A Modified Transposed Farrow Structure

Field of the Invention

The invention relates to a transposed Farrow structure for a receiver for a Software Defined Radio (SDR) system, a feedforward synchronizer for an SDR receiver comprising such a transposed Farrow structure and a method for processing a received signal in an SDR receiver.

Background of the Invention

In communication systems (for example Software Defined Radio (SDR)), the timing in a data receiver must be aware of the timing of the symbols of the incoming data signal. In digitally implemented modems, in some circumstances, the sampling rate can be synchronized to the symbol rate of the incoming signal, but, in other circumstances, the sampling rate cannot be synchronized to the incoming signal and, for one reason or another, the sampling clock at the receiver must remain independent of the symbol timing of the received signal.

In that case, various ways of matching the receiver sampling rate to the incoming signal have been proposed.

One known arrangement (see for example Floyd M Gardner: "Interpolation in Digital Modems – Part I: Fundamentals", *IEEE Transactions on Communications*, Vol 41, No. 3 March 1993, pp 501-507) proposes sampling the received signal at a fixed rate that is not in time with (i.e. asynchronous with) the symbol clock. After that there follows an interpolator which will output the re-sampled data in time with (i.e. synchronous with) the symbol clock. Usually, the time delay is estimated only for the asynchronous samples at the matched filter output and the interpolator has a filter with an impulse response that either precedes or follows the discrete-time matched filter. The sampling rate at the interpolator input is different from the sampling rate at the interpolator output.

166166

G00002

However, this arrangement is not ideally suited to SDR applications, where flexible re-configurability, extensive adaptation and low complexity are required.

An arrangement which provides an improvement over the digital interpolation arrangement described above has been proposed (see Fredric J Harris and Michael Rice: Multirate Digital Filters for Symbol Timing Synchronization in Software Defined Radios", *IEEE Journal on Selected Areas in Communications*, Vol. 19, No. 12, December 2002, pp 2346-2357) and provides a polyphase multi-rate filter used for symbol timing synchronization in SDR systems. In this arrangement, an up-sampled version of the matched filter is used as the interpolation filter and a polyphase decomposition of the up-sampled matched filter is used to gain implementation advantages. The advantages of this arrangement are that no separate interpolation filter is required and that maximum likelihood (ML) timing estimators can easily be incorporated into the timing recovery loop. However, the up-sampling factor of the matched filter may be too large when high accurate timing adjustment is required, and the complexity of the system increases quickly because of the storage increase for the polyphase filter's coefficients. Also, the arrangement only accommodates two functions: interpolation and matched filtering.

One known way to improve on the above known arrangements has been proposed and is named the Farrow Structure. (The Farrow Structure and the Transposed Farrow Structure, which will both be described below with reference to Figures 1 to 3, have been well described in, for example, Tim Hentschel and Gerhard Fettweis: "Continuous-Time Digital Filters for Sample-Rate Conversion in Reconfigurable Radio Terminals", Proc. Of European Wireless 2000, Sep. 12-14, Dresden, Germany, pp55-59.)

Consider an input signal $r(kT_1)$ sampled at one rate, which needs to be converted to an output signal $y(mT_2)$ sampled at a different rate. This is shown schematically in Figure 1. T_1 is the time interval between samples for the input

signal r (k T_1) (k representing the order of the sample of the input signal) and T_2 is the time interval between samples for the output signal y (m T_2) (m representing the order of the sample of the output signal).

The well-known sample rate conversion (SRC) equation relating the output signal to the input signal and describing the filtering and re-sampling of SRC for the case in Figure 1 is:

$$y(mT_2) = \sum_{k=-\infty}^{\infty} r(kT_1) \cdot h(mT_2 - kT_1) \qquad [1]$$

h(t) is the continuous-time impulse response of the required signal, which delivers the signal y(mT_2) at the new sample rate from the input signal r(kT_1) at the old sample rate. Or, putting it another way, h(t) is a function representing the action the required filter performs on the input signal to convert it to the output signal.

Equation [1] describes a time-varying system.

It should be noted that we have Rational Factor SRC if $\frac{T_1}{T_2} = \frac{L}{M}$ $with\ L, M \in N_+$.

