Faculty of Engineering of University of Porto

State of the Art

Adaptive Equalization of Interchip Communication

Author:
Dénis Gaspar Nogueira da Silva

Supervisors:
Prof^{o} Dr^{o}Henrique Salgado
Eng^{o}Luis Moreira @Synopsys
Eng^{o}Sérgio Silva @Synopsys

Department of Electrical Engineering
Faculty of Engineering
University of Porto

Contents

List of Figures

Chapter 1

Introduction

Modern communication systems are based on digital transmission, the primary advantage over analogue systems is that digital signals are easier to regenerate and are more insensible to noisy communication channels.
As link communication speed increases problems derived from channel distortion and multipath become a major issue as they cause Inter Symbol Interference.
Inter Symbol Interference is one of the major obstacles in reliable high speed data communication,so the study of its compensation or elimination becomes crucial.
Equalization is the technique used to reduce the influence of channel distortion in the detection of the information,however equalization must be adaptive as channel characteristics are unknown or time varying.
This thesis has the objective to study different methods of digital and analogue equalization and their applicability in very high speed communication systems such as inter chip communication.

Chapter 2

State of the Art

Inter symbol Interference

Several aspects degrade the probability of error in a communication channel,the channel itself has associated with it a frequency response usually similar to a low pass filter.
The cut-off frequency associated to the channel is a function of the channel characteristics,material of the guiding material ,length,temperature etc.
The maximum throughput that can be achieve in a communication is given by the Shannon?Hartley theorem.

$$C = B\log_2(1 + \frac{S}{N})$$

The theorem states that the channel capacity C in $bits/s$ is a function of the channel Bandwidth in Hz and the signal to noise ratio.
So from the previous equation we can see that we want to make the channel bandwidth as higher as possible so we can achieve the maximum throughput possible.
The channel transfer function is usually represent by means of the Fourier transform,in the next figure we can view the usual low characteristic of a communication channel.

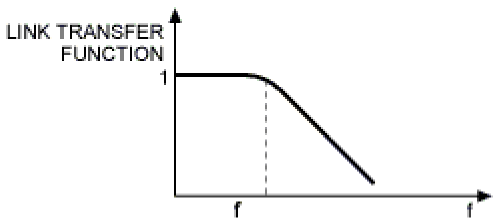

FIGURE 2.1: Channel Low pass Transfer Function

The channel can both introduce Amplitude and Phase distortion altering the transmitted bits. For no amplitude distortion the channel transfer function $H_c(j\omega)$ must be flat for the entire signal spectre.

$$| Hc(j\omega) |= K$$

For no phase distortion the channel group delay must be constant:

$$-\frac{\partial\theta_c(j\omega)}{\partial\omega} = K$$

As the channel introduces distortion the transmitted waveforms become attenuated and the dispersion of the signal causes the duration of the bit to be extended beyond T seconds.The dispersion of the signal can be caused by the distortion or by Multipath inside the channel.

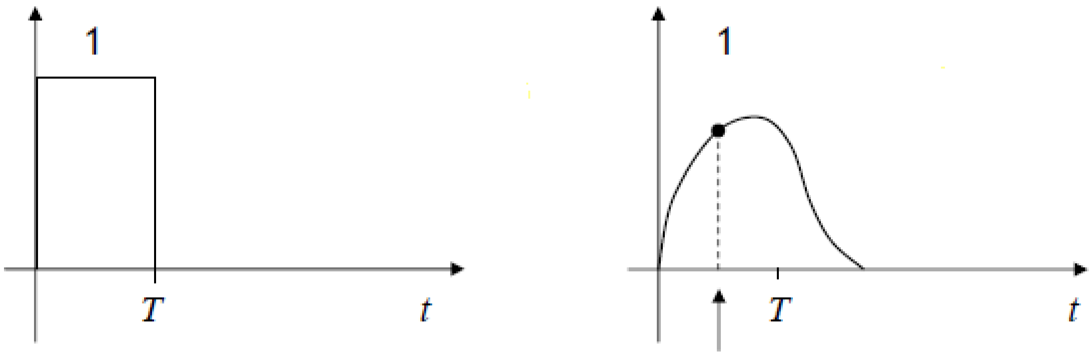

FIGURE 2.2: Bit distortion

When a sequence is transmitted through a channel that introduces dispersion inter Symbol Interference appears.

Inter symbol interference results from the distortion of the waveform of the transmitted bits because bits interfere with the amplitude of neighbouring bits in the sampling instant.ISI can cause the decisor to make incorrect decisions on the detection of a sequence.

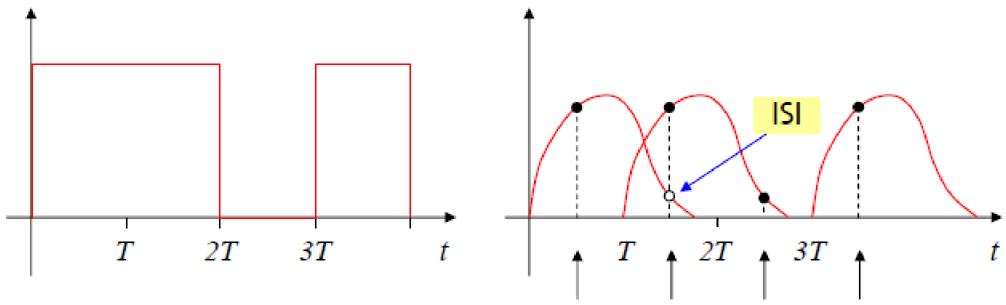

FIGURE 2.3: Inter Symbol Interference

2.1 Nyquist condition

Lets consider the channel as linear and time invariant, so the channel can be represented by its impulse response:

$$H_c(n) = \sum_{k=0}^{+L} h_k \delta(n-k)$$

Where $\delta(n)$ represents the Dirac pulse. The sequence observed after the channel results from the linear convolution of the input sequence and the channel impulse response:

$$y(n) = x(n) * h_c(n)$$

$$y(n) = \sum_{k=-\infty}^{+\infty} x(n) h_c(n-k)$$

So for no ISI one must satisfy the Nyquist criteria:

$$y(nT) = \begin{cases} c & \text{if } x = 0 \\ 0 & \text{if } x \neq 0 \end{cases}$$

Nyquist criteria states that for no ISI the bit waveform can only be different from zero at is own sample time and equal to zero in other sample time,this represents the **Time domain Nyquist Criteria**.

