Key Design Features

- Synthesizable, technology independent VHDL IP Core
- 16-bit signed input data samples
- Automatic carrier acquisition with no complex setup required
- User specified design parameters
- Practical symbol rates of up to 10 Mbits/s
- Typical FPGA sample rates of up to 200 MHz
- Low area footprint ideal for FPGA

Applications

- Software radio
- Medium to long-range telemetry
- SRD and ISM band devices
- Robust, low bandwidth RF applications for small FPGA devices
- Low-cost radio links over a few 100 meters using either wireless or cable - e.g. coax or twisted pair
- Applications where fast carrier acquisition is essential - e.g. where data packets are transmitted in discrete bursts

Generic Parameters

Generic name	*Description*	*Type*	*Valid range*
threshold	Bit decision level threshold for a 0 / 1	integer	0 to 65535
sym_period	Symbol period in sample clocks	integer	0 to 65535
sym_polarity	Swaps symbol polarity i.e. 1 ↔ 0	boolean	TRUE / FALSE

Pin-out Description

Pin name	*I/O*	*Description*	*Active state*
clk	in	Sample clock	rising edge
reset	in	Asynchronous reset	low
en	in	Clock enable	high
x_in [15:0]	in	BPSK 16-bit signed input	data
mag_out [15:0]	out	Magnitude of symbol values at the decoder (16-bit signed)	data
psk_val	out	PSK bit valid strobe	high
psk_out	out	PSK bit out	data

Block Diagram

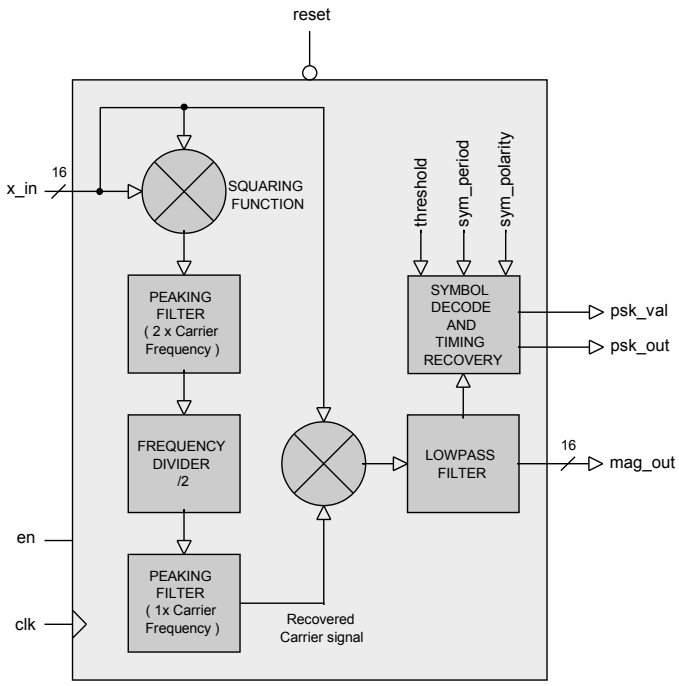


Figure 1: BPSK Demodulator architecture

General Description

BPSK_DEMOD is a Binary-PSK demodulator based on a multiply-filter-divide architecture. The design is robust and flexible and allows easy connectivity to an external ADC.

As the the carrier recovery circuit is open-loop, there is no feedback path or loop-filter to configure. This results in an extremely simple circuit with a very fast carrier acquisition time. The only requirement is that the user set the desired symbol period and a suitable threshold level for the bit decisions at the symbol decoder. The other design parameters including carrier frequency, symbol rate and sampling frequency should be specified by the user before delivery of the IP Core[1].

The input data samples are 16-bit signed (2's complement) values that are synchronous with the system clock. Input values are sampled on the rising edge of *clk* when *en* is high.

Figure 1 shows the basic architecture in more detail. The input signal is first squared in order to generate a harmonic at twice the carrier frequency and zero phase-shift. This squared signal is then filtered and divided in frequency to recover the original carrier. A second filter is employed to isolate a clean carrier signal which is used to demodulate the original input signal. The demodulated input signal is then low-pass filtered before a bit-decision is made at the symbol decoder.