If L = 1 or M = 1, we have Integer Factor SRC (i.e. one of the sample rates is an exact multiple of the other sample rate). If L = M = 1, we have discrete-time convolution (i.e. no SRC at all).

Generally, the complete continuous-time impulse response h(t) must be known. For rational and integer value SRC, described above, the system described by equation [1] varies periodically with time. So, only certain values of h(t) are actually required for the computation.

In order to simplify calculating the samples of h(t) which are required, we look for simple functions describing continuous time impulse responses. It has been found that, since polynomials are such simple functions, polynomial filters are useful in SRC. We limit the class of polynomial filters to piecewise polynomial impulse responses composed from pieces of equal length. Given polynomial pieces of degree Q and length Δ, we may derive an expression for h(t) as follows:

$$h(t)=\sum_{q=0}^{Q} c_q\left(\left\lfloor \frac{t}{\Delta}\right\rfloor\right)\cdot\left(\frac{t}{\Delta}-\left\lfloor \frac{t}{\Delta}\right\rfloor\right)^q, \quad 0\le t<N\Delta \qquad [2]$$

Where c_q is the coefficient for the q-th order polynomial.

There are two choices for Δ which simplify equation [2], namely $\Delta = T_1$ and $\Delta = T_2$. Using equation [2] and equation [1] above and setting $\Delta = T_1$ gives:

$$y(mT_2)=\sum_{q=0}^{Q} x_q(mT_2)\cdot(\mu_m)^q \qquad [3]$$

where

$$x_q(mT_2)=\sum_{k=-\infty}^{\infty} r(kT_1)\cdot c_q\left(\left\lfloor m\frac{T_2}{T_1}\right\rfloor-k\right) \qquad [4]$$

and

$$\mu_m = \frac{mT_2}{T_1} - \left\lfloor \frac{mT_2}{T_1} \right\rfloor \in [0,1) \qquad [5]$$

$\lfloor a \rfloor$ denotes the floor operation i.e. $\lfloor a \rfloor$ denotes the greatest integer a' such that a' ≤ a.

μ_m is the inter-sample position i.e. the distance between the previous input signal and the current output signal (see Figure 1) and has no units.

The above equations [3], [4] and [5] represent an implementation of SRC known as the Farrow structure and Figure 2 shows a schematic illustration of such a structure.

The Farrow structure shown in Figure 2 can also be interpreted as a polyphase interpolator with an infinite number of polyphase branches that are implemented by just one "reference" polyphase branch, and a polynomial description of how to calculate the remaining ones. The "reference" polyphase branch can be obtained by setting $\mu_m = 0$ in equation [3]. It represents the samples of the impulse response h(t) which mark the starting points of the individual polynomial pieces. These starting points (i.e. the "reference" branch) and the order of the connecting polynomial pieces determine the transfer characteristics of the SRC system.

In the Farrow structure equations [3], [4] and [5] above, the higher the order of the polynomial pieces, the better the impulse response h(t) can be matched to the application. If high order polynomials are not feasible, it is also possible to use shorter polynomials of lower order. In that case, more "reference" polyphase branches (i.e. starting points of polynomial pieces) are required. This can be achieved by decreasing the length of the polynomial pieces by a factor J, that is:

$$\Delta = \frac{T_1}{J} \qquad [6]$$

This may be thought of as a generalization of the Farrow structure so is commonly known as the generalized Farrow structure (see for example R W Schafer and L R Rabiner: "A Digital Signal Processing Approach to Interpolation", *Proceedings of the IEEE*, 61(6):692-702, June 1973).

Since equation [6] provides us with a generalization compared with the case where $\Delta = T_1$, we can skip the case where $\Delta = T_2$ and, instead, immediately set

$$\Delta = \frac{T_2}{J} \qquad [7]$$

Substituting equations [6] and [7] into equation [2] above, we have:

$$y(mT_2) = \sum_{q=0}^{Q} \sum_{k=-\infty}^{\infty} x_q(kT_1) \cdot c_q\left(mJ - \left\lceil kJ \frac{T_1}{T_2} \right\rceil \right) \qquad [8]$$

where

$$x_q(kT_1) = r(kT_1) \cdot (\mu_k)^q \qquad [9]$$

and

$$\mu_k = \left\lceil kJ \frac{T_1}{T_2} \right\rceil - kJ \frac{T_1}{T_2} \in [0,1) \qquad [10]$$

Note that when J = 1, we return to the original Farrow structure and μ_k becomes equal to μ_m, the previously defined inter-sample position.

