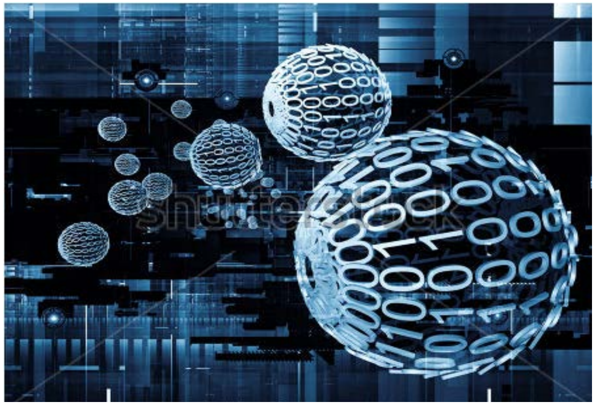
BIRZEIT UNIVERSITY

Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

Modern Communication Systems, ENEE3306

Dr. Mohammad Jubran

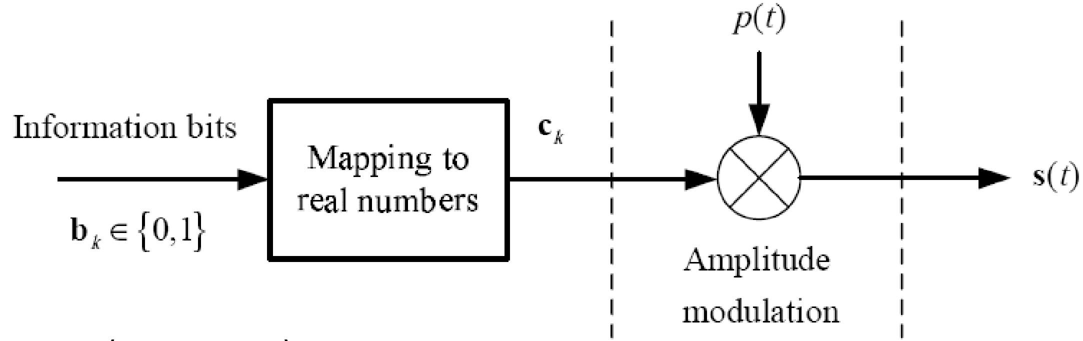


Lecture 2

- Bits (0, 1) are mapped into two voltage levels for direct transmission without any frequency translation.
- Baseband modulation can be defined as a modulation with PSD is handled around $f = 0$ (Hertz)
- Various baseband signaling techniques (line codes) were developed to satisfy typical criteria:
  - Transmission Bandwidth: as small as possible
  - Power Efficiency: Transmitted power must be as small as possible for given BW and probability of error
  - Error Detection and Correction capability: Ex: Bipolar
  - Favorable power spectral density: (no DC component→ imply that AC coupling via transformers may be used)
  - Adequate timing content: Extract timing from pulses (self-synchronization, don't transmit clock)
  - Transparency: Prevent long strings of 0s or 1s
  - Cost and complexity of transmitter and receiver implementations

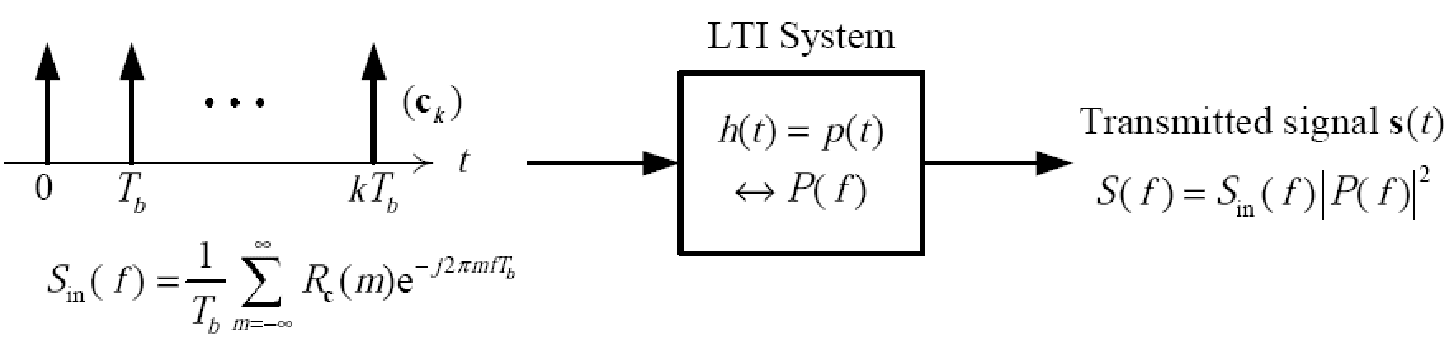
BIRZEIT UNIVERSITY
ECE

- Recall from Chapter 5: PSD of Digital Amplitude Modulation

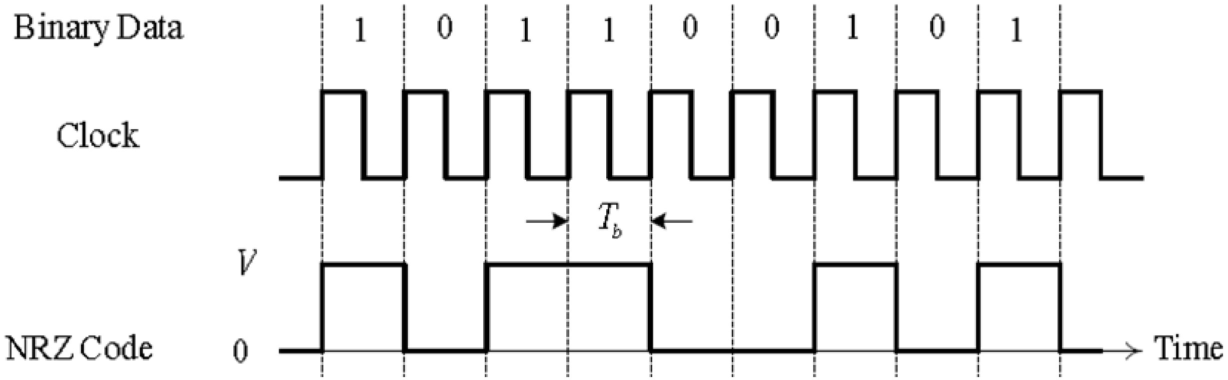


$$S(f) = \lim_{T\to\infty} \frac{E\left\{|S_T(f)|^2\right\}}{2T} = \frac{|P(f)|^2}{T_b} \sum_{m=-\infty}^{m=\infty} R_c(m) e^{-j2\pi f m T_b} = S_{in}(f)|P(f)|^2$$

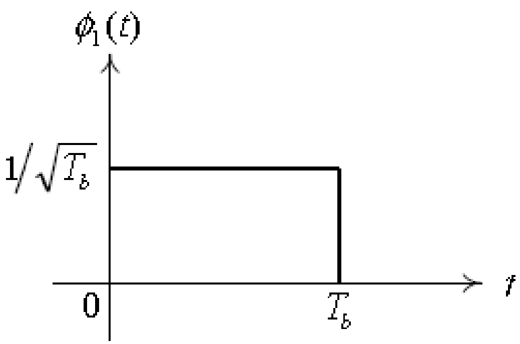
$$S_{in}(f) = \frac{1}{T_b} \sum_{m=-\infty}^{m=\infty} R_c(m) e^{-j2\pi f m T_b}, \quad R_c(m) = E\left\{c_k c_{k-m}^*\right\} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N} c_k c_{k+m}^*$$

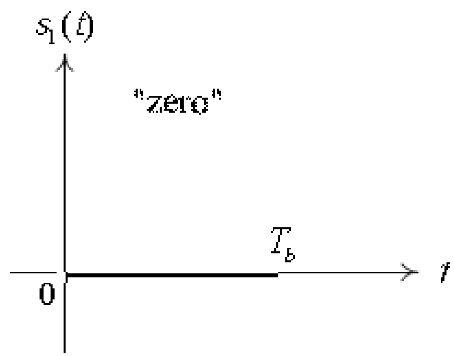


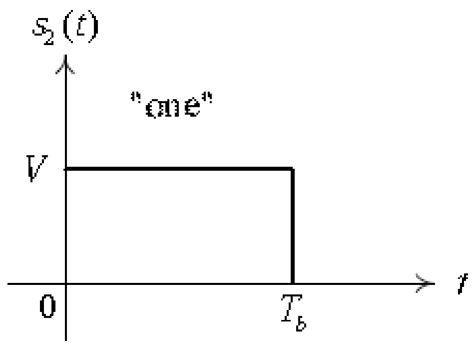
➢ Unipolar Non-Return-to-Zero (NRZ) Line Code

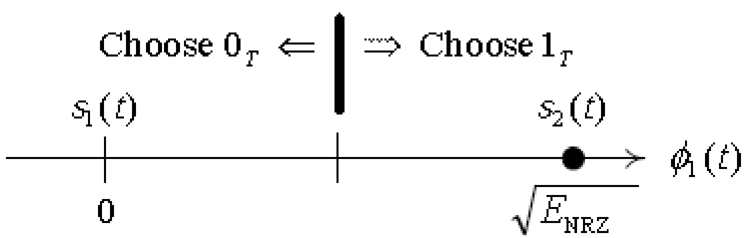


- **Transmission Bandwidth:** doesn't require a lot of bandwidth for transmission
- **Power Efficiency:**

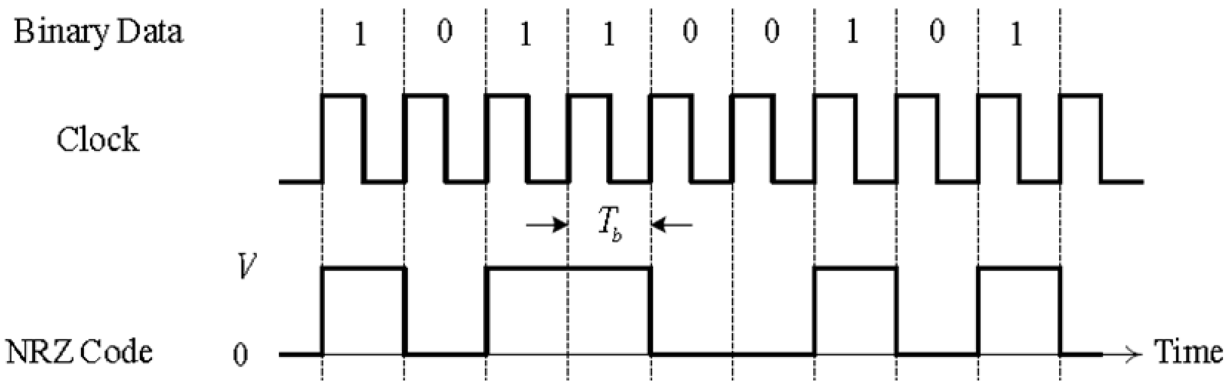




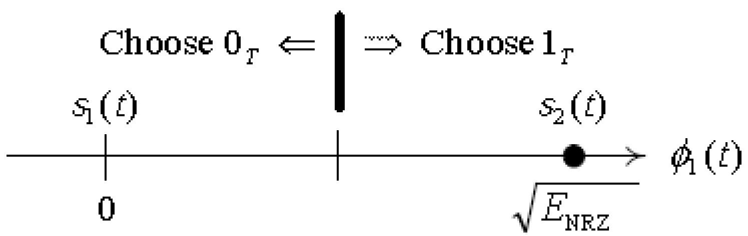




Unipolar Non-Return-to-Zero (NRZ) Line Code



- **Power Efficiency:**



$$\rightarrow P[\text{error}] = Q\left(\frac{\text{distance between the signals}}{2\times\text{noise RMS value}}\right) = Q\left(\frac{\sqrt{E_{NRZ}}}{2\times\sqrt{N_o/2}}\right) = Q\left(\sqrt{\frac{E_{NRZ}}{2N_o}}\right)$$