Considering that the received waveform $y(t)$ equals:

$$y\delta(t) = \sum_{n=-\infty}^{+\infty} y(nT)\delta(t-nT)$$

$y\delta(t)$ represents the sampled received signal.
Taking the Fourier transform we get:

$$Y\delta(f) = \frac{1}{T} \sum_{n=-\infty}^{+\infty} Y(f - \frac{n}{T})$$

Because $Y\delta(t) = \delta(t) \longrightarrow Y\delta(f) = 1$

$$\sum_{n=-\infty}^{+\infty} Y(f - \frac{n}{T}) = T$$

This equation represents the **Frequency domain Nyquist Condition** and it states that for no ISI the folded spectrum of the received signal must be constant.

Assuming a band limited channel to Whz the Nyquist condition has the following implications:

- Suppose that the symbol rate is so high that $1/T > 2W$ no matter how the received spectrum looks like it will always be gaps between spectrum copies and ISI is inevitable.
- If the data rate is slower that $1/T = 2W$ the copies of X(f) will overlap and there is many options for X(f) that make the folded spectrum $\sum_{n=-\infty}^{+\infty} Y(f - \frac{n}{T})$ flat .

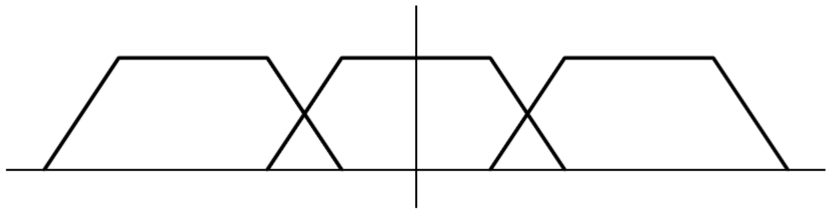

FIGURE 2.4: Overlapping spectrum in the case of $1/T < 2W$

- If $1/T = 2W$ Then the spectral copies of the bit waveform just touch and in order for not to exist ISI the spectrum $Y(f)$ must be rectangular.
  Rectangular spectres can only be achieve by shaping the bits as sync pulses.
  The rate $1/T = 2W$ is know as the Nyquist Rate and imposes the maximum theoretical bit rate that can be transmitted in a channel with bandwidth W.

Strategies to fight ISI

2.2 Fight ISI with bit shaping

We can deduce from the frequency Nyquist condition that the bit shape can help fight the effects of ISI in communication systems.
The more we compress the signalling spectrum the higher the data rate we can transmit because ideally we want to reduce the bandwidth required for some communication link,the spectrum compression is achieved by applying a Nyquist filter to the transmitting waveform. A widely used ISI free is the raised cosine pulse:

$$X(f) = \begin{cases} T & \text{for } 0 \leq\mid f \mid\leq \frac{1-\alpha}{2T} = 0 \\ \frac{T}{2}[1 + \cos\frac{\pi T}{\alpha}(\mid f \mid - \frac{1-\alpha}{2T})] & \text{for} \frac{1-\alpha}{2T} \leq\mid f \mid\leq \frac{1+\alpha}{2T} \\ 0 & \text{for } \mid f \mid > \frac{1+\alpha}{2T} \end{cases}$$

α is the roll-of factor which determines the excess bandwidth from the original rectangular pulse 1/2T The time corresponding function is:

$$x(t) = \frac{\sin\frac{\pi t}{T}}{\frac{\pi t}{T}} \frac{\cos\frac{\pi \alpha t}{T}}{1 - \frac{4\alpha^2 t^2}{T^2}}$$

if $\alpha = 0$ the raised cosine pulse becomes the sync function.

Notice that in the multiples of the sample period the waveform passes through zero resulting in zero ISI.

The roll-of factor decreases the ondulation after the time bit, this ondulation may be harmful for ISI if the sample time suffers from jitter.

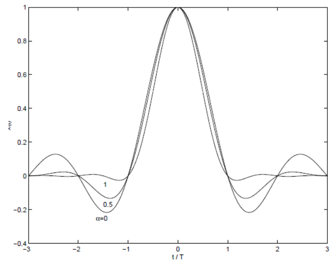

FIGURE 2.5: Raised cosine waveform

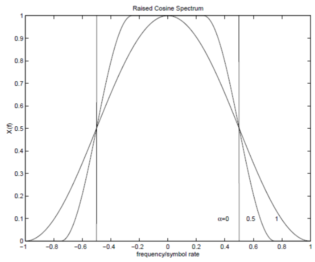

FIGURE 2.6: Raised cosine spectrum

Notice that in the multiples of the sample period the waveform passes through zero resulting in zero ISI.
Nyquist filters are hard to realize in practice because it has an impulse response witch approaching infinity.
The roll-of factor decreases the ondulation after the time bit, this ondulation may be harmful for ISI if the sample time suffers from jitter.

2.3 Maximum length Sequence Estimation

In the maximum likelihood receiver the samples are not modified or reshaped by the receiver ,instead using MLSE the receiver adjusts itself to better deal with the distorted samples The MLSE receiver uses an estimate of the channel modelled as a finite input response (FIR) filter to compute the most likely transmitted sequence.

Let us consider the following channel model:

FIGURE 2.7: Discrete channel model.

Where x represents the transmitted sequence w represents white Gaussian noise $\mathcal{N}(0, \delta^2)$added to the channel.
y represents the output of the channel with impulse response h of length L.

So from the above figure we can write $z = y + w$ and $y = x * h$ resulting in :

$$yi = h0yi + \sum_{j=1}^{L}(hjx_{i-j})$$

Where the summation represents the inter symbol interference. In the MLSE receiver we want to maximize the following expression:

$$P(Z \mid U^{(m*)}) = maxP(Z \mid U^{(m)})$$

Meaning we want determine received sequence z that maximizes the probability $P(Z \mid U^{(m)})$ U(m) represents a possible transmitted sequence.
In the case of binary transmission and in the case o transmited sequence of size M. We have 2^L possible sequences so the computational complexity increases exponentially with the sequence length M,making it impossible to use in real applications.

2.3.1 Viterbi's Algorithm

Viterbi's algorithm uses a simplification of the MLSE algorithm by taking advantage of a special structure called Trellis.
The advantage of a Viterbi decoder compared with the original MLSE is that the complexity of the algorithm is not a function of the sequences length .
The algorithm involves calculating a measure of similarity or distance between the received signal $Z(ti)$ and all the trellis paths entering each state at time ti,discarding those trellis paths that could not possibly be candidates to the maximum likelihood sequence. When two paths enter the same state, the path with best metric is chosen,the method is repeated for all the received bits.