The demodulated BPSK bit-stream appears at the output *psk_out*. Bits are valid on the rising edge of *clk* when both *psk_val* and *en* are high.

1 Please contact Zipcores first to discuss your design parameters. We can also provide a sub-set of programmable design parameters if necessary.

Peaking Filters

The signal path is filtered using a series of peaking filters in order to recover the carrier signal. These filters are precision 2nd-order IIR filters with fully configurable 16-bit coefficients. The coefficients are specified at compile time and should be set correctly for the chosen sample rate and carrier frequency. The characteristics of these filters will largely determine how fast the carrier signal is acquired and how the system responds to jitter and frequency drift between the transmitter and receiver clocks.

Figure 2 below shows the example filter responses for a system with a carrier frequency set to 2 MHz. The first peaking filter isolates the 4 MHz tone after squaring and the second filter recovers the 2 MHz carrier. The sample frequency is set to 50 MHz.

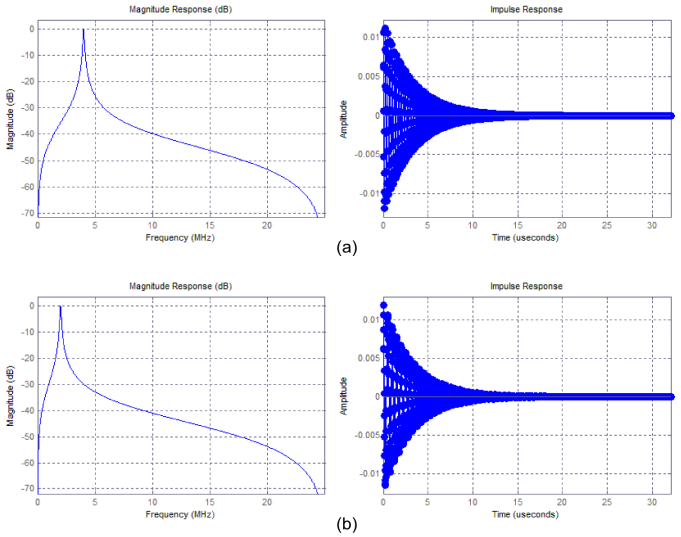


Figure 2: Example peaking filter responses for a 50 MHz sample rate: (a) 4MHz peak, (b) 2 MHz peak

Low-pass Filter

After demodulation, the signal is low-pass filtered to remove any unwanted frequency components above the symbol frequency. The following plot in Figure 3 is for a typical filter response for a symbol rate of 2 Mbps.

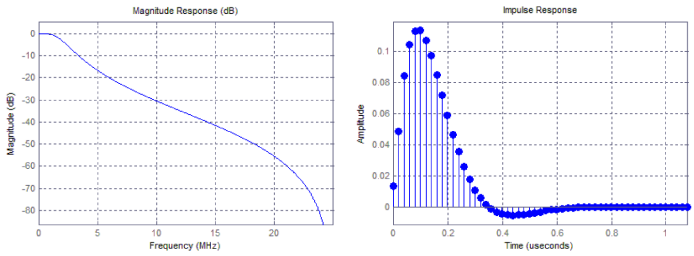


Figure 3: Example low-pass filter response with -3dB cut-off frequency set at symbol rate

Symbol rate, Carrier frequency and Sample rate

In order to recover the phase information correctly from the modulated signal, there must be at least one full wave of the carrier signal per symbol. Figure 4 shows this relationship pictorially.

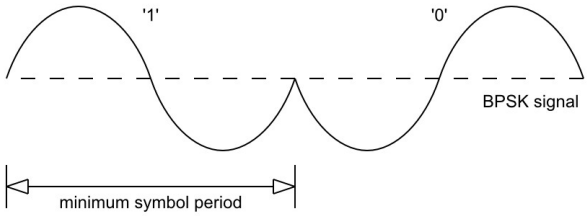


Figure 4: Relationship between carrier sinusoid and symbol period

In other words, the symbol period must be no smaller than the wavelength of the carrier signal. As a general rule, the maximum symbol period is given by:

$$Symbol\ period \geq 1\ /\ Carrier\ frequency$$

For example. If the carrier frequency is set to 2 MHz, then the BPSK symbol period should be set to 0.5 us or greater. This equates to a symbol rate of 2 Mbps or less.