Unipolar Non-Return-to-Zero (NRZ) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:**

$$R_c(m) = E\{c_k c_{k+m}^*\} = \lim_{N\to\infty} \frac{1}{N}\sum_{k=0}^{N} c_k c_{k+m}^*$$

$$R_c(0) = E\{c_k c_k^*\} = \lim_{N\to\infty} \frac{1}{N}\sum_{k=0}^{N} c_k c_{k+m}^* = \lim_{N\to\infty} \frac{1}{N}\times\frac{N}{2} = \frac{1}{2}$$

a_k is equally likely to be 0 or 1

$$R_c(m\neq 0) = E\{c_k c_{k+m}^*\} = \lim_{N\to\infty} \frac{1}{N}\sum_{k=0}^{N} c_k c_{k+m}^* = \sum_{c_k\in\{1,0\}}\ \sum_{c_{k+m}\in\{1,0\}} c_k c_{k+m}^* p(c_k, c_{k+m})$$

$$= (1\times 1)P(c_k = 1, c_{k+m} = 1) + (1\times 0)P(c_k = 1, c_{k+m} = 0)$$

$$+ (0\times 1)P(c_k = 0, c_{k+m} = 1) + (0\times 0)P(c_k = 0, c_{k+m} = 0)$$

$$= P(c_k = 1, c_{k+m} = 1) = \frac{1}{4}$$

For $m \neq 0$: there are 4 possibilities,
- $c_k=1$ and $c_{k+m}=1$
- $c_k=1$ and $c_{k+m}=0$
- $c_k=0$ and $c_{k+m}=1$
- $c_k=0$ and $c_{k+m}=0$

➢ Unipolar Non-Return-to-Zero (NRZ) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:**

$$S_{in}(f) = \frac{1}{T_b}\sum_{m=-\infty}^{m=\infty} R_c(m) e^{-j2\pi f m T_b}$$

$$= \frac{1}{T_b}\left[\frac{1}{2} + \frac{1}{4}\sum_{m=-\infty, m\neq 0}^{m=\infty} e^{-j2\pi f m T_b}\right] = \frac{1}{T_b}\left[\frac{1}{4} + \frac{1}{4}\sum_{m=-\infty}^{m=\infty} e^{-j2\pi f m T_b}\right]$$

$$= \frac{1}{4T_b}\left[1 + \sum_{m=-\infty}^{m=\infty} e^{-j2\pi f m T_b}\right] = \frac{1}{4T_b}\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{m=\infty} \delta\left(f - \frac{m}{T_b}\right)\right]$$

Recall: Poisson summation formula

$$\sum_{n=-\infty}^{\infty} f(n) = \sum_{k=-\infty}^{\infty} F\{f(n)\}$$

➢ Unipolar Non-Return-to-Zero (NRZ) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:**

$$P(f) = VT_b \operatorname{sinc}(\pi f T_b)$$

$$S_{in}(f) = \frac{1}{4T_b}\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{m=\infty}\delta\left(f - \frac{m}{T_b}\right)\right]$$

$$S(f) = |P(f)|^2 S_{in}(f) = \frac{V^2 T_b^2 \operatorname{sinc}^2(\pi f T_b)}{4T_b}\left[1 + \frac{1}{T_b}\sum_{m=-\infty}^{m=\infty}\delta\left(f - \frac{m}{T_b}\right)\right]$$

$$= \frac{V^2 T_b \operatorname{sinc}^2(\pi f T_b)}{4}\left[1 + \frac{1}{T_b}\delta(f)\right] = \frac{V^2 T_b \operatorname{sinc}^2(\pi f T_b)}{4} + \frac{V^2}{4}\delta(f)$$

Notice: $\operatorname{sinc}(\pi f \mathrm{T_b}) = 0$ for any all f=m/Tb other than f=0.

➢ Unipolar Non-Return-to-Zero (NRZ) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density: It has a non-zero Dc component**

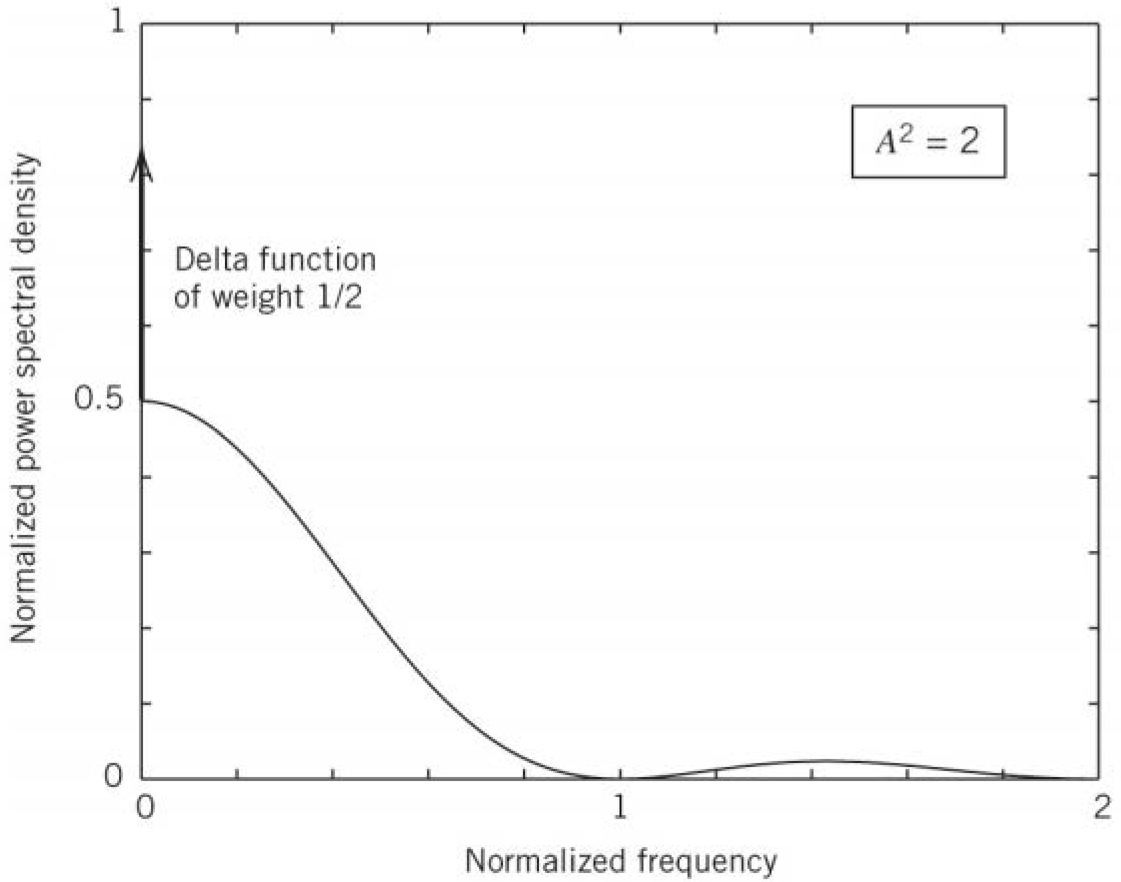
$$DC = 0 \times P_1 + V \times P_2 = V \times P_2 = \frac{V}{2}$$

If a_k is equally likely to be 0 or 1

- **Adequate timing content:** doesn't have any clock component and so is not good for synchronization
- **Transparency:** doesn't prevent long strings of 0s or 1s
- **Cost and complexity:** simple to implement

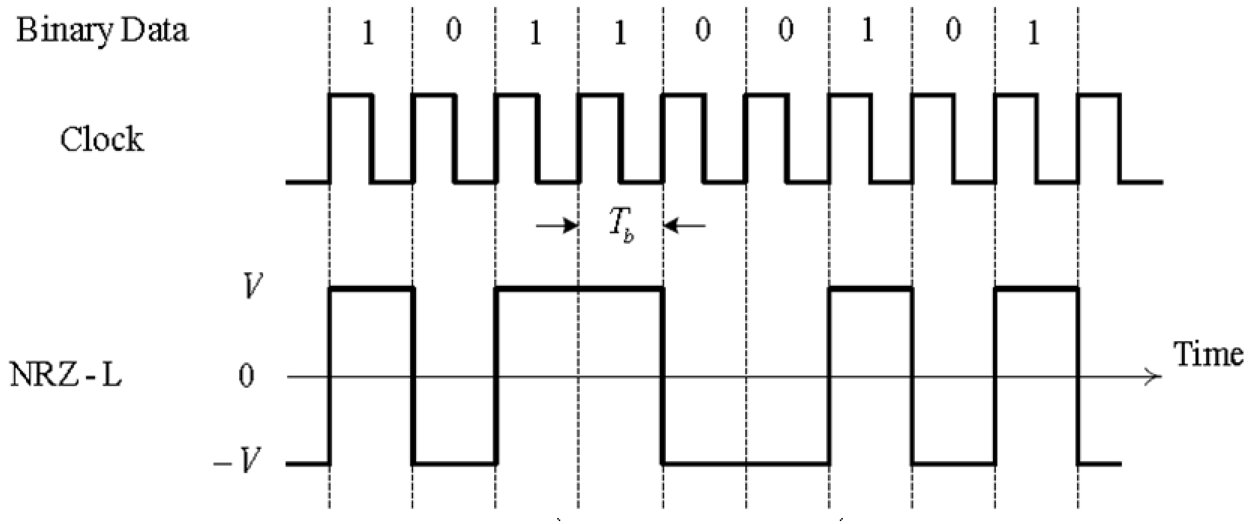
Unipolar Non-Return-to-Zero (NRZ) Line Code

- Power spectra of unipolar NRZ signal. (The frequency is normalized with respect to the bit rate 1/Tb, and the average power is normalized to unity. Only the positive side of the spectrum is shown in the figure).

