Timing Synchronization

Fuyun Ling

fling@twinclouds.com

Outline

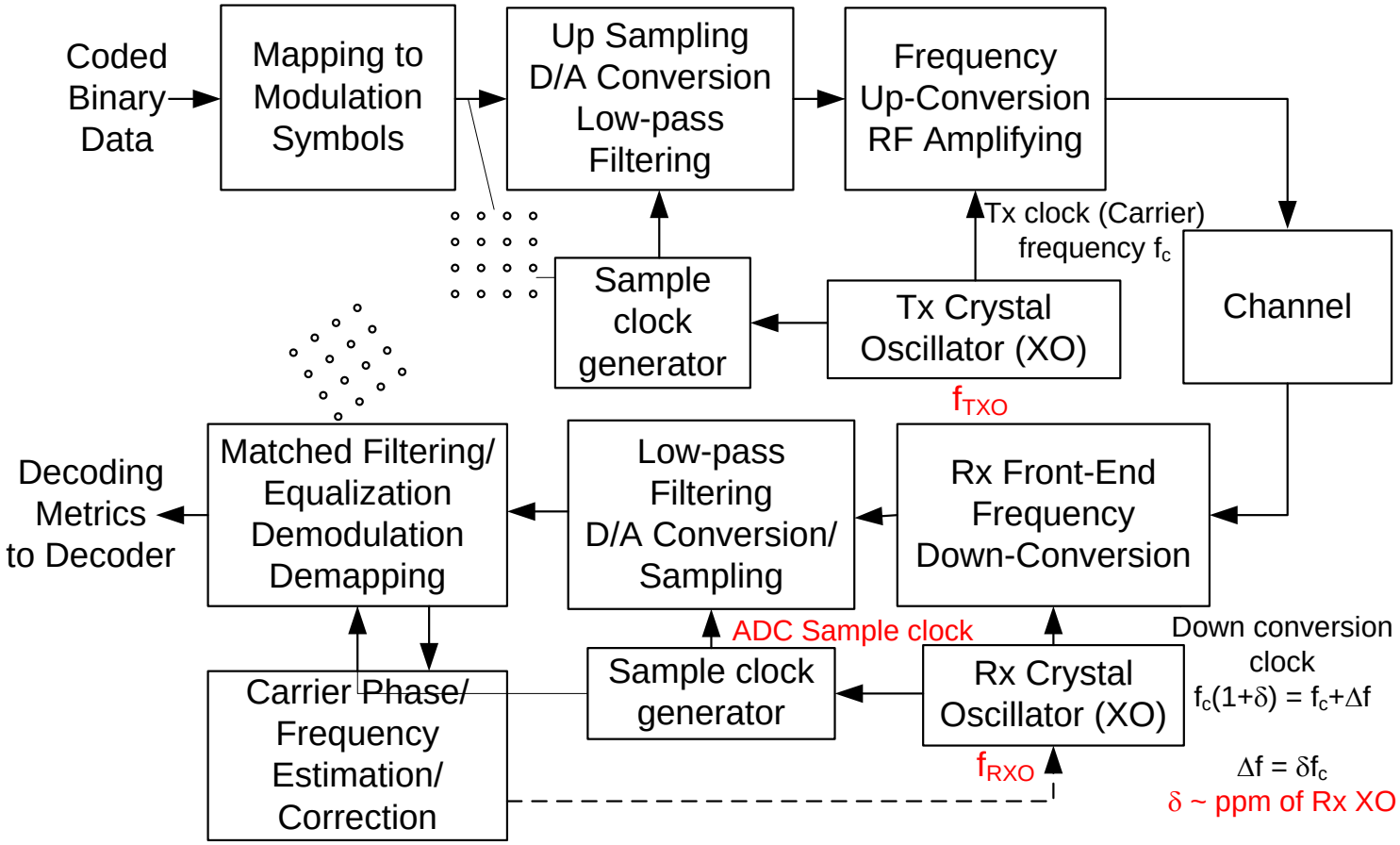
- Overview
- Classic timing synchronization techniques – non-data assisted
  - Squarer Based Timing Recovery
  - Early/Late Gate Timing Recovery
  - Gardener's Algorithm
- Data assisted timing synchronization techniques
  - Mueller-Müller algorithm
  - CDMA Early/Late Gate DLL
  - CIR/EQ Estimates Based Timing synchronization
  - Timing Synchronization in OFDM systems
- Timing adjustments using digital interpolation

fling@twinclouds.com

OVERVIEW OF TIMING SYNCHRONIZATION

fling@twinclouds.com

Block Diagram of a Digital Communication System



fling@twinclouds.com

Continuous time Signal Model

- The passband signal at the receiver frontend:

$$r_{pass}(t) = \mathrm{Re}\left[e^{j2\pi f_c t} \sum_{k=-\infty}^{\infty} a_k h(t - kT - \tau) \right] + z(t)$$

  - a_k: modulation data symbol
  - $h(t)$ overall channel impulse response (CIR)
  - $z(t)$ additive noise
  - T: Tx symbol time interval or baud interval
    - 1/T is the Tx symbol rate, or baud rate
  - τ : time delay introduced in transmission
  - f_c : carrier frequency

fling@twinclouds.com

Continuous time Signal Model (cont.)

- The received baseband signal can be expressed as:

$$r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT - \tau) + z(t)$$

- Here we assume:
  - Perfect carrier synchronization,
  - $\tau < \mathrm{T}$
- The continuous received signal is sampled to generate digital samples for processing
  - The sampling rate may be equal to, or higher than, the symbol rate $1/\mathrm{T}$

fling@twinclouds.com

Objectives of Timing Synchronization (Timing Recovery)

- To achieve best receiver performance, it is necessary:
  (1) the receiver digital sampling frequency need to be synchronous to the Tx symbol rate (timing frequency synchronization)
    - Receiver sampling rate is usually equal to 1/T, m/T or (m/n)/T
  (2) The receiver sampling need to occur at the right time instant (timing phase) relative to the symbol waveform
    - The sampling phase need to be stable
    - It may need to be adjusted (from time to time) to achieve best receiver performance
- It is necessary to achieve both timing *frequency* and *phase* synchronization
  – Timing synchronization is also called timing clock recovery. (We shall use these two terms interchangeably.)

fling@twinclouds.com

Timing Synch Realizations

- In general, a second order loop can be used to achieve both timing frequency and phase synchronization
  - It is called timing locked loop (TLL), timing control loop (TCL), or delay locked loop (DLL)
  - The loop is driven by timing phase error (the difference between the sampling phase and the desired timing phase)
- Timing frequency and phase synchronizations can be jointly implemented but may be optimized independently
  - The objective of timing frequency synch is to track long term average Tx sampling rate
  - Timing phase may need to be adjusted based on short time channel change or time skipping due to, e.g., entering and exiting receiver sleep state
    - Such adjustments should not cause sudden timing frequency change

fling@twinclouds.com

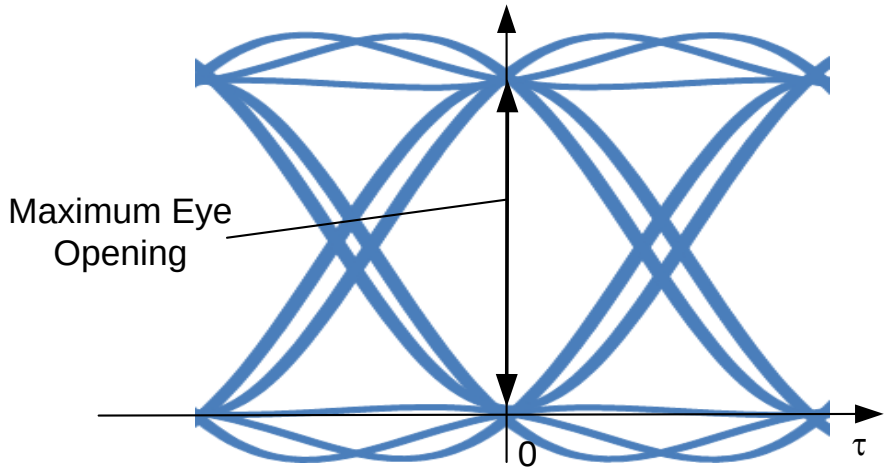
Timing Synch Realizations (cont.)

- Relationship between timing and carrier synchronizations
  - Carrier frequency offset is caused by the differential between modulation and demodulation frequency references
  - Timing frequency offset is caused by the difference between the Tx symbol generating clock and Rx sampling clock frequencies
  - Recovered carrier frequency clock in the receiver can be used for timing clock generation if the same frequency reference is used for demodulation and sampling clock generation
    - Here we assume the transmitter modulation and symbol rate clock are always accurate and locked with each other
  - If different frequency references are used for demodulation and sampling, timing clock recovery need to be performed separately from the carrier clock recovery
  - In all digital implementation, carrier and timing adjustments are usually performed independently

fling@twinclouds.com

Optimal Timing Phase Selection

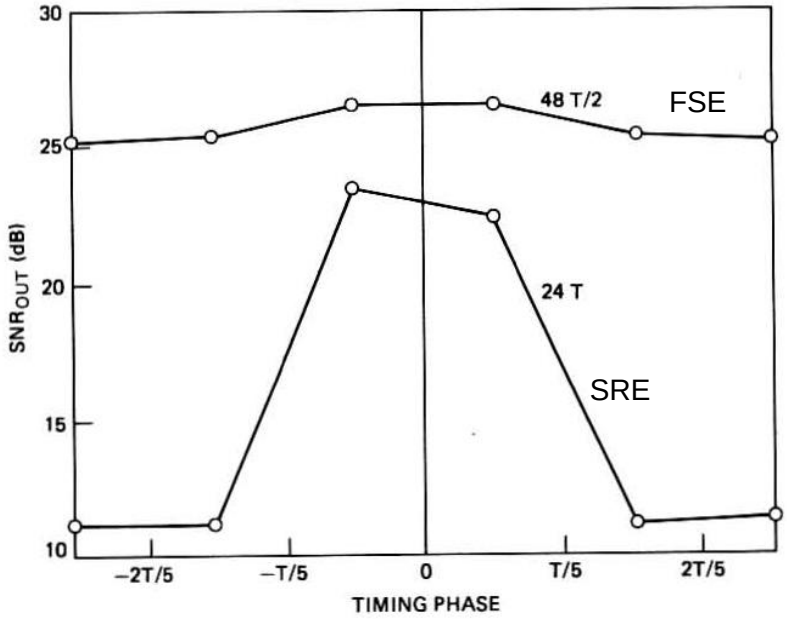
- For a single path AWGN channel with optimal receiver
  - The overall channel impulse response satisfied Nyquist criterion, i.e., there's no intersymbol Interference (ISI) with right sampling timing
  - Overall channel impulse response has, e.g., raised cosine waveform
  - The best receiver performance can be achieved if the timing of sampling is at $\tau = 0$ (maximum eye opening)



fling@twinclouds.com

Optimal timing phase selection (cont.)

