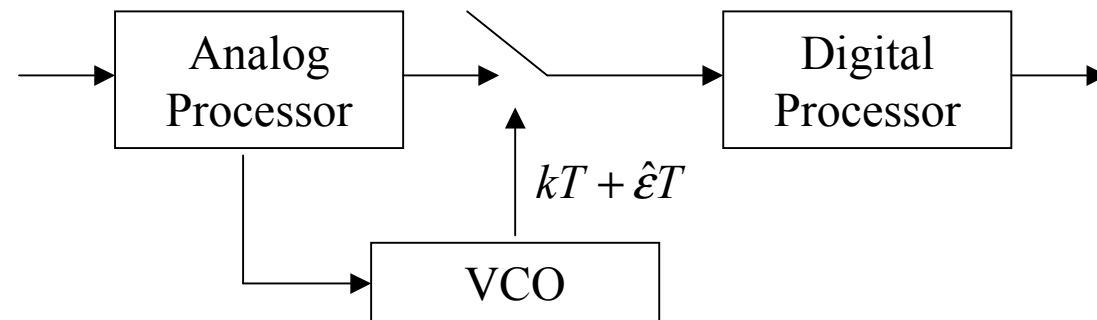


## TIMING ESTIMATION

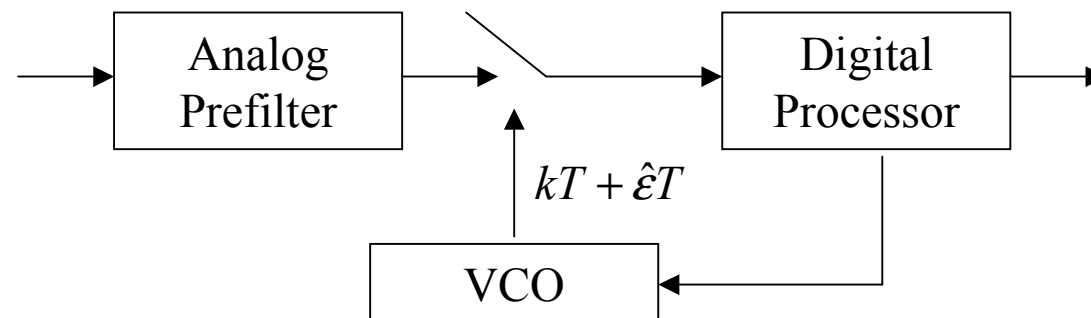
### Analog Timing Recovery:

Sampling is synchronous with received signal at symbol rate.



### Hybrid Timing Recovery:

The timing information is derived from the samples of the receiver.

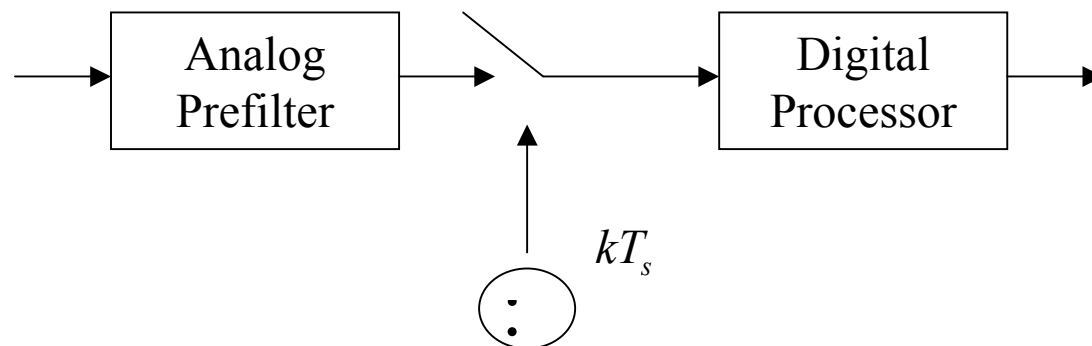


### Digital Timing Recovery:

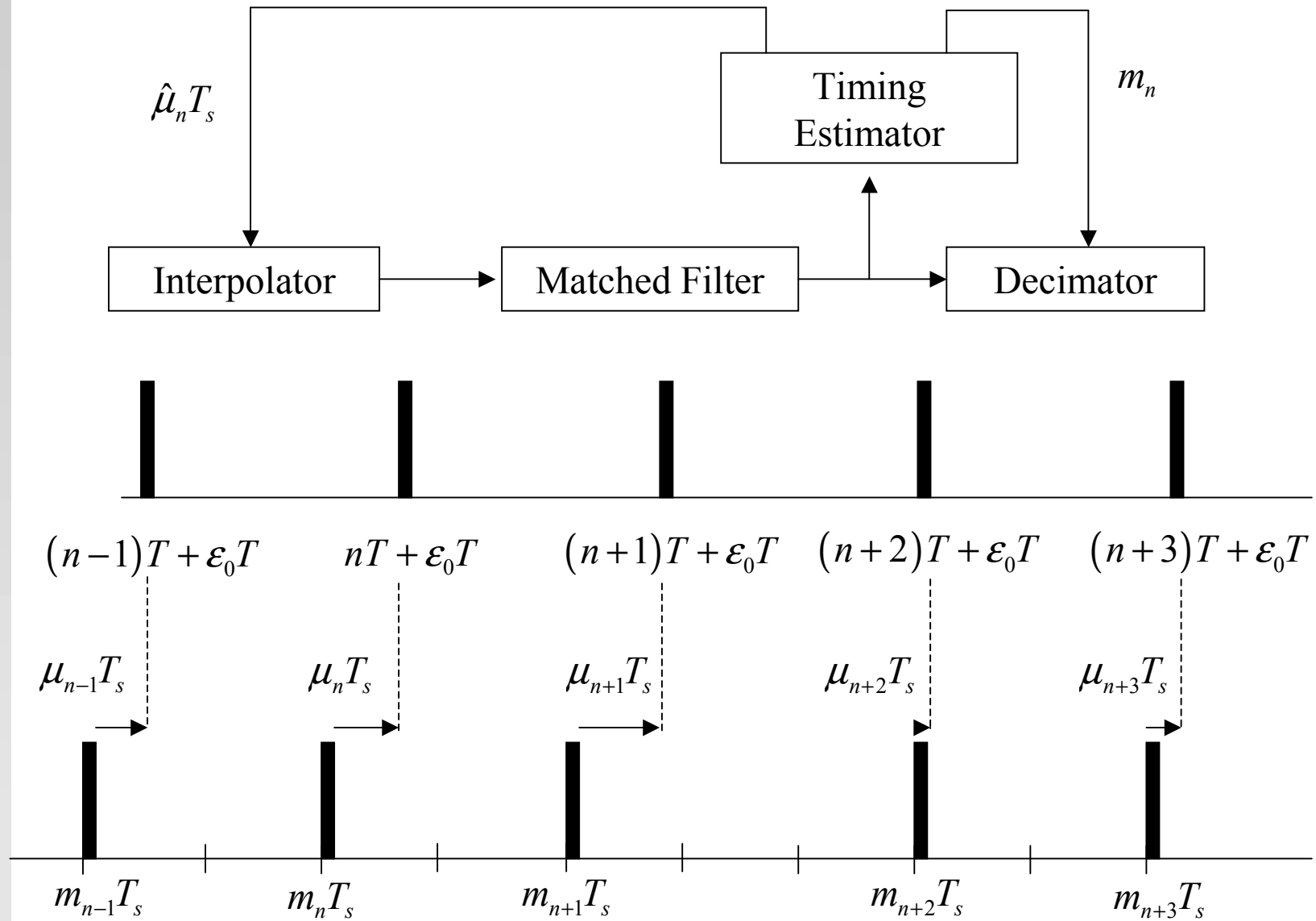
Sampling is asynchronous with received signal.

