COM-1018 DIRECT SEQUENCE SPREAD-SPECTRUM DEMODULATOR 20 Mchip/s. VHDL SOURCE CODE OVERVIEW

Overview

The COM-1018 ComBlock Module comprises two pieces of software:

- VHDL code to run within the FPGA for all signal processing functions.
- C/Assembly code running within the Atmel AT90S8515 or ATMega8515L microprocessor for non application-specific monitoring and control functions.

The VHDL code interfaces to the monitoring and control functions by exchanging byte-wide registers on the Atmel microcontroller 8-bit data bus. The control and monitoring registers are defined in the specifications [1].

The Atmel microprocessor code is generic (i.e. non application specific), not user-programmable and functionally transparent to the user. It is thus not described here.

The COM-1018 VHDL code runs on the generic COM-8000 hardware platform. The schematics [2] for this platform are available in this CD.

Reference documents

[1] specifications: com1018.pdf

[2] hardware schematics: com_8000schematics.pdf

[3] VHDL source code in directory
com-1018_016\src

[4] Xilinx ISE project files
com-1018_016\com1018_ISE41.npl
com-1018_016\com1018_ISE63.npl

[5] .ucf constraint file
com-1018_016\src\root_demod.ucf

[6] .mcs FPGA bit files
com-1018_016\com1018A_016.mcs
com-1018_016\com1018B_016.mcs
com-1018_016\com1018D_016.mcs
com-1018_016\com1018E_016.mcs

Configuration Management

The current software revision is 16.

Configuration Options

In order to provide configuration flexibility without unduly increasing the hardware complexity, some features require generating different firmware versions. In particular, the channel filter (root raised cosine square root) rolloff can take four distinct values: 20%, 25%, 35% and 40%.

Four versions of the *raised_cos4x* root raised filters are included in the source code .src directory. To change the filter:

(a) change the OPTION constant in the *root_demod.vhd* file so that the resulting bit file can later be correctly identified.
(b) Change the RAISED_COS4x statements in three places within the *demod.vhd* file: declaration and two instantiations.

VHDL development environment

The VHDL software was developed using two development environments:

(a) Xilinx ISE 4.1 with Synopsys FPGA Express 3.6 as synthesis tool.
(b) Xilinx ISE 6.3 with XST as synthesis tool.

Target FPGA

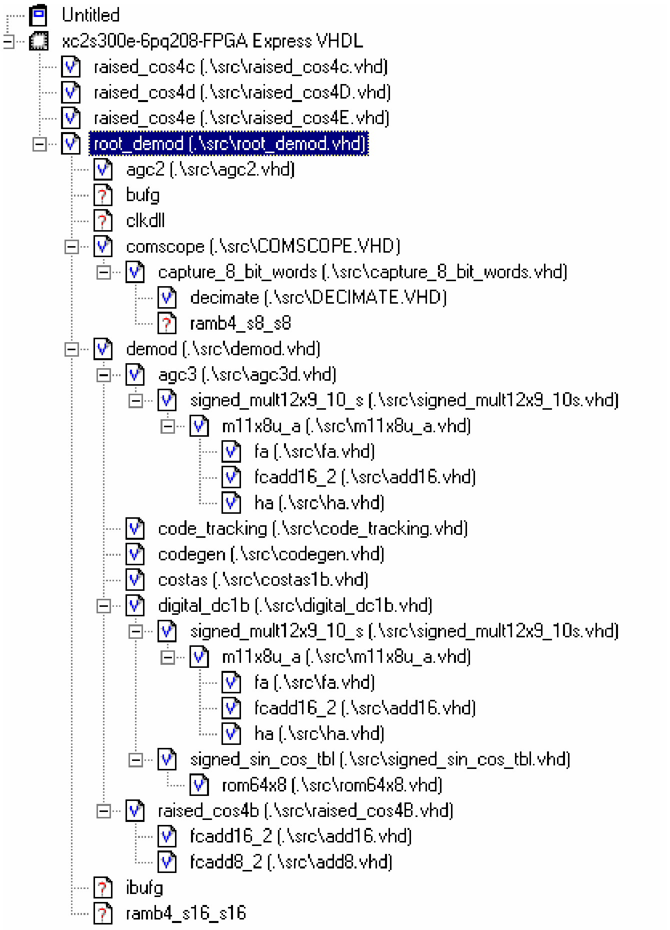
The VHDL code was synthesized for the Xilinx Spartan-IIE XC2S300E-6PQ208 FPGA.