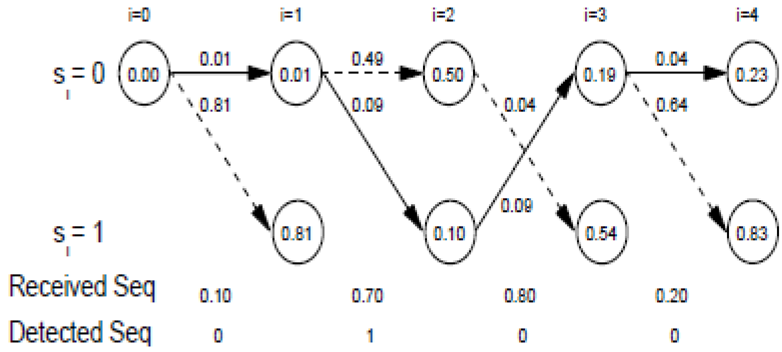

FIGURE 2.8: Graphical illustration of the Viterbi's algorithm

In the diagram the dotted arrows represent the reception of a bit 0 and the other the reception of a bit 1.
The problem of Viterbi's algorithm is that it requires knowledge of the channel transfer function for the calculation of the weight of each path.
The number of operations in the method grows linearly with the size of the sequence l, but the problem is the computation required to store and compute all the paths for the received sequence as the number of states increases with the size of the channel impulse response.

In filter equalization the main idea is to compensate the channel impulse response $Hc(f)$ by introducing a filter $He(f)$ whose impulse response is the inverse of the channel.
In this method the equalizer compensates the distorted pulses by trying to reduce the effects of Inter Symbol Interference.

Equalization with filters can be made in continuous time or in discrete time where the compensation is done using samples of the received waveform.

Figure 2.9: Communication System.

Linear equalizers contain only feedforward elements (transversal equalizers) if they are non linear they contain feedforward and feedback elements(decision feedback equalizers),the major difference between this two types of equalizer is that in decision feedback equalizers current symbol decisions are based on previous detections ,where in transversal equalizers that information is not used.

Equalization by filters can also be divided by the nature of their operation they can be pre-set or adaptive,pre-set means that the setup of the filter coefficients is only made at the beginning of the operation , adaptive requires the filter coefficients to be updated as operation takes place.
Adaptation becomes important as channel characteristics change during operation,the changes can be caused by alteration in local temperature,cable length in guided mediums etc.

2.4 Continuous Time linear Equalizers

In CTLE equalizers the compensation is done in terms of compensating the channel attenuation at high frequency by introducing a high frequency boost,with this strategy we increase the effective bandwidth of the channel.

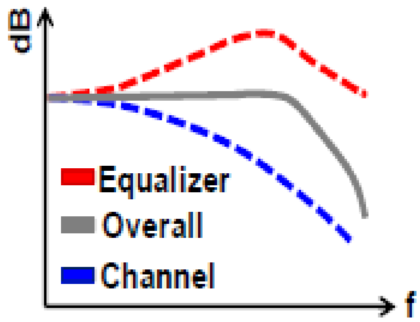

FIGURE 2.10: Compensation Scheme

By increasing the effective bandwidth ISI becomes less significant and it reduces the probability of error. The increase of bandwidth is achieve with the introduction of a zero near the cut-off frequency of the channel :

$$He(s) = K\frac{s+\omega_z}{s+\omega_p} \quad \omega_z < \omega_p$$

Compensating with higher order systems with more than one zero can also be achieved but generically with worse results,because in guided mediums the amount of attenuation is about $-10Dbs/dec$ and by introducing more zeros the overall response becomes over-compensated,also phase linearity also becomes an issue with higher order systems.

2.4.1 Circuit realization of CTLE

As we seen earlier the desired response of the equalizer is a response with a zero and a pole. Equalizers are physically implement with a differential pair:

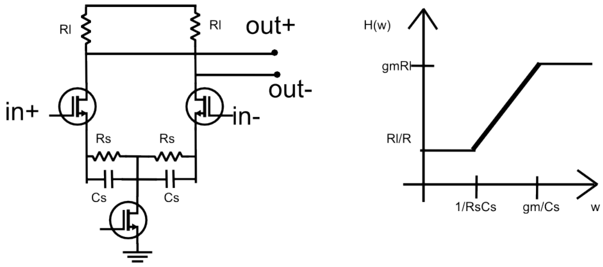

FIGURE 2.11: Simple Equalizer schematic

In this CTLE the DC gain equal Rl/Rs equal to a gain of a common source with source resistance,and the final gain equals $gmRl$ equal to the gain of the common source amplifier.
The position of the zero is controlled with the value of the capacity Cs and the value of the transconductance of the transistors of the differential pair. The actual response does not stabilize at $gmRl$ because of the high frequency poles introduced by the parasitic capacitances of the transistors.

Other circuits that are derived from the previous are also used for CTLE:

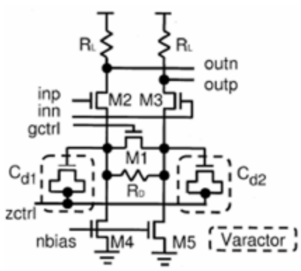

FIGURE 2.12: Equalizer Schematic

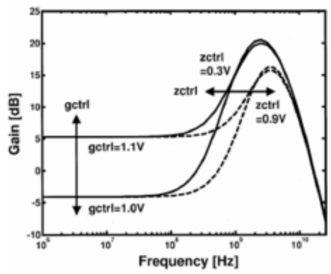

FIGURE 2.13: Equalizer Transfer Function

This circuit was presented in [1] for equalization at $3.5Gbits/s$ and the parameters of the equalizer transfer function can be tuned by providing two control voltages,$zctrl$ controls the frequency of the zero and the voltage $Gctrl$ controls the initial Dc gain.