Symbol rate and sampling rate are incommensurate.

Shifted samples at the matched filter output ( $kT + \hat{\epsilon}T$ ) must be obtained from the signal samples at rate  $1/T_s$



$$nT + \hat{\epsilon}T = \left[ n \frac{T}{T_s} + \hat{\epsilon} \frac{T}{T_s} \right] T_s = (m_n + \hat{\mu}_n) T_s$$



## TIMING ESTIMATION

- The carrier phase is known

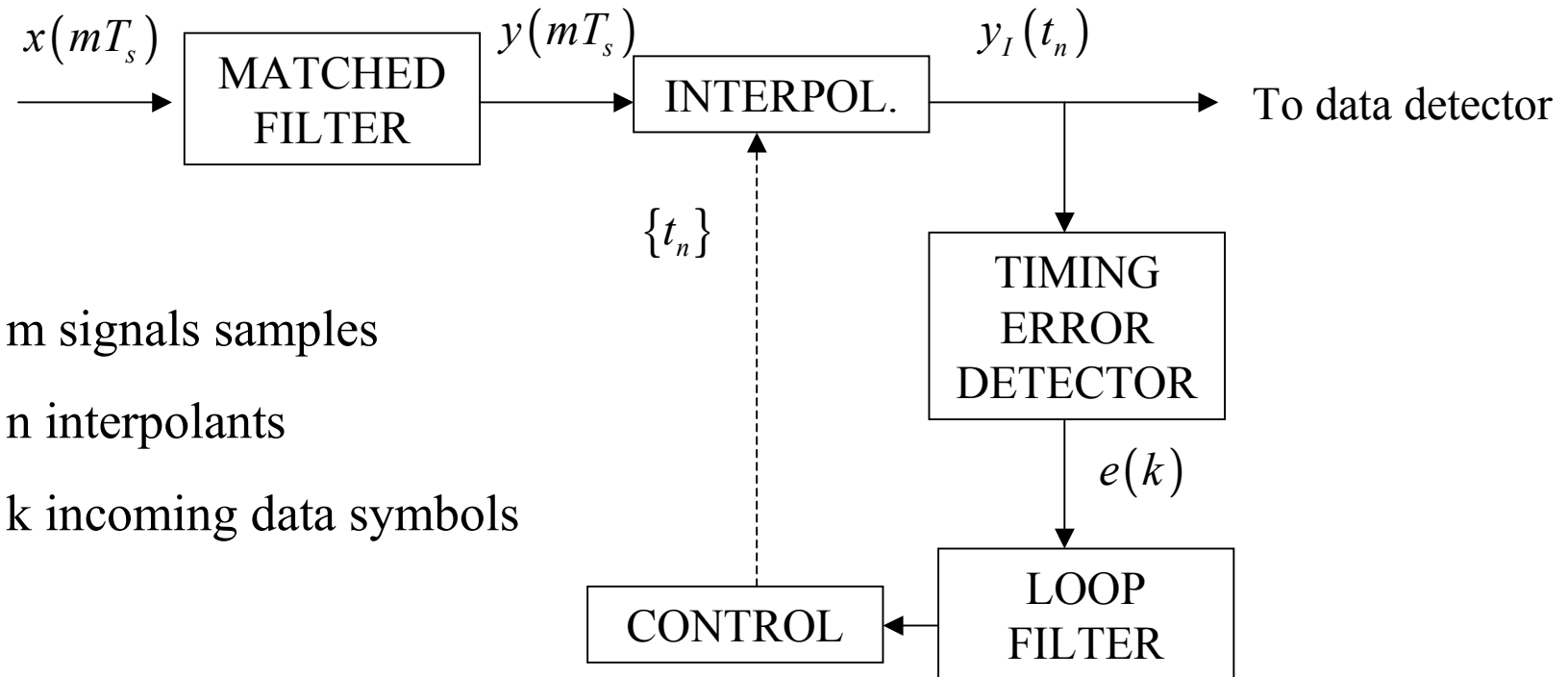
The signal is converted into the IQ components: BASEBAND

- The carrier phase is not known and jointly recovered with timing in a DD approach
- The carrier phase is not known but is recovered later, in a clock-aided system

NDA feedback

NDA feedforward

## TIMING RECOVERY IN BASEBAND TRANSMISSION



m signals samples

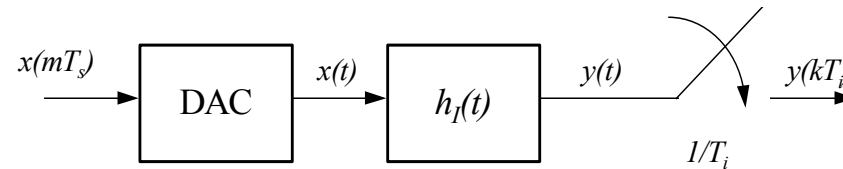
n interpolants

k incoming data symbols

$$e(k) = S(\tau - \hat{\tau}_k) + N(k)$$

## DIGITAL INTERPOLATORS

- The digital interpolation can be described as an DAC plus a re-sampling at the output of an analog interpolation filter:



- The input of the sampler is given by:  $y(t) = x(t) * h_I(t) = \sum_m x(mT_s) \delta(t - mT_s) * h_I(t) = \sum_m x(mT_s) h_I(t - mT_s)$