- For ISI channels, equalizers are needed for good reception performance
  - Performance of symbol rate EQ (SRE) depends on sampling phase
  - Performance of fractional spaced EQ (FSE) is insensitive to sampling phase
  - SRE need precise sampling timing phase to achieve good receiver performance
    - Optimal timing phase difficult to determine in real time
  - FSE only need stable relative timing, i.e., accurate sampling frequency
    - FSE simplifies timing synch implementation



From Gitlin and Weinstein, BSTJ, Feb. 1981

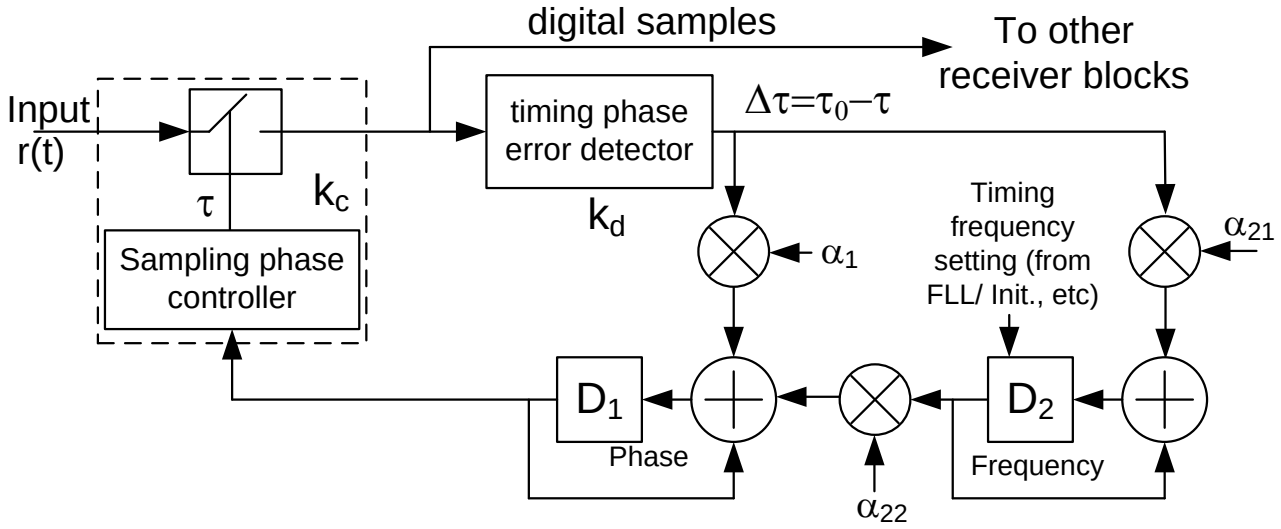
fling@twinclouds.com

Optimal timing phase selection (cont.)

- Above results are for single carrier receiver in general
  - The sampling rate of SRE does not meet the Nyquist criterion of sampling
  - Sampling rate of FSE satisfies the Nyquist criterion
- With proper guard-carriers the OFDM receiver sampling rate (equal to OFDM chip rate) satisfies Nyquist criterion
  - No subchip sampling adjustment is necessary
- The timing phase selection is equivalent to optimal FFT window (the sample block for performing FFT) placement
  - The FFT window should contain one complete period of the data convolve with channel or contains most of the data energy
  - The optimal placement may not be unique

fling@twinclouds.com

A Typical Digital Timing Locked Loop



- Operations of a second order timing locked loop (TLL)
  - The input r(t) is sampled at $r(nT-\tau)$ according to the timing phase value contained in the register D_1.
  - The sample phase is compared to the desired timing phase τ_0 by phase error detector to generate phase error $\Delta\tau$ with a phase unit, e.g., T
    - The phase error detector has a gain k_d (number unit/phase unit)
  - The phase error is scaled by coefficients α_1 and α_2 and fed to the perfect integrators with registers D_1 and D_2

fling@twinclouds.com

A Typical Digital Timing Locked Loop (cont.)

- Operations (cont.)
  - The register D_2 contains the (scaled) timing frequency offset value
    - D_2 can be set, reset or modified by external circuitries, e.g.,
      - initial acquisition block for setting initial frequency value
      - carrier synchronization block for estimated local oscillator frequency offset
  - The register D_1 contains the timing phase offset value (in unit of, or proportional to, T)
    - The scaled value of D_2 is added to D_1 every T (constant phase increment due to frequency offset)
  - The changed D_1 value modifies the timing sampling phase by the sampling phase controller and thus complete the loop
    - The sampling phase controller has a gain k_c (phase unit/number unit)
  - TLL is a case of PLL
    - The first and second order gains are $k_1=\alpha_1 k_c k_d$ and $k_2=\alpha_{21}\alpha_{22} k_c k_d$, respectively

fling@twinclouds.com

A Typical Digital Timing Locked Loop (cont.)

- The linearized system equation is:

$$\frac{\tau(z)}{\tau_0(z)} = \frac{(k_1 - k_2)z - k_1}{1 - (2 + k_1)z + (1 + k_1 + k_2)z^2}$$

  - Using time impulse invariance mapping z = (1-s)$^{-1}$ we convert the system equation to s-domain:

$$\frac{\tau(s)}{\tau_0(s)} = \frac{-k_2 + (k_2 + k_1)s - k_1 s^2}{s^2 + k_1 s + k_2}$$

  - The two roots are: $\sqrt{}$
  - System is stable as long as $k_1 > C$
  - System critical damped if [redacted]
  - System under damped (oscillating when converging) if $k_1 < 2\sqrt{k_2}$
  - Denominator in standard second order linear system form:

$$s^2 + k_1 s + k_2 \cong s^2 + 2s\zeta\omega_n + \omega_n^2$$

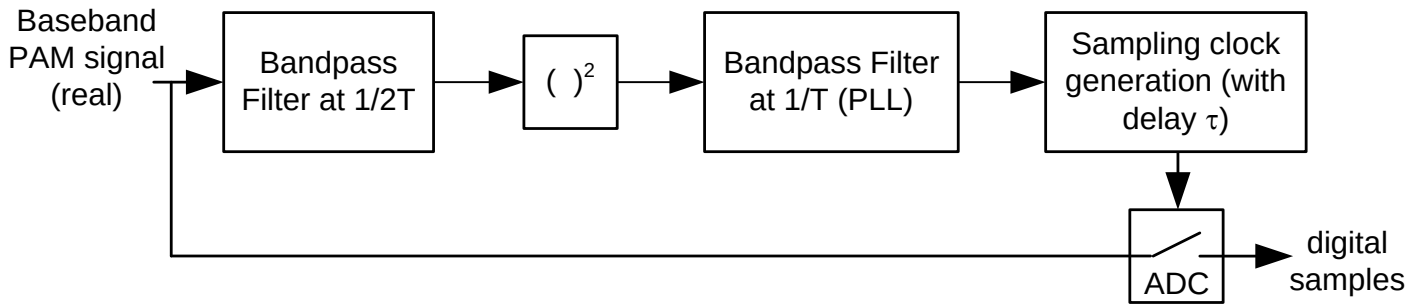
fling@twinclouds.com

CLASSIC TIMING SYNCHRONIZATION TECHNIQUES – NON-DATA ASSISTED

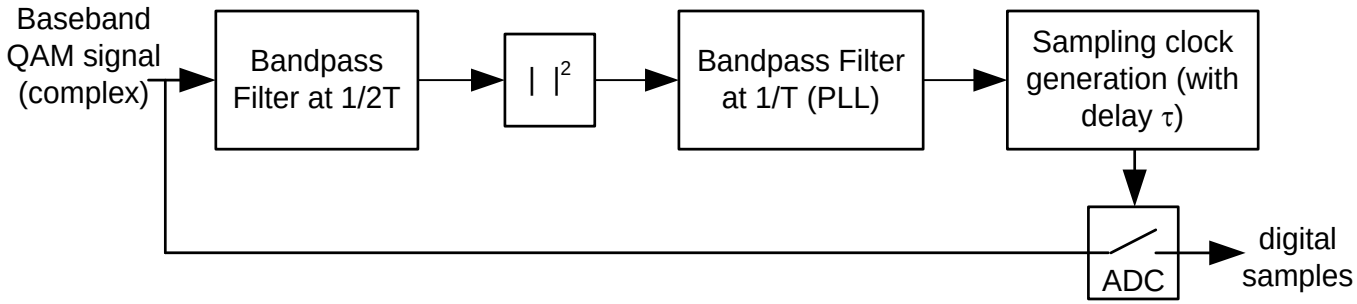
fling@twinclouds.com

Squarer Based Timing Recovery

- Block Diagrams
  - Square-law timing recovery of PAM signal (real)



  - Envelop timing recovery of QAM signal (complex)



fling@twinclouds.com

Squarer Based Timing Recovery (cont.)