Xilinx-specific code

The VHDL source code was written in generic VHDL with few Xilinx primitives. No Xilinx CORE is used. The Xilinx primitives are:

- BUFG
- IBUFG
- CLKDLL (x2)
- RAMB4_S16_S16
- RAMB4_S8_S8

VHDL software hierarchy



```
Untitled
xc2s300e-6pq208-FPGA Express VHDL
    raised_cos4c (.\src\raised_cos4c.vhd)
    raised_cos4d (.\src\raised_cos4D.vhd)
    raised_cos4e (.\src\raised_cos4E.vhd)
    root_demod (.\src\root_demod.vhd)
        agc2 (.\src\agc2.vhd)
        bufg
        clkdll
        comscope (.\src\COMSCOPE.VHD)
            capture_8_bit_words (.\src\capture_8_bit_words.vhd)
                decimate (.\src\DECIMATE.VHD)
                ramb4_s8_s8
        demod (.\src\demod.vhd)
            agc3 (.\src\agc3d.vhd)
                signed_mult12x9_10_s (.\src\signed_mult12x9_10s.vhd)
                    m11x8u_a (.\src\m11x8u_a.vhd)
                        fa (.\src\fa.vhd)
                        fcadd16_2 (.\src\add16.vhd)
                        ha (.\src\ha.vhd)
            code_tracking (.\src\code_tracking.vhd)
            codegen (.\src\codegen.vhd)
            costas (.\src\costas1b.vhd)
            digital_dc1b (.\src\digital_dc1b.vhd)
                signed_mult12x9_10_s (.\src\signed_mult12x9_10s.vhd)
                    m11x8u_a (.\src\m11x8u_a.vhd)
                        fa (.\src\fa.vhd)
                        fcadd16_2 (.\src\add16.vhd)
                        ha (.\src\ha.vhd)
                signed_sin_cos_tbl (.\src\signed_sin_cos_tbl.vhd)
                    rom64x8 (.\src\rom64x8.vhd)
            raised_cos4b (.\src\raised_cos4B.vhd)
                fcadd16_2 (.\src\add16.vhd)
                fcadd8_2 (.\src\add8.vhd)
        ibufg
        ramb4_s16_s16
```

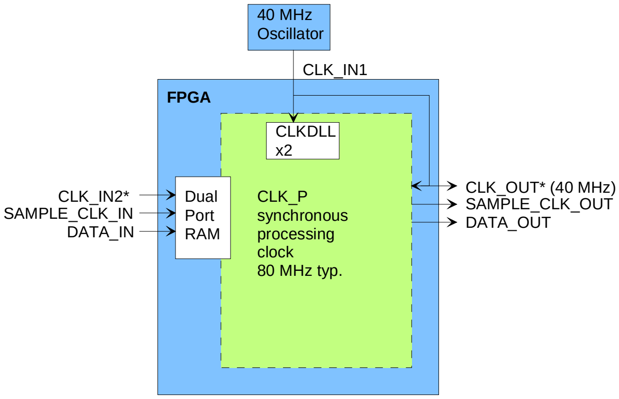
The code is stored with one, and only one, entity per file as shown above.

The root program (highlighted) is *root_demod.vhd*.

Clock / Timing

The software uses two different clocks:

- external clock CLK_IN2 which serves synchronous clock for the input data stream.
- CLK_IN1 is generated by a 40 MHz oscillator on the COM-1018 module. It is used as reference for the output clock and for the double-frequency processing clock. The code is written to meet the timing requirements on the target FPGA at a speed of at least 80 MHz.



(*) denotes edge-trigger signal

Block Diagram

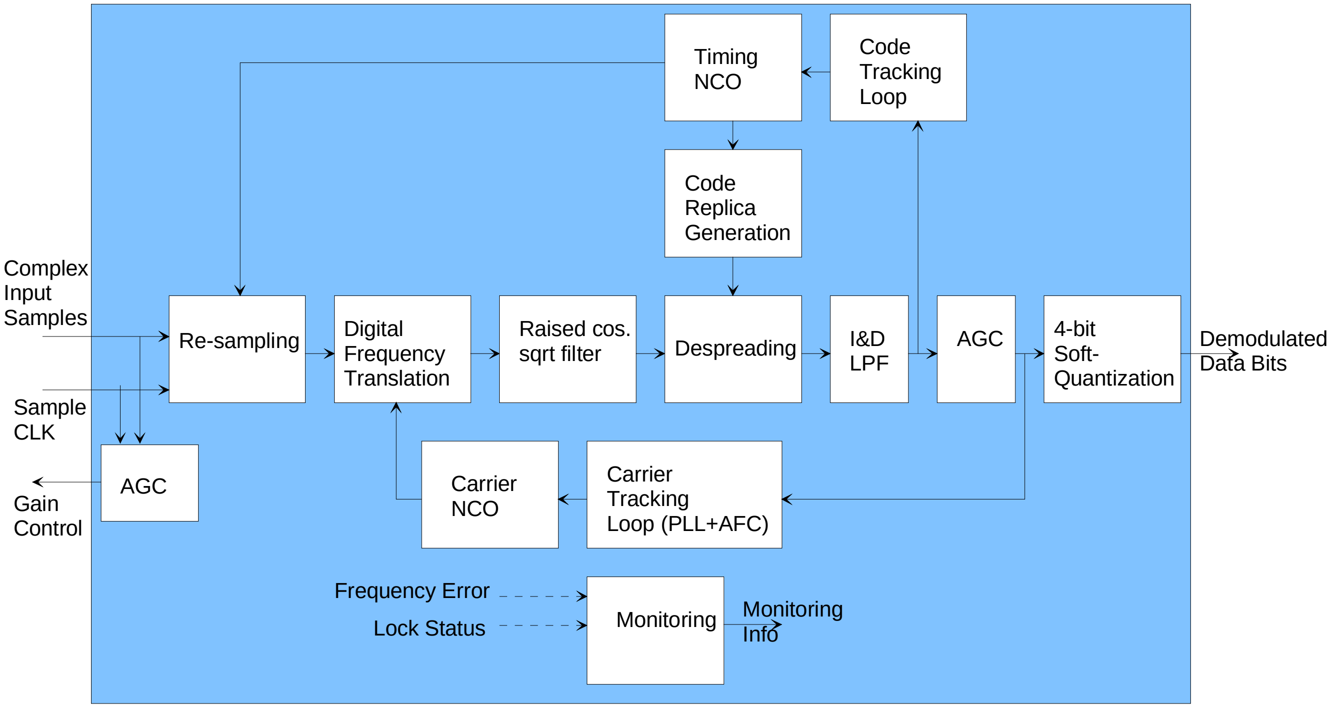
The hierarchical nature of the VHDL code reflects the block diagram below:

- *root_demod* is the root program which includes the demodulator *demod*, the analog AGC *agc2* to prevent saturation at the external A/D converters and ancillary functions such as monitoring and control functions (interface with microprocessor) and *comscope* to capture and display internal signals.
- the main Direct-Sequence Spread-Spectrum functions are encapsulated within *demod*.
- entity *codegen* implements all spreading codes.
- the frequency translation is implemented within *digital_dc1b*. The frequency translation is realized in the form of a complex vector rotation, using sine/cosine lookup tables (*signed_sin_cos_tbl*) and pipeline multipliers (*signed_mult12x9_10_s*) made of half adders *ha* and full adders *fa*.
- spectrum shaping is made by means of two root raised cosine filters *raised_cos4x*, one for each complex axis.
- a conventional early-late gate delay lock loop is implemented in *code_tracking*. This entity includes only the error computation

and the loop gain settings. The NCO and integrate & dump functions are part of the *demod* entity.

- A digital AGC *agc3* is used to normalize the despread signal and to condition the resulting signal in a variety of formats: 4-bit unsigned (soft-quantization), 8-bit signed (for carrier tracking loop processing).
- Most carrier acquisition, carrier tracking and AFC functions are implemented within the *costas* entity. The actual carrier NCO is part of the *demod* entity.

Block Diagram



VHDL Simulation

Representative simulation screens for salient internal signals are captured and discussed below.

Signal Processing

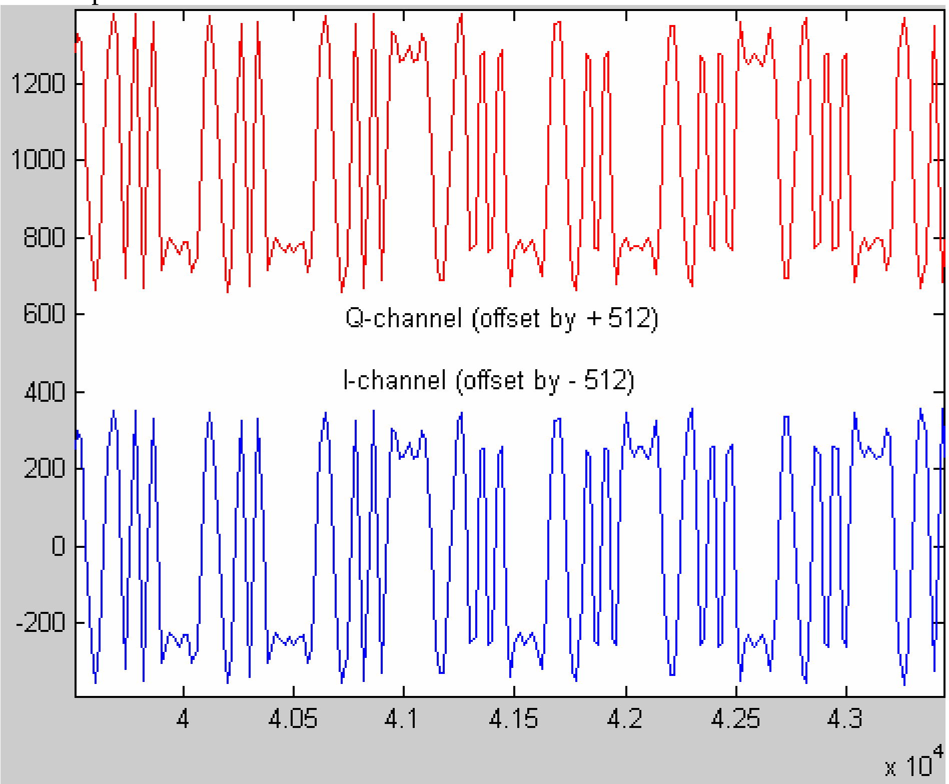
In this section, we capture the key signals during VHDL simulation and plot them. The simulation is for back-to-back modulator-demodulator in ideal conditions (noiseless, no frequency offset, stable gain). The plots refer to the internal VHDL signal names. The internal clock is set at 100 MHz for simplicity (actual clock is 80 MHz).