- After the signal sampling:  $y(kT_i) = \sum_m x(mT_s) h_I(kT_i - mT_s)$

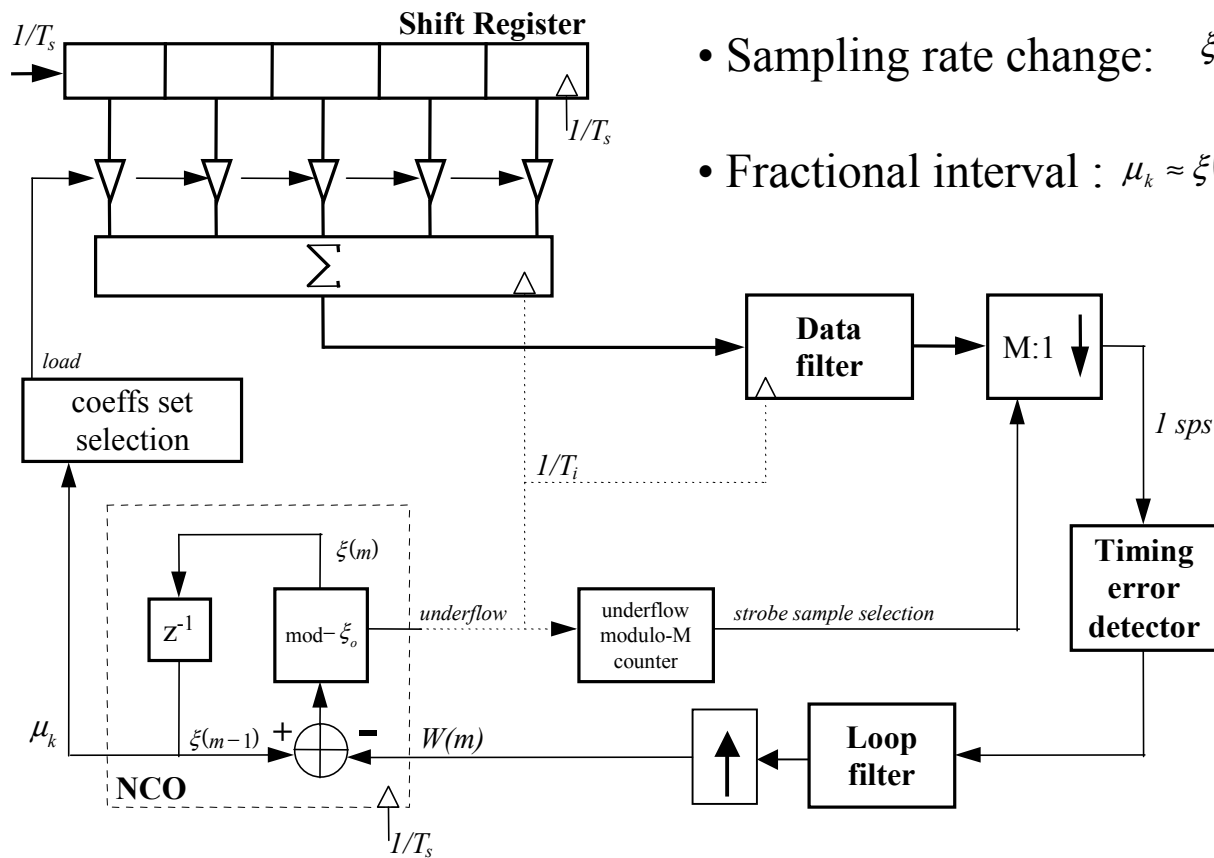
- Defining : Basepoint index:  $m_k = \text{int} \left[ \frac{kT_i}{T_s} \right]$       Fractional index:  $\mu_k = \left[ \frac{kT_i}{T_s} \right] - m_k \Rightarrow \mu_k \in [0,1]$

$$i = m_k - m \quad \boxed{y(kT_i) = y[(m_k + \mu_k)T_s] = \sum_i x[(m_k - i)T_s] h_I[(i + \mu_k)T_s]}$$

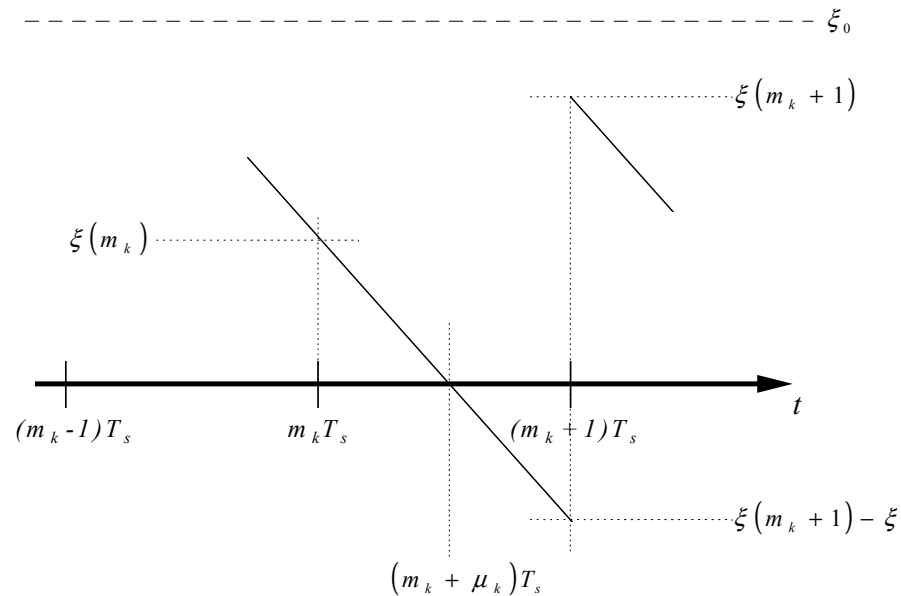
↑
↑  
 Base pointer (integer)      Fractional pointer (real between 0..1)

# NCO INTERPOLATOR CONTROL

- NCO equation:  $\xi[m] = [\xi[m-1] - W[m-1]] \bmod \xi_o$
- $W$  : Control word (step of the counter)
- Sampling rate change:  $\xi_o \approx \frac{T_i}{T_s}$
- Fractional interval :  $\mu_k \approx \xi(m_k)$



## FRACTIONAL EQUATION



$$\xi(m_k T_s + \mu_k T_s) = 0 = \xi(m_k) + \mu_k (\xi(m_k + 1) - \xi_0 - \xi(m_k))$$

$$\mu_k = \frac{\xi(m_k)}{\xi(m_k) - [\xi(m_k + 1) - \xi_0]} = \frac{\xi(m_k)}{W(m_k)}$$