Equations [8], [9] and [10] above describe the Farrow structure when the piece length Δ of the polynomial pieces is defined by equation [7] above, and is known as the Transposed Farrow Structure (TFS). Figure 3 shows a schematic illustration of such a TFS for J=1. Note that, in a real implementation, J is always equal to 1.

The Farrow Structure (FS) performs well in the case of sample rate increase but sample rate decrease is more commonly required in the receiver design. The Transposed Farrow Structure (TFS) provides an improvement over the FS, as it performs well for sample rate decrease, but the effectiveness of the TFS could still be improved.

It is an object of the invention to provide a modified Transposed Farrow Structure and method which reduces or substantially avoids the problems of known arrangements described above.

Summary of the Invention

According to a first aspect of the invention, there is provided a transposed Farrow structure for a Software Defined Radio (SDR) receiver, the transposed Farrow structure being arranged to implement:

sample rate conversion (SRC) for converting a received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$;

timing adjustment using an estimated timing error τ; and

matched filtering of the received signal.

Preferably, the transposed Farrow structure equates the signal y to Q polynomial pieces.

In one preferred embodiment, the transposed Farrow structure comprises a first part for generating the Q polynomial pieces; and a second part for summing up the Q polynomial pieces to produce signal y.

The first part of the transposed Farrow structure may comprise a multiplier and plus network for generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

The first part of the transposed Farrow structure may further comprise an integration and dump circuit for summing the Q polynomials generated by the multiplier and plus network over all orders k of the received signal r to produce the Q polynomial pieces of the signal y,
the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

The second part of the transposed Farrow structure may comprise a Finite Impulse Response Filter for summing up the Q polynomial pieces generated by the integration and dump circuit to produce signal y, the duration of the summation in the Finite Impulse Response Filter being controlled by the overflow factor.

The transposed Farrow structure may further comprise a unit for generating a timing parameter μ from T_1, T_2 and τ.

The transposed Farrow structure may further comprise a unit for generating an overflow factor from T_1, T_2 and τ.

According to a second aspect of the invention, there is provided a feedforward synchronizer for a Software Defined Radio (SDR) receiver, the feedforward synchronizer being arranged to receive a signal r having a sampling rate of $\frac{1}{T_1}$ and comprising:

a feedforward estimator for generating an estimated timing error τ ; and

a transposed Farrow structure arranged to receive the signal r and the estimated timing error τ and being arranged to implement: a) sample rate conversion (SRC) for converting the received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$; b) timing adjustment using the estimated timing error τ ; and c) matched filtering of the signal.

Preferably, the transposed Farrow structure equates the signal y to Q polynomial pieces.

In one embodiment, the transposed Farrow structure comprises a first part for generating the Q polynomial pieces; and a second part for summing up the Q polynomial pieces to produce signal y.

The first part of the transposed Farrow structure may comprise a multiplier and plus network for generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ .

The first part of the transposed Farrow structure may further comprise an integration and dump circuit for summing the Q polynomials generated by the multiplier and plus network over all orders k of the received signal r to produce the Q polynomial pieces of the signal y,

the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

The second part of the transposed Farrow structure may comprise a Finite Impulse Response Filter for summing up the Q polynomial pieces generated by the integration and dump circuit to produce signal y, the duration of the summation in the Finite Impulse Response Filter being controlled by the overflow factor.

The transposed Farrow structure preferably comprises a portion for generating a timing parameter μ from T_1, T_2 and τ.

The transposed Farrow structure preferably comprises a portion for generating an overflow factor from T_1, T_2 and τ.

The feedforward estimator may comprise a maximum likelihood based estimator.

