorithms

Collected Data Set



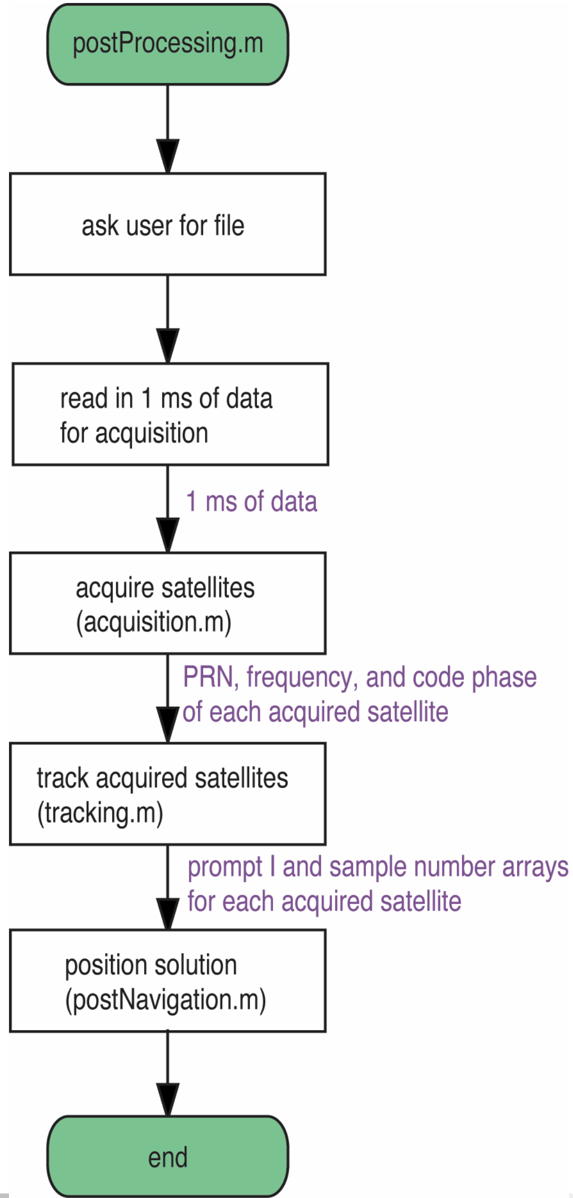


- Collected data set is multiple minutes of data
- Algorithms have been tested with other front ends (sampling and intermediate frequencies
- Software GNSS RX architecture utilizes traditional processing of the data
  - Acquisition, Code & Carrier Tracking, Navigation Data Decoding & Position Solution