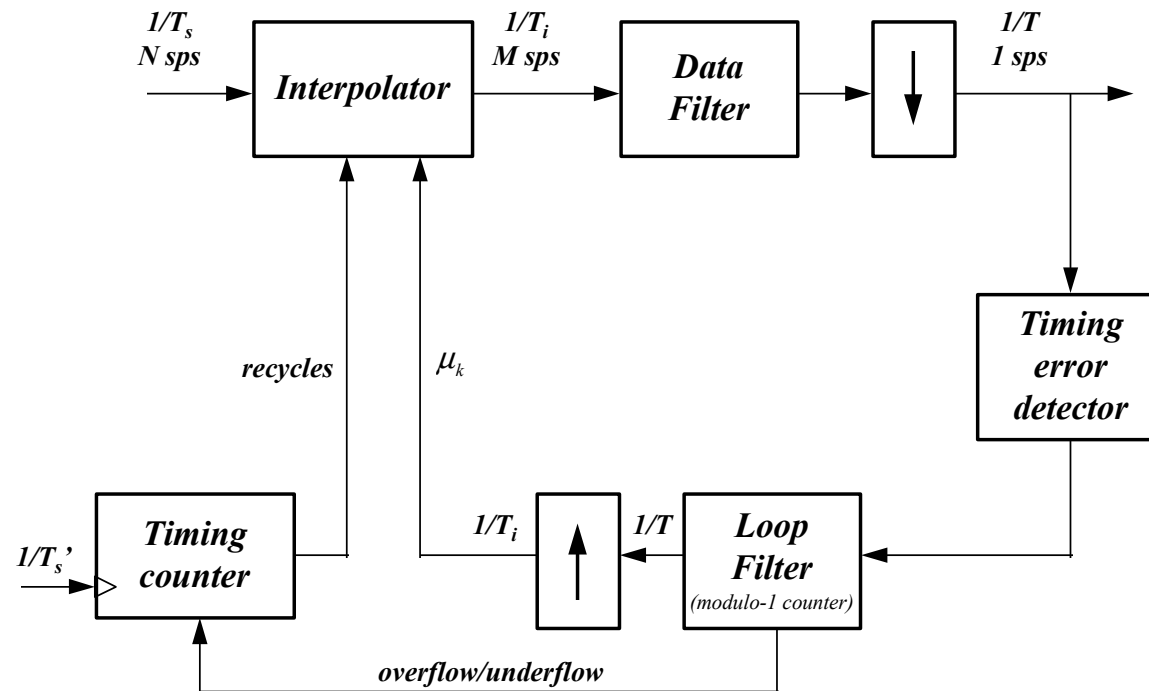
$$\mu_k \approx \xi(m_k)$$



## INTEGER RATIO ALGORITHM

Algorithm for  $T_i/T_s$  INTEGER :

- Input sampling rate  $1/T_s$ .
- At each counter recycle a new interpolated sample is computed.
- After  $\xi_o$  clock 'ticks' a new counter recycle is done.
- Underflow of  $\mu_k$  recycle in  $\xi_o - 1$
- Overflow of  $\mu_k$  recycle in  $\xi_o + 1$



## FARROW INTERPOLATOR (I)

- The impulse response of the interpolator is polynomial:

$$h_I(t) = \begin{cases} \sum_{l=0}^N a_l t^l & |t| \leq \frac{T_0}{2} \\ 0 & |t| > \frac{T_0}{2} \end{cases} \quad \begin{matrix} T_0 = LT_s \\ \text{L length in symbols of the interpolator} \end{matrix}$$

- The interpolation equation is given by:

$$y(kT_s) = y[(m_k + \mu_k)T_s] = \sum_i x[(m_k - i)T_s] h_I[(i + \mu_k)T_s]$$

- Sampling the interpolator impulse response:

$$h_I((i + \mu_k)T_s) = \sum_{l=0}^N a_l T_s^l (i + \mu_k)^l = \sum_{l=0}^N b_l(i) \mu_k^l$$

- And substituting in the previous equation:

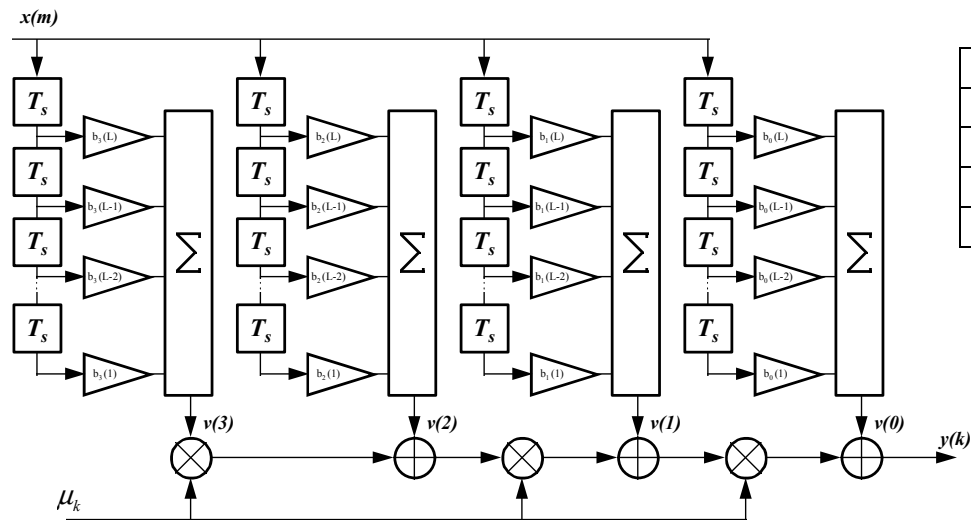
$$\begin{aligned} y(kT_s) &= \sum_i x[(m_k - i)T_s] h_I[(i + \mu_k)T_s] = \sum_i x[(m_k - i)T_s] \sum_{l=0}^N b_l(i) \mu_k^l = \\ &= \sum_{l=0}^N \mu_k^l \sum_i b_l(i) x[(m_k - i)T_s] = \sum_{l=0}^N \mu_k^l v(l) \end{aligned}$$

where we have defined:

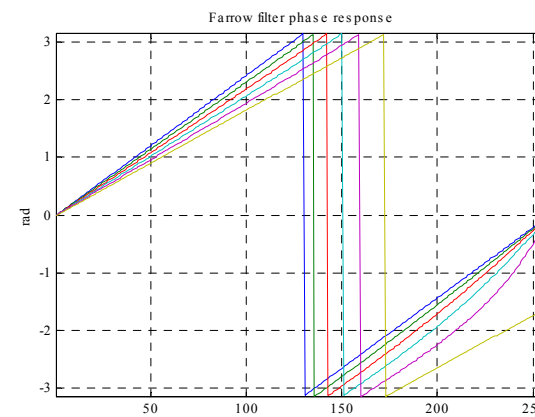
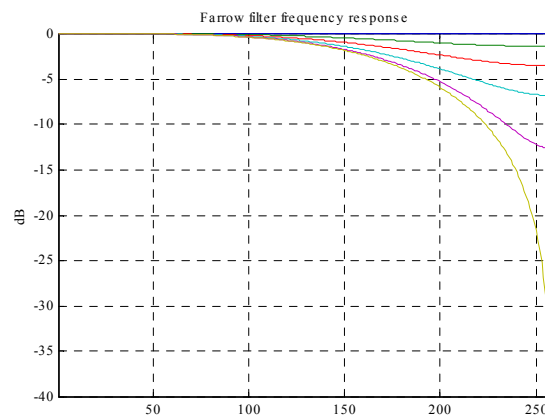
$$v(l) = \sum_i b_l(i) x[(m_k - i)T_s] \quad \text{Time-invariant FIR}$$

# FARROW INTERPOLATOR (II)

- CUBIC Interpolator :  $y[k] = ((v(3) \mu_k + v(2)) \mu_k + v(1)) \mu_k + v(0)$



i	l=0	l=1	l=2	l=3
-2	0	-1/6	0	1/6
-1	0	1	1/2	-1/2
0	1	-1/2	-1	1/2
1	0	-1/3	1/2	-1/6



**SIGNAL PROCESSING IN COMMUNICATIONS GROUP**  
**DEPARTMENT OF SIGNAL THEORY AND COMMUNICATIONS**

$$s_i = \sum_{k=0}^N a_k t_i^k = \mathbf{t}^H \mathbf{a} \Rightarrow \mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ \mathbf{M} \\ a_N \end{bmatrix} \quad \mathbf{t}_i = \begin{bmatrix} 1 \\ t_i \\ \mathbf{M} \\ t_i^N \end{bmatrix}$$

$$\phi^2 = \sum_{i=1}^p (s_i - \mathbf{t}_i^H \mathbf{a})^2 = (\mathbf{s} - \mathbf{T}\mathbf{a})^H (\mathbf{s} - \mathbf{T}\mathbf{a})$$

$$\frac{\partial}{\partial \mathbf{a}^H} \phi^2 = -2\mathbf{T}^H (\mathbf{s} - \mathbf{T}\mathbf{a}) = \mathbf{0} \quad \mathbf{T} = [\mathbf{t}_0 \quad \mathbf{t}_1 \quad \Lambda \quad \mathbf{t}_N]$$

$$\mathbf{a} = (\mathbf{T}^H \mathbf{T})^{-1} \mathbf{T}^H \mathbf{s} = \mathbf{M}\mathbf{s}$$

$$\mathbf{M} = \begin{bmatrix} m_{01} & m_{02} & \Lambda & m_{0P} \\ m_{11} & m_{12} & \Lambda & m_{1P} \\ \mathbf{M} & & & \mathbf{M} \\ m_{N1} & \Lambda & \Lambda & m_{NP} \end{bmatrix}$$

## DD TIMING ERROR DETECTORS

### ML-BASED DETECTOR:

$$\Lambda(\mathbf{r} | \vartheta) = \exp \left\{ \frac{2}{N_0} \int_0^{T_0} r(t) s(t) dt - \frac{1}{N_0} \int_0^{T_0} s(t)^2 dt \right\}$$

$$s(t) = \sum_i \vartheta_i g(t - iT - \vartheta)$$

$$\Lambda(\mathbf{r} | \vartheta) = \left\{ \sum_i \vartheta_i \int_{T_0} r(t) g(t - iT - \vartheta) dt - \frac{1}{2} \sum_i \sum_n \vartheta_i \vartheta_n \int_{T_0} g(t - nT - \vartheta) g(t - iT - \vartheta) dt \right\}$$

$$\frac{d\Lambda(\mathbf{r} | \vartheta)}{d\vartheta} = - \sum_i \vartheta_i \int_{T_0} r(t) g'(t - iT - \vartheta) dt + \sum_i \sum_n \vartheta_i \vartheta_n \int_{T_0} g'(t - iT - \vartheta) g(t - nT - \vartheta) dt$$

$$\frac{d\Lambda(\mathbf{r} | \vartheta)}{d\vartheta} = \sum_{n=0}^{L_0-1} \vartheta'_n \left[ y'(iT + \vartheta) - \sum_{n=-\infty}^{\infty} \vartheta'_n h'((i-n)T) \right]$$

$$r(t) \longrightarrow \boxed{-g'(-t)} \longrightarrow y'(t) @ \int_{-\infty}^{\infty} r(\xi) g'(\xi - t) d\xi$$