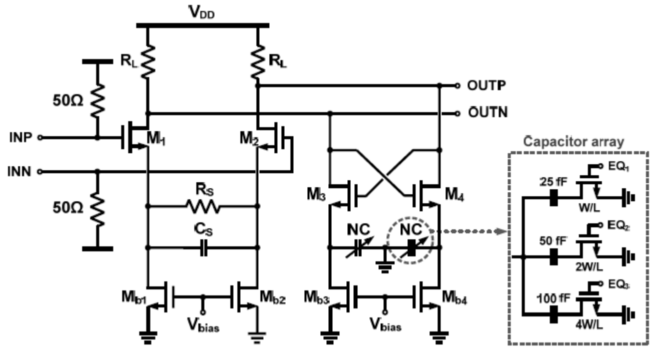

FIGURE 2.14: Equalizer schematic

This circuit was presented in [2]for equalization at $10Gbits/s$,the circuit presents very low power consumption $2.46mW$ during normal function.
The tuning of the equalizer is based in capacitive source degeneration and configures the CTLE with a variety of 8 diferent gain stages separated with a $1.5Dbs$ increment.

The next circuit also allows the tuning of different high frequency boosts controlled with a control voltage $Vctrl$

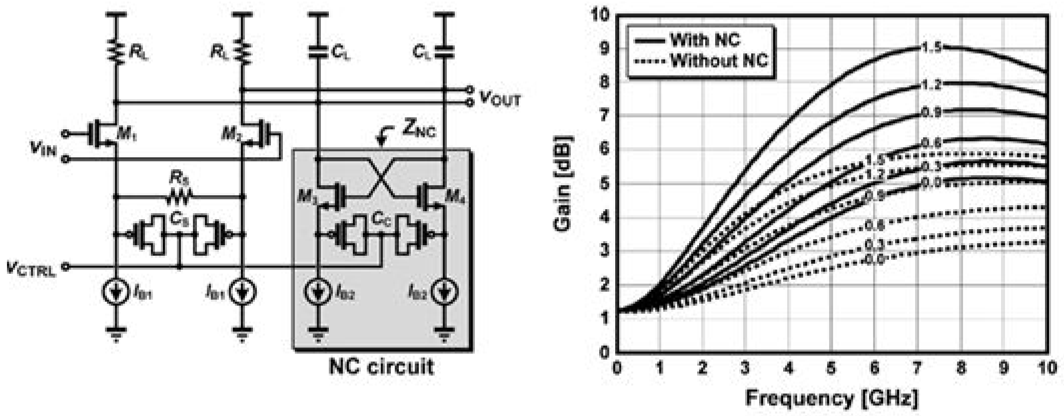

FIGURE 2.15: Equalizer schematic and transfer function

2.4.2 Adaptation techniques for CTLE

In this section some adaptation techniques are studied for the tuning of the CTLE's operation.
As we discussed earlier the problem with CTLE's is finding the equalizer transfer function that better compensates the high frequency loss presented in the channel so that:

$$H_c(jw)H_e(jw) = K, \ \forall\, w < C/2$$

2.4.2.1 Asynchronous Under sampling Histogram

This method is based on the assumption that by under sampling the waveforms coming out of the channel and constructing a histogram based on the amplitude of the samples, the histogram with lowest variance δ^2 represents the better eye opening.

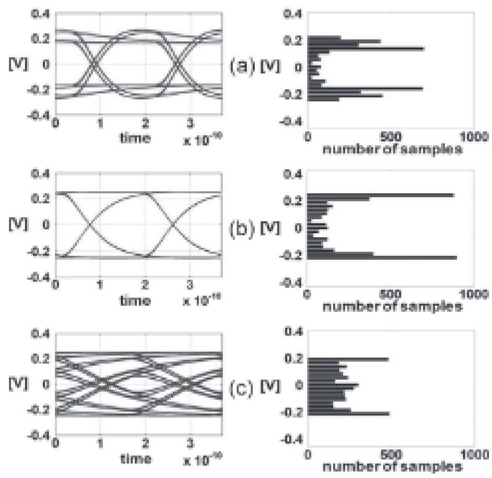

FIGURE 2.16: Relation between eye opening and Under sampling Histogram[3]

In a channel with no noise and no ISI the histogram would only present two peeks representing the two voltages the "0" or the "1" bit.
Adaptation following this method is simple and usually there is a predefined set of coefficients for the equalizer. While in the adaptation period all the CTLE coefficients are tested and a histogram of the samples is made, when all the coefficients are tested we choose the coefficients set that produces the histogram with largest peak and smallest variance.
The problem with this approach is that the adaptation process usually takes some time,because we need to transmit a large sequence and analyse it to make the histogram a valid measure of the channel.

In spite of this ,this adaptation technique is one of the most widely spread for tuning CTLE coefficients,because of its simplicity and low power circuit implementation.

2.4.2.2 Power Sensing

Adaptation through power sensing follows a different approach from the seen earlier.In this process the idea is to adjust the equalizer based on difference between of power in the high frequency and low frequency components of the received signal.

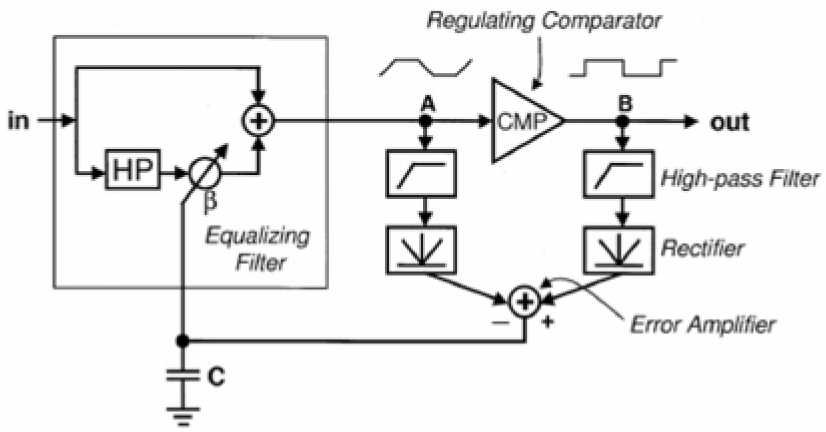

Figure 2.17: Adaptation using power sensing[4]

In this scheme there is two different paths in the receiver the the unity gain path ,and the high frequency boosting gain path ,the boost gain control is controlled by the difference of powers between the received signal and the recovered signal after a regulating comparator. The power is detected with a filter followed by a rectifier and the difference of powers will control the position of the zero of the zero in the equalizer.
Another more complete version of the adaptive CTLE is also presented in the same article.The new version also controls the gain at low frequencies .