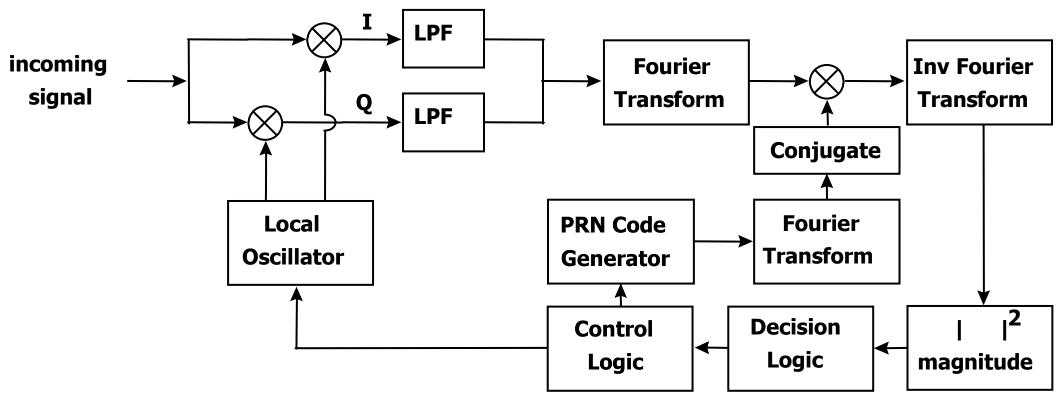
GNSS Software RX Flow Diagram

Start with over view of complete software GNSS RX architecture



GNSS Signal Acquisition – Parallel Code Phase Search

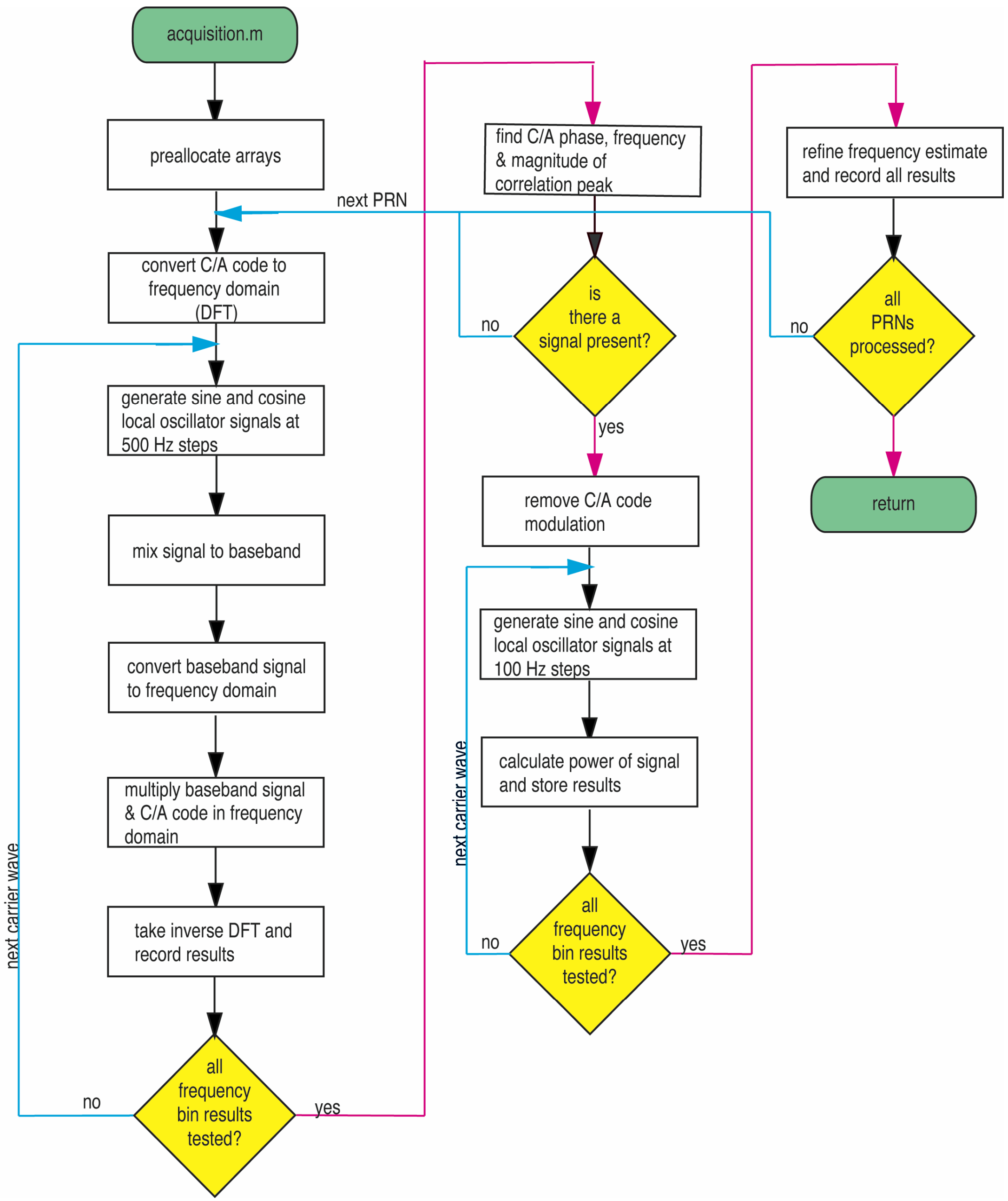
Frequency-domain circular convolution technique