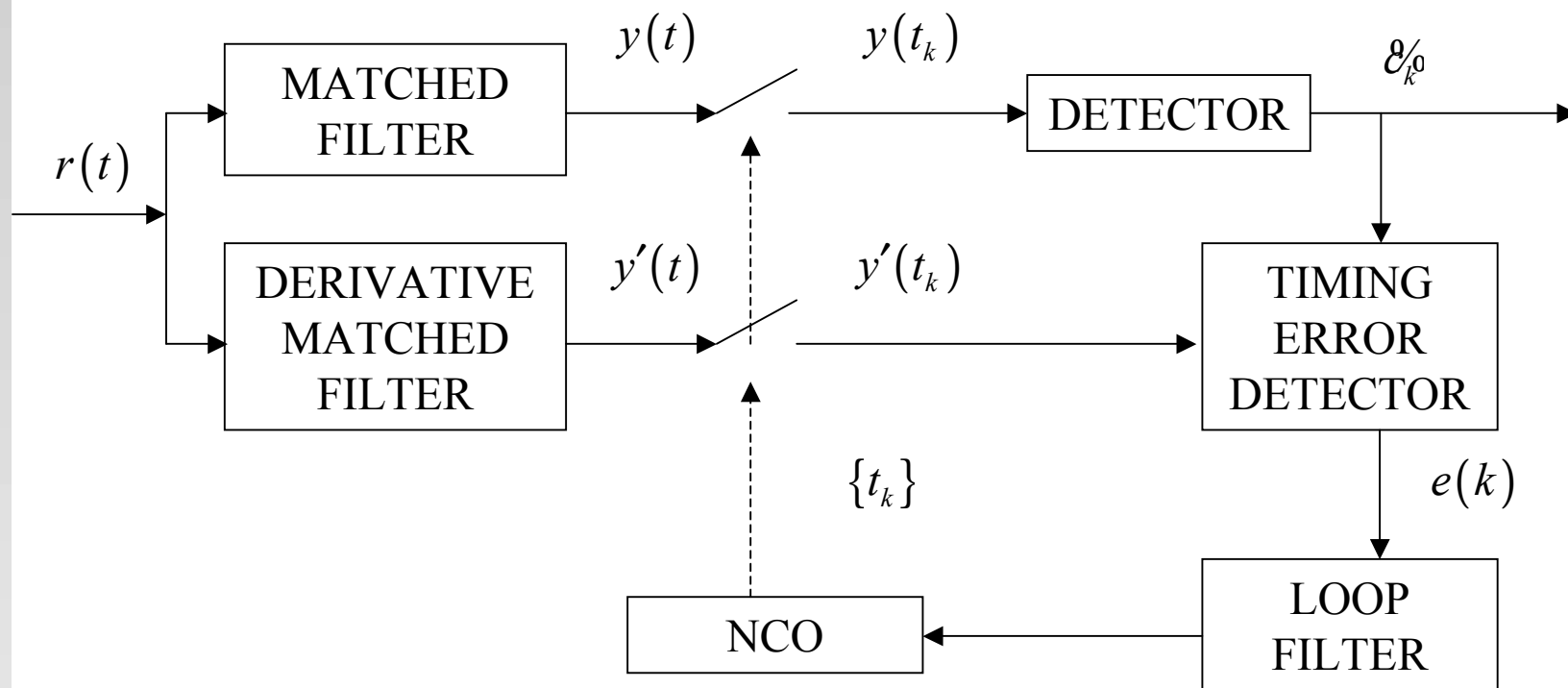
$$-g'(-t) \longrightarrow \boxed{g(t)} \longrightarrow h'(t) @ g(t) * [-g'(-t)]$$

$$e(k) = \vartheta'_k \left[ y'(kT + \vartheta_k) - \sum_n \vartheta'_n h'((k-n)T) \right]$$

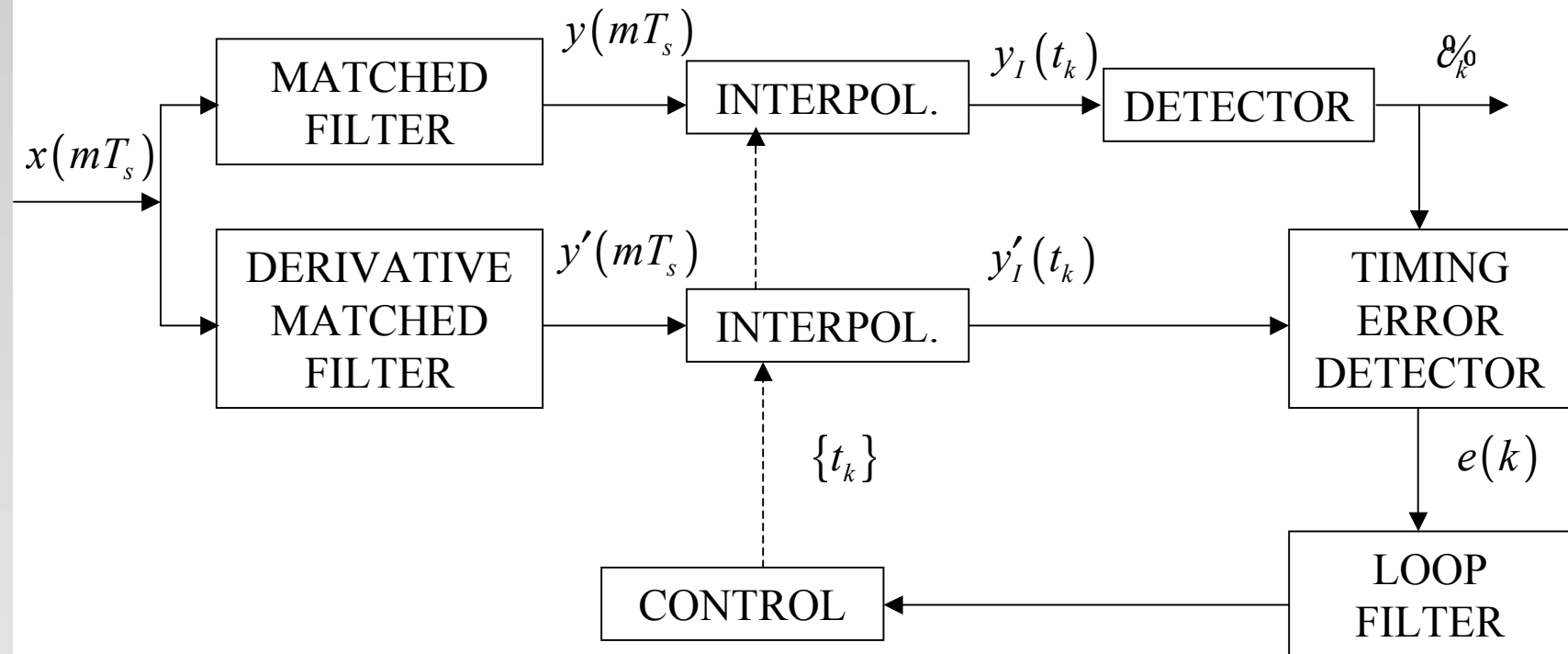
$$e(k) = \vartheta'_k \left[ y'(kT + \vartheta_k) - \sum_{n=k-D}^{k+D} \vartheta'_n h'((k-n)T) \right] \quad \text{True ML error detector by Gardner}$$

$$\boxed{e(k) = \vartheta'_k y'(kT + \vartheta_k)} \quad \left\{ \begin{array}{l} D=0 \\ \frac{1}{2N_0} \int_0^{T_0} |g(t)|^2 dt \quad \text{Independent of } \vartheta \end{array} \right.$$