For this simulation, the COM-1019 DSSS modulator is configured as 19.9 Mchips/s, Barker code, code length 13, PRBS-11 internal data source, QPSK modulation, noiseless, maximum amplitude, no frequency offset.
NCO_CHIP_RATE := x”3FAE14”
SPREADING_FACTOR := x"00000D"
CODE_SEL := "011"
NCO_CENTER_FREQUENCY := x"000000"
SIGNAL_SCALING := "011111111"
AWGN_SCALING := "000000000"
REG18 := x"96"

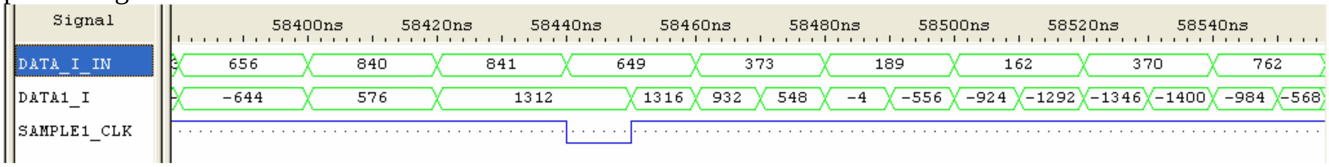
REG19 := x"0C"
REG20 := x"00"

Likewise, the COM-1018 DSSS demodulator is configured as follows:
NOMINAL_CHIP_RATE := x"3FAE14"; -- 19.9 Mchips/s
SPREADING_FACTOR := x"00000D"
CODE_SEL := "011"
NOMINAL_NCO_CENTER_FREQUENCY := x"000000"
REG16 := x"82"
REG17:= x"25"; -- QPSK, serial output, integration over 8 bits for each ½ chip sequential code search.
REG18:= x"80"; -- address 0, output type: shared bus

Input baseband samples DEMOD/DATA_I_IN(9:0) and DATA_Q_IN(9:0)
Sampling rate 80 Msamples/s

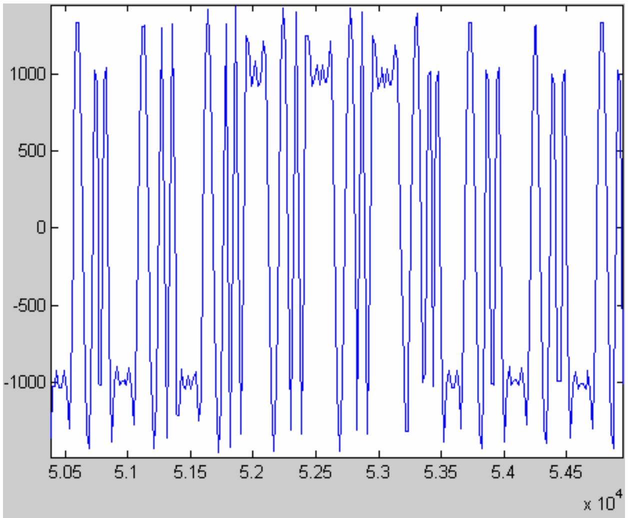


Baseband samples after x2 interpolation, decimation to 4 samples / chip, format conversion to 2's complement and precision extension to 12-bits.
DEMOD/DATA1_I and associated sample clock SAMPLE1_CLK
The sample clock SAMPLE1_CLK is nearly always high because 4*chip rate = 79.2 MHz out of 80 MHz processing clock.

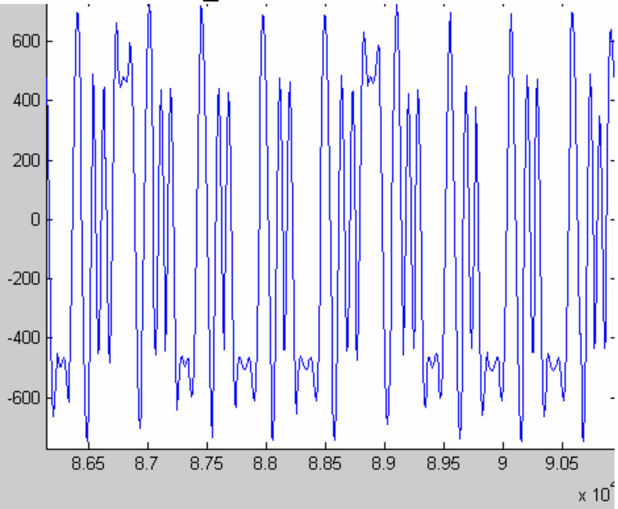


```
Signal        58400ns     58420ns     58440ns     58460ns     58480ns     58500ns     58520ns     58540ns
DATA_I_IN     656         840         841         649         373         189         162         370         762
DATA1_I       -644        576         1312        1316  932   548   -4    -556  -924  -1292 -1346 -1400 -984  -568
SAMPLE1_CLK
```