- Algorithm tests all possible code phases via an FFT/IFFT computation
  - FFT/IFFT computation time is the key to the algorithm
- Provides an exhaustive testing of all possible code phases
- Potential for very rapid acquisition times

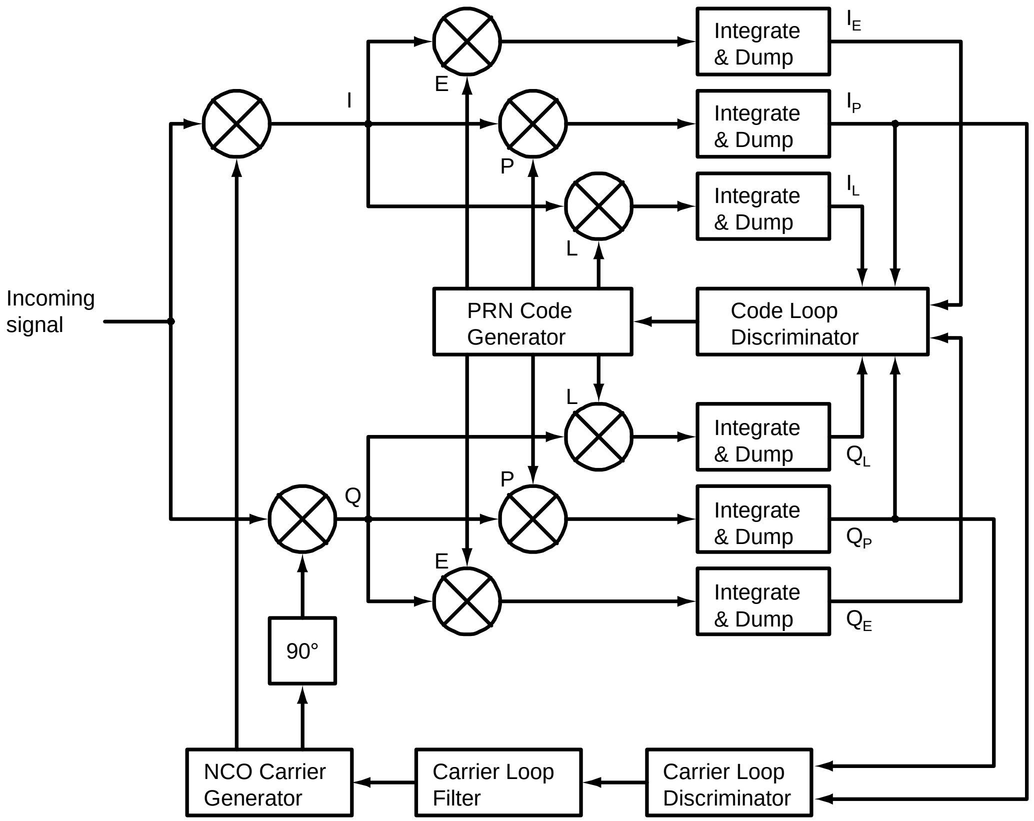
Flow Diagram of Software GPS RX Acquisition

- Perform acquisition on sample collected data set
- Need to know the sampling frequency and resulting intermediate frequency (IF) to enable processing
- Result should return visible satellites, their code phase and carrier frequency estimate