- Analysis of PAM Signal timing recovery
  - The baseband received signal can be written as

$$r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT - \tau) + z(t) \cong x(t - \tau) + z(t)$$

  - The expectation of the squared form of $x(t)$ is:

$$E\left[x^2(t)\right] = \overline{a_k^2} \sum_{k=-\infty}^{\infty} h^2(t - kT) = E\left[x^2(t + mT)\right]$$

    - It is periodic with a period of T
  - It can be shown (Poisson Sum Formula):

$$E\left[x^2(t - \tau)\right] = \frac{\overline{a_k^2}}{T} \mathrm{Re}\left[\sum_l D_l \, \mathrm{e}^{\frac{j2\pi l(t-\tau)}{T}}\right]$$

Where $D_l = \int_{-\infty}^{\infty} P(l/T - f)P(f)df$

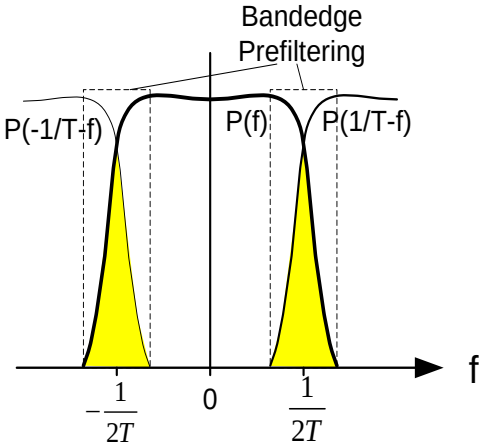
fling@twinclouds.com

Squarer-Based Timing Recovery (cont.)

- Analysis (cont.)
  - For conventional communication signal with an excess bandwidth less than 1/T, $D_l \neq 0$, only for $l = -1, 0, 1$
  - The term with D_0 is a DC term, $D_{-1} = D_1$

$$\Rightarrow E\left[x^2(t-\tau)\right] = \text{const.} + \left(a_k^2/T\right) D_1 \cos\left[2\pi(t-\tau)/T\right]$$

  - $x^2(t)$ is bandpass filtered around 1/2T to remove the DC and interference/ noise terns, to extract the cos(2πt/T) component
  - A sampling clock with frequency of f = 1/T can be generated with a PLL locked to cos(2πt/T)



fling@twinclouds.com

Squarer-Based Timing Recovery (cont.)

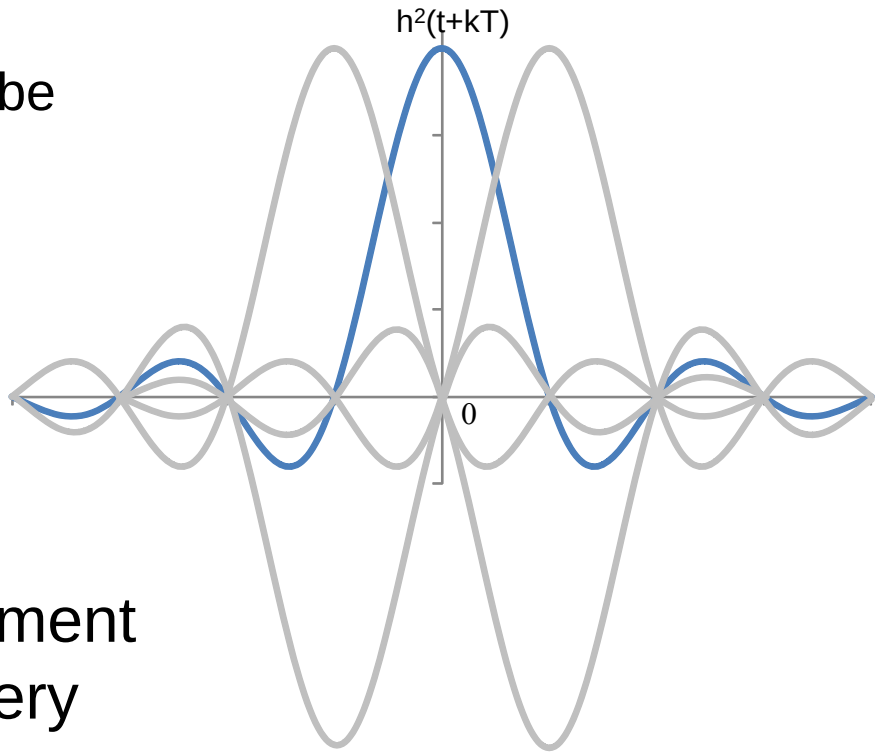
- Discussion
  - Only signal at the transitional bands (at bandedges) are useful
    - Performance can be improved by bandpass prefiltering to remove the other parts of the signal
  - It will not work if there's no or little transitional band signal
  - It does not necessary lock to the optimal timing phase
  - Analysis of envelop timing recovery is similar
  - Intensive analog processing is needed for such implementations
    - High sampling rate will be desirable if to implement using digital signal processing (DSP) but requires high computational complexity
  - For today's modem implementations, it is advantageous to implement timing recovery using DSP with lower sampling rate
    - Timing recovery using baud-rate samples is especially attractive
  - This method is not data assisted. It can be performed independently of demodulation/decoding process.
  - It is widely used in wireline modem implementations

fling@twinclouds.com

Early/Late Gate Timing Recovery

- Assume h(t) similar to a raised cosine waveform
- The right timing phase at the maximum eye opening has a derivative equals to zero
  - The optimal timing phase can be computed recursively

$$\hat{\tau}_{n+1} = \hat{\tau}_n + \Delta \frac{dr^2(t \mid \hat{\tau}_n)}{dt}$$



- This is an approximate form of ML estimate
- This is another way to implement squarer based timing recovery
- Lead to Early/Lage Gate implementation

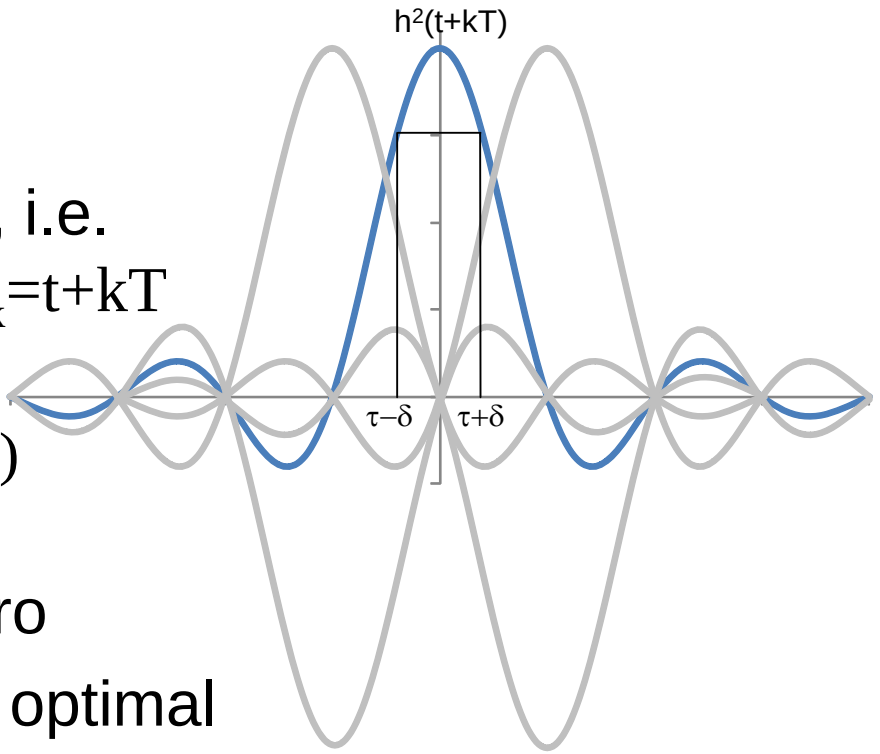
fling@twinclouds.com

Early/Late Gate Timing Recovery (cont.)

- Approximated derivative by:

$$\frac{dr^2(t\,|\,\tau)}{dt} \approx \frac{r^2(\tau+\delta)-r^2(\tau-\delta)}{2\delta}$$

- We use 3 samples every T, i.e. $r(t_k)$, $r(t_k+\delta)$ and $r(t_k-\delta)$, for $t_k=t+kT$ with $\delta < T/2$
- Use $m(t_k) = r(t_k+\delta) - r(t_k-\delta)$ as the input to the timing loop that drives $m'(t_k)$ to zero
- In steady state, t_k's are the optimal sampling timing phases and $r(t_k)$ are the samples for demodulation/decoding



fling@twinclouds.com

Gardener's Algorithm

- This algorithm was proposed by Floyd Gardener in 1986
  - It was derived based on the squarer timing recovery
- The input data are sampled at every T/2
  - The samples at integer multiple of T, $r(nT)$, are used by the demodulator/decoder
  - The timing error detection (TED) function is defined by

$$m_G(t_n) = \mathrm{Re}\left[r^*(nT) \times \left\{ r(nT + T/2) - r(nT - T/2) \right\} \right]$$

  - It is input to a timing loop to drive $m_G(t_n)$ to zero
- This algorithm operates on the sample sequence spaced T/2, which is practical feasible for implementation
  - Acceptable rate (1/T will be even better)
  - Such sequence is convenient for timing adjustment using digital interpolation techniques described below

fling@twinclouds.com

Gardener's Algorithm (cont.)

