DYNAMICAL ANALYSIS OF ALL-DIGITAL SYMBOL TIMING RECOVERY IN TWISTED PAIR BROADBAND RECEIVERS

Jan Maris  Patrick Schaumont  Serge Vernalde  Marc Engels  Ivo Bolsens
IMEC - VSDM, Kapeldreef 75, B-3001 Leuven, Belgium
e-mail: {maris, schaun, vernalde, engels, bolsens}@imec.be

Abstract: The use of all-digital symbol timing recovery is presented for a contemporary high speed modem application. This digital algorithm increases the integration and reduces the equalizer complexity. A dynamical analysis is presented, and the influence of important channel impairments is investigated. We propose a modern architecture that includes this algorithm. Finally, we present the results of an implementation exploration.

I. INTRODUCTION
In the fast evolving world of broadband access networks, VLSI implementation is a necessity. Each functional block of the modem is characterized by a cost and a performance feature. By trading off these characteristics for each block, the designer is able to devise an optimal global architecture. We present a design exploration for symbol timing synchronisation algorithms for twisted pair modems.

II. OVERVIEW
The applications currently under study are continuous mode modem connections over standard subscriber loops, such as twisted pair and coax cables. The bitrates for these applications are typically several tens of megabits per second and are realized with QAM modulation schemes in order to make maximal use of the available bandwidth. For these range of operation speeds, we present the use of all-digital symbol timing synchronisation [2, 3]. Digital synchronisation algorithms have important advantages over conventional symbol synchronisation algorithms, such as full-digital analysis and design of the loop algorithm, use of a crystal oscillator instead of a VCXO, and reduced delay in the control loop which allows faster tracking [1]. Moreover, by making use of a dedicated solution for timing recovery, the constraints put on the equalizer can be reduced, which will also be reflected in the complexity of the modem.
This study is done for the DAVIC framework [8]. However, most of the results and conclusions are also usable for similar standards (e.g. IEEE 802.14).
In the subsequent sections we will present the basic structure of all-digital interpolation based timing recovery. This is then analysed in the presence of channel attenuation and channel group delay variation. In the last section, we will present the results of the VLSI implementation of this functional block.

III. DYNAMICAL ANALYSIS

In [1] a statistical analysis of the timing recovery was presented. For convenience, we present again the modem architecture which is used, now extended with an equalizer section. In fig. 1 the timing synchro-

![Diagram](image-url)

Figure 1: Modem architecture with baseband digital timing recovery.

nisation block is shown as a part of the baseband modem functionality. This architecture features a $K$ times oversampled symbol rate which is decimated by the timing recovery block producing samples at the baud rate $R = 1/T$. The data filter is matched to the expected received pulse shape and maximizes the signal to noise ratio of the incoming signal. The AGC restores the desired power level. The complex equalizer receives timing corrected symbols and compensates for the residual frequency response distortion in the channel. This type of linear equalizer allows for pipelined VLSI implementations which are needed to meet the required data throughput. Moreover, this architecture allows for separated and independent feedback paths. This reduces both the algorithm and implementation complexity.
The timing recovery loop for a simple data branch (I or Q) is shown in more detail in fig. 2. In an all-digital modem, the continuous input waveform is only available as a stream of discrete samples. Using the band-limited properties of the input waveform, the digital actuator then uses interpolation on input samples to derive the samples of the output grid [2].
The interpolation is done with a polynomial (linear, quadratic or cubic) [1]. The interpolation filter is preceded by a precompensation filter that reduces the non-ideal interpolation behaviour [2, 1]. This synchronisation algorithm is a variable rate process. This is demonstrated in fig. 3, where a stream of output samples is to be generated at a slower pace than the input samples. Here, a quadratic interpolation (IP) is used, that uses a basepoint set of three subsequent input samples and generates an interpolant using a fractional part of a control value $\text{frac}(\phi_{err}) = \mu_k$. The control value indicates the abscis of the interpolant with relation to the basepoint set and is provided by the integrator in the feedback loop. As time proceeds, the basepoint set slides along the input samples to evaluate new interpolants. After generation of two output samples however (fig. 3), the required interpolant is outside the timespan of the basepoint set at that time. This is reflected by the control value $\phi_{err,k}$ that exceeds the sample period. The decimator uses this to decide to select the preceeding ($\phi_{err,i} < 0$) or succeeding sample ($\phi_{err,i} > 1$) instead of the nominal input sample.