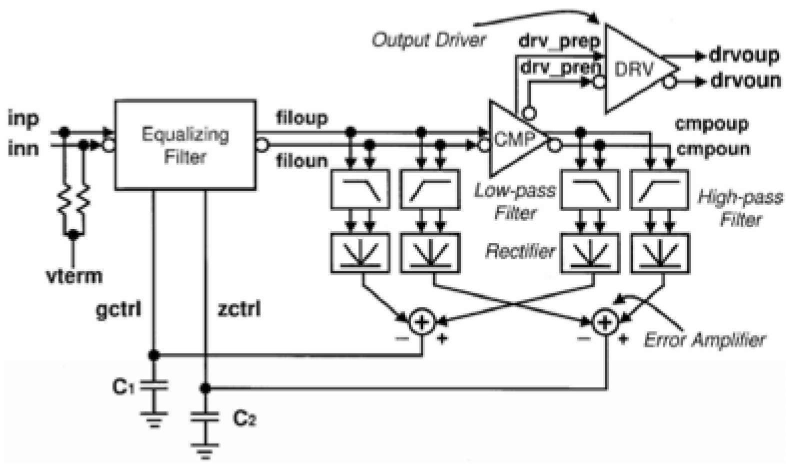

FIGURE 2.18: Adaptation using power sensing [1]

2.5 Discrete Time Equalizers

In Discrete Time equalizer the compensation is based on samples of the received sequence,as we seen earlier, if the impulse response of the channel is known the optimal receiver uses the MLSE method ,but this method is not practical in very high speed communication due to the computation complexity of the algorithm.
In this section the compensation of ISI is accomplish using discrete filtering,we will discuss techniques of adaptation using training sequence.

Consider the following simplification of the discrete channel.

FIGURE 2.19: Discrete channel model

Where:

$$y(n) = \sum_{k=-\infty}^{+\infty} x(k)h_c(n-k) \ \ and \ \ z = w + y$$

The idea is to introduce a filter in series with the channel so that:

$$h_c(n) * h_e(n) = \delta(n)$$

By taking the z transform this results in:

$$H_e(z)H_c(z) = 1$$

Graphically the problem with discrete equalization can be resumed in the next figure:

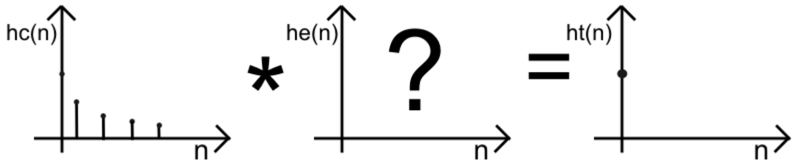

FIGURE 2.20: Discrete equalization

2.5.1 Transversal equalizers

By the properties of the linear convolution we can see that for exact cancellation we need an IIR filter equalizer to make the perfect cancelation of ISI,as the final tap of the overall response will never be exactly equal to zero.

Example Consider the channel impulse response with a single post Tap introducing ISI we want to find the coefficients of a three tap equalizer that reduces the ISI.

By making the linear convolution ,flipping and sliding the equalizer response over the channel response :

We obtain the following equations for the overall response:

$$\begin{cases} he(0)hc(0) = 1 \\ hc(1)he(0) + he(1)hc(0) = 0 \\ he(2)hc(0) + he(1)hc(1) = 0 \\ he(2)hc(1) \neq 0 \end{cases}$$

If we consider he(0)=1 we get a value for the remaining ISI equal to

$$hc(1)\frac{h(1)^2}{h(0)^2}$$

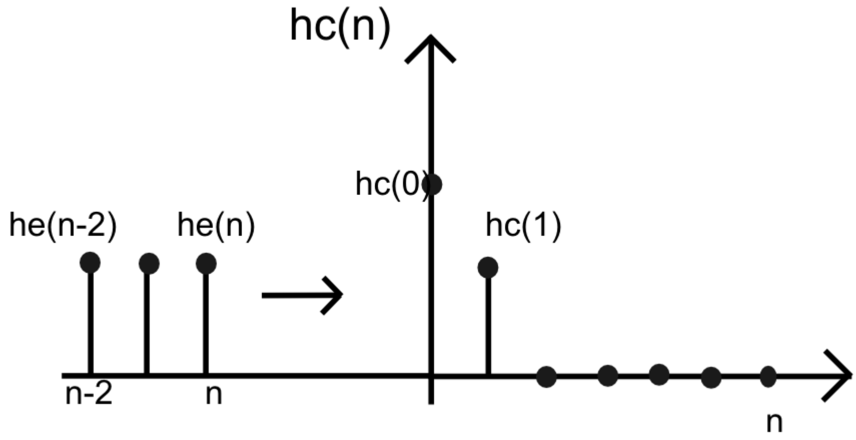

FIGURE 2.21: Channel response

For the case of a N tap equalizer and a two tap channel response we get the expression for the last tap of the overall response equal to:

$$ht(2+N-1) = he(1)\frac{he(1)}{he(0)}^{N-1}$$

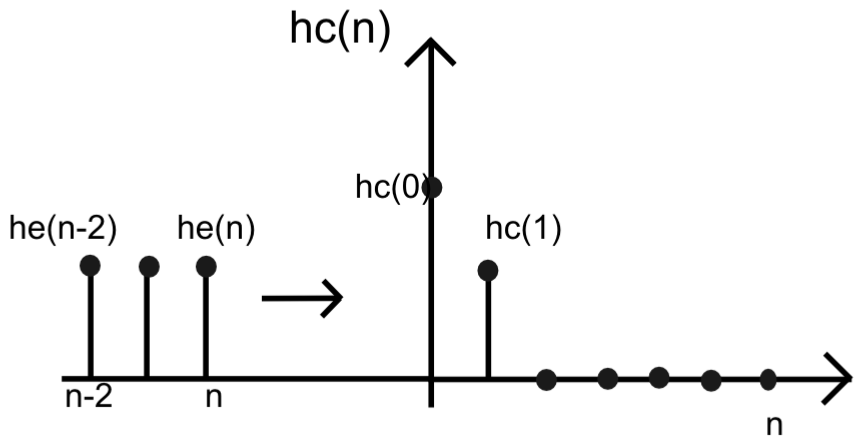

FIGURE 2.22: Channel response

The transversal filter is one of the most popular form of an easy adjustable equalizing filter and the impulse response of the equalizer is the same as the filter coefficients

$$z(n) = \sum_{k=-N}^{+N} c_k x(n-k)$$

The next figure represents the basic structure of a linear equalizer.