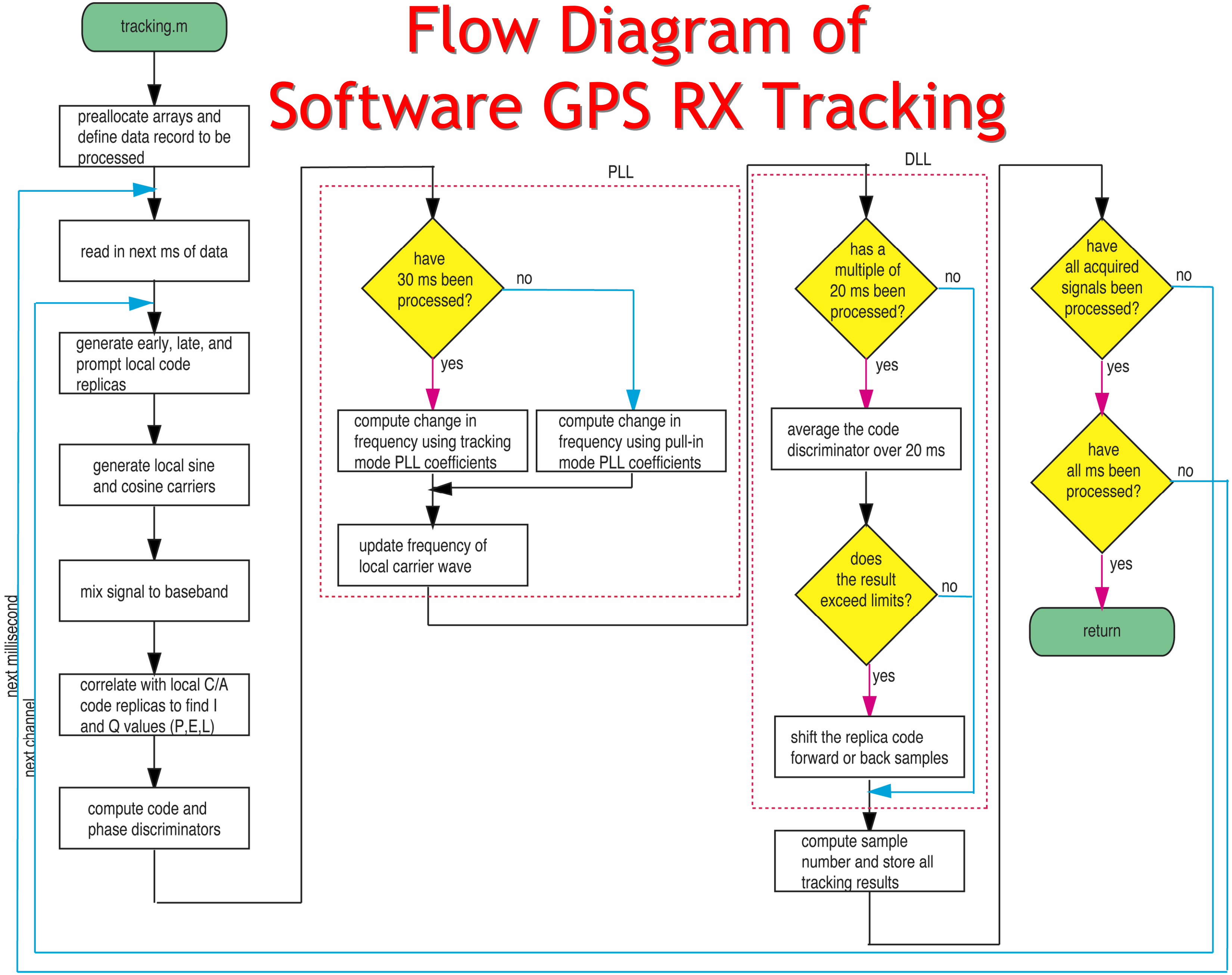


Complete Tracking Block

- Combined code and carrier tracking loops



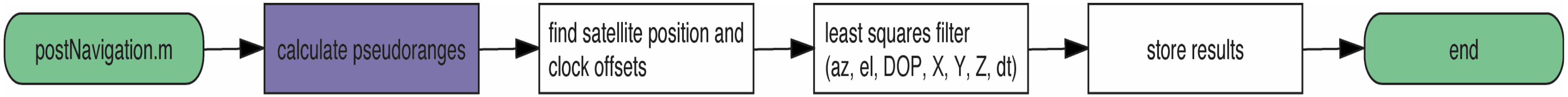
Flow Diagram of Software GPS RX Tracking