## TIMING RECOVERY WITH SYNCHRONOUS SAMPLING



# TIMING RECOVERY WITH NON-SYNCHRONOUS SAMPLING





## EARLY-LATE DETECTOR (ELD)

$$y'(kT + \vartheta_k^o) \approx \frac{1}{T} \left[ y(kT + T/2 + \vartheta_{k+1/2}^o) - y(kT - T/2 + \vartheta_{k-1/2}^o) \right]$$

$$\vartheta_{k+1/2}^o \rightarrow \vartheta_k^o$$

$$\vartheta_{k-1/2}^o \rightarrow \vartheta_{k-1}^o$$

$$e(k) = \vartheta_k^o \left[ y(kT + T/2 + \vartheta_k^o) - y(kT - T/2 + \vartheta_{k-1}^o) \right]$$

$$\vartheta_{k+1}^o = \vartheta_k^o + \gamma e(k)$$

## MUELLER AND MUELLER DETECTOR (MMD)

$$e(k) = \hat{c}_{k-1} y(kT + \hat{t}_k) - \hat{c}_k y((k-1)T + \hat{t}_{k-1})$$

$$y(t) = \sum_i c_i h(t - iT - \hat{t}_i) + n(t)$$

Assuming:

-small tracking errors  $\hat{t}_k \approx \hat{t}_{k-1} \approx \tau$

-correct decisions  $\hat{c}_k \approx c_k$

-negligible thermal noise

$$e(k) = \sum_i c_{k-1} c_i h((k-i)T) - \sum_i c_k c_i h((k-1-i)T) = 0$$

$$h(0) = 1$$

$$h(kT) = 0 \quad \text{for } k \neq 0$$



## NDA TIMING ERROR DETECTORS

### ML-BASED DETECTOR

$$\Lambda(\mathbf{r} | \theta, \phi) = \exp \left\{ \frac{2}{N_0} \int_0^{T_0} r(t) s(t) dt - \frac{1}{N_0} \int_0^{T_0} s(t)^2 dt \right\}$$

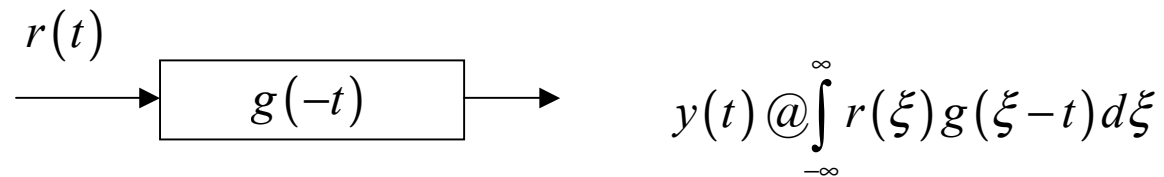
$$s(t) = \sum_i \theta_i g(t - iT - \theta)$$

Low SNR:

$$\Lambda(\mathbf{r} | \theta, \phi) \approx 1 + \frac{2}{N_0} \int_0^{T_0} r(t) s(t) dt + \frac{2}{N_0^2} \left[ \int_0^{T_0} r(t) s(t) dt \right]^2$$

$$\int_0^{T_0} r(t) s(t) dt = \sum_i \theta_i \int_0^{T_0} r(t) g(t - iT - \theta) dt$$

$$\int_0^{T_0} r(t) \delta(t) dt \approx \sum_{i=0}^{L_0-1} \delta_i y(iT + \hat{t}) \quad L_0 \approx T_0 / T$$



Performing the expectation with respect to the symbols:

$$\Lambda(\mathbf{r} | \hat{t}) \approx \sum_{i=0}^{L_0-1} y^2(iT + \hat{t})$$

The optimum timing estimate corresponds to the sampling epoch  $\hat{t}$  that maximizes the energy of the sequence  $\{y(iT + \hat{t})\}$

$$\Lambda'(\mathbf{r} | \theta) \approx 2 \sum_{i=0}^{L_0-1} y(iT + \theta) y'(iT + \theta)$$

$$e(k) \approx y(kT + \theta_k) y'(kT + \theta_k)$$

$$\theta_{k+1} = \theta_k + \gamma e(k)$$

## NDA-ELD

$$e(k) \approx y(kT + \theta_k) [y(kT + T/2 + \theta_k) - y(kT - T/2 + \theta_{k-1})]$$

## GARDNER DETECTOR

$$e(k) \approx \{y[(k-1)T + \theta_{k-1}] - y(kT + \theta_k)\} y(kT - T/2 + \theta_{k-1})$$

# FEEDFORWARD ESTIMATION SCHEMES

## NDA ML BASED ALGORITHM

$$\Lambda(\mathbf{r} | \vartheta, \phi) = \exp \left\{ \frac{2T_s}{N_0} \sum_{k=0}^{NL_0-1} x(kT_s) \vartheta(kT_s) - \sum_{k=0}^{NL_0-1} \vartheta^2(kT_s) \right\}$$

$$\vartheta(t) = \sum_i \vartheta_i g(t - iT - \vartheta)$$

$$\Lambda(\mathbf{r} | \vartheta) \approx \sum_{k_1=0}^{NL_0-1} \sum_{k_2=0}^{NL_0-1} x(k_1 T_s) x(k_2 T_s) F(k_1, k_2, \vartheta)$$

$$F(k_1, k_2, \vartheta) = \sum_i g(k_1 T_s - iT - \vartheta) g(k_2 T_s - iT - \vartheta) \quad \text{Periodic function of } \vartheta \text{ of period } T$$

$$F(k_1, k_2, \vartheta) = \sum_m F_m(k_1, k_2) e^{j2\pi m \vartheta / T}$$

$$F_m(k_1, k_2) = \frac{1}{T} \int_0^T F(k_1, k_2, \vartheta) e^{-j2\pi m \vartheta / T} d\vartheta$$

Dem: If the signal is bandlimited to  $1/T$   $F_m(k_1, k_2) = 0, |m| \geq 2$

$$t = k_1 T_s - iT - \nu \quad N @ T / T_s$$

$$F_m(k_1, k_2) = \frac{1}{T} e^{-j2\pi m k_1 / N} \sum_i \int_{k_1 T_s - (i+1)T}^{k_1 T_s - iT} g(t) g[t - (k_1 - k_2) T_s] e^{-j2\pi m \tau t / T} dt$$

$$\frac{1}{T} e^{-j2\pi m k_1 / N} \underbrace{\int_{-\infty}^{\infty} g(t) g[t - (k_1 - k_2) T_s] e^{-j2\pi m \tau t / T} dt}_{\int_{-\infty}^{\infty} G(\nu) G[f - \nu] e^{-j2\pi(k_1 - k_2)(f - \nu) T_s} d\nu \Big|_{f = -m/T}}$$

$$F_m(k_1, k_2) = \frac{1}{T} e^{-j2\pi m k_2 / N} \int_{-\infty}^{\infty} G(\nu) G^*[\nu + m/T] e^{j2\pi(k_1 - k_2)\nu T_s} d\nu$$

