# REAL TIME IMPLEMENTATION OF A SYMBOL TIMING RECOVERY ALGORITHM FOR A NARROWBAND WIRELESS MODEM

J. H. Grimm, R. N. Kumar, J. Kusuma, and J. V. Krogmeier

School of Electrical and Computer Engineering Purdue University West Lafayette, IN 47907-1285 http://yake.ecn.purdue.edu/~cominfo/

### ABSTRACT

This paper examines some of the complexity issues arising in the real time implementation of the symbol timing subsystem of a narrowband wireless modem. The modem prototype has been designed for 4 kHz channels in the 220-222 MHz land mobile band. In an effort to achieve high bandwidth efficiency, the modem architecture employs transmitter diversity, pilot symbol assisted modulation, and trellis coded modulations. An experimental, non-real time system has been implemented and extensively field tested demonstrating bandwidth efficiencies in excess of 3 bits per second per Hz. The work on this project is currently focused on the real-time implementation of the baseband receiver functions using the Texas Instruments C54x fixed point digital signal processor. Here we report on some of the performance/complexity tradeoffs present in the design of a DSP implementation of the digital filter and square symbol timing recovery algorithm.

## 1. INTRODUCTION

The goal of high bandwidth efficiency in wireless data transmission is often met by using large signal constellations, trellis coded modulations, and transmitted references for coherent detection [1]. However, the use of more sophisticated modulation places increasingly stringent requirements on the synchronization systems in the receiver including the symbol timing recovery which is performed at the first stage of the baseband receiver. For satisfactory performance in modulations with large constellations (e.g., 64- and 128-QAM), timing uncertainty must be held to a very small fraction of the signaling interval in order to attain satisfactory performance.

If the received signal has a discrete spectral component at a harmonic of the symbol frequency, the symbol timing algorithm may be as simple as a filter tuned to this harmonic. However, most situations require that the transmission of such spectral components be avoided. In a practical transmission scheme, a discrete spectral component can be produced by inserting a non-linear element before the tuned filter in the timing recovery circuit. This ad-hoc timing recovery method is known as the filter and square algorithm in the literature [2, 3, 4]. Timing jitter is caused by the combined action of thermal noise and the random nature of the transmitted data. At reasonably high signal-to-noise



Figure 1: Matched filter and timing recovery algorithm.

ratio (SNR), performance is therefore limited by the data dependent jitter which is strongly influenced by the shape of the data pulse at the input to the nonlinear element.

Digital realizations of receivers for synchronous data transmission are of growing importance as the capabilities of digital signal processors increase. To take advantage of this trend as much of the receiver function as possible should be should be digital. In other words, the input signal should be sampled by a free running oscillator and all subsequent receiver processing should be done in discrete time on the samples.

# 2. THE SYMBOL TIMING ARCHITECTURE

The symbol timing estimator and matched filter architecture is shown in Fig. 1. The input to the matched filter consists of samples of the noncoherent in-phase and quadrature channels of the received baseband signal r(t). The sampling rate  $1/T_s$  is chosen as a multiple  $N_s$  of the symbol rate 1/T. The objective of the timing estimator is to determine the optimal point to sample the output of the matched filter as shown in the figure. Not only is there an unknown time delay between transmitter and receiver but also the transmit and receive clocks are not synchronous. Thus the optimum matched filter sample time will drift. The timing estimation algorithm must be able to track this. The delay block in Fig. 1 is used to compensate for additional delay introduced by the timing estimation algorithm itself.

The architecture of the timing estimator is shown in Fig. 2. The timing estimator is based upon the filter and square algorithm [2, 3]. The algorithm extracts a timing waveform  $z(nT_s)$  whose zero crossings correspond to the optimal sampling instants for the matched filter output. More detail on the subblocks in the algorithm are presented



Figure 2: Architecture of timing estimation block.

below.

#### 2.1. The Prefilter

Franks and Bubrouski [2] have shown that by using the appropriate prefilter with the magnitude squared nonlinearity the data dependent noise is entirely eliminated. The requirements are:

- 1. prefiltered pulses have conjugate symmetry about the frequency 2/T and are bandlimited to  $1/4T \le |f| \le 3/4T$ , and
- 2. the transfer function of the postfilter has conjugate symmetry about the symbol rate 1/T and a bandwidth less than 1/T.

Denoting the square-root Nyquist pulse shape p(t), a simple discrete time prefilter that will meet these conditions is

$$h(nT_s) = \cos(2\pi nT_s/T)p(nT_s) * p(-nT_s)$$

This is simply the Nyquist pulse modulated in frequency to the symbol rate. Other researchers have derived similar constraints for other nonlinearities [3].