- An intuitive view of the algorithm
  - Assuming the symbol waveform close to a raised cosine function
  - At the optimal sampling time, $r(nT)$ is an estimate of symbol a_n
  - The TED $m_G(t_n)$ is approximately a scaled version of

$$h(nT + T/2) - h(nT - T/2)$$

  - When h(t) is symmetric, nT is the optimal sample timing if and only if $m_G(t_n) = 0$
- It is not data assisted
  - It is very similar to the data assisted Early-Late timing algorithm widely used in digital receivers, such as receivers used in wireless CDMA receivers
  - It may not work well if for low SNR and/or high ISI cases

fling@twinclouds.com

DATA ASSISTED TIMING SYNCHRONIZATION TECHNIQUES

fling@twinclouds.com

All-Digital/Symbol-Rate Timing Recovery

- For today's modem implementation, all digital processing at lower sampling rate is preferable
  - Symbol rate (1/T) is probably the practically lowest possible rate
- Assume single path AWGN channel with no ISI

$$r(t) = \sum_{k=-\infty}^{\infty} a_k \, p(t - kT - \tau) + z(t)$$

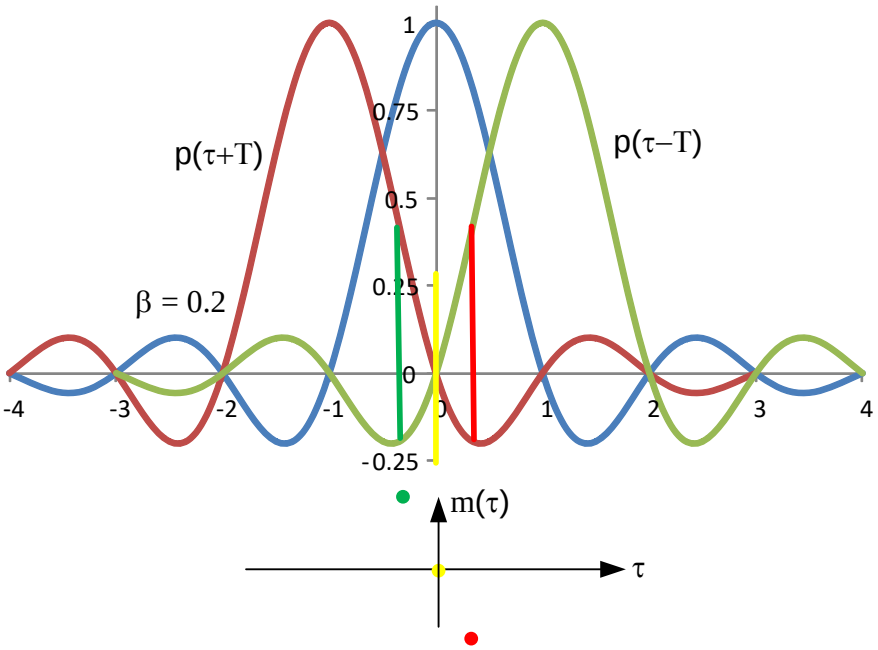
  - p(t) satisfies Nyquist criterion, e.g., a raised cosine waveform
  - We construct the timing discriminator function for $t = kT$:

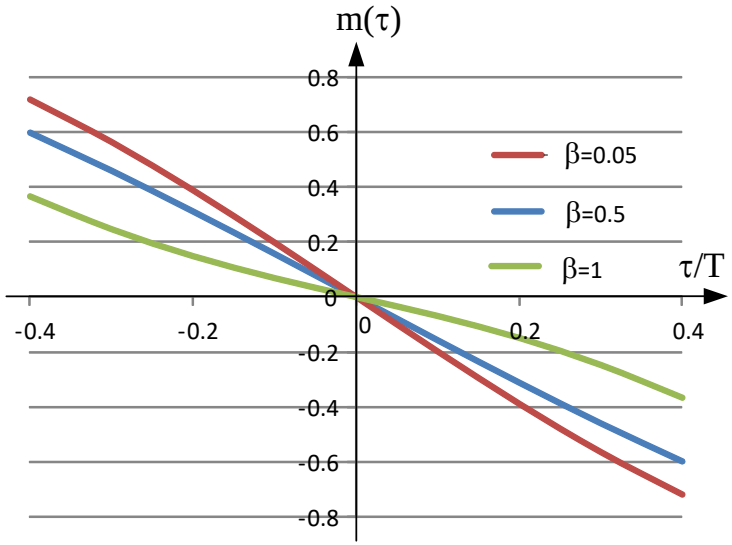
$$m(\tau) = p(\tau + T) - p(\tau - T)$$

  - It is a monotonic decreasing function of τ with $m(0) = 0$
    - It's shape (slop) is a function of the excess bandwidth β

fling@twinclouds.com

All Digital-Symbol Rate TR (cont.)





- To computer m(t), we define: $\varphi_k(\tau) = r(k)a_{k-1} + r(k-1)a_k$, where $r(k) = r(kT)$. It can be shown: $E\left[\varphi_k(\tau)\right] = m(\tau)$
- This is the Mueller-Müller algorithm widely used for timing recovery with symbol rate processing

fling@twinclouds.com

All Digital Symbol Rate TR (cont.)

- Discussion
  - The output m(t) can be used to drive a TLL for timing phase and frequency synch
  - For signals over channels that satisfy Nyquist criterion (no ISI), the timing phase generated is optimal (maximum eye opening)
  - For general ISI channels, the timing generated will be stable but may not be optimal
    - Additional methods to determine optimal timing delay may be used for generating sample for generating Tx symbol estimates
  - It is a data-assisted method. Tx symbols a_k's and their rough timing (to within a fraction of a symbol interval) should be known.
    - Suitable to be used in training mode.
    - Tentative decisions may be used instead of true Tx symbols.

fling@twinclouds.com

CDMA Early/Late Gate Delay Locked Loop

- In a CDMA system, the SNR at chip level can be very low at operation point, the early/late gate timing recovery method described above does not work well
- For CDMA receivers the early/late gate approach is modified by incorporating pilot PN sequence despreading
  - CDMA receiver usually implemented as a Rake structure
  - Each rake finger corresponds to a single path of a multipath channel with certain delay and its CIR similar to RCOS
  - The T_c spaced early/late samples of each rake finger correlate with the corresponding PN sequences (despread) to improve SNR
    - The despread output approximately follows the channel path CIR
  - It is usually called the delay locked loop (DLL) in CDMA literature
  - Each Rake finger has its own DLL
  - It is the most popular timing recovery approach in CDMA systems

fling@twinclouds.com

CDMA Early/Late Gate DLL (cont.)

- CDMA DLL operation
  - The received signal are sampled at every $T_c/2$ (T_c is chip interval)
  - The approximate optimal timing (delay) τ_i of the i-th finger is determined by searchers when the i-th path is found
  - Assume each CDMA pilot symbol consists of N_p chips
  - The n-th CDMA pilot symbol corresponds to the chip sample sequence $r(nN_pT_c + kT_c + \tau_{n,i}), k = 0,..., N_p - 1$ for the i-th finger
  - The early and late gate outputs of the i-th finger is formed by

$$g_i^{(E/L)}(n) = \sum_{k=0}^{N-1} p_p(n,k) r(nNT_c \mp T_c/2 + kT_c + \tau_{n,i})$$

    where $p_p(n,k)$ is the k-th chip value of the n-th pilot symbol
  - The delay error metric is $m_i(n) = \left|g_i^{(E)}(n)\right| - \left|g_i^{(L)}(n)\right|$
  - $m_i(n)$ is fed to the i-th DLL to control the sampling delay for driving $m_i(n)$ towards to, and maintaining at, zero

fling@twinclouds.com

CDMA Early/Late Gate DLL (cont.)

- The operations of CDMA DLLs (cont.)
  - When $m_i(n')$ is zero, the delay $\tau_{n',i}$ is optimal, which is used for pilot and data despreading
  - The pilot despread output of center samples:

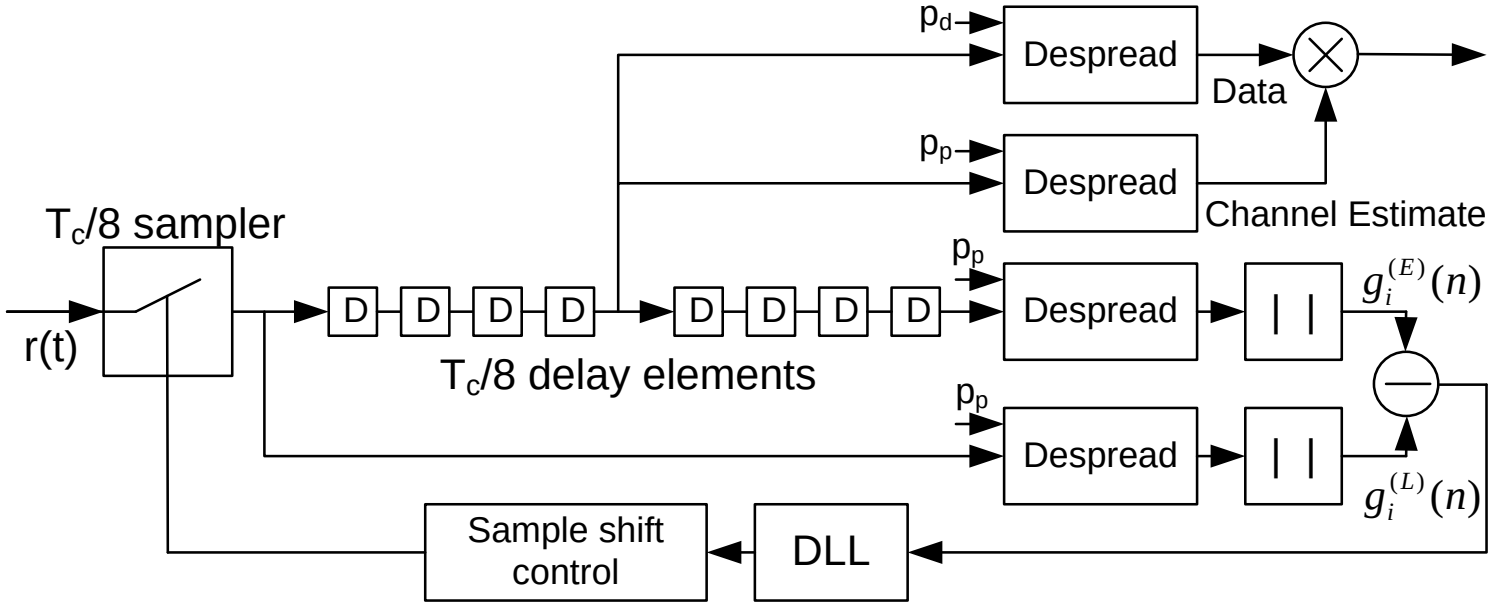
$$g_i(n') = \sum_{k=0}^{N_p-1} p_p(n,k) r(nNT_c + kT_c + \tau_{n',i})$$

    is the channel estimate of the i-th path
  - The despread output of center samples by data spreading sequence are data symbol estimates distorted by channel
  - The despread data symbols are weighed by phase and magnitude of the complex conjugate of the channel estimate to generate data symbol estimates for finger combining and demodulation.
  - When CDMA receivers operate at low SNRs, as most likely the case, a timing phase resolution of $T_c/8$ would be sufficient

fling@twinclouds.com

CDMA Early/Late Gate DLL (cont.)