If  $G(f)$  is bandlimited to  $1/T$   $F_m(k_1, k_2) = 0, |m| \geq 2$



$$F_1(k_1, k_2) = \frac{1}{T} e^{-j2\pi k_2/N} \int_{-\infty}^{\infty} G(v) G^* [v + 1/T] e^{j2\pi(k_1 - k_2)vT_s} dv$$

$$F_1(k_1, k_2) = \frac{1}{T} e^{-j\pi(k_1 + k_2)/N} \int_{-\infty}^{\infty} \underbrace{G\left(f - \frac{1}{2T}\right) G^*\left(f + \frac{1}{2T}\right)}_{Q(f)} e^{j2\pi(k_1 - k_2)fT_s} df$$

$$\uparrow$$

$$v = f - \frac{1}{2T}$$

$$Q(f)$$

$$F_1(k_1, k_2) = \frac{1}{T} q[(k_1 - k_2)T_s] e^{-j\pi(k_1 + k_2)/N}$$

$$F(k_1, k_2, \vartheta) = F_0(k_1, k_2) + 2 \operatorname{Re} \left\{ F_1(k_1, k_2) e^{j2\pi\vartheta T} \right\}$$

$$\Lambda(\mathbf{r} | \vartheta) \approx \operatorname{Re} \left\{ e^{j2\pi\vartheta T} \sum_{k_1=0}^{NL_0-1} \sum_{k_2=0}^{NL_0-1} x(k_1 T_s) x(k_2 T_s) F_1(k_1, k_2) \right\}$$

$$\vartheta = \frac{T}{2\pi} \arg \left\{ \sum_{k_1=0}^{NL_0-1} \sum_{k_2=0}^{NL_0-1} x(k_1 T_s) x(k_2 T_s) F_1(k_1, k_2) \right\}$$

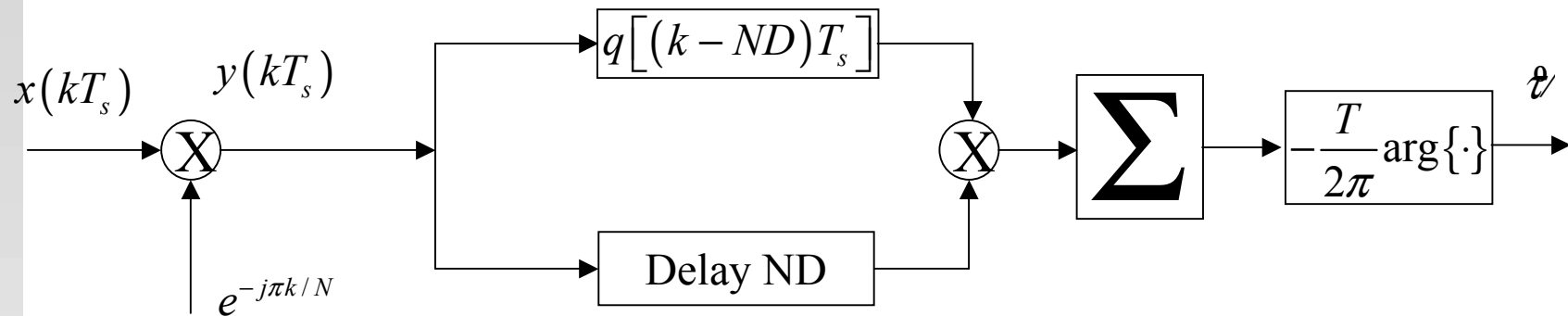
$$\vartheta = -\frac{T}{2\pi} \arg \left\{ \sum_{k=0}^{NL_0-1} y(k T_s) z(k T_s) \right\}$$

$$y(k T_s) @ x(k T_s) e^{-j2\pi k / N}$$

$$z(k T_s) @ \sum_{k_2=0}^{NL_0-1} y(k_2 T_s) q[(k - k_2) T_s]$$

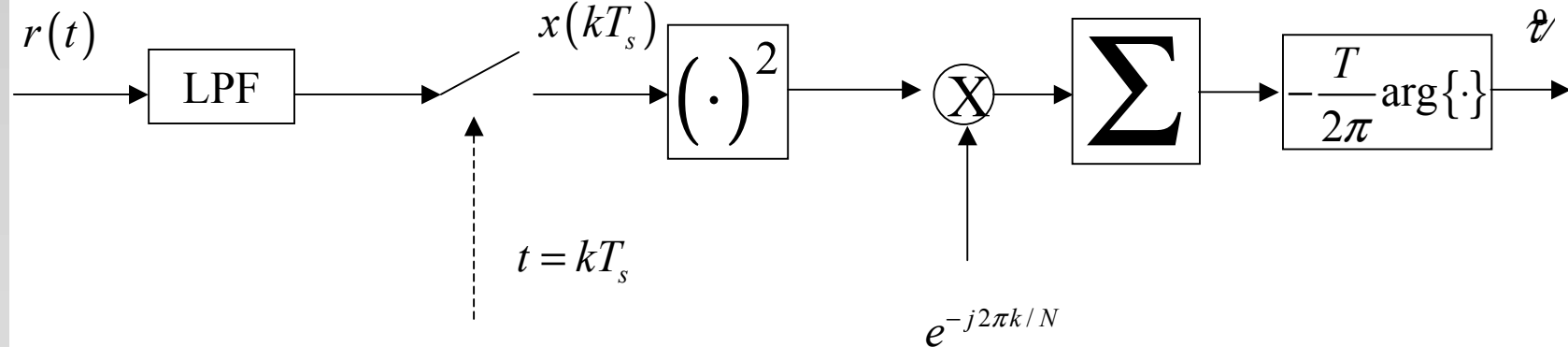
$$z[(k - ND)T_s] = y(kT_s) * q[(k - ND)T_s]$$

$$\vartheta = -\frac{T}{2\pi} \arg \left\{ \sum_{k=ND}^{N(L_0+D)-1} y[(k - ND)T_s] z[(k - ND)T_s] \right\}$$



Oversampling N=2

## OERDER AND MEYR ALGORITHM



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

Oversampling  $N=4$

With rolloff=0.75 the ML based algorithm and the Oerder-Meyr algorithm are essentially equivalent and close to the MCRB.

With rolloff=0.25 the Oerder-Meyr algorithm is slightly inferior and the differences tend to increase with the alphabet sizes.