The feedforward estimator may comprise an unwrapped estimator for compensating for clock offset.

The feedforward estimator may comprise a differentiator.

The feedforward synchronizer may further comprising an anti-aliasing filter.

According to a third aspect of the invention, there is provided a method for processing a received signal r having a sampling rate of $\frac{1}{T_1}$, in a Software Defined Radio (SDR) receiver comprising a transposed Farrow structure, the method comprising the steps of:

performing, in the transposed Farrow structure, sample rate conversion (SRC) for converting received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$;

performing, in the transposed Farrow structure, timing adjustment using an estimated timing error τ; and

performing, in the transposed Farrow structure, matched filtering of the received signal.

Preferably, the transposed Farrow structure equates the signal y to Q polynomial pieces.

The method may comprise the steps of generating the Q polynomial pieces; and summing up the Q polynomial pieces to produce signal y.

The step of generating the Q polynomial pieces may comprise generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

The step of generating the Q polynomial pieces may further comprise summing the Q polynomials over all orders k of the received signal r to produce the Q polynomial pieces of the signal y, the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

The step of summing up the Q polynomial pieces to produce signal y is preferably controlled by the overflow factor which controls the duration of the summation.

The method may further comprise the step of generating a timing parameter μ from T_1, T_2 and τ.

The method may further comprise the step of generating an overflow factor from T_1, T_2 and τ.

Features described in relation to one aspect of the invention may be applicable to another aspect of the invention.

Brief Description of the Drawings

Known arrangements have already been described with reference to Figures 1 to 3, of which:

Figure 1 is a schematic diagram illustrating the relationship between input and output sampling rates;

Figure 2 is a diagram of the Farrow structure described by equations [3], [4] and [5]; and

Figure 3 is a diagram of the transposed Farrow structure described by equations [8], [9] and [10].

Exemplary embodiments of the invention will now be described with reference to Figures 4 to 11, of which:

Figure 4 is a diagram of a modified transposed Farrow structure (TFS) according to the invention;

Figure 5 is a diagram of a feedforward synchronizer incorporating the modified TFS of Figure 4;

Figure 6 is a diagram of the feedforward estimator of Figure 5;

Figure 7 is a diagram of the unwrapped estimator of Figure 5;

Figure 8 is a diagram of the differentiator of Figure 5;

Figure 9 is a BER performance comparison of the arrangement of Figure 5 with known arrangements;

Figure 10 shows the estimated timing error in the presence of timing phase error and clock offset; and

Figure 11 shows the constellation of π/4-DQPSK before and after timing synchronization.

Detailed Description of Preferred Embodiments

The invention provides a modified Transposed Farrow Structure (TFS). From equation [8], we have:

$$y(mT_2)=\sum_{q=0}^{Q}\sum_{k=-\infty}^{\infty}x_q(kT_1)\cdot c_q\left(\left\lfloor m-k\frac{T_1}{T_2}\right\rfloor\right) \qquad [11]$$

As before, $y(mT_2)$ is the output and c_q is the coefficient for the q-th order (q running from 0 to Q). As before:

$$x_q(kT_1)=r(kT_1)\cdot(\mu_k)^q \qquad [10]$$

To simplify notation, we set:

$$\mu_k=2\mu_k'-1 \qquad [12]$$

which gives

$$x_q(kT_1)=r(kT_1)\cdot(2\mu_k'-1)^q \qquad [13]$$

and

$$\mu_k' = m - \frac{kT_1}{T_2} - \left\lfloor m - \frac{kT_1}{T_2} \right\rfloor \qquad [14]$$

h(t), the function representing the action of the TFS is given by

$$h(mT_2 - kT_1) = \sum_{q=0}^{Q} c_q \left(\left\lfloor m - k\frac{T_1}{T_2} \right\rfloor \right) \left(2\left(m - k\frac{T_1}{T_2} - \left\lfloor m - k\frac{T_1}{T_2} \right\rfloor \right) - 1 \right)^q \qquad [15]$$

or more simply as

$$h(mT_2 - kT_1) = \sum_{q=0}^{Q} c_q \left(\left\lfloor m - k\frac{T_1}{T_2} \right\rfloor \right) (2\mu_k' - 1)^q \qquad [16]$$