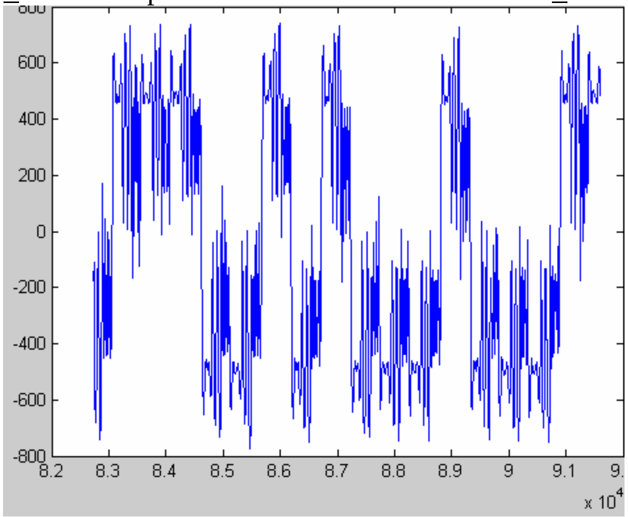
Plot of DEMOD/DATA1_I below:



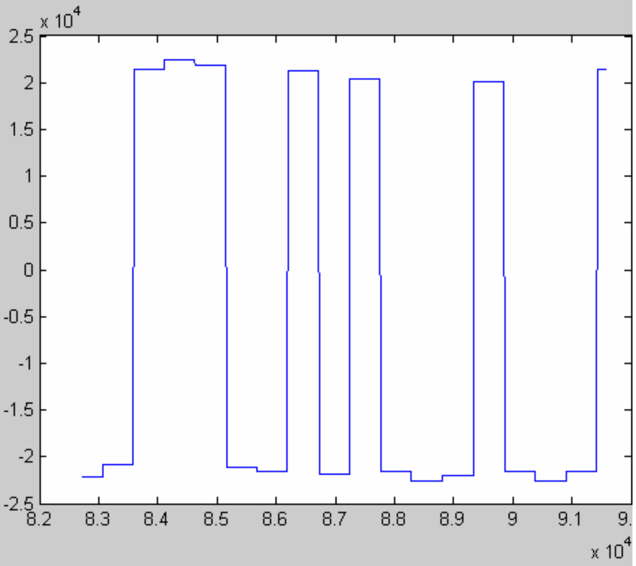
Baseband samples after phase/frequency compensation, root raised cosine filtering. The sampling rate is 4 samples / chip. Once the code tracking loop is locked, the amplitude at the center of chip is constant (amplitude +500 or -500 approximately) as the combined transmit/receive filters do not theoretically cause any intersymbol interference. This is a plot of DEMOD/DATA3_I below:



Despreading with the CODE3_D2 code replica results in DEMOD/DATA10_I:



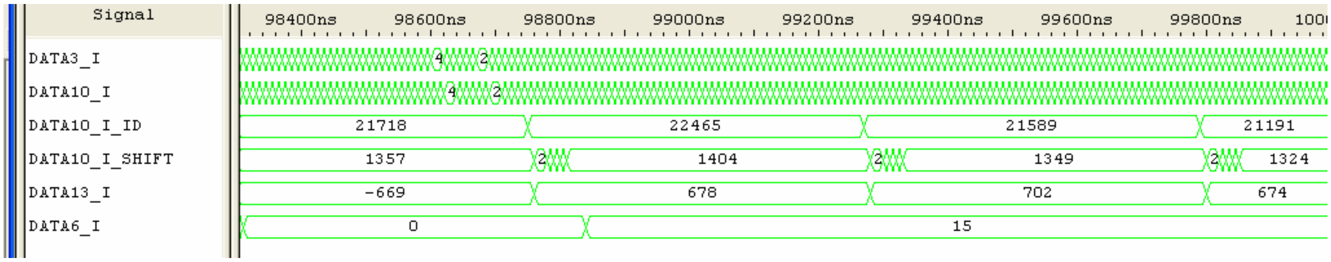
The despread signal is subjected to low-pass filtering by an integrate and dump circuit synchronized with the bit clock. The result is DEMOD/DATA10_I_ID:



The demodulated signal is then normalized in two steps:

(a) a first coarse gain adjustment based on the spreading factor (by shifting DATA10_ID n times to the right, where 2^n-1 is the code length).