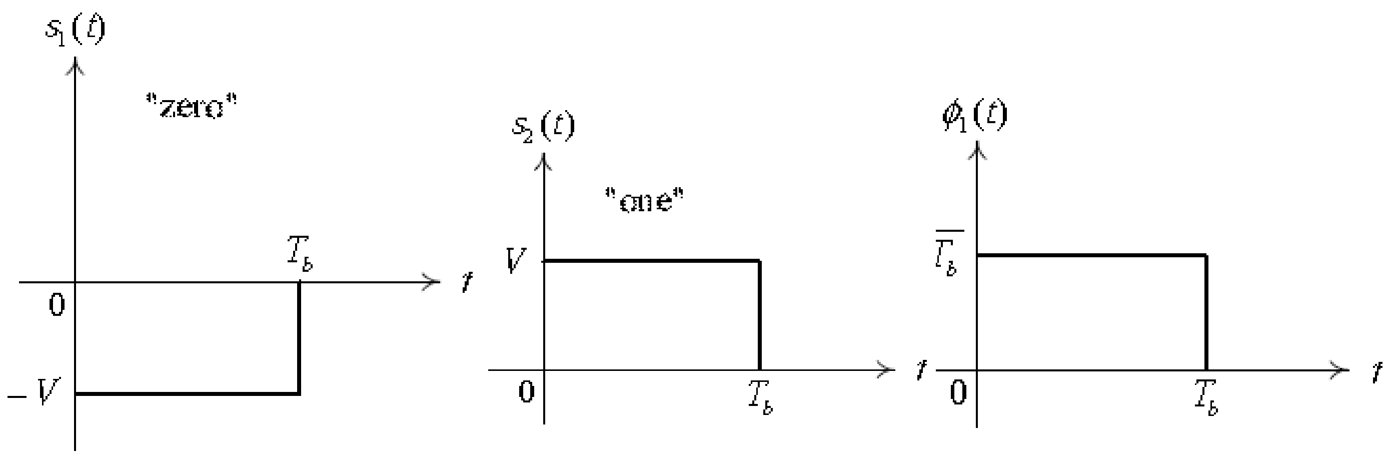


$$S(f) = \frac{V^2 T_b \operatorname{sinc}^2(\pi f T_b)}{4} + \frac{V^2}{4}\delta(f)$$

➢ Polar Non-Return-to-Zero (NRZ-L) Line Code



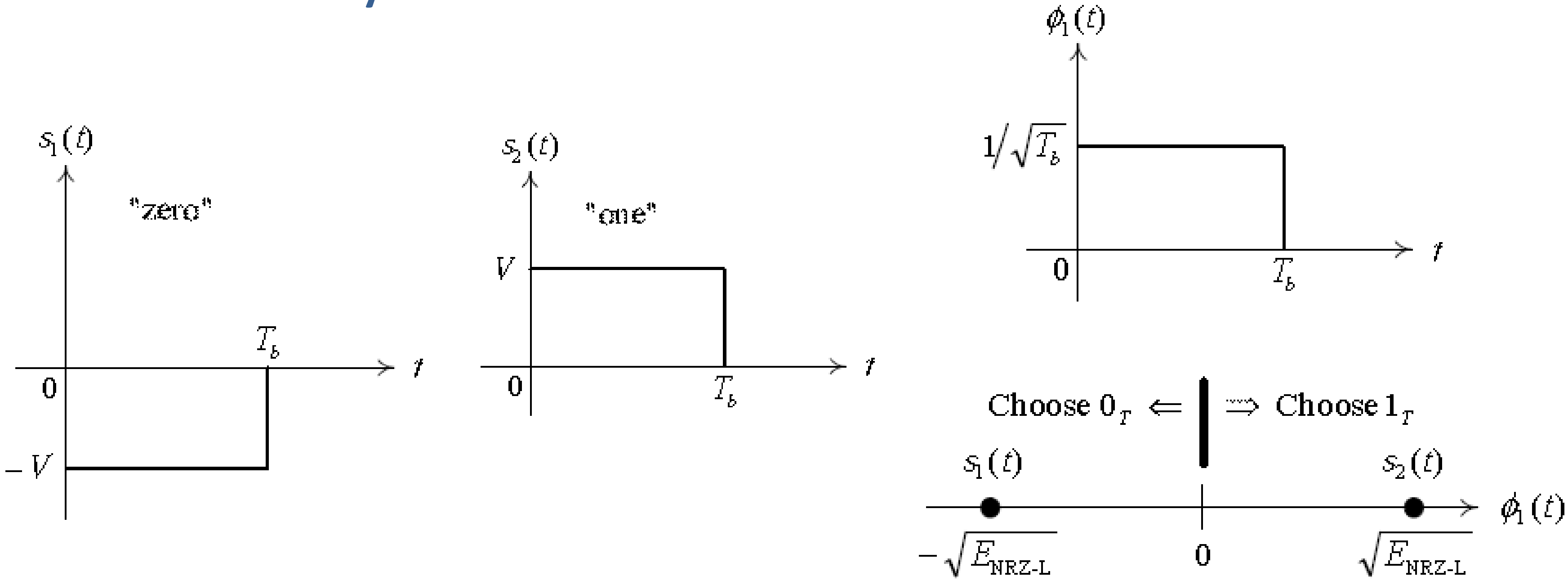
- **Transmission Bandwidth:** doesn't require a lot of bandwidth for transmission
- **Power Efficiency:**



BIRZEIT UNIVERSITY
ECE

➤ Polar Non-Return-to-Zero (NRZ-L) Line Code

- **Transmission Bandwidth:** doesn't require a lot of bandwidth for transmission
- **Power Efficiency:**



$$P[\text{error}] = Q\left(\frac{\text{distance between the signals}}{2\times\text{noise RMS value}}\right) = Q\left(\frac{2\sqrt{E_{\text{NRZ-L}}}}{2\times\sqrt{N_o/2}}\right) = Q\left(\sqrt{\frac{2E_{\text{NRZ-L}}}{N_o}}\right)$$

BIRZEIT UNIVERSITY
ECE

Polar Non-Return-to-Zero (NRZ-L) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:**

$$R_c(m) = E\{c_k c_{k+m}^*\} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N} c_k c_{k+m}^*$$

$$R_c(0) = E\{c_k c_k^*\} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N} c_k c_{k+m}^* = \lim_{N\to\infty} \frac{1}{N} \times N = 1$$

a_k is equally likely to be -1 or 1

$$R_c(m \neq 0) = E\{c_k c_{k+m}^*\} = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N} c_k c_{k+m}^*$$

$$= \sum_{c_k \in \{1,-1\}} \sum_{c_{k+m} \in \{1,-1\}} c_k c_{k+m}^* p(c_k, c_{k+m}) = (1\times 1) P(c_k = 1, c_{k+m} = 1)$$

$$+ (1\times -1) P(c_k = 1, c_{k+m} = -1)$$

$$+ (-1\times 1) P(c_k = -1, c_{k+m} = 1)$$

$$+ (-1\times -1) P(c_k = -1, c_{k+m} = -1)$$

$$= 0$$

For $m \neq 0$: there are 2 possibilities,

- $c_k=1$ and $c_{k+m}=1$
- $c_k=1$ and $c_{k+m}=-1$
- $c_k=-1$ and $c_{k+m}=1$
- $c_k=-1$ and $c_{k+m}=-1$

Polar Non-Return-to-Zero (NRZ-L) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:**

$$S_{in}(f) = \frac{1}{T_b}\sum_{m=-\infty}^{m=\infty} R_c(m) e^{-j2\pi f m T_b} = \frac{1}{T_b} R_c(0) = \frac{1}{T_b}$$

$$P(f) = V T_b \operatorname{sinc}(\pi f T_b)$$

$$S(f) = S_{in}(f)\left|P(f)\right|^2 = V^2 T_b \operatorname{sinc}^2(\pi f T_b),$$

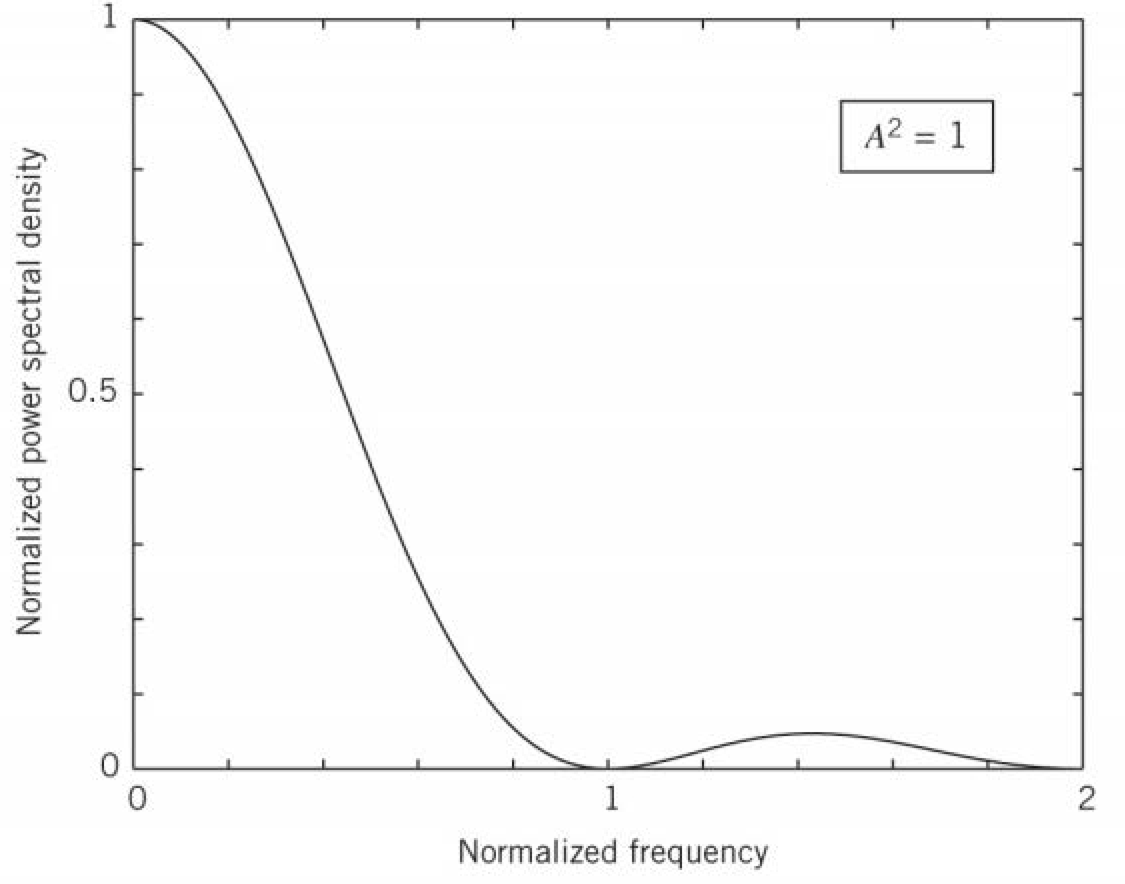
$$DC = -V \times P_1 + V \times P_2 = V(P_2 - P_1) = 0$$

If a_k is equally likely to be 0 or 1

- **Adequate timing content:** doesn't have any clock component and so is not good for synchronization
- **Transparency:** doesn't prevent long strings of 0s or 1s
- **Cost and complexity:** simple to implement

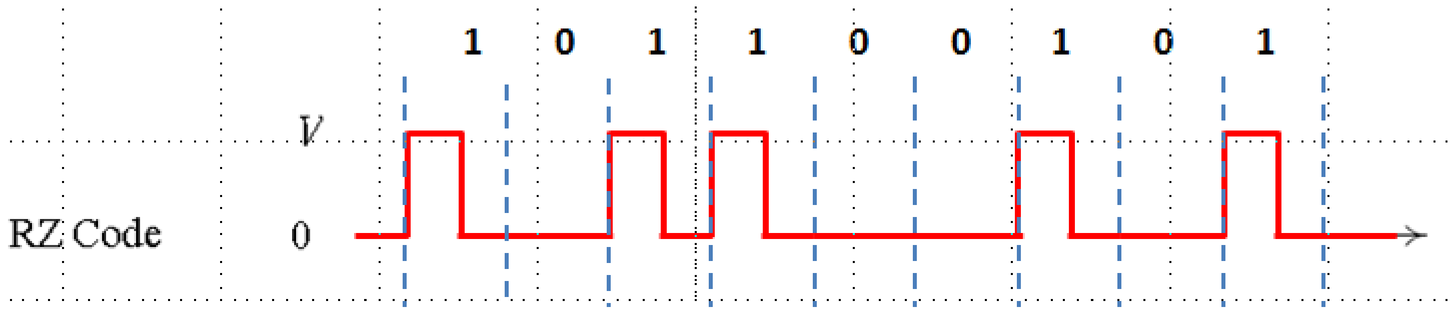
➢ Polar Non-Return-to-Zero (NRZ-L) Line Code

$$S(f) = S_{in}(f)\left|P(f)\right|^2 = V^2 T_b \operatorname{sinc}^2(\pi f T_b),$$