Figure 4 shows implementation of the modified TFS according to one embodiment of the invention. The modified TFS structure includes a position sequence generator 401 receiving the SRC ratio $\frac{T_1}{T_2}$ and the timing error $\tau(k)$. Also included is a multiplier and plus network 403 receiving μ_k (= $2\mu_k'$ – 1) from the position sequence generator 401 and also receiving the input signal r(k). Also included are Integration and Dump (I&D) circuits 405, just as in the TFS of Figure 3. Finally, there is a delay and adder unit – finite impulse response (FIR) filter 407, producing the output y(m). ov(m) is the overflow signal and will be discussed further below.

The modified TFS of Figure 4 is used to implement the equation:

$$y(mT_2)=\sum_{q=0}^{Q}\sum_{k=-\infty}^{\infty}x_q(kT_1)\cdot c_q\left(m-\left\lceil k\frac{T_1}{T_2}\right\rceil\right)$$

which is equation [8] with J=1. Essentially, the multiplier and plus network generates the $x_q(kT_1)\cdot c_q\left(m-\left\lceil k\frac{T_1}{T_2}\right\rceil\right)$, I&D circuits perform the inner summation over k and the FIR filter performs the outer summation over q.

In the multiplier and plus network 403, x_q for q = 0 to Q is implemented in accordance with equation [13] from inputs r(k) and μ_k. Then, Q multipliers

$$x_q(kT_1)c_q\left(\left\lfloor m-k\frac{T_1}{T_2}\right\rfloor\right)$$

are implemented for each input r(k). These are the outputs of the multiplier and plus network 403. Then, in the duration that $\left(\left\lfloor m-k\frac{T_1}{T_2}\right\rfloor\right)$ remains the same, the I&D circuits 405 perform the summation over k for each output from the multiplier and plus network 403. The FIR filter performs the summation over q thereby producing the left hand side of equation [8] i.e. y(m) as its output.

A tunable parameter for the modified TFS of Figure 4 is μ_k' and this is determined by the SRC ratio $\frac{T_1}{T_2}$ and derived at 401. The estimated timing error $\tau(k)$ is also used to control μ_k' as can be seen at 401. By regenerating tunable parameter μ_k' to integrate the timing error $\tau(k)$ as well as the SRC ratio $\frac{T_1}{T_2}$, the modified TFS can combine the functions of SRC, matched filtering and timing adjustment without a huge increase in complexity.

However, as k and m (i.e. the orders of the sample of the input and output signals) increase, the interpolation time grows unboundedly so the computation accuracy will soon deteriorate and the interpolation control will fail. Thus, a more suitable, but equivalent, computational method is needed. One way to do this is to use two separate timing parameters: the fractional interval μ_k and the basepoint index n_k. In that case, we have:

$$n_{k+1} = n_k + \text{int}\left[\mu_k + \frac{T_1}{T_2} + \tau(k)\right] \qquad [17]$$

$$\mu_{k+1} = frac\left[\mu_k + \frac{T_1}{T_2} + \tau(k)\right]\mu \qquad [18]$$

Thus, the basepoint index at a particular order depends on the basepoint index at the previous order, the fractional interval at the previous order, the timing error at the previous order and the SRC ratio. The fractional interval at a particular order only depends on the fractional interval at the previous order, the timing error at the previous order and the SRC ratio. We can regard $\text{int}\left[\mu_k + \frac{T_1}{T_2} + \tau(k)\right]$ in equation [17] as the number of input samples to be shifted into the interpolator until the next interpolant is computed.

In this embodiment, we introduce a new overflow accumulator, index_k where

$$index_{k+1} = \left(index_k + \frac{T_1}{T_2} + \tau(k)\right) \bmod 1 \qquad [19]$$

Note that, in equation [19] we have included the timing error $\tau(k)$so that we can combine the three functions: SRC, matched filtering and timing adjustment. This

is clear from the fact that the SRC factor, the timing error and the overflow accumulator for the previous order are included in the definition of the overflow accumulator. We could define $index_{k+1}$ in another way by removing $\tau(k)$ from equation [19] so that only SRC and matched filtering are combined.