(b) the resulting DATA13_I then undergoes fine gain adjustment through the AGC3 module. The 4-bit soft quantized output is named DATA6_I. The information bit is the most significant bit of DATA6_I. The lower three bits indicate the signal quality. Thus “0000” indicates a strong zero, “1111” a strong one, “0111” a weak zero, etc.



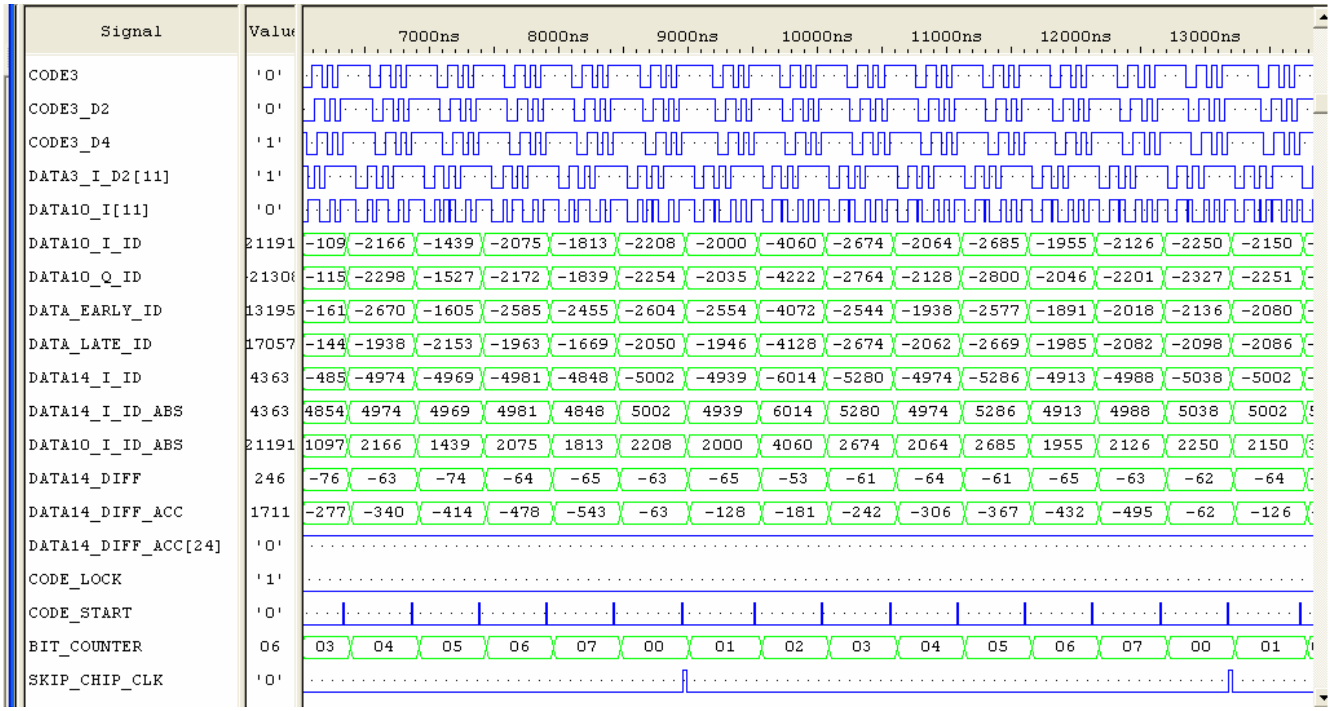
In addition to the 4-bit soft-quantized demodulator output, the AGC3 module also generates 8-bit quantized demodulated signals for internal use by the carrier tracking loop.