- Rake finger block diagram



  - Normally, 8 $T_c/8$ samples are shifted into the delay line when processing a chip sample
  - If an adjustment is needed, 7 or 9 samples will be shifted into the delay line instead of 8

fling@twinclouds.com

Channel-Impulse-Response/Equalizer Coefficient Estimates based Timing Synchronization

- CIR/EQ for communication over multipath channel
  - To achieve desirable performance for communication over multipath channels at higher SNR, it is necessary to use some form of equalization techniques
  - Equalizer coefficients can be generated directory from the received signal or derived from estimated CIR coefficients
  - An OFDM receiver can also be viewed a form of equalizer
  - Timing synchronizations in the receiver of a single carrier system is to determine the starting position of received data signal samples to convolve with the equalizer coefficients
    - Samples back-off may be needed.
  - In an OFDM system, time synch is to determine the position of the FFT window on the data signal sample sequence

fling@twinclouds.com

CIR/EQ Estimates Based Timing Synch (cont.)

- Pilot/reference signals in data transmission
  - In such communication systems, known symbols, called pilots or references, are often sent together with data signal
    - Such pilot signals are usually sent at the beginning of data transmission, during data burst, or distributed along with data signals
  - Pilot signals are used for initial acquisition and other synchronization tasks.
  - Pilot signals are also used for channel estimation, i.e., to generate the estimates of channel impulse responses (CIR) and/or Equalizer coefficients, directly or indirectly
  - Once the reception is established, if data detection has little or no error, CIR and EQ coefficients can be generated/updated using the detected data symbols

fling@twinclouds.com

CIR/EQ Estimates Based Timing Synch (cont.)

- Implementation of timing phase synch in such systems
  - In communication systems with pilot signal embedded in data transmission blocks
    - CIR/EQ coefficients are generated or updated every data block and proper data segments are then selected
    - Namely timing phase synch is done through CIR/EQ estimation for each received data block
  - In systems the pilot signal is only sent at the beginning of data transmission
    - CIR/EQ is estimated at initialization and updated during receiving using decision recovered
    - Additional increment or decrement may be needed occasionally where the coefficients shifts due to various reasons

fling@twinclouds.com

CIR/EQ Estimates Based Timing Synch (cont.)

- Determine optimal timing sampling phase
  - For sub-Nyquist rate, e.g. symbol rate, sampling processing
    - It is important to select optimal sampling phase to ensure best possible receiver performance, but it is difficult in general
    - One simple but effective method is to have multiple samples per symbol but only use the T-spaced stream with the highest energy
  - For Nyquist rate, e.g., fractional symbol rate sampling (single carrier system) or OFDM system with proper guard-carriers
    - No sub-sample-spacing adjustment is needed
- Implementation of timing frequency synchronization
  - Timing frequency is usually estimated at the initialization stage
    - Timing frequency error can be determined from the timing phase error change from sample to sample or from burst to burst
    - Timing frequency error may also be corrected based on estimated carrier frequency error

fling@twinclouds.com

CIR/EQ Estimates Based Timing Synch (cont.)

- Implementation of timing frequency synch (cont.)
  - In data mode, the residual timing frequency error is reflected in and will be corrected when there is a consistent increase of positive or negative timing phase error increase
  - It should be noted the timing frequency error has less impact to receiver performance than carrier frequency error
    - E.g. if there is 2 ppm XO error, the impact will only be seen, say, after 10^4 samples
  - Timing frequency adjustment should have a long time constant
  - Timing frequency adjustment should not respond to occasional large phase error adjustment
    - Which could due to sudden appearing disappearing of a new path in a wireless fading environment or due to other reasons, such as the coefficients drifting in a fractional spaced equalizer

fling@twinclouds.com

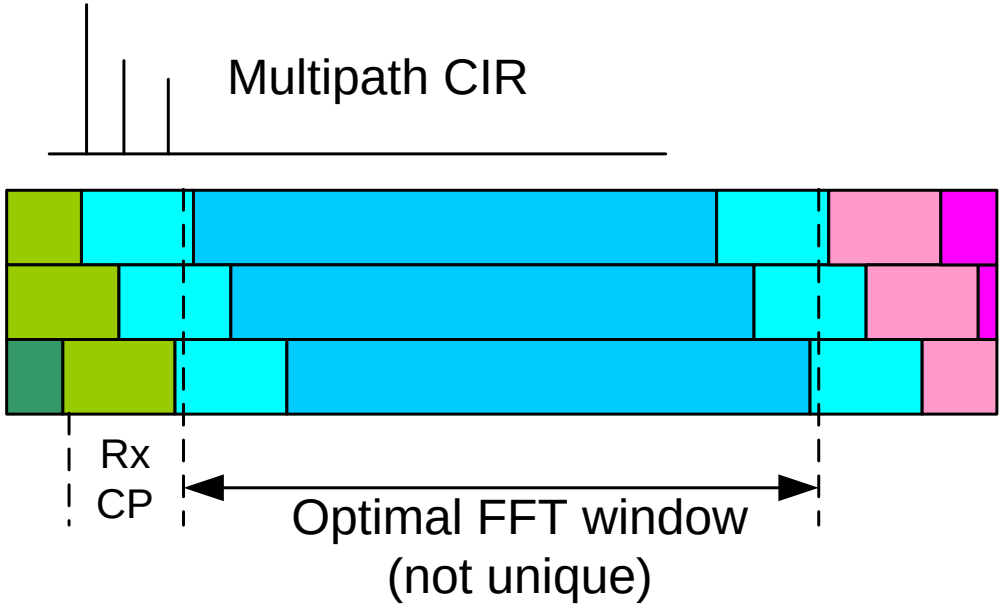
Timing Synchronization in OFDM systems

- In OFDM system, timing synchronization need to address two aspects: Sampling frequency and Positioning of the FFT window
- OFDM is usually tolerant to sampling frequency error
  - It may only need to be fine-tuned during data mode
- Due to that Nyquist rate sampling is commonly used, no subsample sampling phase optimization is needed
  - The timing adjustment is at a resolution of an OFDM chip
- Thus, most important and challenging task is to determine the position of FFT window, which is determined by CIR.
  - FFT window position can be initialized by
    - Initial estimation of CIR using special sequences/TDM pilots, if available
    - Estimate based on cyclic preamble (CP)

fling@twinclouds.com

Timing Synchronization in OFDM systems (cont.)

- The criterions of determining the position of FFT window:
  - The RX CP shall contain the first arriving path (FAP)
    - If possible add some back-off
  - The Rx CP should contain the last arriving path (LAP)
  - If not possible to satisfy both, Rx CP should cover the paths with most of the energy



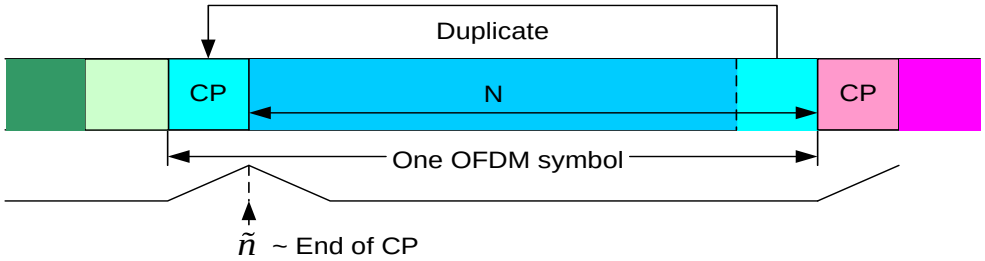
fling@twinclouds.com

OFDM Timing Synchronization (cont.)

- Timing Synch initialization
  - Special sequence/TDM pilot Method
    - Generate an estimate of CIR
    - Determine initial FFT window position based on such estimated CIR
  - CP based Delay-and-correlation Method
    - CP is a repetition of the last portion of the OFDM symbol
    - delay-and-correlation:

$$D(n) = \sum_{k=0}^{N_{CP}-1} r^*(n+k) \times r(n+k+N), \quad N = \text{FFT size}, r(n): \text{ received signal samples}$$

    - Determine OFDM symbol timing with: $\tilde{n} = \arg\max_n D(n)$



fling@twinclouds.com

OFDM Timing Synchronization (cont.)

- Timing tracking in data mode
  - In most OFDM systems, FDM pilots are inserted for frequency domain channel estimation (FCE) for demodulation
  - For timing tracking, time domain CIR can be generated from FCE by iFFT
  - For every OFDM symbol or every burst OFDM symbols, CIR is used to determine the optimal OFDM Rx timing according to the above criterions
    - FFT windows will adjusted by integer number of samples accordingly
  - If such adjustments are consistently in one direction, it is an indication of sample clock frequency error
    - Sample clock frequency need to be adjusted to correct such error
    - Sample clock frequency should not be changed if such adjustments are random or only happen rarely
      - It could due to other reasons, e,g, multipath appearing or disappearing, or due to receiver sleep, rather than sampling frequency error

fling@twinclouds.com

TIMING ADJUSTMENTS USING DIGITAL INTERPOLATION

fling@twinclouds.com

Introduction

- To correct and change receiver timing, sampling phase need to be adjusted
- In early digital receiver designs such changes are done by changing the sampling instant of the AD converter
- Nyquist (1928) showed that an analog signal can be totally recovered from its samples if generated at a rate higher than twice of the highest frequency component
  - They are called Sampling Theorem and Nyquist Rate
  - It also implies that samples with a different sampling phase can be generated from a Nyquist rate sample sequence
    - This technique is called Digital Interpolation in the literature
- From 1960's the research on and usage of digital signal processing (DSP) had greatly increased

fling@twinclouds.com

Introduction (cont.)

- Along with the popularity of DSP, digital interpolation also attracted more and more attention
  - Papers on digital interpolation started appearing in early 1970's, e.g., Schafer and Rabiner (1973)
  - The most well know literature on this topic is the book: *Multirate Digital Signal Processing* by Crochiere and Rabiner, (1983).
- In mid 1980's it was proposed to use digital interpolation to generate a sample sequence from another sequence with a different rate for timing recovery in echo cancellation modems, e.g., Shahid Quresh, Codex Corp., Jan. 1985
- This approach was implemented in Codex V.32 modem commercialized in 1986/1987 time frame
  - It is probably the earliest commercial modem products with timing recovery based on digital interpolation technique

fling@twinclouds.com

Introduction (cont.)