Equation [19] means that the sequence $index_k$ is always < 1 for k = 0,..., . The overflow signal ov(m) which was mentioned previously (one of the 401 outputs in Figure 4) is defined as:

$$ov(m) = index_k + \frac{T_1}{T_2} + \tau(k) \geq 1 \qquad [20]$$

As already mentioned, the I&D circuits 405 in Figure 4 perform the summation of the multiplier and plus network outputs over k. The duration of the summation is controlled by the overflow signal. The integration operation continues until the overflow occurs. Then the dump operation is performed. Moreover, the overflow signal also controls the timing when the output y(m) is generated. Accurate overflow signal generation is the key for a successful TFS implementation.

Equations 19 and 20 give us:

$$\mu_k' = 1 - index_k \qquad [21]$$

These new definitions allow the modified TFS to actually be implemented in hardware and the method is applicable to both rational and non-rational SRC systems integrated with timing adjustment interpolation. μ_k' is tunable and incorporates the estimated timing error τ. The range of μ_k' is defined as (0,1] rather than [0,1) as in conventional methods. The overflow condition is also changed. In the described embodiment, two separate steps are employed, to control the overflow accumulator $index_k$ and generate inter-sample position μ_k' respectively.

Figure 5 shows a timing recovery loop incorporating the modified TFS of Figure 4. Figure 5 illustrates a feedforward synchronizer 500 comprising anti-aliasing filter 501, feedforward estimator 503, unwrapped estimator 505, differentiator 507 and modified TFS (according to Figure 4) 509. The output of the differentiator 507 is the timing error estimate $\tau(k)$ which, as described above, is used, along with the input signal r(k) as input to the modified TFS 509. The output of the modified TFS 509 is y(m). (Note that, for baseband, the AAF is optional.)

Timing synchronization can be classified into two categories: feedback and feedforward. A feedforward synchronizer is preferred and described since feedback synchronization is not ideal for bursty transmission and can also face problems of false locking and locking losing in a poor channel condition.

Referring to Figure 5, the received signal is fed to anti-aliasing filter (AAF) 501 and is then sampled at some rate $\frac{1}{T_1}$. The samples from the AAF then input the feedforward estimator 503. The feedforward estimator 503 is shown in detail in Figure 6.

Figure 6 shows a Maximum-Likelihood (ML) based estimator. The ML based estimator and the O&M method (as described in, for example, Umberto Mengali and N. D'Andrea, "Synchronization Techniques for Digital Receivers (Applications of Communications Theory)", *Plenum Publishing Corporation*, 1997) are two known non-data aided feedforward estimators which are essentially equivalent in performance. In this embodiment, a ML-based estimator is used.

In the estimator of Figure 6, the samples from AAF 501 follow two separate branches. In the upper branch, the samples are first complex conjugated at 601.

Then, the samples are multiplied by $e^{\frac{-j2\pi k}{N}}$ at operator block 603. Then, the samples are filtered at filter 605. In the lower branch, the samples are multiplied by $e^{\frac{-j2\pi k}{N}}$ at operator block 607 then delayed at delay block 609.

The outputs from the two branches are then multiplied together at operator block 611 and then the product is accumulated at block 613. The argument of the accumulator output gives the timing estimate within a factor of $\frac{T}{2\pi}$.

Alternatively, we could use an O&M estimator instead of a ML-based estimator. In that case:

$$\hat{\tau} = \frac{-T}{2\pi}\arg\left\{\sum_{k=0}^{NL_0-1}\left|x(kT_1)\right|^2 e^{\frac{-j2\pi k}{N}}\right\}$$

Where $x(KT_1)$ are samples from the matched filter. The O&M algorithm seems more simple to implement than the ML-based scheme as it does not involve any filtering of $x(kT_1)$. However, an oversampling of 4 is needed in an O&M estimator, whereas oversampling of 2 is sufficient with the ML-based method. In practice, the ML-based scheme is actually not as complex as expected, because the digitized signal generally comes from an efficient AAF followed by an analogue to digital converter.