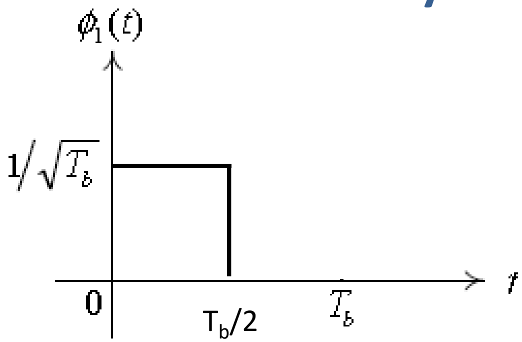


(b)

➢ Return-to-Zero (RZ) Line Code

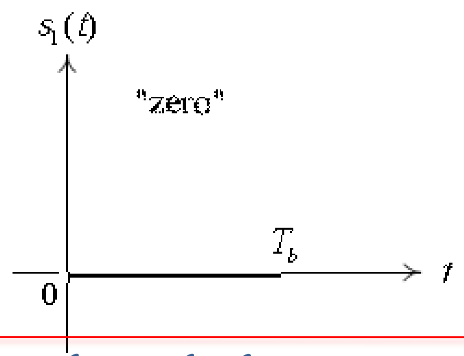


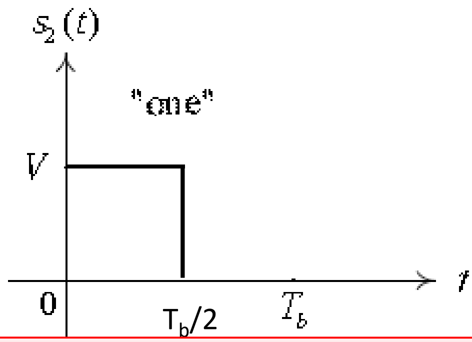
- **Transmission Bandwidth:** requires more bandwidth than NRZ code
- **Power Efficiency:**

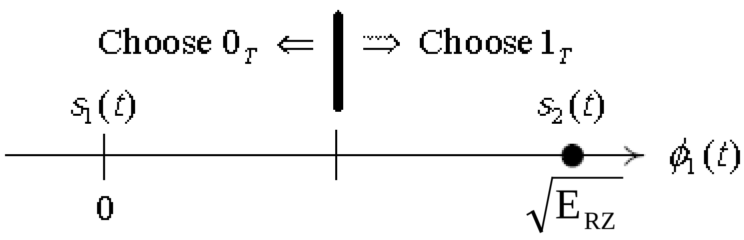


$$P[\text{error}] = Q\left(\frac{\text{distance between the signals}}{2\times\text{noise RMS value}}\right)$$

$$= Q\left(\frac{\sqrt{E_{RZ}}}{2\times\sqrt{N_o/2}}\right) = Q\left(\sqrt{\frac{E_{RZ}}{2N_o}}\right)$$







➢ Return-to-Zero (RZ) Line Code

- **Error Detection and Correction: doesn't have**
- **Power spectral density:** for equal probable 0 and 1

$$S(f)=|P(f)|^2 S_{in}(f)=\frac{V^2(T_b/2)^2\,\text{sinc}^2(\pi f(T_b/2))}{4T_b}\left[1+\frac{1}{T_b}\sum_{m=-\infty}^{m=\infty}\delta\left(f-\frac{m}{T_b}\right)\right]$$

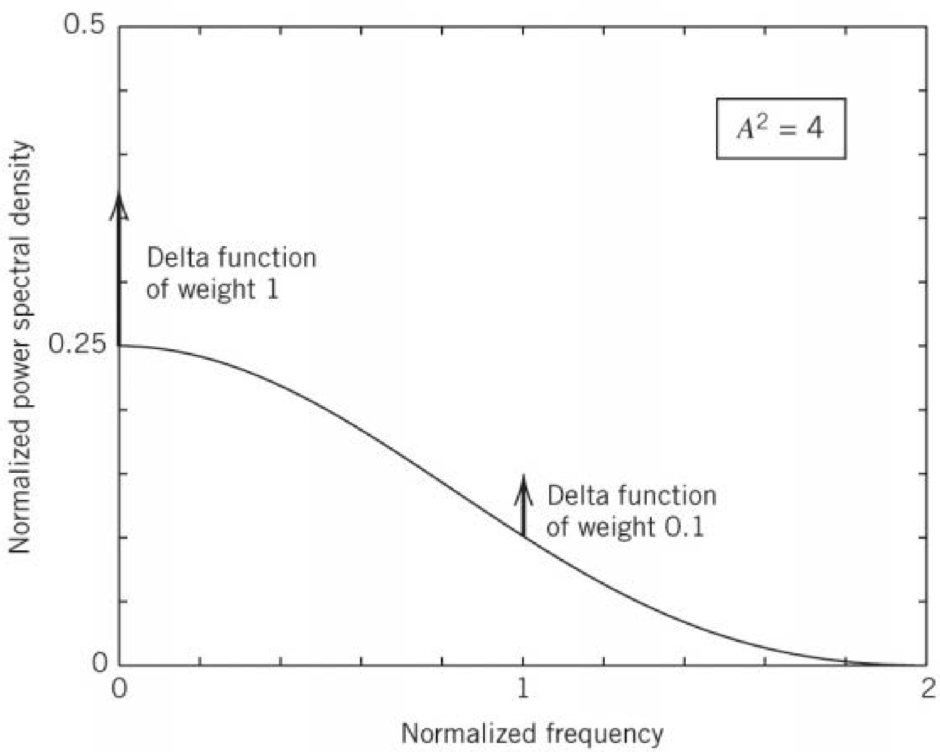
$$=\frac{V^2T_b\,\text{sinc}^2(\pi f(T_b/2))}{16}\left[1+\frac{1}{T_b}\sum_{k=-\infty}^{k=\infty}\delta\left(f-\frac{2k}{T_b}\right)+\frac{1}{T_b}\sum_{n=-\infty}^{n=\infty}\delta\left(f-\frac{2n+1}{T_b}\right)\right]$$

$$=\frac{V^2T_b\,\text{sinc}^2(\pi f(T_b/2))}{16}+\frac{V^2}{16}\delta(f)+\frac{V^2\,\text{sinc}^2\left(\pi\frac{2n+1}{2}\right)}{16}\left[\sum_{n=-\infty}^{n=\infty}\delta\left(f-\frac{2n+1}{T_b}\right)\right]$$

$$=\frac{V^2T_b\,\text{sinc}^2(\pi f(T_b/2))}{16}+\frac{V^2}{16}\delta(f)+\frac{V^2}{16}\times\frac{4}{((2n+1)\pi)^2}\left[\sum_{n=-\infty}^{n=\infty}\delta\left(f-\frac{2n+1}{T_b}\right)\right]$$

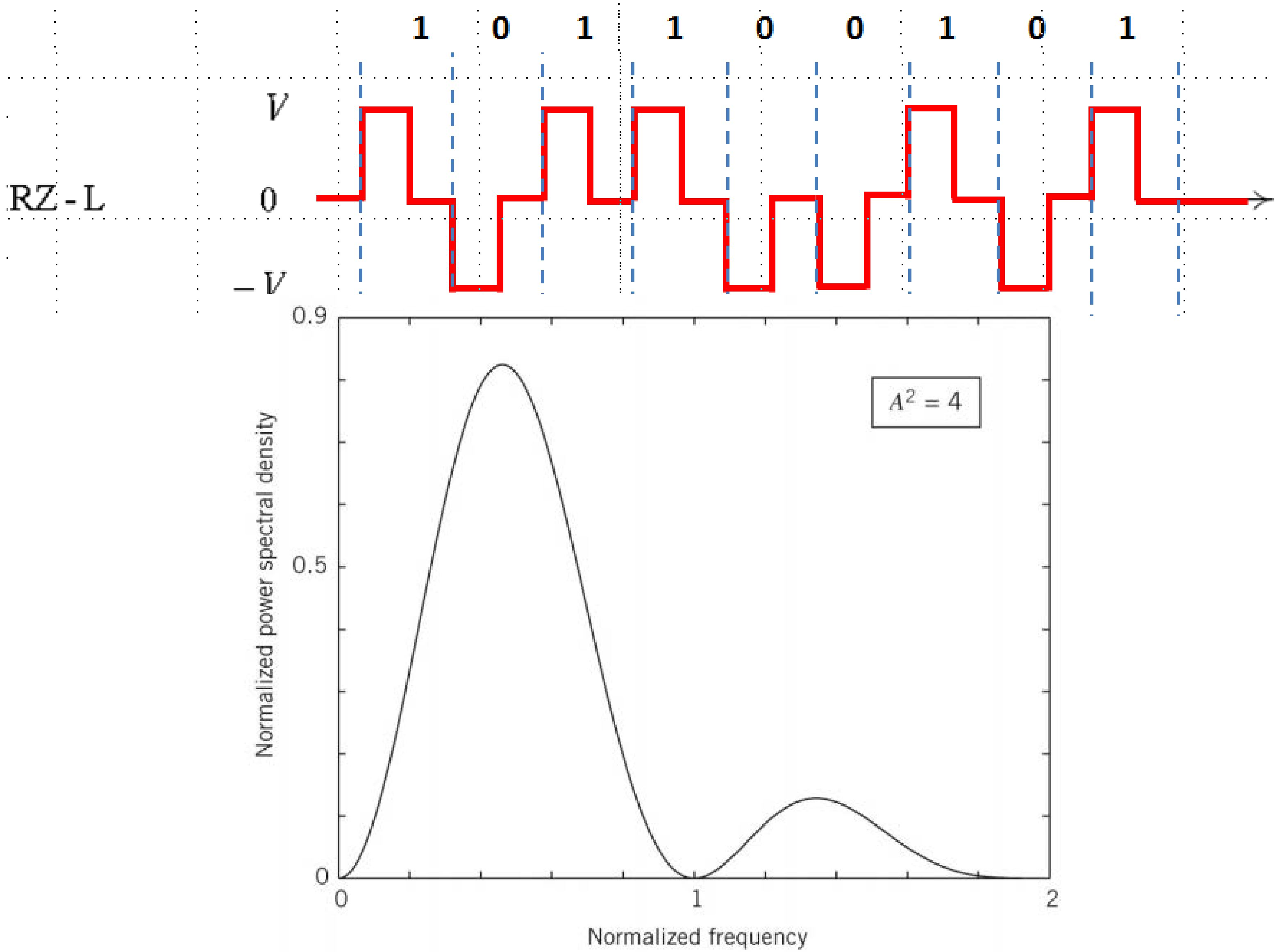
➢ Return-to-Zero (RZ) Line Code

- **Power spectral density**



- **Adequate timing content:** doesn't have any clock component and so is not good for synchronization
- **Transparency:** doesn't prevent long strings of 0s or 1s
- **Cost and complexity:** simple to implement