- Floyd Gardener did similar work independently for European Space Agency also in mid or late 1980's
  - I am not sure exactly what was the time frame of this work
  - His papers: Interpolation in digital modems part I and II published in IEEE T-Comm 1993 March/June had become the most widely cited reference on this topic
- Digital interpolation based timing recovery has become standard technique in today's digital modem implementation

fling@twinclouds.com

Introduction (cont.)

- Part of the original Codex Memo

codex CONFIDENTIAL MEMORANDUM

TO: List

DATE: January 14, 1985

FROM: Shahid Qureshi

SQ: 85-02

SUBJECT: Note on Reduction of Future V.32 Analog Circuitry

Attached are handwritten notes on a subject that I have been contemplating off and on (in my spare time) for the last few months. Please let me know if you have any questions. I can review the subject with all of you if you so desire. Comments are solicited.

/llf

List: H. Chalmers
L. Gonzalez
D. Kline
F. Ling
J. Pasco-Anderson
J. Payton
B. Schissler
M. Sridhar

3

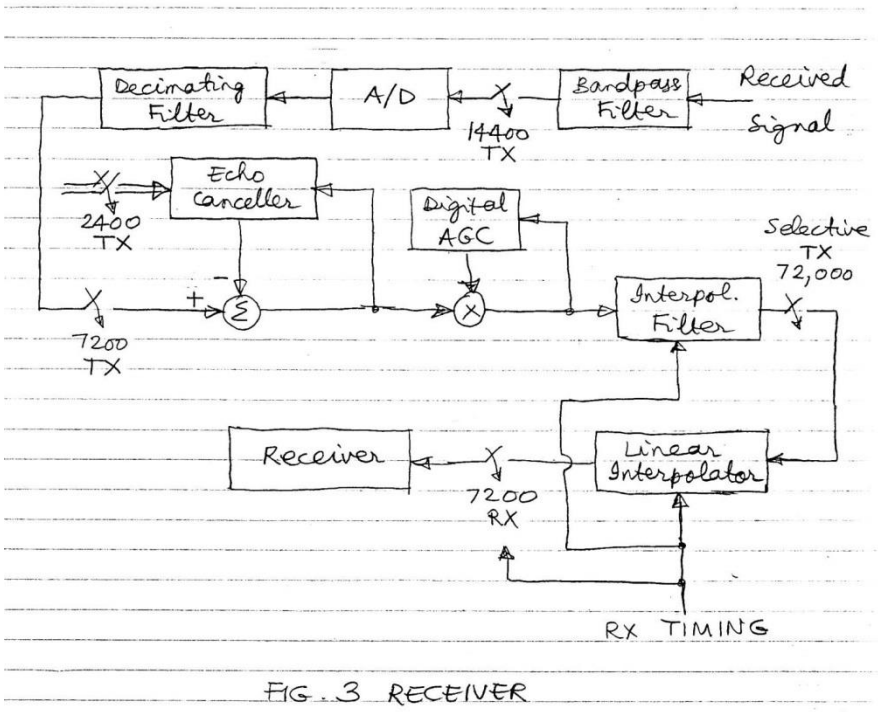


FIG. 3 RECEIVER

fling@twinclouds.com

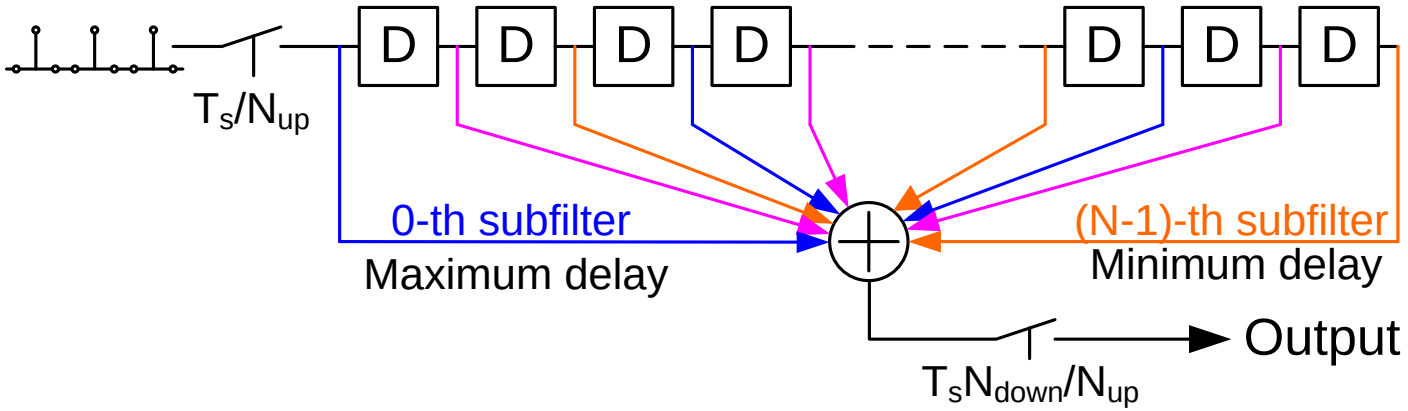
Sampling Phase Control and Rate Conversion

- Sampling phase can be controlled digitally as follows:
  - A T_s-spaced digital sample sequence passes through filter FIR_1 with a group delay $g_1(f)$, as the reference sequence.
  - Passing the input sequence through another filter (FIR_2) with the same amplitude and phase response but a group delay of $g_1(f)+\delta T_s$, the output's sampling phase is changed by δT_s (δ can be positive or negative) relative to is the reference sequence
    - In such way, the modem receiver timing phase is adjusted
- Rate conversion by sampling phase shift
  - We generate a sequence such that each output sample with a negative group delay $-\Delta T_s$ relative to the previous sample, the output sequence's sampling frequency is $1/\Delta T_s$
    - The sample to sample delay has to be negative for a valid sequence
  - The sampling frequency ratio of the input and output is $1/\Delta$, which can be greater or less than one

fling@twinclouds.com

Rational Digital Rate conversion

- Considering up-sampling an input sequence by adding N_{up} -1 zeros and passing through a low-pass FIR filter
  - Low-pass filter is for rejecting the images of the repeated spectrum due to zero insertion
- The output is down-sampled by taking every N_{down} outputs of the FIR filter to achieve an rational rate conversion with a conversion ratio of $R_c = N_{up}/N_{down}$



fling@twinclouds.com

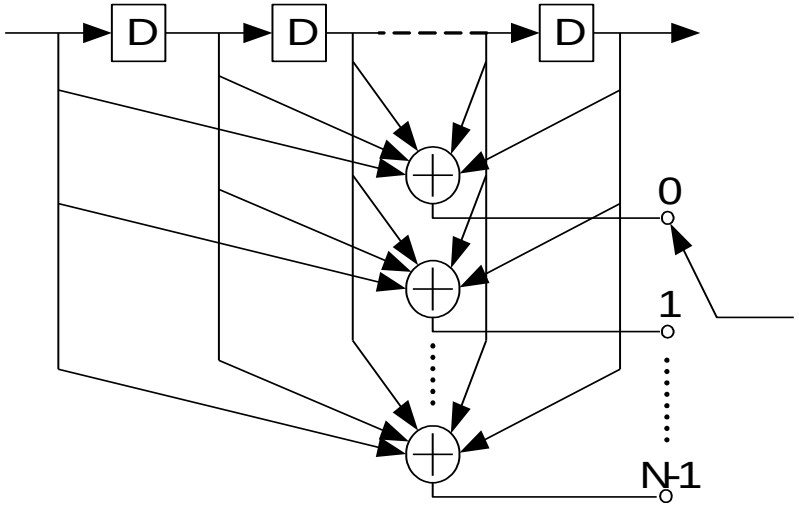
Rational Digital Rate conversion (cont.)

- Remarks
  - R_c can be greater than 1 (interpolation) or less than 1 (decimation)
  - If $R_c < 1$, the low-pass filter need to prevent aliasing
  - The low-pass filter can be considered as consisting of N_{up} subfilters, each of which has L coefficients
  - Because the input to the filter has N_{up}-1 zeros for every non-zero sample, the complexity is proportional to L to generate one output
    - Its complexity is thus proportional to LxN_{down} independent of N_{up}
  - We can view the FIR filter as having N_{up} subfilters, 0, …, N_{up}-1
    - The subfilters have almost identical frequency responses, just differing by group delays. The differences are constants in passband
    - The difference of group delays between k-th and (k+1)-th subfilters is equal to T_s/N_{up}, with the k-th subfilter has higher delay
  - The low-pass filter can be usually designed by using McClellan-Parks, LS or other FIR filter designing methods

fling@twinclouds.com

Poly-Phase Filter Bank

- An efficient implementation of the rate converter is the poly-phase filter bank
  - The FIR subfilters discussed above are used here
  - The output is generated by the subfilter selected,