Referring again to Figure 5, the output of the estimator 503 is fed into unwrapped estimator 505. The unwrapped estimator is shown in detail in Figure 7. The purpose of the unwrapped estimator is to solve the problem caused by large clock offset. The timing error $\tau(k)$ varies in time due to the clock offset between the transmitter and the receiver. As $\hat{\tau}$ (i.e. the output of estimator 503 and input

to unwrapped estimator 505) is restricted to $-\frac{T_s}{2} \Rightarrow \frac{T_s}{2}$, it occasionally exhibits jumps of Ts seconds by jumping from one clock to another. If this is not properly accounted for, some strobes will be missed or duplicated. To solve this, the estimates $\hat{\tau}$ must be "unwrapped" – see Umberto Mengali and N. D'Andrea, "Synchronization Techniques for Digital Receivers (Applications of Communications Theory)", *Plenum Publishing Corporation*, 1997.

Referring again to Figure 5, the output of the unwrapped estimator 505 is fed into the differentiator 507. The differentiator is shown in detail in Figure 8. Because the estimated timing error is accumulated at the frequency of the input clock and the feedforward synchronization is an open loop, a differentiator of the timing estimate is needed between the feedforward estimator (503, 505) and the modified TS (509) to counteract the accumulation effect.

Experimental data using the feedforward synchronizer of the present invention are shown in Figures 9, 10 and 11. A baseband transceiver is used with π/4 – DQPSK modulation.

Figure 9 shows a comparison of the BER of the synchronization scheme of the invention with a traditional feedback synchronization scheme. The conventional feedback scheme adopts the early-late gate timing estimator with an oversample factor of 4, a separate matched filter and an interpolation filter (the Farrow structure). In fact, that scheme has one of the best performances of any feedback scheme. It can be seen that the proposed feedforward scheme has equivalent performance as the traditional feedback method.

Figure 10 shows the estimated timing error with timing phase error (left hand side) and with timing phase error and clock offset (right hand side). It can be seen that the proposed timing synchronization scheme can estimate the timing

error with good accuracy and, at the same time, it can also track the clock frequency offset, which causes the phase to increase or decreases linearly.

Figure 11 shows the π/4 – DQPSK constellation when both timing phase error and clock offset are introduced into the system. The left hand side shows before timing synchronization and it can be seen that the constellation is not discernible. The right hand side shows the constellation with the timing synchronization of the invention; here a clearly distributed constellation is visible. There is a marked improvement to the constellation when the proposed scheme is adopted.

Claims:

1. A transposed Farrow structure for a Software Defined Radio (SDR) receiver, the transposed Farrow structure being arranged to implement:

sample rate conversion (SRC) for converting a received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$;

timing adjustment using an estimated timing error τ; and

matched filtering of the received signal.

2. A transposed Farrow structure according to claim 1, wherein the transposed Farrow structure equates the signal y to Q polynomial pieces.

3. A transposed Farrow structure according to claim 2 comprising:

a first part for generating the Q polynomial pieces; and

a second part for summing up the Q polynomial pieces to produce signal y.

4. A transposed Farrow structure according to claim 3 wherein the first part comprises:

a multiplier and plus network for generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

5. A transposed Farrow structure according to claim 4 wherein the first part further comprises:

an integration and dump circuit for summing the Q polynomials generated by the multiplier and plus network over all orders k of the received signal r to produce the Q polynomial pieces of the signal y,

the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

6. A transposed Farrow structure according to claim 5 wherein the second part comprises:

a Finite Impulse Response Filter for summing up the Q polynomial pieces generated by the integration and dump circuit to produce signal y,
the duration of the summation in the Finite Impulse Response Filter being controlled by the overflow factor.

7. A transposed Farrow structure according to any one of the preceding claims comprising a unit for generating a timing parameter μ from T_1, T_2 and τ.

8. A transposed Farrow structure according to any one of the preceding claims comprising a unit for generating an overflow factor from T_1, T_2 and τ.