The transversal equalizer consists in a delay line with 2N T-second taps (T=symbol

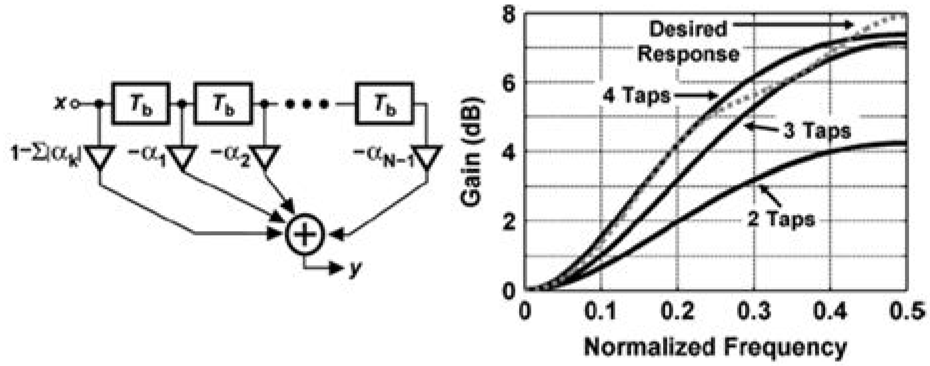

FIGURE 2.23: Linear equalizer

duration) saving the samples of previous received symbols.
The output of the equalizer is calculated through the weighted sum of the saved samples by the filter coefficients, being the central the main contribution to the value of the output. The central tap corresponds to the current symbol to be calculated as other taps produce echoes to cancel the ISI in the current symbol.

The basic limitation of the linear equalizer is that it performs poorly on channels having spectral nulls,and it performs Noise enhancement.
So the basic limitation of Linear equalization results from the impossibility that in practice one cannot make an IIR filter only with a transversal equalizer.

2.5.2 Zero Forcing equalizers

The Zero forcing equalizer makes the equalizer transfer function equal to the inverse of the channel transfer function:

$$He(z) = \frac{1}{Hc(z)}$$

In the zero forcing solution the coefficients are chosen so that:

$$z(k) = \begin{cases} 1 \ for \ k = 0 \\ 0 \ for \ \ k = \pm 1, \pm 2, ..., \pm N \end{cases}$$

Where the number of taps of the equalizer equals 2N.
So in the zero tap equalizer one can only guarantee zero Inter Symbol Interference to the 2N adjacent bits of the sequence in relation to the current sampled bit.
Let us consider the following array definition:

$$Z = \begin{bmatrix} z(-N) \\ . \\ . \\ z(0) \\ . \\ . \\ .z(N) \end{bmatrix} \quad C = \begin{bmatrix} c_{-N}) \\ . \\ . \\ c_0 \\ . \\ . \\ .c_N \end{bmatrix} \quad X = \begin{bmatrix} x(0) & x(-1) & . & . & x(-N) \\ x(1) & x(0) & . & . & .x(-N+1) \\ . & . & . & . & . \\ x(N) & x(N-1) & . & . & x(0) \end{bmatrix}$$

Where:
Z represents a vector with the received samples to be presented to the element making the symbol decision
C is the array with the equalizer coefficients.
X represents the samples present in the equalizer,X is a Toeplitz matrix
We can write the following equation:

$$Z = XC \rightarrow C = X^{-1}Z$$

By solving this equation we can make the ISI equal to zero in the 2N side lobes .
The Length of the filter who performs zero forcing equalization is a function of the smearing introduced by the channel.
With a finite number of taps the ISI is minimized and the solution is optimal in terms of reduction of ISI at the sampling points.The ZF solution requires initial eye opening to perform equalization and the process uses a an estimate of the impulse response of the channel.

2.5.3 Decision Feedback equalizers

The main problem with the equalization using transversal filters was the impossibility of obtaining an IIR equalizer response,decision feedback equalizers can realize an IIR response because it uses both a forward and a feedback filter.
The idea behind the DFE is that if the values of the past decisions are known(decisions are assumed to be correct) then the ISI introduced by these symbols is subtracted in the current decision.
In the DFE the feedback filter uses previous quantized samples and because of this the output of the feedback filter is free of noise.
The sequence produced at the output of the channel equals:

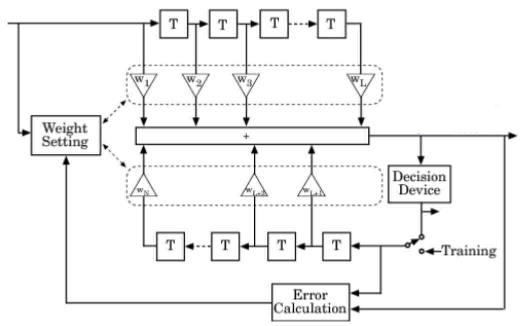

FIGURE 2.24: Structure of the Decision Feedback Equalizer

$$y(k) = y(k)hc_0 + \sum_{j \neq k} y_j hc_{k-j}$$

y(k) represents the received symbol at time k
The summation represents the ISI so the basic idea of the decision feedback equalizer is to simply subtract the ISI.

$$z(k) = y(k) - \sum_{j \neq k} y_j hc_{k-j}$$

The transfer function of the DFE equals:

$$H_e(z) = \frac{A(z)}{1 + B(z)} = \frac{\sum_{n=0}^{M} a_n z^{-n}}{\sum_{n=0}^{N} b_n z^{-n}}$$

Where **bn** and **an** represent the value of the coefficients of the feedback filter and transversal filter,**M** and **N** is the size of the transversal and feedback filter.

Again the problem with equalization is to find the coefficients of both filter in order to achieve the best reduction of ISI fed into the decision device. Assume that the output of the equalizer is given by:

$$z_k = \sum_{j=-M}^{0} y_{k-j} h_{e,j} + \sum_{j=1}^{N} z_{k-j}^{d} h_{e,j}$$

The first summation gives the influence of the transversal filter on the symbol zk and the second the influence of the feedback filter on the output ,note that the feedback filter uses already decided symbols z_{k-j}^{d} thus meaning that the feedback samples don't have the influence of noise.