  - Then the filter index increments by N_{up}, modulo N_{up}, to select next one
  - A new sample is shifted into the delay line when each modulo operation occurs
- Poly-phase filter-bank can be used for sampling phase control with the resolution of phase shift equals to T_s/N_{up}

fling@twinclouds.com

Arbitrary Sampling Phase Shift

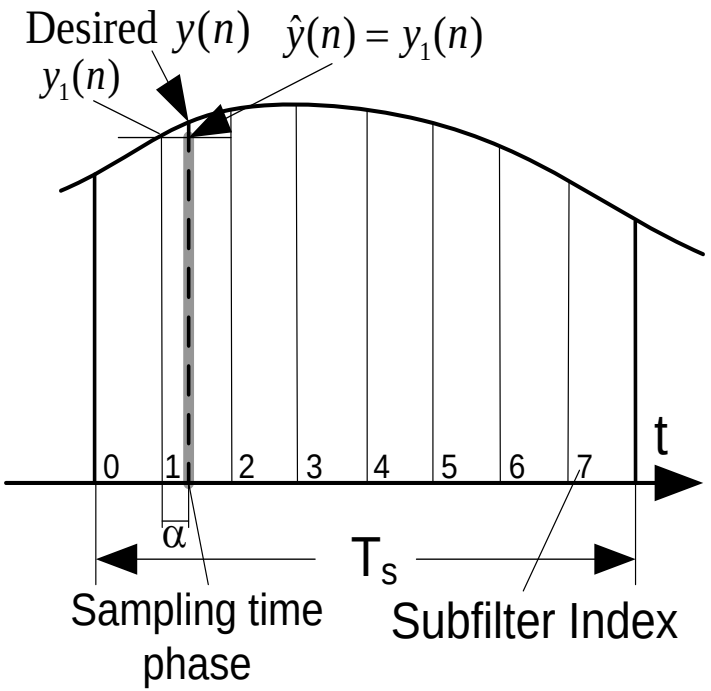
- To achieve higher resolution of sampling phase shift, a large number of subfilters will be needed
  - This will increase the storage requirement
- An efficient way is to use two stage interpolation
  - Linear interpolation approximation is most popular for the second stage. Zero-th and high order approximations can also be used
- The linear interpolation (first order approximation):
  - Assuming the desired sampling phase of sample n is equal to φ
  - Compute the outputs $y_k(n)$ and $y_{k+1}(n)$ of subfilters k and k+1 with sampling phases $\varphi_k < \varphi < \varphi_{k+1}$ and $(\varphi - \varphi_k) / (\varphi_{k+1} - \varphi) = \alpha / (1 - \alpha)$
  - The desired output is computed by $y(n) = (1-\alpha) y_k(n) + \alpha y_{k+1}(n)$
- The zero-th order approximation:

$$if\ (\alpha \le 1-\alpha)\ y(n) = y_k(n);\ \ else\ y(n) = y_{k+1}(n)$$

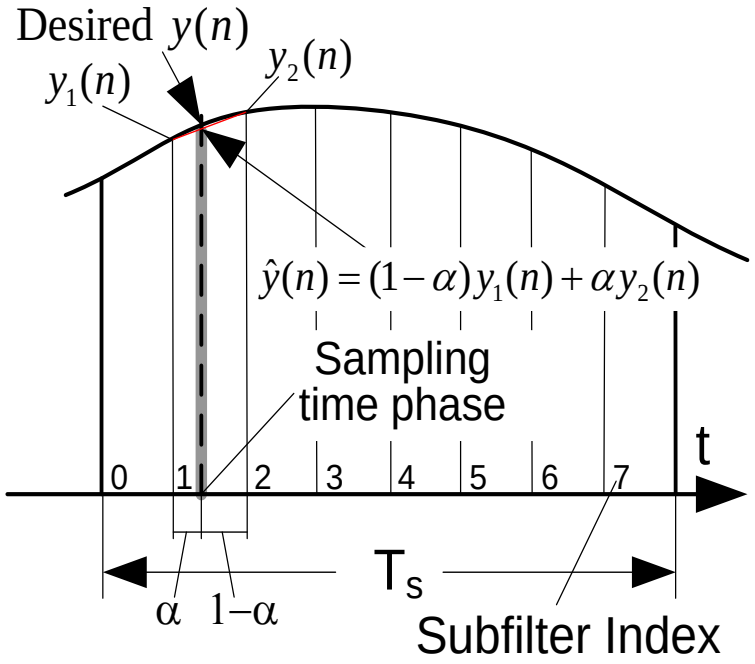
- Higher order spline approximations can also be used

fling@twinclouds.com

Arbitrary Sampling Phase Shift (cont.)



Zero-th order approximation



First order approximation (Linear Interpolation)

fling@twinclouds.com

Arbitrary Sampling Phase Shift (cont.)

- Distortion analysis
  - Analysis of zero-th order approximation
    - Total signal power:

$$P_s = \frac{1}{2\pi}\int_{-\omega_y}^{\omega_y} |Y(\omega)|^2 \, d\omega = \frac{A^2\omega_y}{\pi}$$

      - For flat signal spectrum with magnitude $|Y(\omega)|=A$ in range $[-\omega_x, \omega_x]$
    - Assuming U subfilters, the maximum error $|\alpha/U| \le 0.5/U$. Error in frequency domain: $Y(\omega)\left(e^{j\omega\tau} - e^{j\omega(\tau-\alpha/U)}\right) \approx -Y(\omega)e^{j\omega\tau}\omega\alpha/U$, the total error due to distortion is:

$$P_e = \frac{1}{2\pi}\int_{-\omega_y}^{\omega_y} \left|Y(\omega)e^{j\omega\tau}\omega\alpha/U\right|^2 d\omega \le \frac{1}{2\pi}\int_{-\omega_y}^{\omega_y} A^2\left(\frac{0.5}{U}\right)^2 \omega^2 d\omega = \frac{A^2\omega_y^3}{12\pi U^2}$$

    - The signal to distortion ratio $\gamma_{sd} = P_s/P_e \ge 12U^2/\omega_y^2$
    - Example:
      - For $\omega_x = 0.8\pi$, γ_{sd} = 31622 (45 dB)
      - $U = \omega_y\sqrt{10^4/12} \approx 129$ subfilters

fling@twinclouds.com

Arbitrary Sampling Phase Shift (cont.)

- Distortion analysis (cont.)
  - Analysis of first order approximation (linear interpolation)
    - Total signal power is the same as before
    - The frequency response of the filter with linear interpolation is

$$Y(\omega)\left((1-\alpha)e^{j\omega(\tau-\alpha/U)}+\alpha e^{j\omega(\tau+(1-\alpha)/U)}\right)$$

    - The error in frequency domain is:

$$Y(\omega)\left(e^{j\omega\tau}-(1-\alpha)e^{j\omega(\tau-\alpha/U)}-\alpha e^{j\omega(\tau+(1-\alpha)/U)}\right)$$

$$\approx Y(\omega)e^{j\omega\tau}\alpha(1-\alpha)\omega^2/U^2$$

    - Using $\alpha(1-\alpha)\le 0.25$, we have an upper bound of the total error energy

$$P_e^{(1)}=\frac{1}{2\pi}\int_{-\omega_y}^{\omega_y}\left|Y(\omega)e^{j\omega\tau}\alpha(1-\alpha)\omega^2/U^2\right|^2 d\omega<\frac{A^2\omega_y^5}{320\pi U^4}$$

    - The signal to distortion ratio is then higher than

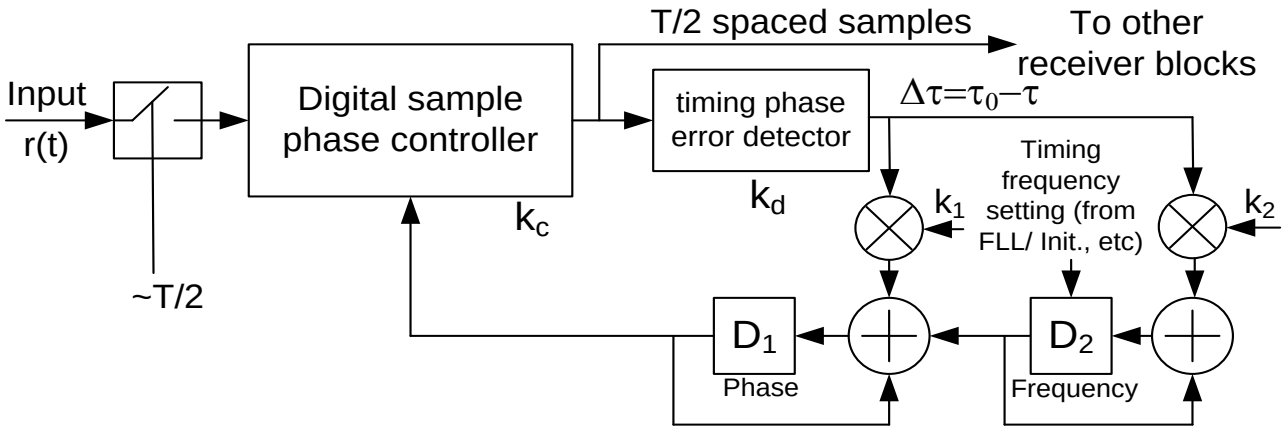
$$\gamma_{sd}=P_s\Big/P_e^{(1)}\ge 320U^4\Big/\omega_y^4$$

    - For ω_y=0.8π, γ_{sd}=45dB, $U\approx\omega_y\sqrt[4]{\frac{\gamma_{sd}}{320}}=0.8\pi\sqrt[4]{\frac{31622}{320}}\approx 8$ subfilters

fling@twinclouds.com

A Design Example of All Digital Timing Loops

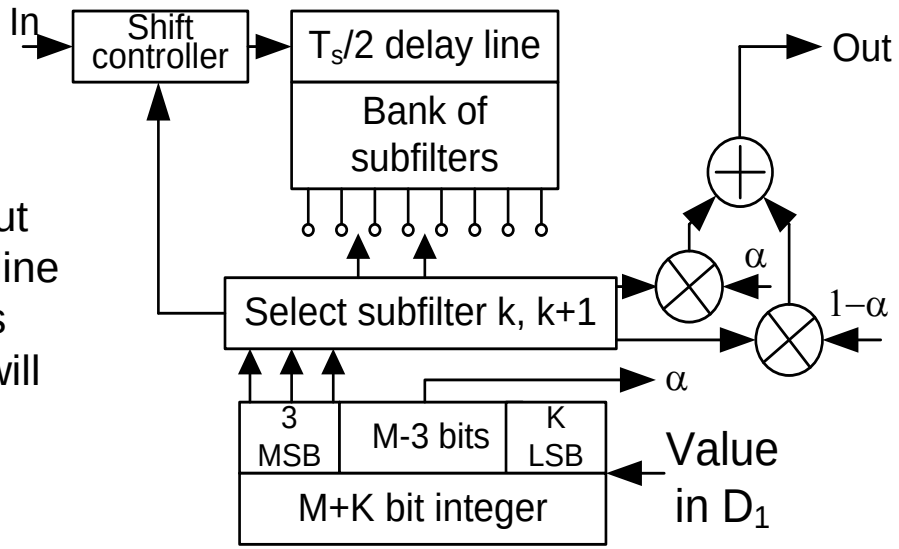
- We consider a (non-adjustable) XO based design
  - The input is at a fixed sampling rate of $1/T_s$, approximately equal to $2/T$
  - It is essentially a second order loop. Its components are:
    - D_2 contains estimated sampling frequency error
    - D_1 contains desired sampling phase represented by an integer number corresponding to time delay τ, $\tau \in [0, T_s)$.
    - Phase controller controls the sampling phase according to the value of D_1(gain k_c)
    - Timing phase error detector computes the difference between the current phase and the desired phase (various implementations, gain k_d)
    - The units of k_d and k_c are timing-unit/integer-unit and integer-unit/timing-unit, respectively.
  - The output samples are at $2/T$ (usually close to input sampling frequency)



fling@twinclouds.com

All Digital Timing Loop Design (cont.)