In addition, it is observed by experiment that the most reliable results are achieved when the system sampling frequency is at least 10 times the carrier frequency of the BPSK source signal. This ensures there are enough samples for the clean recovery of the squared carrier signal.

$$Sampling\ frequency \geq Carrier\ frequency * 10$$

Symbol decoding and Timing recovery

The symbol decoder block extracts the symbol timing information and symbol values from the received BPSK signal. In order for the symbol decoder to function correctly, the generic parameters *sym_period* and *threshold* must be set appropriately.

The symbol period is specified as an integer number of clock cycles for the chosen sampling frequency. The threshold is a relative magnitude, and is used to determine the presence of a symbol (0 or 1) at the decoder. Decreasing the threshold increases the sensitivity of the decoder. Increasing the threshold decreases sensitivity.

Setting the threshold too high or too low may result in incorrect bit decisions. The best threshold level is dependent on a number of factors such as the amplitude of the input signal, the carrier frequency and the symbol period.

The output signal *mag_out* may be used to determine the best threshold level to set for a given set of parameters. It may also be used to plot an eye-diagram of the symbol before decoding and determine the SNR of the decoded bit.

Functional Timing

Figure 5 shows the operation of the BPSK demodulator during normal operation. The clock-enable signal has been de-asserted for one clock cycle to show the functionality of a a stall in the pipeline. The inputs and outputs are sampled on the rising edge of *clk* when *en* is high. The demodulated output bits are valid when *psk_val* is high.

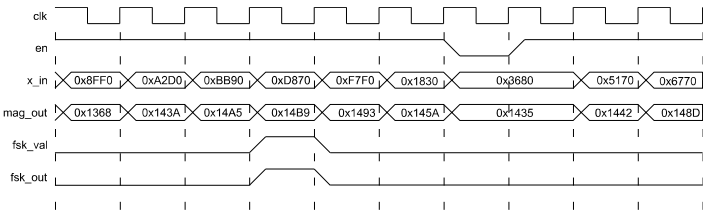


Figure 5: BPSK demodulator timing waveforms

Source File Description

All source files are provided as text files coded in VHDL. The following table gives a brief description of each file.

Source file	***Description***
sincos16.vhd	SIN/COS look-up table
dds16.vhd	16-bit DDS component
iir_biquad.vhd	Basic IIR filter component
iir_peaking.vhd	IIR peaking filter
iir_lowpass.vhd	IIR low-pass filter
bpsk_sym_dec.vhd	Symbol decoder
bpsk_sym_gen.vhd	Random symbol generator
bpsk_demod.vhd	Top-level component
bpsk_demod_bench.vhd	Top-level test bench

Functional Testing

An example VHDL testbench is provided for use in a suitable VHDL simulator. The compilation order of the source code is as follows:

1. sincos16.vhd
2. dds16.vhd
3. iir_biquad.vhd
4. iir_peaking.vhd
5. iir_lowpass.vhd
6. bpsk_sym_dec.vhd
7. bpsk_sym_gen.vhd
8. bpsk_demod.vhd
9. bpsk_demod_bench.vhd

The VHDL testbench instantiates the demodulator component and also a separate DDS that is used to generate a BPSK source signal.

In the example test provided, the system clock period is set to 50 MHz which is the sampling frequency for the simulation. The carrier frequency is set to 2 MHz and the symbol rate is set to 2 Mbps.

During the course of the test, the component 'bpsk_sym_gen.vhd' generates a randomized sequence of 1's and 0's which are used to modulate the 2 MHz carrier. The simulation must be run for at least 3 ms during which time the input bit stream and demodulated output bit stream are captured in the files *bpsk_demod_in.txt* and *bpsk_demod_out.txt*. These two files may be compared to verify that the bits have been demodulated correctly.

In addition, the output magnitudes at the symbol sample points are captured in the file *bpsk_demod_mag.txt*. These magnitudes may be used to calculate the Signal-to-Noise ratio (SNR) at the symbol decoder.