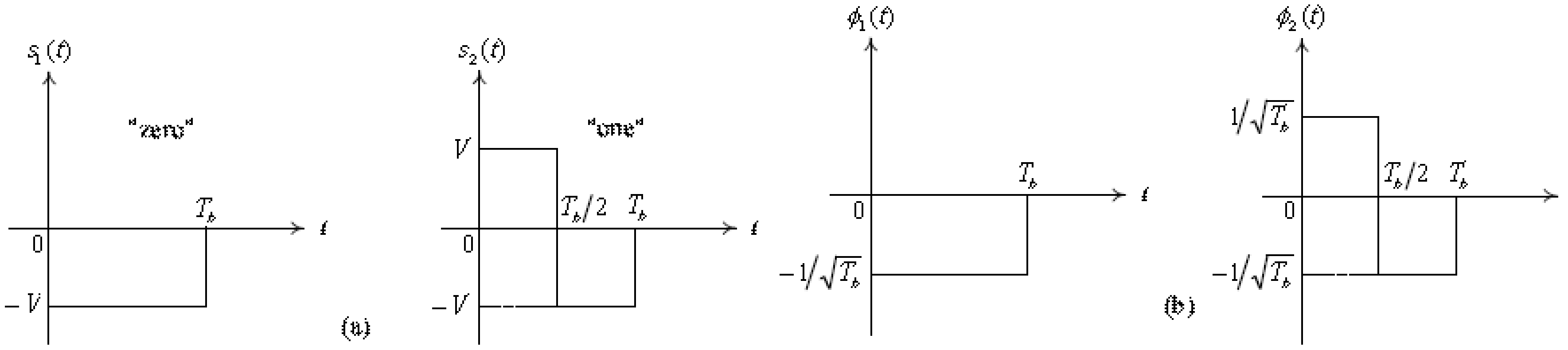
➢ Polar Return-to-Zero (RZ-L) Line Code

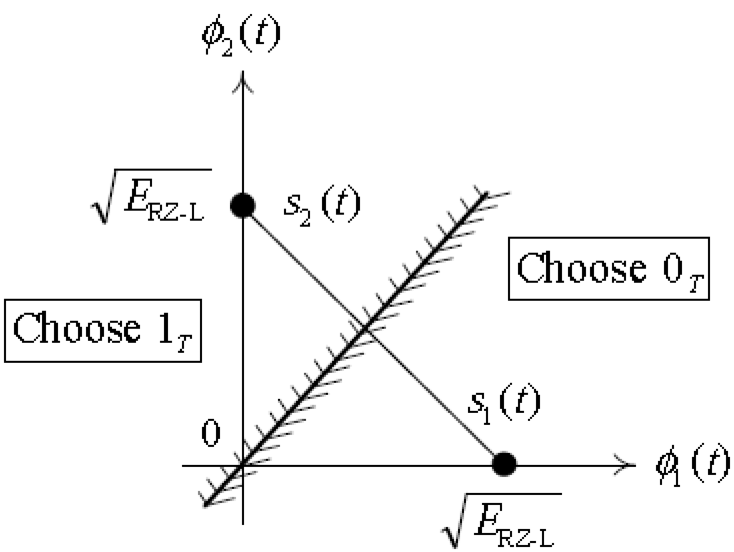


BIRZEIT UNIVERSITY
ECE

- **Polar Return-to-Zero (RZ-L) Line Code**

$$P[\text{error}] = Q\left(\frac{\text{distance between the signals}}{2\times\text{noise RMS value}}\right)$$