By defining the following vectors:
$IF = \begin{bmatrix} z_{k-1}^{d} & z_{k-2}^{d} & z_{k+N-1}^{d} & . & . & z_{k-N}^{d} \end{bmatrix}^T$
$IB = \begin{bmatrix} y_{k-1} & y_{k+N-1} & y_{k+N-1} & . & . & y_k \end{bmatrix}^T$
$hE, B = \begin{bmatrix} h_{E,-M} & h_{E,-M+1} & . & . & h_{E,0} \end{bmatrix}^T$
$hE, F = \begin{bmatrix} h_{E,1} & h_{E,2} & . & . & h_{E,N} \end{bmatrix}^T$

Where:

- IF represents the vector with the previous N decisions
- IB represents the vector with the M samples at the entrance of the equalizer
- HE,B represents the transversal filter coefficients
- HE,F represents the Feedback filter coefficients

The filter coefficients are chosen to minimize the square error function $E[Zk - Xk]^2$. Resulting in the optimal filter coefficients:

$$hE, B = (E(IBIB^T) - E[IBIF^T]E[IFIB^T])^{-1}E(IFIK)$$

$$hE, F = -E[IFIB^T]hE, B$$

Where E denotes expectation
If we not know the value of the expectation values á priori we can estimate them using a training sequence as we will see in the next chapter.

2.6 Adaptation of discrete equalizers

As operation takes place the channels characteristics change and the optimum equalization parameters that were calculated at the beginning of the transmission might not be optimum later on.One solution would be do calculate the equalizer coefficients with a certain period of time ,but the majority of adaptation algorithms needs the transmission of a training sequence that would consume much time that could be used transmitting necessary information.
The tap weights of the equalizer can be updated periodically or continually when performed continually the adaptation is referred as decision directed.
Decision directed equalization uses some kind of algorithm for adjusting the filter coefficients ,most kind of are based in the minimization of the quadratic error,decision directed equalizer can have problems with convergence if the initial probability of error exceeds one percent.In this case, the equalizer taps need to be initialized using an alternate process.
Let us consider the following structure of a system with adaptive equalization

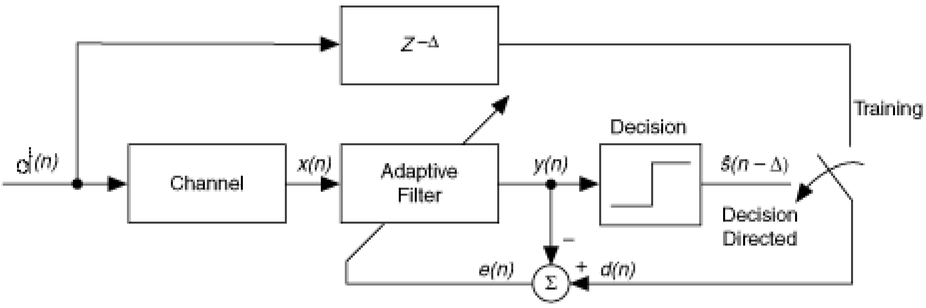

Figure 2.25: Adaptive equalization structure

In the structure we can see that the adaptation process is done in two stages ,first calculate the initial coefficients of the equalizer with a training sequence and then switch to decision directed mode.
We see that the adaptive filter is updated using an estimate of the error signal $[d(n) - y(n)]$. The most common method for adaptive equalization is know as the **Least Mean Squares Algorithm (LMS)**.The LMS algorithm consists on the minimization of the quadratic error between the desired response and the signal before the decision device.

Let us consider the following model:

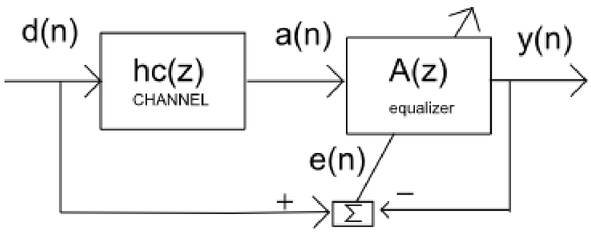

FIGURE 2.26: Adaptive equalization scheme with transversal filter

The quadratic error is from now on defined as ε.

$$\varepsilon = E[e^2[n]] = E[d[n] - y[n]^2]$$

By the properties of the expected value we get:

$$\varepsilon = E[d^2[n]] + E[y^2[n]] - 2E[y[n].d[n]]]$$

In the case of no feedback structure the quadratic error becomes:

$$y(n) = \sum_{j=0}^{N} c_j a(n-j) = c^T a(n) \longrightarrow \varepsilon = E[d^2[n]] - 2c^T p + c^T Rc$$

In this equation the quadratic error becomes a dependence of:

- $E[d^2[n]]$ represents the variance of the desired response
- $2c^T p$ where p represents the cross correlation between the desired response and y(n)
- $c^T Rc$ where R represents the self correlation of y(n)

The expression encountered before defines the dependence of the quadratic error in function of the filter coefficients and its called **performance surface**.

The perfect equalizer is found when the quadratic error is minimum for that coefficients.So we define the gradient vector as:

$$\nabla(\varepsilon) = \frac{\partial E[e^2[n]]}{\partial c} = -2p + 2Rc$$

The local minimum is found when $\nabla(\varepsilon) = 0$

$$c_{opt} = R^{-}1p; \quad \varepsilon_m in = E[d^2[n]] - p^T c_{opt}$$

Now considering that the coefficients are being updated in time we can define the update dynamics in function of the gradient of the quadratic error:

$$c_{[n+1]} = c_{[n]} - \mu \frac{\partial E[e^2[n]]}{\partial c} \Leftrightarrow c_{[n+1]} = c_{[n]} - 2\mu E[e(n)a(n)]$$

This is know as the **gradient algorithm** ,where μ represents the adaptation step. The problem with the gradient algorithm is the calculation of the estimate of $E[e(n)a(n)]$ so the LMS algorithm uses a simplification that consists in replacing the average values of e(n) and a(n) by their instant values witch results in :

$$c_{[n+1]} = c_{[n]} - 2\mu e(n)a(n) \quad , \quad 0 < \mu < \frac{1}{NE[a^2(n)]}$$

