Fox Sniffer
A low Cost GPS System for Small Satellites
Tom Clark, K3IO
AMSAT Space Symposium
Orlando Oct. 27, 2012
In April, as we were preparing a proposal for NASA support, we needed a high accuracy GPS receiver, I came up with this idea: We could split the GPS into two SDR pieces and transfer “raw bits” via telemetry to the ground. This would allow for a much more accurate position determination than possible in an onboard \( \mu \)P and require much less power.
Then I found a single $3 chip which (with a few “glue” parts) implemented my concept:
Does it really work if we throw away all the amplitude information and just use the Sign Bit?

The answer is yes!

- For a “weak” signal, the S/N is degraded by a factor of $\pi/2 = 1.57 \approx 2\text{dB}$.
  - This is known as the van Vleck correction
  - The use of one-bit sampled data has been very common in Radio Astronomy

Why put GPS inside a satellite?

- Eliminate the period of chaos just after launch while the ID of the satellites is sorted out by NORAD
- Hopefully the Keps produced will be more accurate than NORAD’s TLEs - and minute changes in drag caused by varying Space Weather can be tracked.
- Formation flying and attitude determination
- **Because it CAN be done**
Basic concept of GPS

A GPS receiver calculates its position by precisely timing the signals sent by GPS satellites high above the Earth. Each satellite continually transmits messages that include:

- the time the message was transmitted
- satellite position at time of message transmission

The receiver uses the messages it receives to determine the transit time of each message and computes the distance to each satellite using the speed of light. Each of these distances and satellites’ locations define a sphere. The receiver is on the surface of each of these spheres when the distances and the satellites’ locations are correct. These distances and satellites’ locations are used to compute the location of the receiver using the navigation equations. This location is then displayed, perhaps with a moving map display or latitude and longitude; elevation information may be included. Many GPS units show derived information such as direction and speed, calculated from position changes.
GPS L1 Signal Structure - at the GPS S/C end:

10.23 MHz Atomic Clock

X140 Multiplier

BPSK Phase Modulator

PA

10.23 Mb/s P/Y Code

1.023 Mb/s CA Code

50 bits/s Data

Data Computer

Ignore P/Y

Decode these two bit streams
The GPS & Galileo signal structure at L1 = 1575.42 MHz

The GPS signal is a Binary Phase-Shift Keying (BPSK) modulated spread spectrum signal with a chip rate of 1.023 MSPS. The Galileo signal is a Binary Offset Carrier (BOC) modulated spread-spectrum signal with subcarrier at 1 MHz offset and a chip-rate of 1.023 MSPS.
With the SE4150, the Master Oscillator = 16,368 kHz, and the sample clock is at 4092 kHz = 16368/4

Therefore 1 second of data = 4092 kb/sec which means that 4092 kb is a logical data “chunk”.

For a TLM rate in the range [10-100]kb/s, it will take [400-40] seconds to downlink one “chunk”.

All this design assumes that Fox will have such data rates possible using the TX in the SDX.
Some ideas/notes concerning the downlink telemetry:

Start with 4Mbit/s data from SE4150 data/clock port:
0100101011001010010101010001010101010001010101010001010101010011010101010...

Add a few byte sync vector that includes the time on the s/c clock to mark the start of a data “word”:
abchmsxyz010010101101010101000101010001001010100010101010100010101010101010...

Stack these up into a series of long data words (each ~1000-10000 bytes long:
abchmsxyz01010101010010010100010101010101010101010101010101010101010101010101...
abchmsxyz00010100101101011010100010110101010101010101010101010101010101010101...
abchmsxyz110100110111101101110101101000101010101010101010101010101010101010101...

And send a one second = 4,092,000 bit “chunk” on a 10-100 kbit/sec telemetry channel, taking [400-40] sec

The goal is to preserve the number of bits in each “chunk” (i.e. to preserve the sampling time of each bit as read on the local s/c clock associated with the 16.348 MHz GPS receiver)
The GPS satellites orbit at 20,200 km altitude twice/day. Therefore the speed of a GPS satellite orbit is 

\[ S_{GPS} = 2\pi \left( 20200 + 6378 \right) \text{km}/43200 \text{sec} \text{ or } S_{GPS} = 6.80 \text{ km/sec}. \]

Similarly, a LEO satellite in orbit at 400 km and a period \( \sim 100 \text{ minutes} \) will have an orbital speed of 

\[ S_{LEO} = 2\pi \left( 400 + 6378 \right) \text{km}/6000 \text{sec} \text{ or } S_{LEO} = 7.1 \text{ km/sec}. \]

Therefore, depending on the geometry between a GPS and a LEO satellite, we can see a speed difference in the range \( (\pm 6.8) - (\pm 7.1) \) or a total range \( \sim \pm 14 \text{ km/sec} \). Typically \( \sim 14 \) GPS satellites can be seen from a given LEO.

By contrast, the Earth’s E/W rotation is at a speed \( = (40000)/86400 = 463 \text{ m/sec} \) at the equator. Fast jet aircraft fly at \( < 400 \text{ m/s} \). Therefore the GPS speed seen on the Earth is less than \( \pm 5 \text{ km/sec} \) because of geometry. This leads to the conclusion that the calculations associated with a GPS in orbit are \( \sim 5 \) times more demanding than for terrestrial navigation and the results are needed about 10 times as often.
On the ground, processing will be done in software something like the, probably written in MATLAB:

RAW DATA FROM S/C:
1s = 4 Mb

FT

Converse with the C/A Spreading Code Of each satellite

F

2-D Data Spectrum

Loop Back

Τ Τ’ Β Β’

Observables For Each GPS

2-D Results Spectrum

KEPS

The processing of one second of s/c data will probably take < 1 second on the ground given a high-end LINUX workstation.
Paul, PE1NUT provided me this sample of a similar processing approach on data from a sampling SDR.
The penultimate cubesat will fly GPS.

Will it be an AMSAT project?

73 de Tom, K3IO & the Fox!