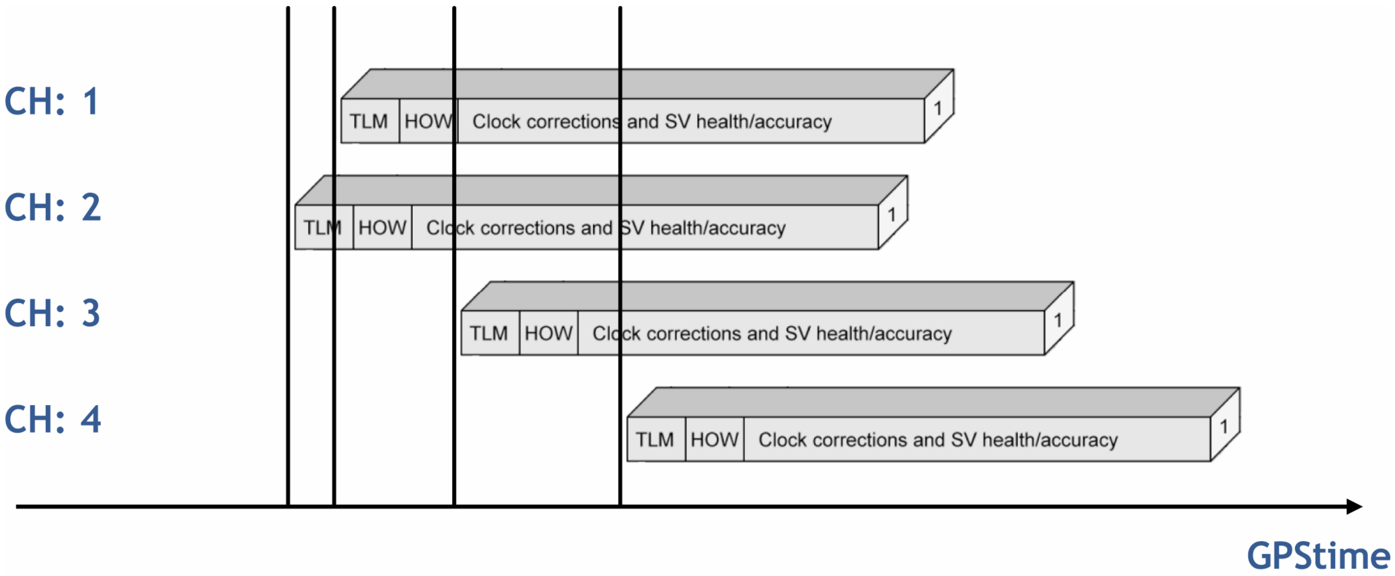
Navigation Data Decoding

- The final signal processing function of the receiver is to decode the 50 Hz navigation data stream
- The bits are clearly visible in the inphase channel of the Costas loop
- Processing proceeds as follows:
  - Bit synch – determine the start/stop of each bit
  - Frame synch – determine the start/stop of the navigation data frames
  - Data decode – extract the necessary parameters from the transmitted ‘1’s and ‘0’s in the first three subframes (required for position solution)
- The ICD-200 and GPS signal specification are outstanding references and describe in detail the structure of the navigation data message