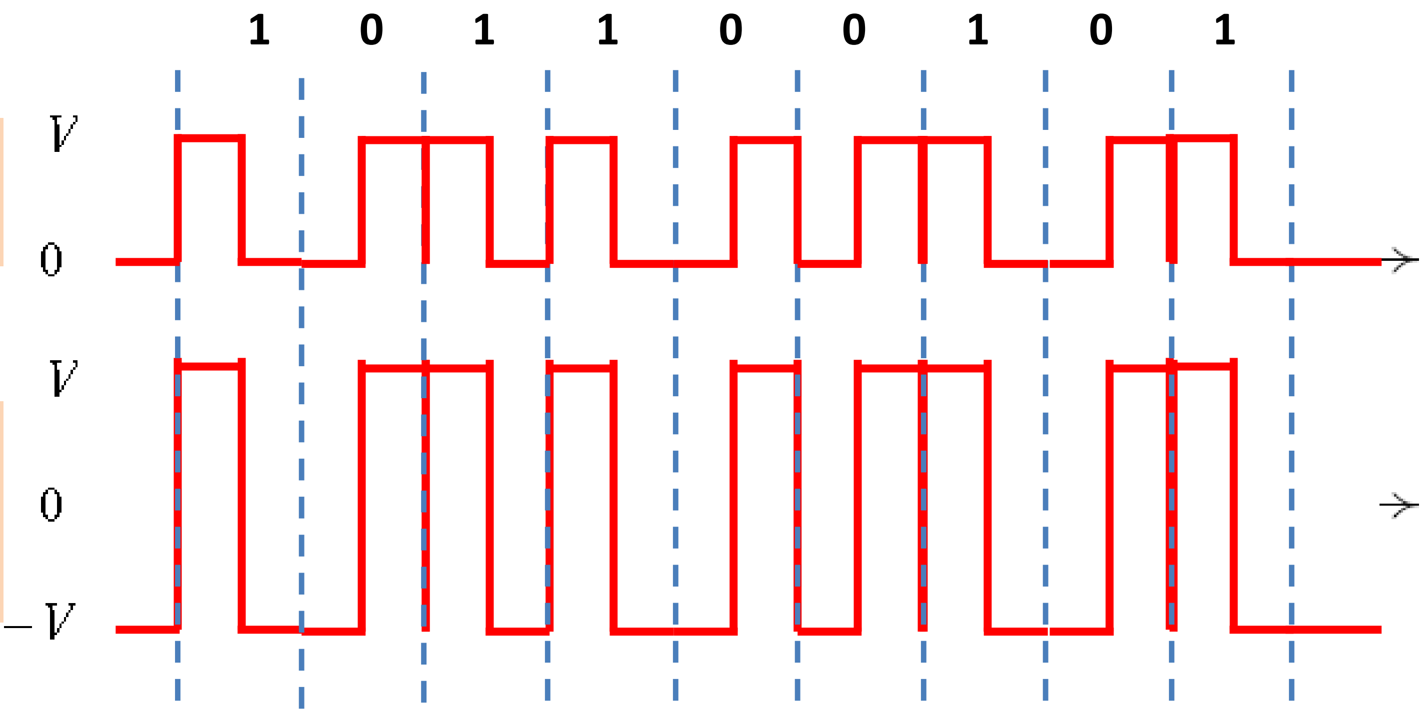


$$P[\text{error}]_{\text{RZ-L}} = Q\left(\sqrt{E_{\text{RZ-L}}/N_0}\right).$$

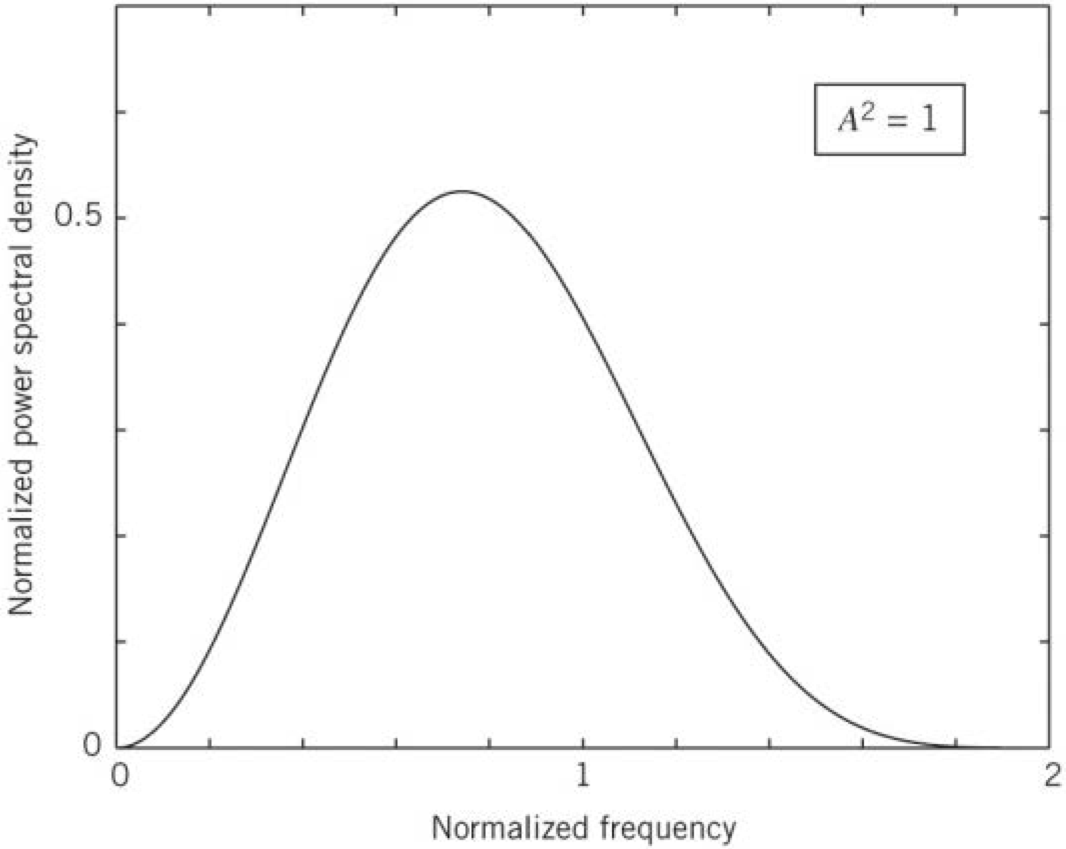
➢ Bi-Phase (Manchester) Line Code

Bi-Phase
Manchester Coding

Bi-Phase-L
Polar Manchester Coding



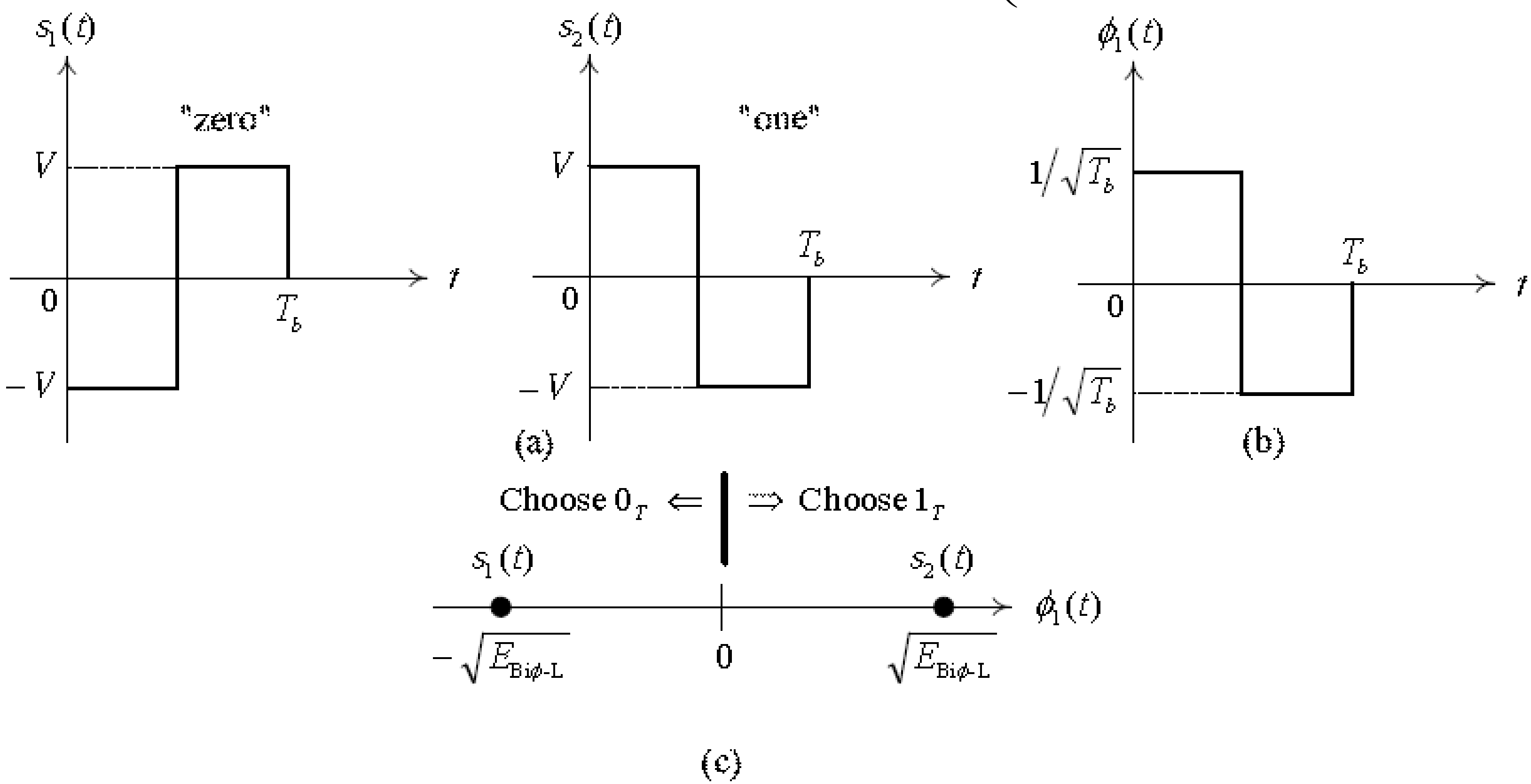
➢Bi-Phase (Manchester) Line Code



BIRZEIT UNIVERSITY
ECE

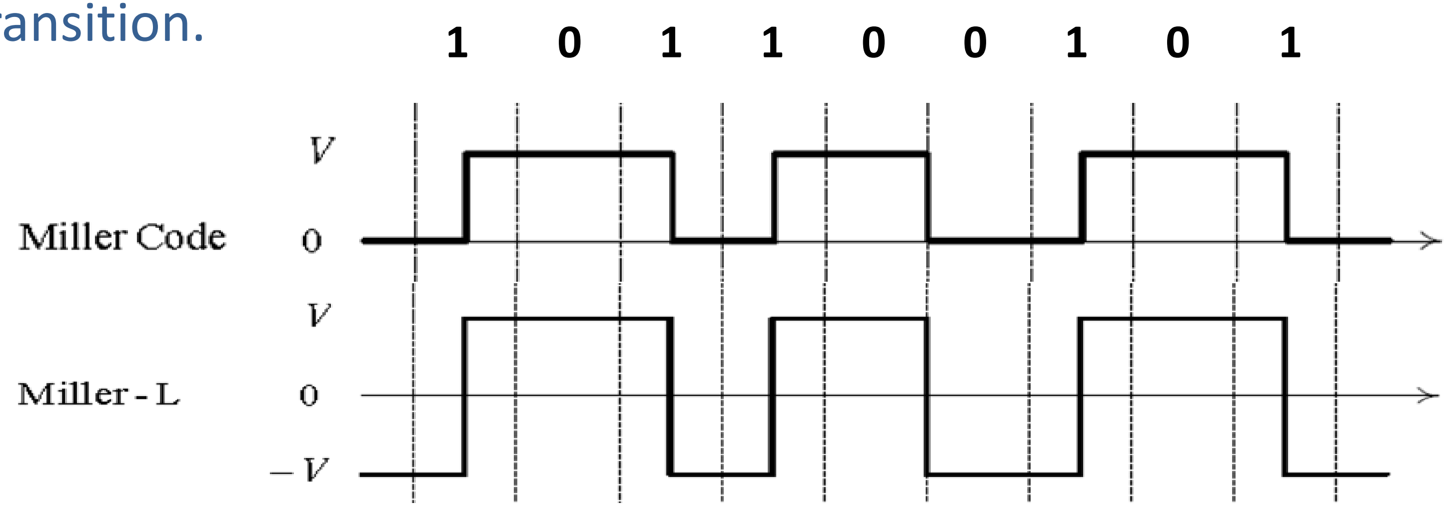
➢Bi-Phase (Manchester) Line Code

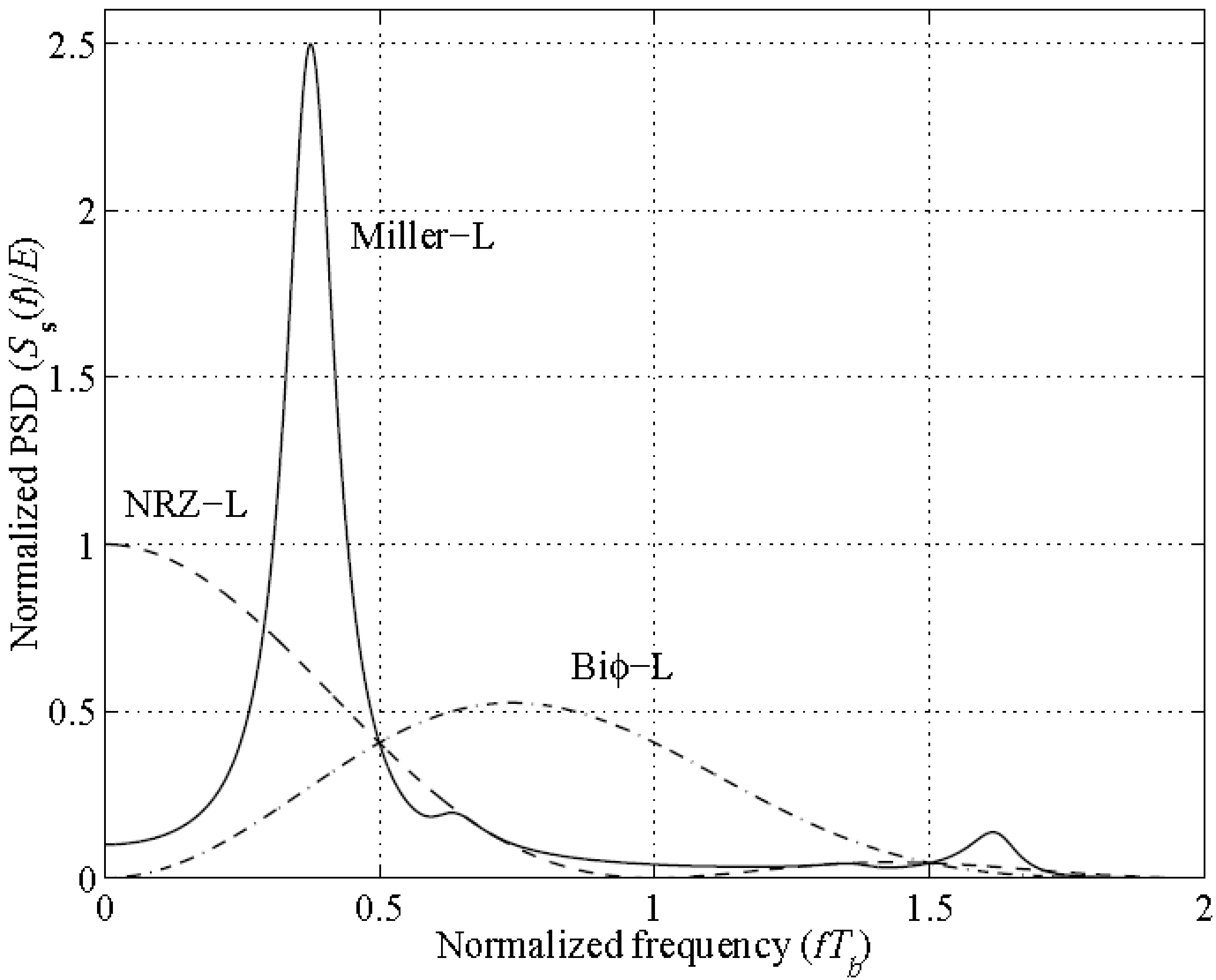
$$P[\text{error}] = Q\left(\frac{\text{distance between the signals}}{2\times\text{noise RMS value}}\right)$$



$$P[\text{error}]_{\text{Bi}\phi\text{-L}} = Q\left(\sqrt{2E_{\text{Bi}\phi\text{-L}}/N_0}\right).$$

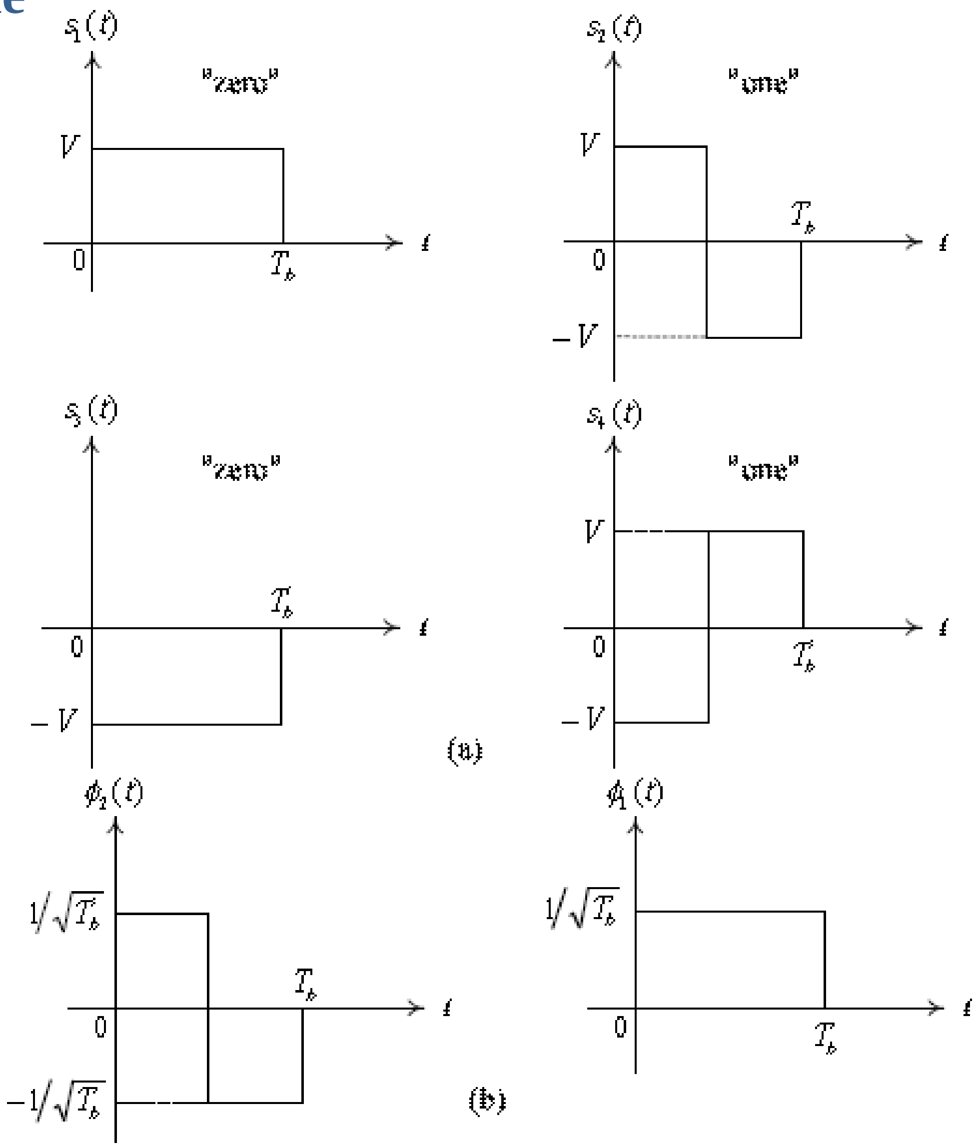
- Has at least one transition every two bit interval and there is never more than two transitions every two bit interval.
  - Bit "1" is encoded by a transition in the middle of the bit interval. Depending on the previous bit this transition may be either upward or downward.
  - Bit "0" is encoded by a transition at the beginning of the bit interval if the previous bit is "0". If the previous bit is "1", then there is no transition.