Code Acquisition

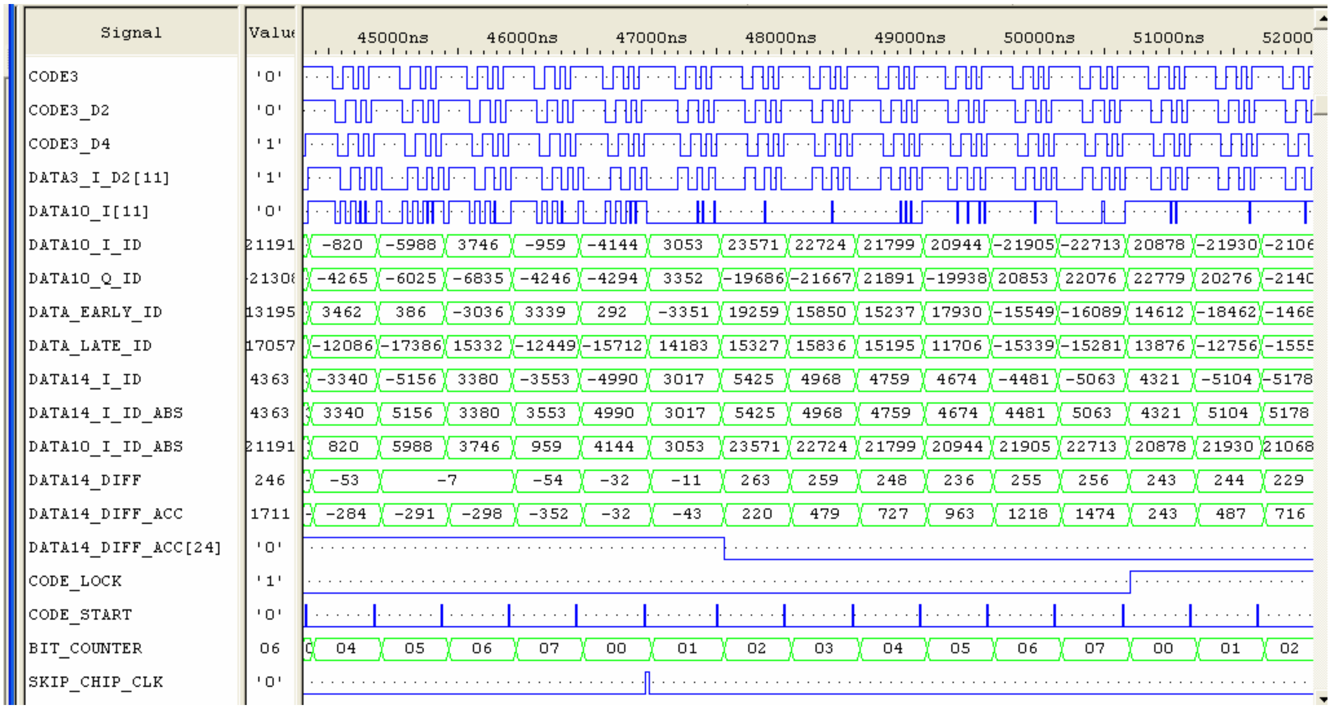
The code acquisition algorithm is a text-book sequential search: the code replica CODE3_D2 is shifted to the right by steps of ½ chip every N symbol periods, where the dwell time N is specified in control register REG17(6:4). This simulation specified a dwell time of 8 symbols.

During the dwell time, the input signal DATA3_I is despread by the code replica CODE3_D2, resulting in DATA10_I. The despread signal is subsequently subject to an integrate and dump low-pass filter. The I&D output is DATA10_I_ID.

The salient signals are shown below while the demodulator is in code acquisition mode:



Code acquisition signals. Acquisition mode



Code acquisition signals. At the time of acquisition.

The last code replica phase adjustment (SKIP_CHIP_CLK) results in a near-perfect alignment of the code replica (CODE3_D2) and the received signal DATA3_I_D2 spreading code. The despread signal DATA10_I is no longer noise-like. Consequently, the energy in the DATA10_I_ID_ABS received signal after dispreading/integrate & dump/ absolute value is greater than the noise-based variable threshold 1.25*DATA14_I_ID_ABS. This difference is integrated over the 8-bit integration period before confirming code lock (reason: the algorithm has to be sturdy in presence of noise).