#### 2.2. The Nonlinearity

Various authors have noted that other nonlinearities such as magnitude, magnitude to the fourth power, etc., provide better performance in some cases. Lee and Messerschmitt [5] have suggested that the magnitude nonlinearity gives better performance for high order quadrature amplitude modulation (QAM) constellations.

### 2.3. The Postfilter

Three possibilities for the postfilter are: an infinite impulse response (IIR) filter, a finite impulse response (FIR) filter, or a phase locked loop (PLL). While the IIR filter is attractive for its low complexity, it is not acceptable for timing recovery because it does not have linear phase. The drawback of an FIR implementation of the postfilter is that a prohibitively large number of taps would be required to provide an adequately narrow bandwidth.

A PLL can meet both performance and complexity requirements in the postfilter application (see Fig. 3). An



Figure 3: Digital phase locked loop as postfilter.

input normalization is required to insure that the PLL receives a signal with constant power. One possibility for normalization is to subtract the sample mean and divide by the sample standard deviation. A less computationally complex normalization would consist of a mean normalization followed by a hard limiter.

The loop filter is the following first order IIR filter:

$$v(nT_s) = v((n-1)T_s) + e(nT_s) - \alpha e((n-1)T_s)$$

where

$$\begin{aligned} \alpha &= (1 - e^{-4\zeta \pi f_n T_s})/k \\ k &= 2(1 - e^{-\zeta 2\pi f_n T_s} \cos(2\pi f_n T_s \sqrt{1 - \zeta^2})) \end{aligned}$$

The PLL damping factor  $\zeta$  and the undamped natural frequency  $f_n$  have been chosen to be  $\zeta = 0.9$  and  $f_n = 5$  Hz. The modem operates at a symbol rate of 3570 Hz using a square-root raised cosine pulse shape with roll-off  $\beta = 0.15$ . The timing waveform  $z(nT_s)$  is the output of the voltage controlled oscillator (VCO) in Fig. 3.

#### 2.4. The Sampler

It is unlikely that the timing estimate will coincide with one of the digital samples, so the sampler needs to interpolate between the nearest two samples. Ideal interpolation requires many clock cycles; if the number of samples per symbol  $(N_s)$  is sufficiently large a linear or quadratic interpolator will be adequate.

#### 3. COMPLEXITY CONSIDERATIONS IN DSP IMPLEMENTATION

#### 3.1. Hardware Design

The system is implemented on a TI TMS320C541 16 bit fixed point DSP which has 5k of on-chip RAM and an 80 Mhz clock rate. A combination of C and assembly code is used in the implementation. The C54x allows treatment of non-integer numbers as fractions simply by setting a register flag. We use a combination of C and assembly code in the implementation. The experimental system uses a personal computer (PC) as the interface between an intermediate frequency to baseband digital down conversion (DDC) board and the DSP evaluation board. An application has been written which reads data from the DDC board, passes it to the DSP board, waits until DSP processing for that data set is finished, and then reads it back to save it into a file. The data passed between the DDC and the DSP consists of 8-bit in-phase and 8-bit quadrature samples.

Subsystem	$\operatorname{Cost}$	Operation
I/O	2	Read/write I/O
Matched Filter	$2(N_sN_w+1)$	MAC
Prefilter	$2(N_s N_{wp} + 1)$	MAC
Nonlinearity	2	absolute value
	1	add
PLL Normalization	$N_s$	add
(Hard limiter)	1	divide
	1	branch test
PLL	4	multiply
	3	add/subtract
	1	sine lookup
Zero Crossing	$2N_s$	branch tests
Detector	1	divide
	2	add/subtract
Interpolator	1	multiply
	2	add/subtract

Table 1: Estimated Computational Complexity.

Table 2: Estimated Total Complexity on C54x.

Operation	Total Cost	Cycles per
		instruction
sine lookup	1	2
divide	4	22
MAC	$2N_s(N_w + N_{wp} + 2)$	1
square	1	1
multiply	5	1
add/subtract	12	1
absolute value	1	1
branch tests	$2N_s$	5
read/write I/O	2	20

## 3.2. Complexity Estimation

The estimation of the number of operations needed to process data is presented in Tables 1 and 2. Estimated memory requirements are given in Table 3. The TI C54x DSP can do multiplication, addition, and calculation of circular array index in 1 cycle, which significantly improves processing speed.

#### 3.3. Complexity Reduction

Implementation of the variance normalization in the PLL subsystem would take 3 divide operations and a square root.

Table 3: Estimated Memory Requirements.

Subsystem	Memory
Matched Filter	$N_s N_w + 1$
Prefilter	$N_s N_{wp} + 1$
Delay	$N_s N_{wp}$
PLL Normalization	2
PLL	4



Figure 4: BEP for DFS timing recovery, 64 and 256 QAM in AWGN,  $N_s = 4$ , absolute value nonlinearity, and hard limiter PLL normalization.