Performance

The output signal *mag_out* was captured over a series of 10,000 symbols for a carrier frequency of 2 MHz and a sample rate of 50 MHz. The symbol periods were set to 100, 50 and 25 samples representing a bit rate of of 500 kbps, 1 Mbps and 2 Mbps respectively.

In all three cases, the input BPSK signal was generated using a 16-bit DDS component, utilizing the 16-bit dynamic range available at the demodulator inputs.

The signal amplitudes in the *mag_out* capture file were overlaid over 2 symbol periods in order to realize the eye diagrams in Figure 6. The eye diagrams were noted for the size of the 'eye-openings' and the time variations at the zero crossing points.

In all three test cases, the size of the eye-openings were seen to be excellent and the time variations at the zero crossing points were observed to be minimal.

In addition, the SNR at the symbol sampling points (the eye mid-points) were measured. For comparison, the SNR was measured for each symbol rate. In all three test cases, the Bit Error Rate (BER) was observed to be zero over the 10,000 symbols. The SNR at the symbol sample points was calculated using the following formula:

$$SNR = 20\log\frac{\bar{A}_1 - \bar{A}_0}{\sqrt{\sigma_1^2 + \sigma_0^2}}$$

Where values A_1 and A_0 signify the mean signal amplitudes at the logic '1' and logic '0' levels. Values σ_1 and σ_0 are the standard deviations from the mean at the logic '1' and '0' levels. The results for the different bit rates in the example described are shown in the table below[2].

Symbol period	***Bit rate***	***SNR at decoder***
25 samples	2 Mbps	30.7 dB
50 samples	1 Mbps	31.1 dB
100 samples	500 kbps	31.9 dB

2 The values for SNR are indicative of this particular example only. The exact SNR may vary depending on the implementation.