2.7 Adaptation of Decision Feedback Equalizers

The adaptation of the DFE can be made using the LMS algorithm

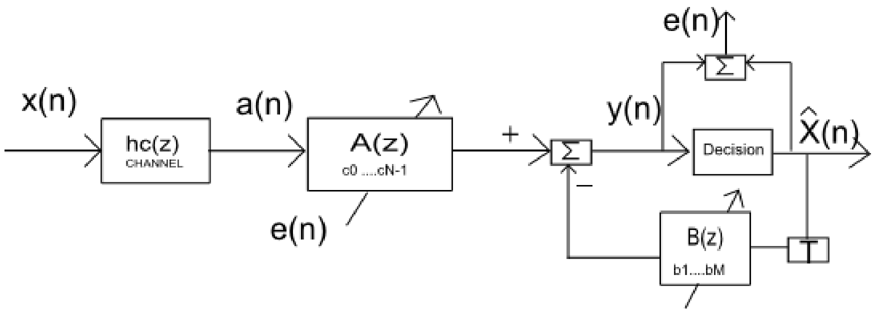

FIGURE 2.27: Decision feedback Equalizer adaptation scheme

$$c_{[n+1]} = c_{[n]} - 2\mu e(n)a(n)$$

$$b_{[n+1]} = b_{[n]} - 2\mu e(n)\hat{x}(n-1)$$

The above expressions represent the expression for the adaptation of the coefficients of both the transversal filter A(z) and the feedback filter B(z).

The simplification used in LMS for the value of the gradient makes that the coefficients do not converge to the optimal solution instead it oscillates around it.This oscillation is called Gradient noise.
Recursive least squares (RLS)makes the coefficient convergence much faster than MLS but it requires much more computation for iteration thus not making it suitable for high speed systems.
Other algorithms use different approaches for the calculation of the gradient and can make the adaptation step variable,despite all ,the LMS simplification is the most used.

2.8 Further simplification of the MLS algorithm

Other simplifications can be made to the LMS algorithm namely on the equation for the coefficients adaptation,several variants can be chosen accordingly for:

- Simplify the algorithm implementation
- Accelerate the convergence speed of the coefficients
- Reduce the Gradient Noise

2.8.1 Normalized MLS

In the MLS the equation for the coefficients update depends on $\mu e(n)a(n)$,containing the entrance vector $a(n)$,which makes the convergence speed dependent of the input signal power.
The idea behind this alteration is to alter the expression in a way to make the update coefficients step more independent of the signal power:

$$c_{[n+1]} = c_{[n]} + \frac{2\hat{\mu}}{b+ \parallel a(n) \parallel^2} e(n)a(n), \quad \parallel a(n) \parallel^2 = \sum_{j=0}^{N-1} a^2(n-j)$$

$\parallel a(n) \parallel^2$ represents the euclidean norm of the input signal.
$0 < \hat{\mu} < 1$ to guarantee convergence.
$b > 0$ just to make impossible the division by 0.

2.8.2 Signal algorithm

This simplification further simplifies the calculation of the gradient $\nabla(\varepsilon)$ by just considering the signal of e(n) or a(n) or both.

$$sign(x) = \begin{cases} 1 & x > 0 \\ -1 & x < 0 \end{cases}$$

$$c_{[n+1]} = c_{[n]} - 2\mu sign(e(n))a(n)$$

$$c_{[n+1]} = c_{[n]} - 2\mu e(n)sign(a(n))$$

$$c_{[n+1]} = c_{[n]} - 2\mu sign(e(n))sign(a(n))$$

The double Signal algorithm makes the calculation of the e(n) by diving the scale into 4 steps instead of two. Other alterations of the Original LMS include,the Leaky LMS,Dead Zone LMS ,LMF algorithm and others.

Chapter 3

Description of problems to solve

This thesis is oriented for the study of equalization methods for high speed communication systems ,board to board or chip to chip.
The thesis will study implementation of both analogue and digital equalizers designed for the compensation of communication channels used in inter chip communication. The work will consist in the approach of the following :

- Continuous Time Linear Equalization : study of different circuits implementation and adaptation techniques.
- Equalization through digital filter :study of both transversal and decision feedback equalizers.Influence of the length of the filter,the adaptation method and cost of the physical implementation.
- Study of algorithms and implementations for automatic equalizer adaptation,namely the Least mean Squares algorithm
- Requirements for training sequences:Length of the training sequence and methods for its generation
- Matlab/ADS simulations using synthetic and real signals

Chapter 4

Work Plan

Preparation for the Master Thesis:

- State of the art and revision of bibliography.
- Summary of continuous time linear equalization techniques, emphasis on ASIC implementation.
- Summary of equalization methods through digital filter, emphasis on ASIC implementation.
- Summary of algorithms for automatic equalizer adaptation. Should cover specifically DFE systems.

Master Thesis will be conducted in Synopsy's facilities in Tecmaia:

- Implementation of a proof of concept using simulation software and a model of the equalization system and channel impairments. 6W
- Definition of training sequences requirements or digital transmission stream characteristics required for adaptive equalization. 4W
- Comparison regarding performance and implementation cost, performance vs. complexity and power consumption. 4W
- Master Thesis 4W

References

1- Jong-Sang Choi ,Moon-Sang Hwang Deog-Kyoon Jeong,"A 0.18μm CMOS 3.5Gb/s Continous Time Adaptive Cable Equalizer Using Enhanced Low Frequency Gain Control Method"

2-Wang-Soo Kim and Woo-Young Choi,"A 10 Gb/s low power Adaptive Continuous-Time Linear equalizer Using assynchronous Under-Sampling histogram"

3- Wang-Soo Kim, Chang-Kyung Seong, Student Member, IEEE, and Woo-Young Choi, Member, IEEE,"A 5.4-Gbit/s Adaptive Continuous-Time Linear Equalizer Using Asynchronous Undersampling Histogram"

Silvio A Abrantes ,Processamento adaptativo de Sinais ,Fundacao Calouste Gulbenkian

Bernard Skylar ,Digital Communications:Fundamentals and Applications, 2th Edition, Prentice Hall January 2001

P.S.R. Diniz, Adaptive Filtering, Kluwar Academic Publishers, Norwell, Massachusetts, 2002.

Wong and Lok, Theory of Digital Communications

Amal Ekbal,Stanford University "EE379A: Lecture 12 FIR Equalizer Design"

Mohammad Havaei, Nandivada Krishna Prasad, and Velleshala Sudheer,"Elimination of ISI Using Improved LMS Based Decision Feedback Equalizer "

Texas Intruments,"The Benefits of Using Linear Equalization in Backplane and Cable Applications"

Sam Palermo Analog and Mixed-Signal Center Texas AM University,"Lecture 18: RX FIR and CTLE Equalization"