A Software-Defined Radio Transmitter for Variable-Coded Modulation on a CubeSat

By

John Mullet, B.A.

A Project Submitted in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Electrical Engineering

University of Alaska Fairbanks

May 2020

APPROVED:

Denise L. Thorsen, Committee Chair
Dejan Raskovic, Committee Member
Katrina Bossert, Committee Member
Richard Wies, Chair

Department of Electrical and Computer Engineering
Abstract:

The large volume of satellites sharing the same spectrum and the complexities of communications in Low-Earth Orbit (LEO) pose challenges to the downlink of large volumes of data on a platform that is bandwidth, power, and time limited. LEO satellites operate in a highly variable communications environment due to variations in inter-satellite or satellite-to-ground geometries, weather, and interference. Therefore, there is motivation for implementing satellite communication techniques that manage these issues to increase the data throughput. One such technique is variable-coded modulation which shows improvement by taking advantage of the dynamic nature of a satellite link. As part of the Air Force Research Laboratory University Nanosatellite Program, and in collaboration with NASA, this project focuses on the development of an S-band software defined radio for CubeSats that utilizes variable-coded modulation defined by the Digital Video Broadcasting-Satellite-Second Generation standard. This project defense discusses the initial development and testing using GNU Radio, and the challenges for full implementation, as well as the current status of the transmitter, and future work.
A Software Defined Radio Transmitter for Variable-Coded Modulation on a CubeSat

John Mullet
Outline

• CubeSat Communications Platform (CCP)
  • Link analysis
  • Variable-coded modulation

• Software Defined Radio (SDR)
  • Signal path
  • DVB-S2

• Initial SDR Development Approach

• Testing
  • Issues with initial approach

• Current Status and Next Steps
CubeSat Communications Platform (CCP)

- Air Force Research Labs (AFRL) – University Nano-Satellite Program (UNP)
- CCP Mission Objectives
  1. Characterize the on-orbit beam pointing error, power efficiency, and thermal load of the high-gain antenna payload. (Retrodirective Antenna)
  2. Characterize the on-orbit performance of the variable coded modulation payload with respect to the information throughput capabilities. (Software Defined Radio)
- Collaboration: NASA Near Earth Network (NEN)
- SpaceOps 2020 Conference Paper
CCP Link Analysis

General RF Parameters

- **Carrier Frequency**
  - S-Band
  - 2.2 GHz

- **Bandwidth**
  - 5 MHz

Ground Station

- **Antenna Size**
  - NEN: 11 m
  - UAF: 1.2 m

- **G/T**
  - NEN: 19 dB/K
  - UAF: 6 dB/K

- **Symbol Rate**
  - NEN: 3.7 M Sym/s
  - UAF: 3.7 M Sym/s

Spacecraft

- **Transmit Power**
  - 0 dBW

- **Antenna Gain**
  - 6 dB

- **EIRP**
  - 5.5 dBW
UAF Student Ground Station

Link Margin Versus Angle from Zenith
NASA Wallops Ground Station

Link Margin Versus Angle from Zenith

-90 -80 -70 -60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90

Angle from Normal (°)

Link Margin (dB)

BPSK
QPSK
8PSK
16APSK
32APSK
64APSK

6dB Margin
Variable-Coded Modulation (VCM)

- Variable link conditions
  - Distance from satellite to ground station
  - Weather

- Constant coding and Modulation (CCM)
  - Plans for “worst-case”

- Variable coded modulation (VCM)
  - Takes advantage of better conditions
CCP – Software Defined Radio (SDR)

• Three protocols
  • DVB-S2, CCSDS, VITAMIN

• DVB-S2:
  • QPSK: 1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10
  • 8PSK: 3/5, 2/3, 3/4, 5/6, 8/9, 9/10
  • 16APSK: 2/3, 3/4, 4/5, 5/6, 8/9, 9/10

• CCSDS/VITAMIN:
  • BPSK: 1/6, 1/4
  • QPSK:
    • Turbo: 1/6, 1/4, 1/3, 1/2
    • AR4JA LDPC: 1/2, 2/3, 4/5
    • C2 LDPC: 7/8
  • 8PSK:
    • AR4JA LDPC: 1/2, 4/5
    • C2 LDPC: 7/8
  • 16APSK:
    • AR4JA LDPC: 1/2, 2/3, 4/5, 7/8
SDR Signal Path
DVB-S2 Functional Blocks

- Baseband header insertion
- Baseband scrambler
- Outer BCH Coding
- Inner LDPC Coding
- Bit Interleaver
- Constellation Mapping
- Physical Layer Header Insertion
- Physical Layer Randomizer
- Quadrature Modulation and Filtering
DVB-S2 Outer Encoding—BCH

• Bose-Chaudhuri-Hocquenghem (BCH) Code

• Simplified example encoding:
  • Message: 1011
  • Generator polynomial: \( g(X) = 1 + X^2 \)
  • Procedure
    • Message polynomial: \( m(X) = 1 + X^2 + X^3 \)
    • Multiply message by \( X^2 \): \( X^2 m(X) = X^2 + X^4 + X^5 \)
    • Divide by \( g(X) \), the remainder is \( p(X) \)
    • The codeword polynomial is: \( p(X) + X^2 m(X) \)
      \[ 0 + X + X^2 + 0X^3 + X^4 + X^5 \]
    • Coded bits: 011011
DVB-S2 Outer Encoding—BCH