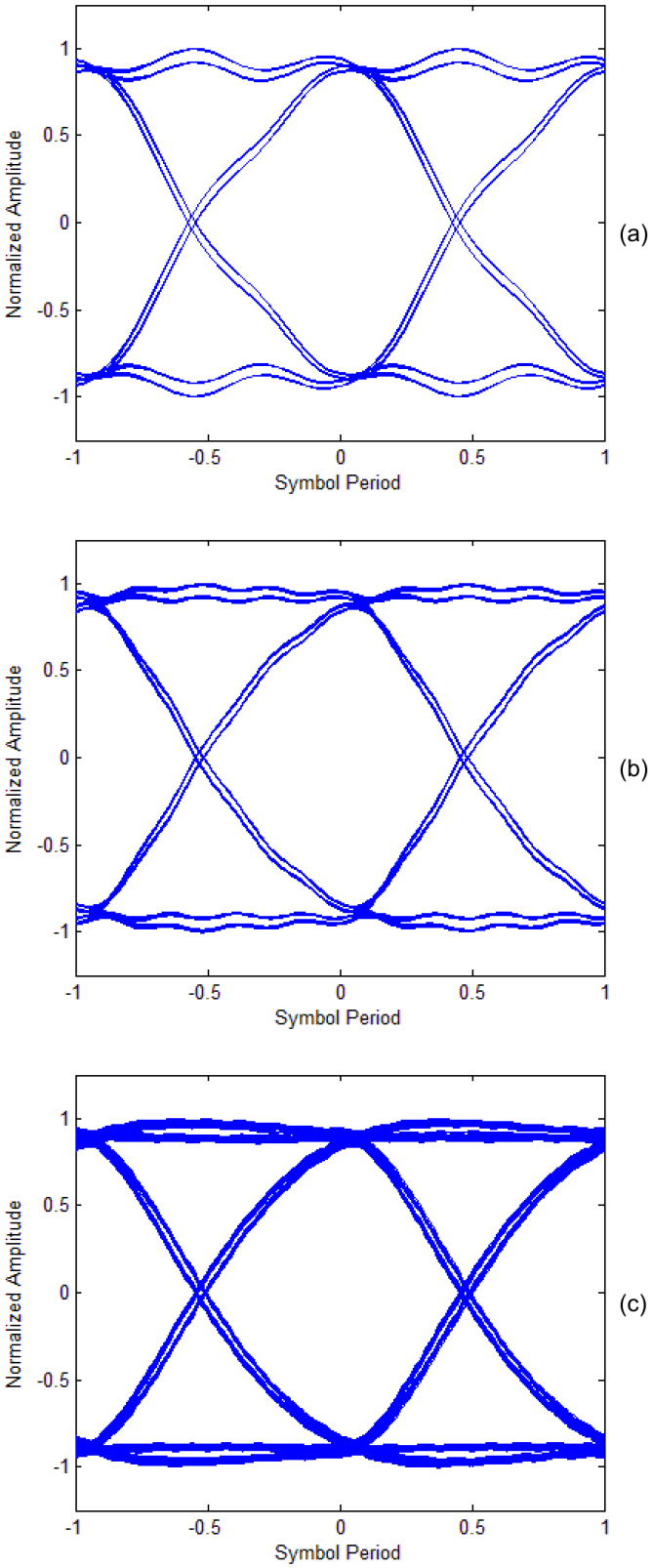


Figure 6: Eye diagrams for symbols at the decoder with different symbol rates: (a) 2Mbps, (b) 1 Mbps & (c) 500 kbps

Development Board Testing

The BSPK demodulator IP Core was implemented and tested on an FPGA-based development board. The board featured a Xilinx® Virtex4 FPGA with a DAC (AD9772A) and an ADC (AD6644) from Analog Devices®. The whole system was running at a sample rate of 50 MSPS.

The board was set up to modulate frames (packets) of data using BPSK. The digital signal was converted to analogue and transmitted via twisted pair cable over a distance of 20m. The signal was then demodulated on the same board and verified to ensure that the transmitted frames were correct.

The sample frequency was set to 50 MHz with the BPSK carrier frequency set to 2 MHz. Different symbol rates were chosen from 500 kbps to 2 Mbps. Both the Modulator and Demodulator were implemented separately on the FPGA with asynchronous 50 MHz clocks. This was done to ensure that the demodulator could tolerate frequency drift or phase differences between the transmitter and receiver clocks.