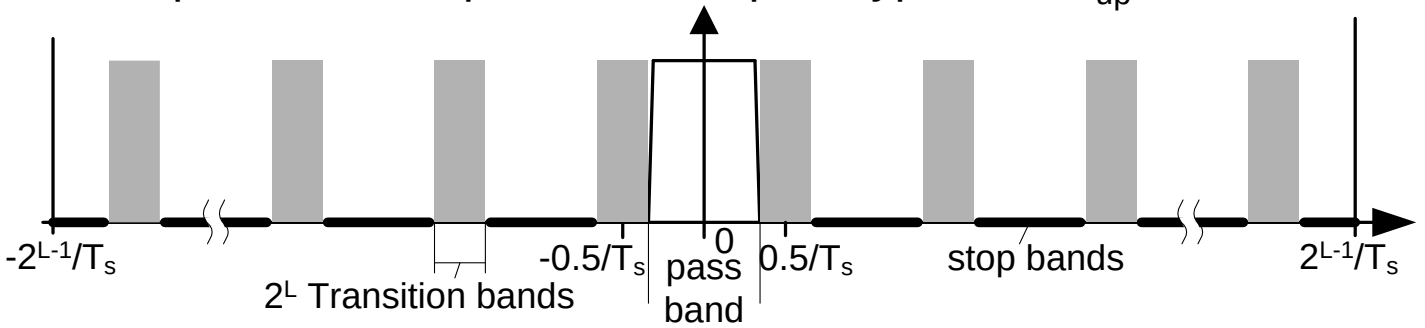
- Operations of the digital sample phase controller
  - It uses a poly-phase filter bank with 2^L subfilters. (L = 3 in the figure)
  - The phase register has M+K bits, which is interpolated as an unsigned integer with maximum value is equal to one
    - The value of the L MSBs, k, determines which two subfilter to use
    - The middle M-L bits (unsigned) is the value of $0 \le \alpha < 1$
    - The K LSBs are needed for phase error accumulation as k_2 is small
    - The number is unsigned. It will wrapping around when overflow
  - By defining phase-error of $T_s/2$ to be one, the gain $k_c = 1$
  - Shift controller operations
    - Normally, a sample is shifted in for every output sample
    - When D_1 overflows (k changes from 2^L-1 to 0) an additional input sample is shifted into the delay line
    - When D_1 underflows (k changes from 0 to 2^L-1) no new sample will be shifted in (shift hold)



fling@twinclouds.com

All Digital Timing Loop Design (cont.)

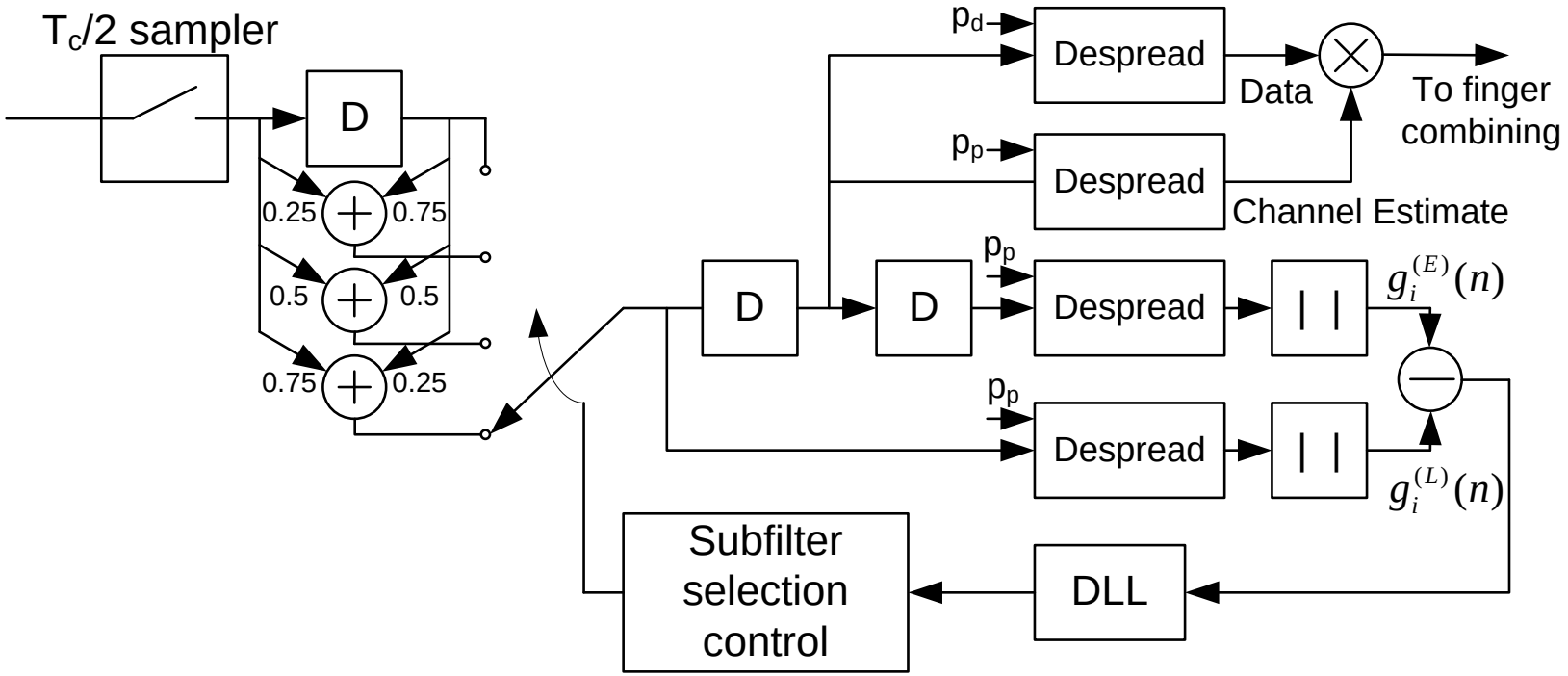
- Design of the Interpolation filter for poly-phase filter bank
  - Signal is sampled at T_s.
  - The filter's passband is $[-\omega_y, \omega_y]$, which is of interest for interpolation
    - The passband should be as flat as possible, but also as necessary
  - Stop bands are the image bands of the passband due to zero insertion.
    - The attenuation of stop bands determines the uniformity of the subfilters
  - The transitional bands are the bands between the stop and pass bands
    - It can be chosen as "don't care," with very low weighting, for FIR filter design.
    - The aliased signal in these bands does not affect the results of interpolation
  - The filter can be designed using various FIR filter design tools
  - An example of an interpolation filter prototype with $N_{up}=2^L$ subfilters:



fling@twinclouds.com

CDMA DLL with $T_c/2$ Samples and Linear Interpolation

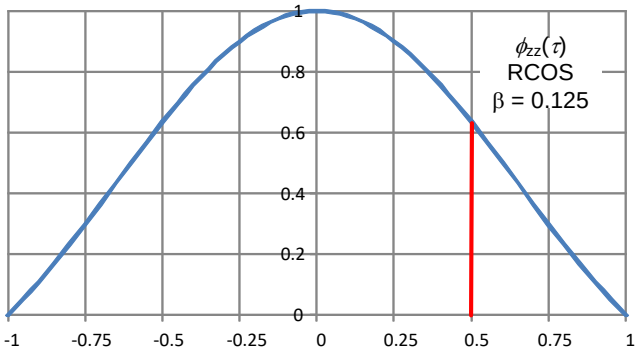
- For low SNR applications, four 2-coefficients linear interpolating subfilters can be used to generate $T_c/8$ samples from $T_c/2$ input
- The subfilter selection is controlled by the phase register of DLL



fling@twinclouds.com

CDMA DLL with $T_c/2$ Samples and Linear Interpolation (cont.)

- The simple linear interpolating coefficients shown in the above figure are: $(a, b) = (1.0, 0.0), (0.75, 0.25), (0.5, 0.5), (0.25, 0.75)$
- They should work fine for demodulation, however they have different gains to random data input
  - It will cause problem when the output need to be normalized by the power of the random input data, e.g., for searcher with fixed threshold
    - The samples at t and t+0.5T are correlated with correlation coefficient ρ
    - The output power is equal to $(a_2+b_2+2ab\rho)$ times the input power
    - If the received filter is a square-root RCOS filter with 1/8 excess bandwidth, the correlation coefficient $\rho = 0.635$ (see figure)
    - The normalized coefficients are $(a', b') = (1.0, 0.0), (0.807, 0.269), (0.553, 0.553), (0.269, 0.807)$
    - We may also determine the coefficients by designing a 7 tap lowpass filter
  - We can solve this problem also by using thresholds depending on filter index



fling@twinclouds.com

THE END

fling@twinclouds.com

Acknowledgement

The materials in this presentation are mostly not new. I just pulled them from the literature and from my experience accumulated during the past 30 years. Even most of these experience were also learned from others as well.

Thus, here, I would like to thank all the researchers and engineers who contributed to this technical topic. The credits should go to the respective contributors. However, because there are so many of them, I really cannot mention all. Of cause, I should responsible for any mistake in the presentation and would sincerely appreciate if someone would take time to point out to me.

Sincerely,

Fuyun Ling

fling@twinclouds.com

THANK YOU!

fling@twinclouds.com