The input samples are taken at a rate $F_s = KR$ of a free running clock. The timing error can be both related to a constant phase mismatch of the sampling instant, or worse, a varying phase mismatch due to a deviation between the transmitter and receiver clock. In this study, the clock oscillators of the receiver and the transmitter feature 50ppm frequency tolerance, so that the total worst case frequency error is 100ppm.

This algorithm is implemented with simple digital logic. The proposed circuit is based on a modulo $K$ counter and a comparator, fig. 4.

![Figure 2: Interpolation based symbol timing recovery control loop](image)

![Figure 3: Digital interpolation for resampling](image)

![Figure 4: Generation of strobing conditions](image)
3. introduction of self-noise by the timing error detector,
4. propagated channel noise,
5. delay distortions in the channel.

In table 1 the results of the dynamical analysis are presented. This study was done for three cases. A constant worst case phase error (1), a constant phase error averaged over uniform distributed \( \mu \) (2), and a worst case (100ppm) frequency error (3). These results where obtained with the quasi-analytical performance calculation method [7], based on the simulation output in the tracking regime. The simulation was done with floating point number representation. The acquisition time is typically 1000 symbols, depending on the loop parameters. As can be seen, the quadratic compensated interpolator yields 0.25dB degradation with respect to the theoretical QAM-16 performance. In [1] it was shown that the silicon area doubles per order of the interpolation polynomial. This is clearly not so for the performance.

### IV. AWGN INFLUENCE

In the application for these modems, the channel suffers from an AWGN level of -140dBm. This distorts the TED operation while the TED also adds self-noise of its own. We will show that these sources do not degrade the loop operation significantly. Indeed, straightforward calculations show that noise filtered through the TED is still zero mean, and that the resulting variance is in the order of the squared variance of the input. Moreover, the loopfilter is of low-pass nature, and it removes the residual high frequency noise components.

Although the integration of noise yields a zero mean noise process, it has bad performance for low frequencies, which could interfere with the low-pass timing error estimate. However, any temporal deviation on the timing estimation due to integrated noise will be detected by the TED so that the closed loop behaviour guarantees proper maximum likelihood operation.

It can be concluded that the noise does not affect the timing synchronisation. Experiments showed that the average value of the loopfilter output \( \tau \) is independent of the channel noise level (table 2). This table also shows that the acquisition time \( t_{acq} \) is not significantly influenced by the introduction of channel noise.

<table>
<thead>
<tr>
<th>( \frac{E_b}{N_0} ) (dB)</th>
<th>( \tau )</th>
<th>( \sigma_\tau )</th>
<th>( t_{acq} ) samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>With channel noise</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.0391</td>
<td>0.0000</td>
<td>3029</td>
</tr>
<tr>
<td>12</td>
<td>0.0298</td>
<td>0.0015</td>
<td>3111</td>
</tr>
<tr>
<td>13</td>
<td>0.0298</td>
<td>0.0013</td>
<td>3029</td>
</tr>
<tr>
<td>14</td>
<td>0.0299</td>
<td>0.0019</td>
<td>3007</td>
</tr>
<tr>
<td>15</td>
<td>0.0245</td>
<td>0.0017</td>
<td>2804</td>
</tr>
<tr>
<td>16</td>
<td>0.0302</td>
<td>0.0001</td>
<td>3094</td>
</tr>
<tr>
<td>Without channel noise</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \infty )</td>
<td>0.0300</td>
<td>0.0294</td>
<td>2590</td>
</tr>
</tbody>
</table>

Table 2: Loop variance and acquisition time for various channel noise levels (\( K = 3 \), linear compensated interpolation)

### V. ATTENUATION

Broadband applications on the existing copper infrastructure also suffer from frequency dependent attenuation, e.g. [9]. This effect was modeled with a linear phase FIR filter and we investigated its effect on the loop. Two important observations can be made:

1. The amplitude distortion over the received bandwidth can reach as much as 25dB. Therefore, equalisation is unavoidable.