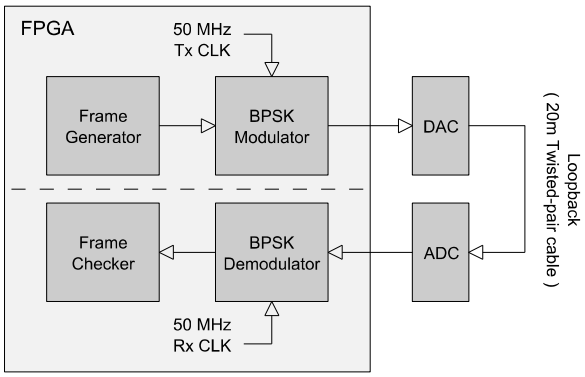


Figure 7: Test setup for the BPSK demodulator IP Core

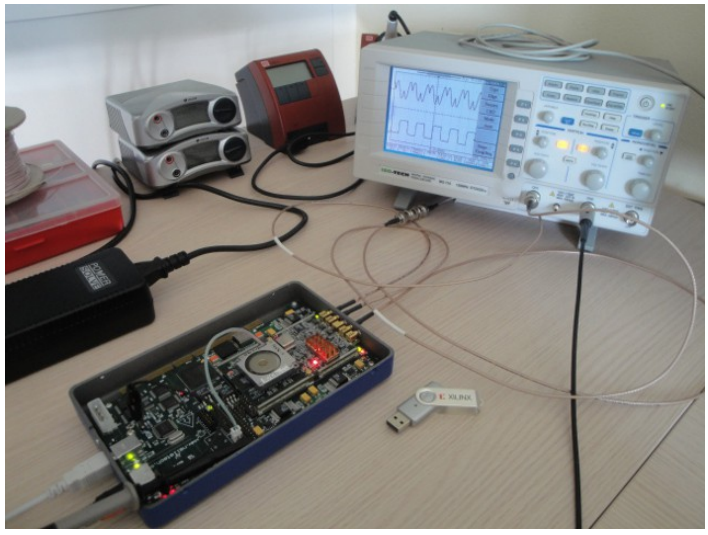
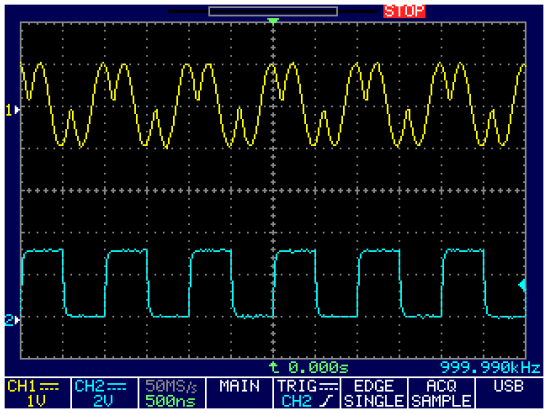
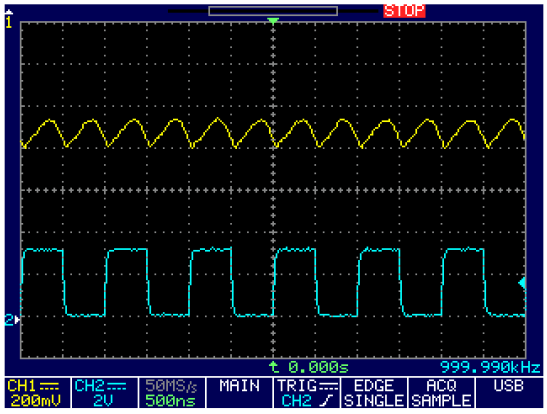


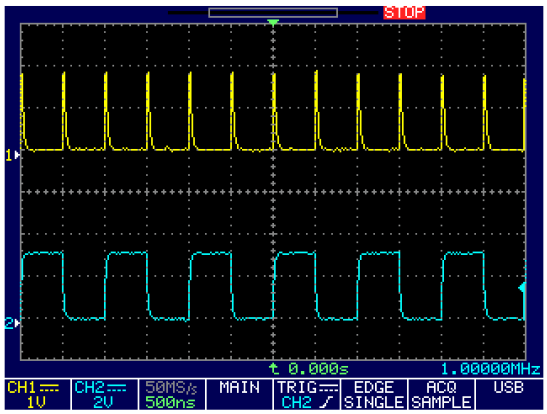
Figure 8: Dev-board test setup showing demodulated BPSK signal as measured by the oscilloscope



(a)



(b)



(c)

Figure 9: Scope traces for a 2 Mbps bitstream using the devboard test setup: (a) BPSK source, (b) Magnitude at the decoder, (c) psk_val pulses

Figure 9 shows some example scope traces for the described test setup. In this test, the carrier signal was modulated with the bitstream 1,0,1,0,1,0, … etc. The bitstream was seen to be recovered correctly at a data rate of 2 Mbps.

Synthesis and Implementation

The files required for synthesis and the design hierarchy is shown below:

- bpsk_demod
  - iir_peaking
    - iir_biquad
  - iir_lowpass
    - iir_biquad
  - bpsk_sym_dec

The VHDL core is designed to be technology independent. However, as a benchmark, synthesis results have been provided for the Xilinx® Virtex 6 and Spartan 6 FPGA devices. Synthesis results for other FPGAs and technologies can be provided on request.

Note that depending on the exact specification of the internal filters, the total number of hardware multipliers in the design may be slightly higher or lower for different implementations.

Trial synthesis results are shown with the generic parameters set to: threshold = 1000, sym_period = 25 and sym_polarity = true. The resource usage is specified after place and route.

VIRTEX 6

Resource type	***Quantity used***
Slice register	557
Slice LUT	1109
Block RAM	0
DSP48	18
Occupied slices	
Clock frequency (approx)	200 MHz

SPARTAN 6

Resource type	***Quantity used***
Slice register	557
Slice LUT	1109
Block RAM	0
DSP48	18
Occupied slices	
Clock frequency (approx)	200 MHz

Revision History

Revision	*Change description*	*Date*
1.0	Initial revision	15/09/2011
1.1	Modified synthesis results in line with minor source-code changes	03/10/2011
1.2	Added dev-board testing description including scope traces	29/02/2012
1.3	Added symbol polarity generic and optimized the IIR filters for speed	26/02/2015