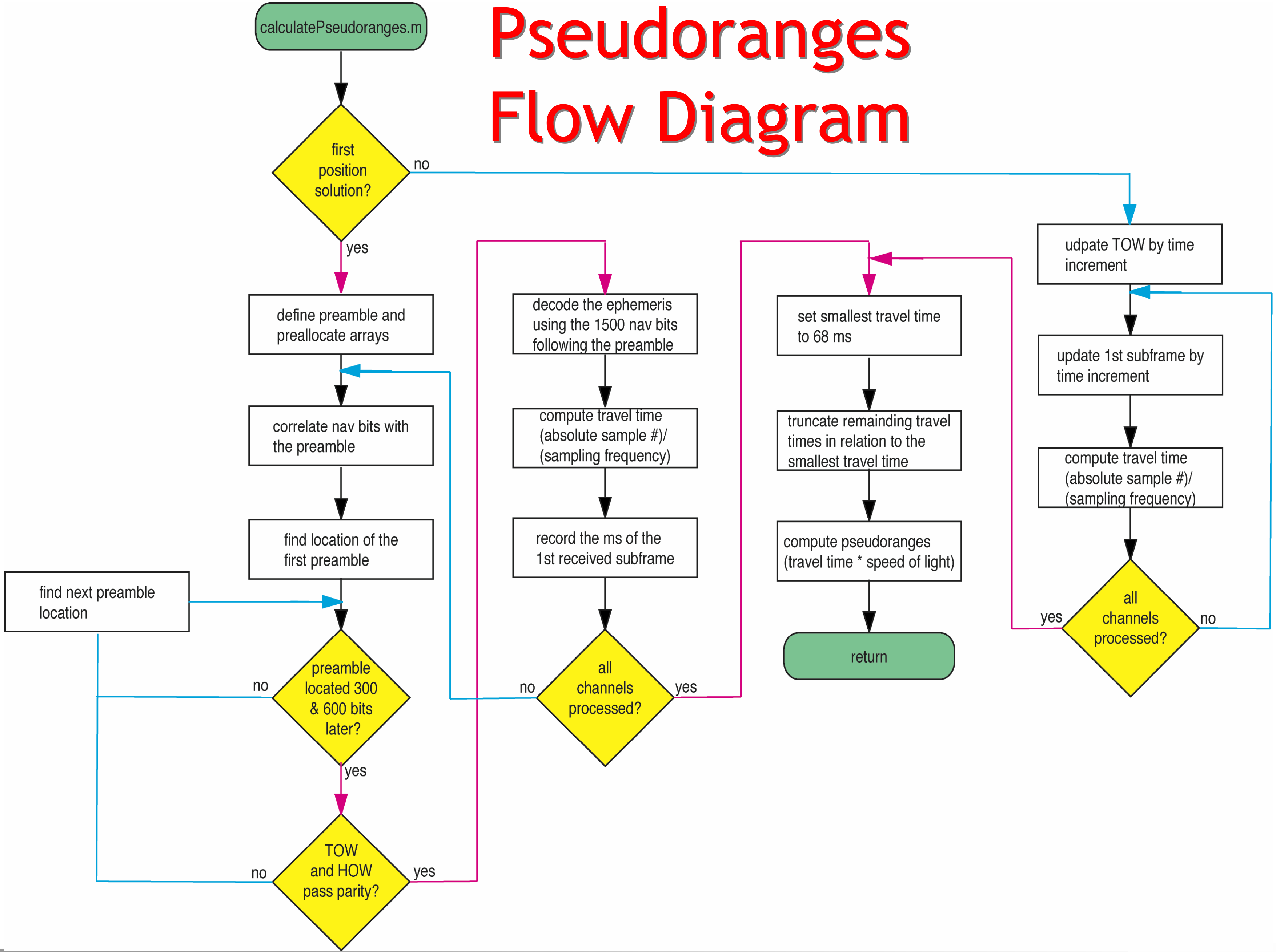
Calculating Pseudoranges



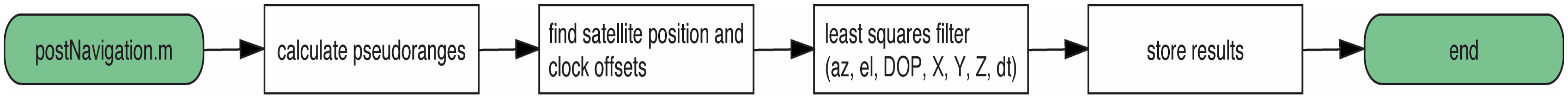
- Timestamp the start of each subframe



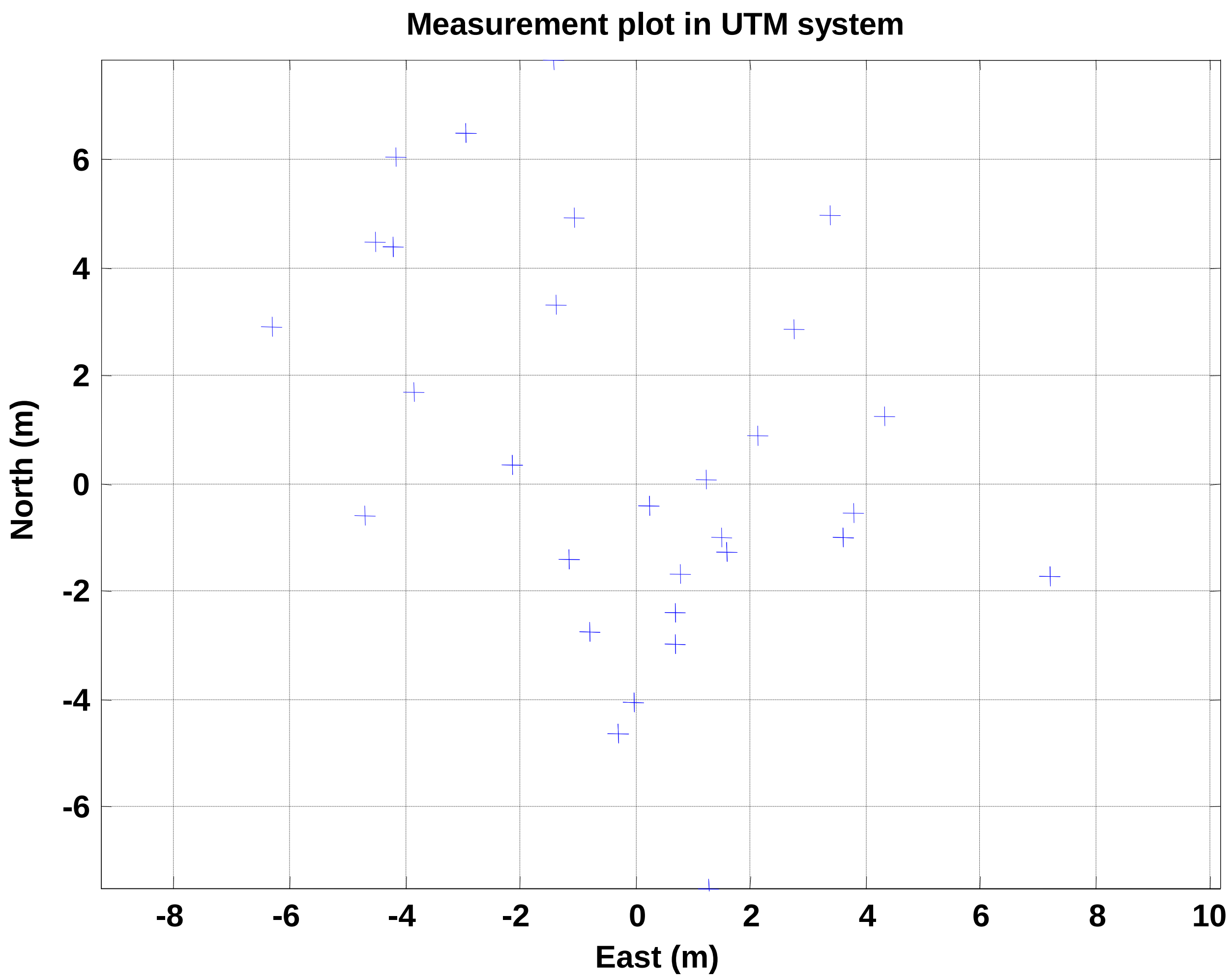
Pseudoranges Flow Diagram



Position Solution Flow Diagram



Receiver Position Computation



- Position solutions generated at 1 Hz rate for 38.192 MHz data set
- Shown are the results for the first 30 second block of data

Receiver/Code Comments

- Post-processing MATLAB version
  - Focus is on algorithm research and development
  - Provide non-real time processing yet not excessively slow
    - Computation speed approximately 6-12 times real-time (sampling frequency dependent)
  - ~500 lines of code

- Goal is to augment the knowledge concerning signals and algorithms

Summary & Conclusions

- Book will be available early 2006
  - Should provide basis for software GNSS receiver courses
- Current receiver developments
  - Support for Galileo signals
  - Support for EGNOS signals
- Will make available a reference textbook & complete GPS/Galileo GPL Matlab framework to be used for algorithm development and testing