• More complex than the example
• DVB-S2 uses a generator polynomials of degree 128, 160, or 192
• Depending on LDPC code rate, can correct 8, 10, or 12 code bit errors
• Example: For an LDPC code rate of 1/4 → 
  \((n, k) = (16200, 16008)\)

\[
\begin{align*}
g_1(x) &= 1 + x^2 + x^3 + x^5 + x^{16} \\
g_2(x) &= 1 + x^4 + x^5 + x^6 + x^8 + x^{16} \\
g_3(x) &= 1 + x^2 + x^3 + x^4 + x^5 + x^6 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{16} \\
g_4(x) &= 1 + x^2 + x^4 + x^5 + x^6 + x^9 + x^{11} + x^{12} + x^{14} + x^{16} \\
g_5(x) &= 1 + x + x^2 + x^3 + x^5 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{16} \\
g_6(x) &= 1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{13} + x^{14} + x^{15} + x^{16} \\
g_7(x) &= 1 + x^2 + x^5 + x^6 + x^8 + x^9 + x^{10} + x^{11} + x^{12} + x^{13} + x^{15} + x^{16} \\
g_8(x) &= 1 + x + x^2 + x^5 + x^6 + x^8 + x^9 + x^{12} + x^{13} + x^{14} + x^{16} \\
g_9(x) &= 1 + x^5 + x^7 + x^8 + x^{10} + x^{11} + x^{16} \\
g_{10}(x) &= 1 + x + x^2 + x^5 + x^7 + x^8 + x^{10} + x^{12} + x^{13} + x^{14} + x^{16} \\
g_{11}(x) &= 1 + x^2 + x^3 + x^5 + x^9 + x^{11} + x^{12} + x^{13} + x^{16} \\
g_{12}(x) &= 1 + x + x^5 + x^6 + x^7 + x^9 + x^{11} + x^{12} + x^{16}
\end{align*}
\]

BCH polynomials
DVB-S2 Inner Encoding—Low-Density Parity Check (LDPC)

A simple example:

**Variable nodes**

```
      0  0  1  0  0  1
      1  0  0  1  1  0
```

**Check nodes**

```
      1  1  1  1  0  0
      0  1  0  0  1  1
      0  0  1  1  0  1
```

**Parity check matrix**

\[
H = \begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 0
\end{bmatrix}
\]

**Parity check matrix**

\[
G = \begin{bmatrix}
1 & 0 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 1
\end{bmatrix}
\]

**Check matrix**

\[
U = mG = \begin{bmatrix}
1 & 0 & 1 \\
1 & 1 & 1 \\
1 & 1 & 0
\end{bmatrix} \begin{bmatrix}
1 & 0 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
0 & 1 & 1 & 1 & 0 & 1
\end{bmatrix}
\]

\[
H = \begin{bmatrix}
1 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 0
\end{bmatrix} = [I_{n-k} \mid P^T]
\]

\[
G = \begin{bmatrix}
1 & 0 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 1
\end{bmatrix} = [P \mid I_k]
\]
DVB-S2 Inner Encoding—LDPC Encoding

• Normal baseband frame sizes are 64800 bits (value of $n$)

• For 1/4 rate $\rightarrow$

$$ (n, k) = (64800, 16200) $$

1. Accumulate information bit $i_0$ at parity bit addresses in row 1
2. Repeat for $i_m$ where $m = 1, 2, \ldots 359$ for parity address $(x + m \mod 360 \times q) \mod (n - k)$
3. Repeat for the next group of 360 using row 2
4. Repeat until all information bits are exhausted
5. Final parity follows: $p_i = p_i \oplus p_{i-1}$
6. The codeword, $c = (i_0, i_1, \ldots, p_{n-k}, p_{n-k-1})$

<table>
<thead>
<tr>
<th>Code Rate</th>
<th>$q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/4</td>
<td>135</td>
</tr>
<tr>
<td>1/3</td>
<td>120</td>
</tr>
<tr>
<td>2/5</td>
<td>108</td>
</tr>
<tr>
<td>1/2</td>
<td>90</td>
</tr>
<tr>
<td>3/5</td>
<td>72</td>
</tr>
<tr>
<td>2/3</td>
<td>60</td>
</tr>
<tr>
<td>3/4</td>
<td>45</td>
</tr>
<tr>
<td>4/5</td>
<td>36</td>
</tr>
<tr>
<td>5/6</td>
<td>30</td>
</tr>
<tr>
<td>8/9</td>
<td>20</td>
</tr>
<tr>
<td>9/10</td>
<td>18</td>
</tr>
</tbody>
</table>
Initial SDR Development

- Started with DVB-S2
- NASA Near-Earth Network (NEN) partnership
- Initial approach
  - GNU Radio
  - B205 mini-I
  - Beaglebone Black
Testing

• GNU Radio had sample code for DVB-S2 (CCM only)
Testing

• Required data to be a data stream of MPEG-2 video packets
  • Developed script to have any data look like an MPEG-2 packet stream
  • Similar script required to convert it back to original data

• Did not work for NASA
  • Wanted input to be pseudorandom noise
    • PN 15 Code
      • Initialized to all 1s
      • Looped 8 times to align to a byte boundary

Pattern is 32767 bits in length.
Issues with GNU Radio

• NASA’s DVB-S2 receiver does not work with MPEG-2 data streams

• Big Issues
  • A data rate of 15 Mbit/s is not achievable with GNU Radio
  • Changing the GNU Radio code or creating it from scratch would be significant

• A new approach is required
  • Matlab/Simulink HDL Coder and FPGA
Matlab/Simulink HDL Coder—Current Status and Next Steps

• Currently developed Matlab functions:
  • Baseband header insertion
  • Baseband scrambler
  • Outer BCH Coding
  • Inner LDPC Coding
  • Bit Interleaver

• Next steps
  • Verify current code compatibility with HDL coder
  • Physical layer header insertion, physical layer randomizer, modulation and filtering
  • Transfer SDR lead to the next person
Questions