9. A feedforward synchronizer for a Software Defined Radio (SDR) receiver, the feedforward synchronizer being arranged to receive a signal r having a sampling rate of $\frac{1}{T_1}$ and comprising:

a feedforward estimator for generating an estimated timing error τ ; and

a transposed Farrow structure arranged to receive the signal r and the estimated timing error τ and being arranged to implement: a) sample rate conversion (SRC) for converting the received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$; b) timing adjustment using the estimated timing error τ ; and c) matched filtering of the signal.

10. A feedforward synchronizer according to claim 9, wherein the transposed Farrow structure equates the signal y to Q polynomial pieces.

11. A feedforward synchronizer according to claim 10 wherein the transposed Farrow structure comprises:

a first part for generating the Q polynomial pieces; and

a second part for summing up the Q polynomial pieces to produce signal y.

12. A feedforward synchronizer according to claim 11 wherein the first part of the transposed Farrow structure comprises:

a multiplier and plus network for generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

13. A feedforward synchronizer according to claim 12 wherein the first part of the transposed Farrow structure further comprises:

an integration and dump circuit for summing the Q polynomials generated by the multiplier and plus network over all orders k of the received signal r to produce the Q polynomial pieces of the signal y,

the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

14. A feedforward synchronizer according to claim 13 wherein the second part of the transposed Farrow structure comprises:

a Finite Impulse Response Filter for summing up the Q polynomial pieces generated by the integration and dump circuit to produce signal y,

the duration of the summation in the Finite Impulse Response Filter being controlled by the overflow factor.

15. A feedforward synchronizer according to any one of claims 9 to 14 wherein the transposed Farrow structure comprises a portion for generating a timing parameter μ from T_1, T_2 and τ.

16. A feedforward synchronizer according to any one of claims 9 to 15 wherein the transposed Farrow structure comprises a portion for generating an overflow factor from T_1, T_2 and τ.

17. A feedforward synchronizer according to any one of claims 9 to 16 wherein the feedforward estimator comprises a maximum likelihood based estimator.

18. A feedforward synchronizer according to any one of claims 9 to 17 wherein the feedforward estimator comprises an unwrapped estimator for compensating for clock offset.

19. A feedforward synchronizer according to any one of claims 9 to 18 wherein the feedforward estimator comprises a differentiator.

20. A feedforward synchronizer according to any one of claims 9 to 19 further comprising an anti-aliasing filter.

21. A method for processing a received signal r having a sampling rate of $\frac{1}{T_1}$, in a Software Defined Radio (SDR) receiver comprising a transposed Farrow structure, the method comprising the steps of:

performing, in the transposed Farrow structure, sample rate conversion (SRC) for converting received signal r having a sampling rate of $\frac{1}{T_1}$ to a transmitted signal y having a sampling rate of $\frac{1}{T_2}$;

performing, in the transposed Farrow structure, timing adjustment using an estimated timing error τ; and

performing, in the transposed Farrow structure, matched filtering of the received signal.

22. A method according to claim 21, wherein the transposed Farrow structure equates the signal y to Q polynomial pieces.

23. A method according to claim 22 comprising the steps of:

generating the Q polynomial pieces; and

summing up the Q polynomial pieces to produce signal y.

24. A method according to claim 23 wherein the step of generating the Q polynomial pieces comprises:

generating Q polynomials for each order k of the received signal r, the Q polynomials being dependent on a timing parameter μ and the received signal r, the timing parameter μ being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

25. A method according to claim 24 wherein the step of generating the Q polynomial pieces further comprises:

summing the Q polynomials over all orders k of the received signal r to produce the Q polynomial pieces of the signal y,

the duration of the summation in the integration and dump circuit being controlled by an overflow factor, the overflow factor being dependent on SRC factor $\frac{T_1}{T_2}$ and the timing error τ.

26. A method according to claim 25 wherein the step of summing up the Q polynomial pieces to produce signal y is controlled by the overflow factor which controls the duration of the summation.

27. A method according to any one of claims 21 to 26 further comprising the step of generating a timing parameter μ from T_1, T_2 and τ.

28. A method according to any one of claims 21 to 27 further comprising the step of generating an overflow factor from T_1, T_2 and τ.