A simpler normalization consisting of a mean correction followed by thresholding was chosen. There was no significant performance penalty paid for the simplification.

An attractive alternative to the magnitude squared nonlinearity is to add the absolute values of the in-phase and quadrature components. Not only does this require less clock cycles, but it also requires less dynamic range which is an important issue for fixed point DSPs.

QAM requires the system to have linear phase, so an FIR square-root raised cosine filter was chosen for the Nyquist pulse shape. The pulse was truncated at  $N_w$  symbol periods left and right of the origin giving a total of  $N_s N_w + 1$  filter coefficients. If  $N_s$  is too small the interpolation operation of the sampler will induce too much intersymbol interference (ISI), and if  $N_w$  is too small the pulse shape will no longer meet the Nyquist criterion.

## 4. ILLUSTRATION OF COMPLEXITY/PERFORMANCE TRADEOFFS

The timing recovery subsystem was implemented in the Signal Processing Worksystem (SPW) to quantify tradeoffs between complexity and performance. Simulations demonstrated that for all scenarios tested there was negligible loss in bit error probability (BEP) performance when using the absolute value nonlinearity in place of the magnitude squared nonlinearity, and negligible loss when using 4 samples per symbol. Furthermore, using the hard limiter to normalize the input to the DPLL did not degrade performance noticeably.

BEP curves for uncoded QAM are shown in Figs. 4 and 5 along with the analytical BEP. The gray bit mapping from [6] was used to facilitate calculation of the exact analytic BEP. The result in Fig. 5 is for slow varying, frequency non-selective Rayleigh fading with ideal channel state information (CSI).

Several observations may be made from these BEP plots.



Figure 5: BEP for DFS timing recovery, 64 and 256 QAM in Rayleigh fading,  $N_s = 4$ , absolute value nonlinearity, and hard limiter PLL normalization.



Figure 6: 256 QAM scatter plots in the presence of no AWGN. Left  $N_w = 10$ , right  $N_w = 14$ .

Truncating the matched filter to  $N_w = 12$  symbol periods (6 on either side of the origin) causes tolerable degradation in BEP and results in a FIR filter length of 49 samples. Using  $N_w = 14$  is closer to ideal performance and only costs 8 more coefficients. Note that in fading the BEP when  $N_w = 8$  levels off at high SNR - if the ISI induced by filter truncation is too severe, bit errors will result even when SNR is extremely high. Figure 6 shows the effect of truncation length on the signal constellation.

## 5. CONCLUSION

This timing recovery system was also incorporated in a wireless narrowband modem optimized for Rayleigh fading channels. The system includes transmitter diversity, trellis coded modulation, and pilot symbol assisted modulation [7]. The timing recovery circuit has performed well for all tests to date.

With the PLL bandwidth set around 5 Hz the timing recovery system takes about 300 msec to converge. The loop bandwidth may be increased to trade off convergence speed for steady state estimation variance. Fast acquisition time is usually a major requirement for time division multiple access (TDMA) systems. However, if all users share a common pulse shape and are synchronized, timing recovery could be obtained from the signal of other users and the convergence time would not be a problem.

This paper has demonstrated the feasibility of implementing symbol timing recovery on a DSP by enumerating some of the performance/complexity tradeoffs.

## ACKNOWLEDGEMENT

This work was supported by the National Academy of Sciences ITS-IDEA program and by an equipment donation from Texas Instruments, Inc. through the TI Elite Labs program.

#### 6. REFERENCES

- J. K. Cavers. An analysis of pilot symbol assisted modulation for Rayleigh faded channels. *IEEE Trans. Veh. Tech.*, vol. VT-40:686–693, November 1991.
- [2] L. E. Franks and J. P. Bubrouski. Statistical properties of timing jitter in pam timing recovery schemes. *IEEE Trans. Commun.*, COM-22:913-920, July 1974.
- [3] A. N. D'Andrea, U. Mengali, and M. Moro. Nearly optimum prefiltering in clock recovery. *IEEE Trans. Commun.*, COM-34:1081-1088, November 1986.
- [4] M. Oerder and H. Meyr. Digital filter and square timing recovery. *IEEE Trans. Commun.*, COM-36:605-611, May 1988.
- [5] E. A. Lee and D. G. Messerschmitt. Digital Communications. Kluwer Academic Publishers, Boston, 1988.
- [6] J. P. Seymour. Improved Synchronization in the Mobile Communications Environment. PhD thesis, Purdue University, West Lafayette, IN, 1994.
- [7] M. P. Fitz, J. V. Krogmeier, J. Grimm, J. A. Gansman, T.-A. Chen, and T. M. Magnusen. The 220 MHz ITS spectral allocation: Potential, pitfalls, and applications. *IEEE Communications Magazine*, 34(10):42-54, October 1996.