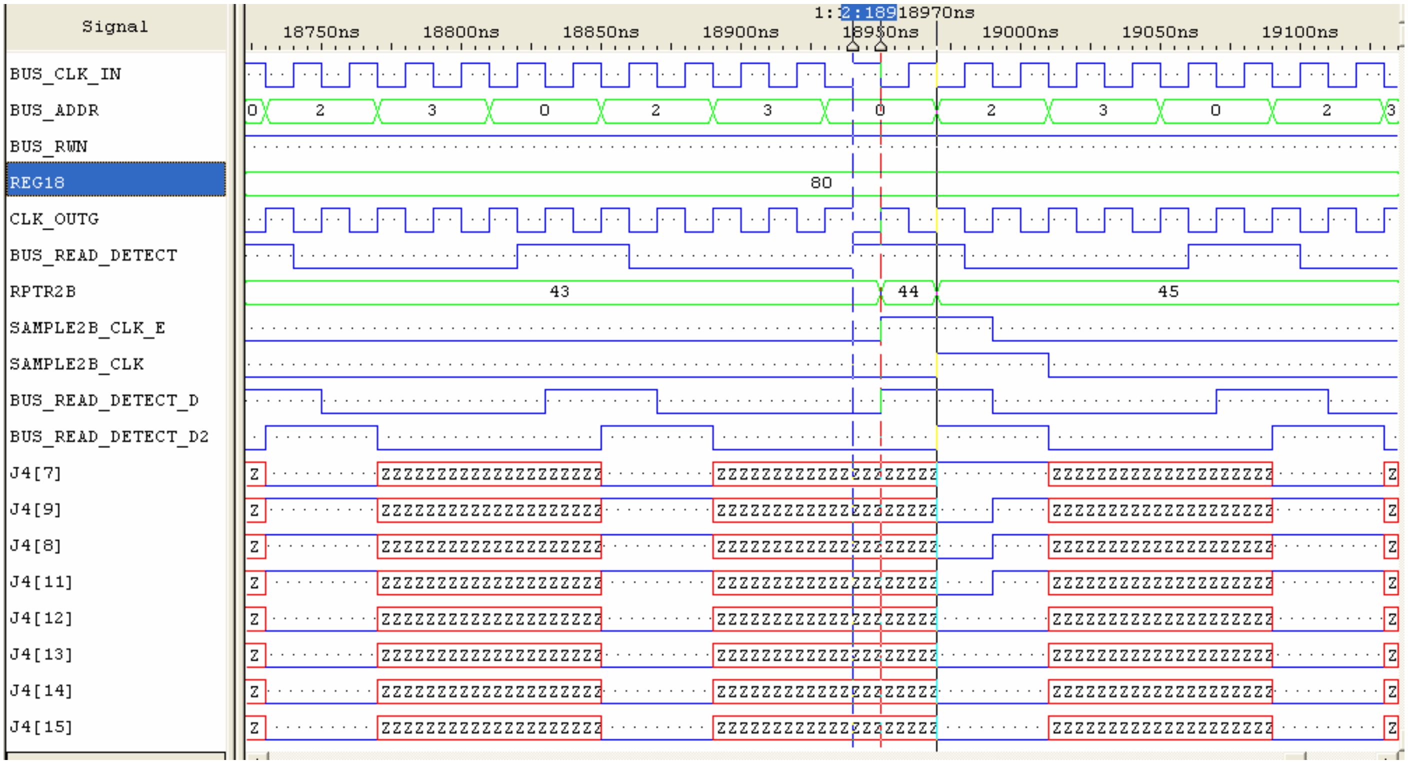
Output Data Bus

Definition:

Output Module Interface Shared bus, REG18(7) = ‘1’	**Definition**
BUS_CLK_IN	40 MHz input reference clock for use on the synchronous bus.
BUS_ADDR[3:0]	Bus address. Input (since this module is a bus slave). Designates which slave module is targeted for this read transaction. Read at the rising edge of BUS_CLK_IN
BUS_RWN	Read/Write#. Input (since this module is a bus slave). Indicates whether a read (1) or write (0) transaction is conducted. Read at the rising edge of BUS_CLK_IN. Read and Write refer to the bus master’s perspective.
BUS_DATA[15:0]	Bi-directional data bus. Input when BUS_RWN=‘0’. Output when BUS_RWN=‘1’. Read latency is 2 BUS_CLK_IN periods. Minimum read cysle is 3 BUS_CLK_IN periods. Reading can be continuous. Functional definition during read: • bit 0 BIT_CLK_OUT. '1' when DATA_I_OUT is available • bits(4:1) DATA_I_OUT[3:0] demodulated data stream. • bit 5 RX_LOCK. • bits(10:6) test points. • bits(15:11) undefined.

The COM-1018 can be configured to forward the demodulated data and ancillary information (soft-quantized bits, lock status, test points) to a subsequent ComBlock over a shared data bus. One usage example for the shared data bus is to route multiple demodulated data streams from multiple demodulators to a signal diversity combiner.

The COM-1018 demodulator is a bus ‘slave’. It only responds to read commands when the BUS_ADDR matches the unique module bus address, as configured in control register REG18. The read latency is 2 bus clocks. The key signals are shown in the VHDL simulation below:



In the simulation above, the COM-1018 is configured to respond to bus address 0. The bus master periodically reads two samples from the COM-1018 demodulator. The COM-1018 drives the data bus with a latency of 2 bus clocks. At all other times, it sets the data bus signals J4(18:7)in high impedance.

FPGA Occupancy

```
Design Summary
--------------
   Number of errors:      0
   Number of warnings:   19
   Number of Slices:                 3,070 out of  3,072   99%
   Number of Slices containing
      unrelated logic:                  34 out of  3,070    1%
   Number of Slice Flip Flops:       3,597 out of  6,144   58%
   Total Number 4 input LUTs:        4,247 out of  6,144   69%
      Number used as LUTs:                         4,138
      Number used as a route-thru:                   109
   Number of bonded IOBs:               54 out of    142   38%
      IOB Flip Flops:                                 32
   Number of Block RAMs:                 5 out of     16   31%
   Number of GCLKs:                      4 out of      4  100%
   Number of GCLKIOBs:                   4 out of      4  100%
   Number of DLLs:                       1 out of      4   25%
Total equivalent gate count for design:  151,420
Additional JTAG gate count for IOBs:  2,784
```

Contact Information

MSS • 18221 Flower Hill Way #A •
Gaithersburg, Maryland 20879 • U.S.A.
Telephone: (240) 631-1111
Facsimile: (240) 631-1676
E-mail: info@comblock.com