2. Fractionally spaced equalizers can correct static timing phase errors, but no frequency errors, because this requires variable decimation.

Therefore, a symbol spaced equalizer with 15 taps is used in conjunction with a separate symbol timing loop.

The performance evaluation is summarized in table 3 for case 1 (worst case phase error) and case 3 (100 ppm frequency error). These results show that an attenuated channel does not degrade the performance significantly. Moreover, these results take into account the AGC and the equalizer operation. The presence of the equalizer clarifies also the considerable performance gain for the linear interpolation. The equalizer compensates the residual interpolation error.

<table>
<thead>
<tr>
<th>( \frac{E_b}{N_0} ) (dB)</th>
<th>( \tau )</th>
<th>( \sigma_\tau )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.38</td>
<td>0.33</td>
</tr>
<tr>
<td>3</td>
<td>0.37</td>
<td>0.38</td>
</tr>
</tbody>
</table>

Table 3: \( \frac{E_b}{N_0} \) degradation (dB) at \( P_e = 10^{-6} \) for constant phase error and 100ppm frequency error. Attenuated channel: 300m UTP-Cat.3.

### VI. PHASE DISTORTION SENSITIVITY
Broadband access infrastructure feature localized group delay variations. In some parts of the band, this variation reaches $\tau_g = 100\text{ns}/\text{MHz}$. This effect distorts the pulse severely so that the TED becomes biased. The loop equilibrium point is shifted away from $\mu = 0$. Proper loop functionality cannot be guaranteed anymore. Simulations showed that for $K = 3$, the bias is as much as $3T_s/31.25\text{ns}$. As a possible solution, the timing information is to be calculated after equalization (and resampling).

VII. IMPLEMENTATION

In this section the results of a design exploration towards implementation of this algorithm into synthesizable silicon architectures are presented. The results of this study were obtained with the Cathedral-3 silicon compiler [5]. The complexity is expressed as silicon area (no intercell routing included). The interpolator was implemented with the Farrow structure [3, 1].

The target broadband application requires a baud rate of $R = 6.48$ Mbaud. Since we use an oversampling ratio of $K = 3$, the sample rate is $F_s = 19.44\text{MHz}$. Quadratic interpolation was chosen. The algorithm was described in C++ making use of the OCAPI framework [6]. The wordlengths are not optimized, instead, different values were used to evaluate the complexity dependency.

Two implementation alternatives were investigated based on custom accelerator datapaths. The first one (fig. 5a) is a single cycle processor with maximal parallelism. The second alternative (fig. 5b) uses two separate processors. One handles the datapath on the I and the Q with shared hardware and the other processor handles the common control data of the feedback loop.

The results are summarized in table 4. The two processor alternative yields more optimal results. However, for the shared hardware alternative, a clock of 38.88 MHz is required, so that the power consumption will be higher.

VIII. CONCLUSIONS

In this paper we demonstrated the use of full-digital symbol synchronization for a 6.48 Mbaud modem application over twisted pair media. This feedback control loop algorithm can compensate 100 ppm frequency errors under real twisted pair channel conditions, such as AWGN noise and attenuation. This variable rate algorithm can be realized with simple control logic. Furthermore, it is shown that the equalizer in the proposed architecture can be kept simple, if attenuation is the main impairment. Finally, a design implementation exploration showed the use of a synthesis framework to evaluate various architectures.

Acknowledgements

This work was carried out under the Flemisch Government Impulse Program for Information Technology in coorporation with Alcatel Telecom, Belgium. Mark Engels is a senior research assistant of the Belgian National Fund for Scientific Research.

REFERENCES