2.5
2
1.5
1
0.5
0
0
0.5
1
1.5
2
Normalized PSD ($S_s(f)/E$)
Normalized frequency (fT_b)
Miller−L
NRZ−L
Biϕ−L

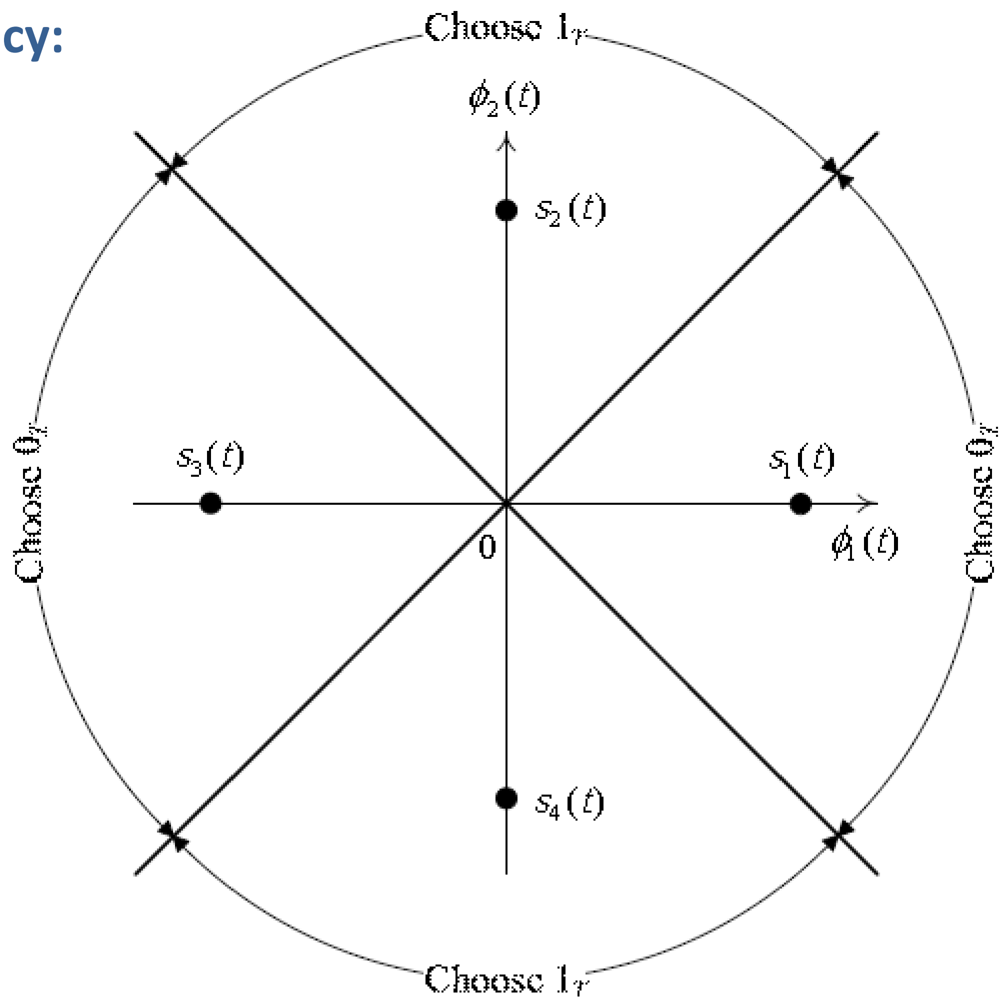
➤Miller Line Code

- **Power Efficiency:**



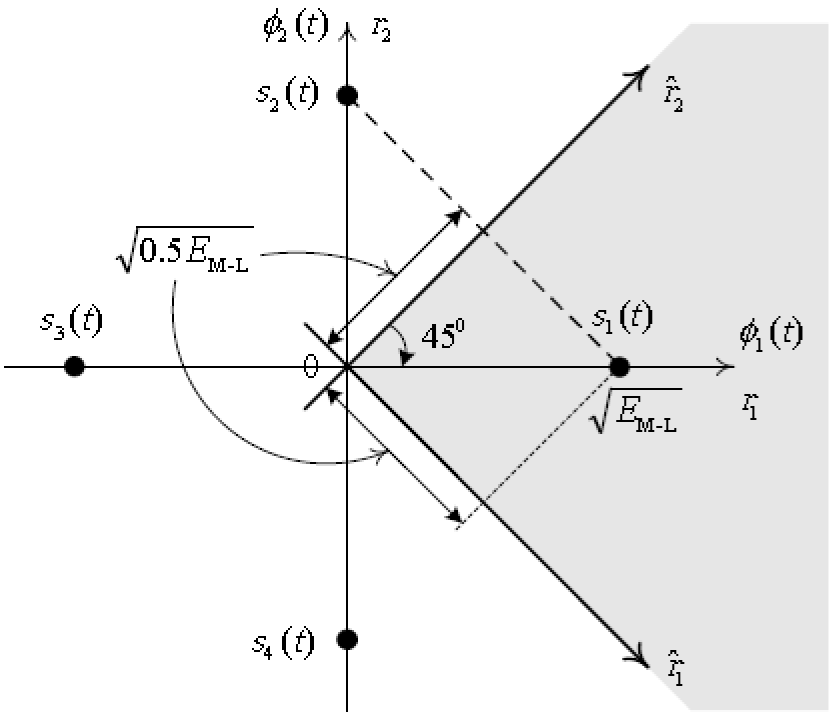
➢Miller Line Code

- **Power Efficiency:**



➢ Miller Line Code

- **Power Efficiency:**

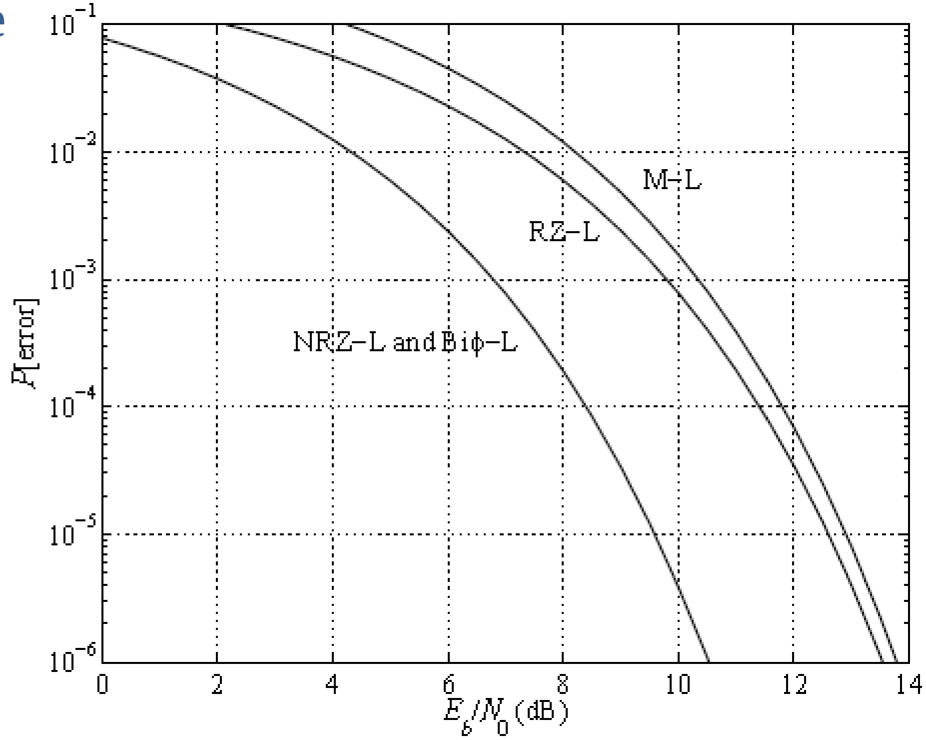


$$P[\text{error}]_{\text{M-L}} = 1 - \left[1 - Q\left(\sqrt{E_{\text{M-L}}/N_0}\right)\right]^2$$

BIRZEIT UNIVERSITY
ECE

➢ Miller Line Code

- **Power Efficiency:**



$$E_{\text{NRZ-L}} = E_{\text{RZ-L}} = E_{\text{Bi}\phi\text{-L}} = E_{\text{M-L}} = V^2 T_b \equiv E_b \text{ (joules/bit)}.$$

$$P[\text{error}]_{\text{NRZ-L}} = P[\text{error}]_{\text{Bi}\phi\text{-L}} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right),$$

$$P[\text{error}]_{\text{RZ-L}} = Q\left(\sqrt{\frac{E_b}{N_0}}\right), \quad P[\text{error}]_{\text{M-L}} \approx 2Q\left(\sqrt{\frac{E_b}{N_0}}\right).$$

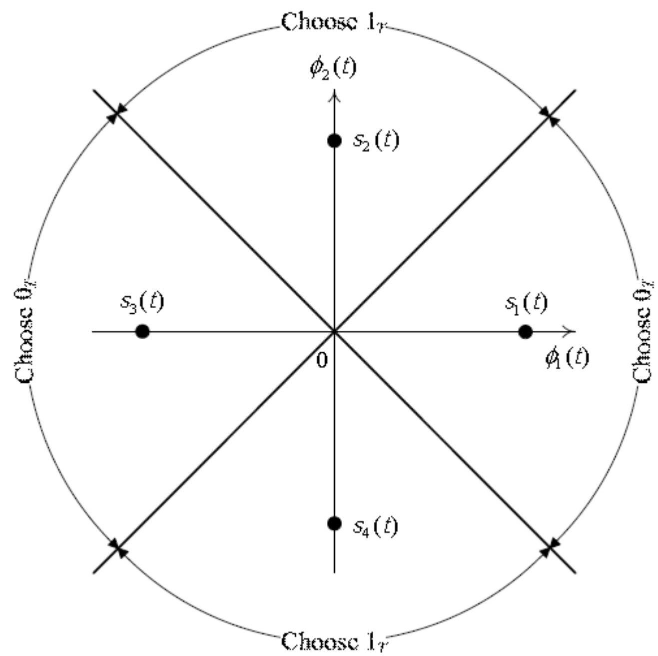
➢ Optimum Sequence Demodulation for Miller Signaling

- Sequence demodulation exploits memory in Miller modulation.
- Example: The four Miller signals have unit energy and the projections of the received signals on to $\phi_1(t)$ and $\phi_2(t)$ are

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$$d_i^{2\,(j)} = \left[\left(r_1^{(j)} - S_{i1}^{(j)}\right)^2 + \left(r_2^{(j)} - S_{i2}^{(j)}\right)^2\right].$$

Transmitted signal	Distance squared			
	$0 \rightarrow T_b$	$T_b \rightarrow 2T_b$	$2T_b \rightarrow 3T_b$	$3T_b \rightarrow 4T_b$
$s_1(t)$				
$s_2(t)$				
$s_3(t)$				
$s_4(t)$				



➢ Optimum Sequence Demodulation for Miller Signaling

- Sequence demodulation exploits memory in Miller modulation.
- Example: The four Miller signals have unit energy and the projections of the received signals on to $\phi_1(t)$ and $\phi_2(t)$ are

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

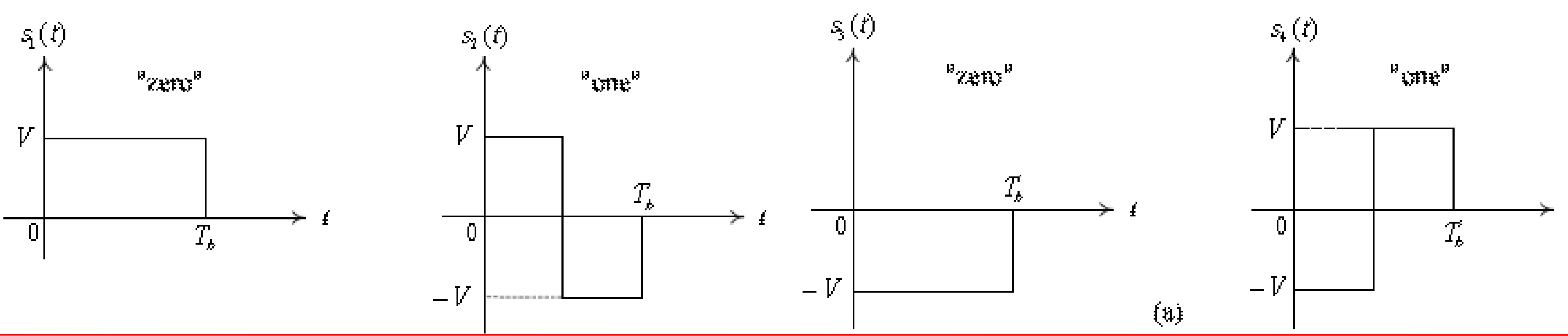
Transmitted signal	Distance squared			
	$0 \rightarrow T_b$	$T_b \rightarrow 2T_b$	$2T_b \rightarrow 3T_b$	$3T_b \rightarrow 4T_b$
$s_1(t)$	1.6	1.28	2.8421	4.42
$s_2(t)$	2.0	3.28	0.6221	2.02
$s_3(t)$	0.8	2.08	**0.4021**	**0.02**
$s_4(t)$	**0.4**	**0.08**	2.6221	2.42

➤Optimum Sequence Demodulation for Miller Signaling

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$

$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

Transmitted signal	Distance squared			
	$0 \to T_b$	$T_b \to 2T_b$	$2T_b \to 3T_b$	$3T_b \to 4T_b$
$s_1(t)$	1.6	1.28	2.8421	4.42
$s_2(t)$	2.0	3.28	0.6221	2.02
$s_3(t)$	0.8	2.08	**0.4021**	**0.02**
$s_4(t)$	**0.4**	**0.08**	2.6221	2.42



Optimum Sequence Demodulation for Miller Signaling

- Sequence demodulation exploits memory in Miller modulation.
- Example: The four Miller signals have unit energy and the projections of the received signals on to $\phi_1(t)$ and $\phi_2(t)$ are

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

Transmitted signal	Distance squared			
	$0 \to T_b$	$T_b \to 2T_b$	$2T_b \to 3T_b$	$3T_b \to 4T_b$
$s_1(t)$	1.6	1.28	2.8421	4.42
$s_2(t)$	2.0	3.28	0.6221	2.02
$s_3(t)$	0.8	2.08	**0.4021**	**0.02**
$s_4(t)$	**0.4**	**0.08**	2.6221	2.42

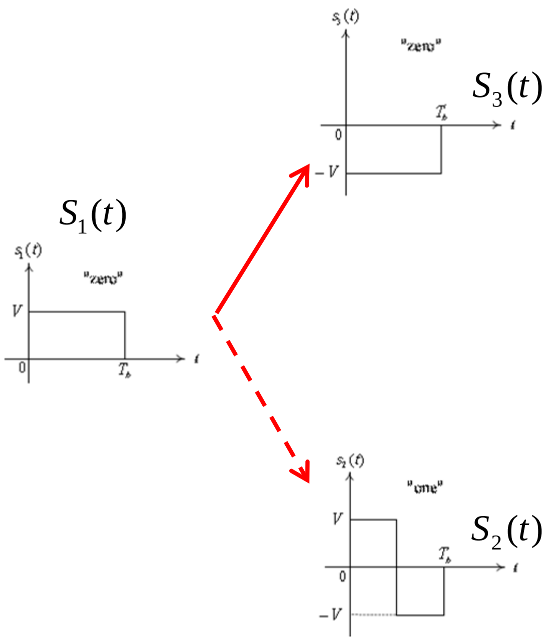
$\{s_4(t), s_4(t), s_3(t), s_3(t)\}$ is not a *valid* transmitted sequence!

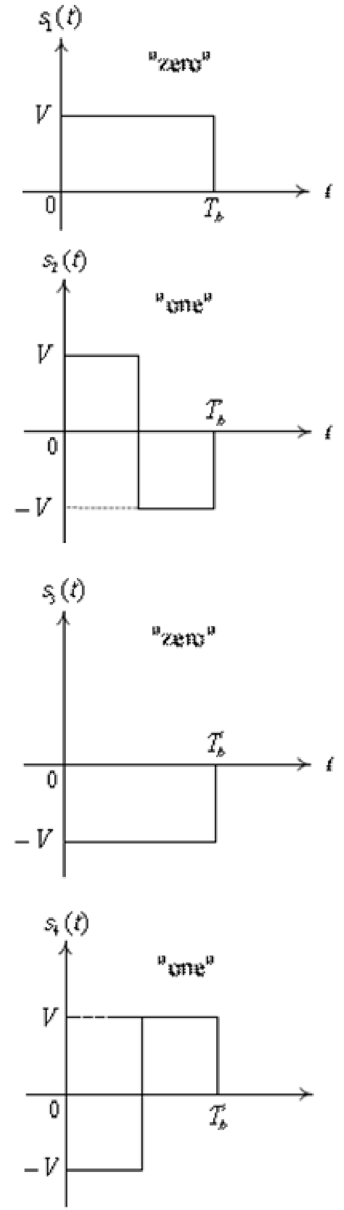
➢ Optimum Sequence Demodulation for Miller Signaling

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$$\begin{aligned} d_i^2 &= \int_0^{nT_b} [r(t) - S_i(t)]^2 \mathrm{d}t = \sum_{j=1}^{n} \int_{(j-1)T_b}^{jT_b} [r_j(t) - S_{ij}(t)]^2 \mathrm{d}t \\ &= \sum_{j=1}^{n} \left[\left(r_1^{(j)} - S_{i1}^{(j)}\right)^2 + \left(r_2^{(j)} - S_{i2}^{(j)}\right)^2\right]. \end{aligned}$$

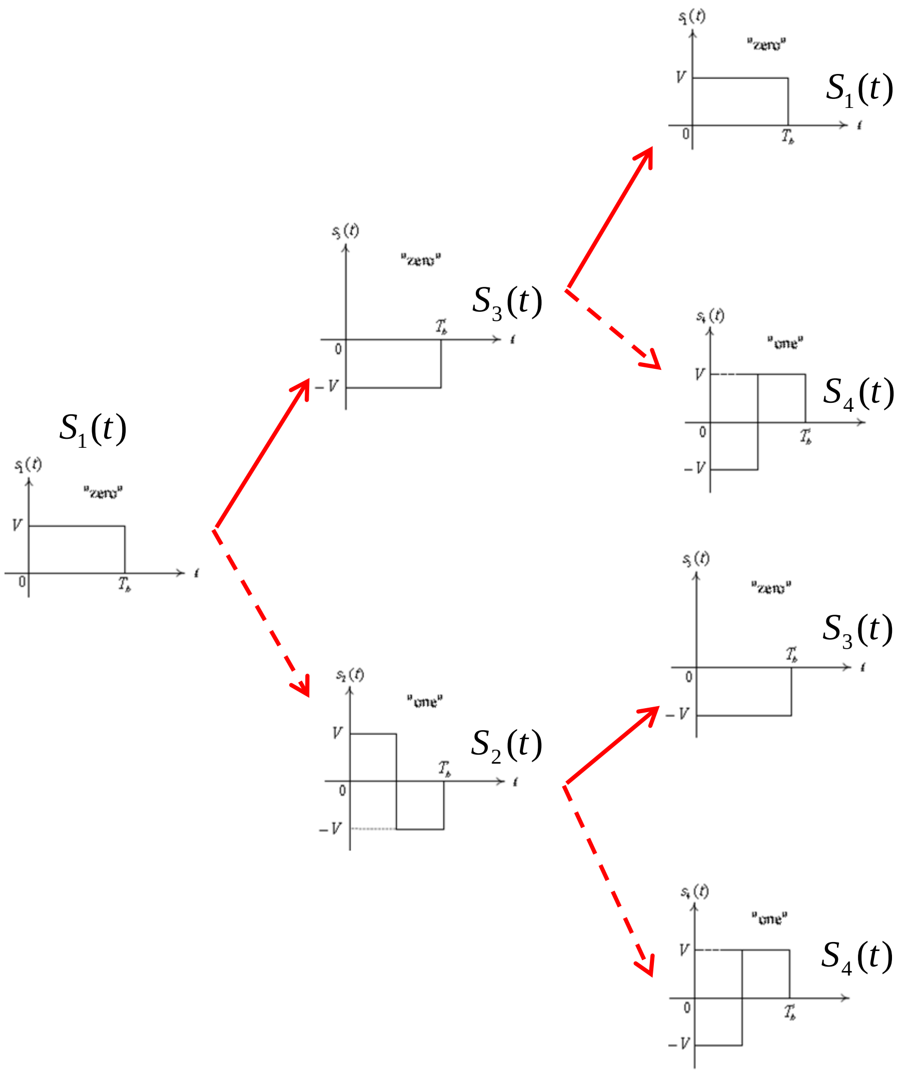
State Diagram

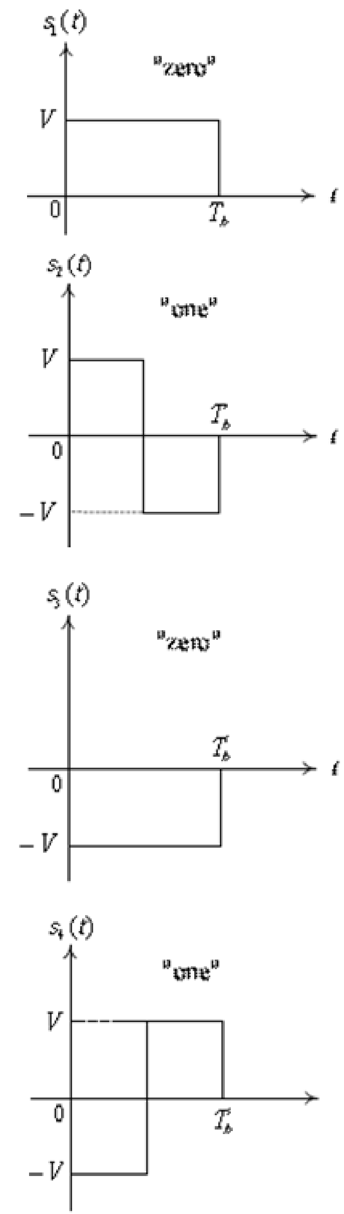




BIRZEIT UNIVERSITY
ECE

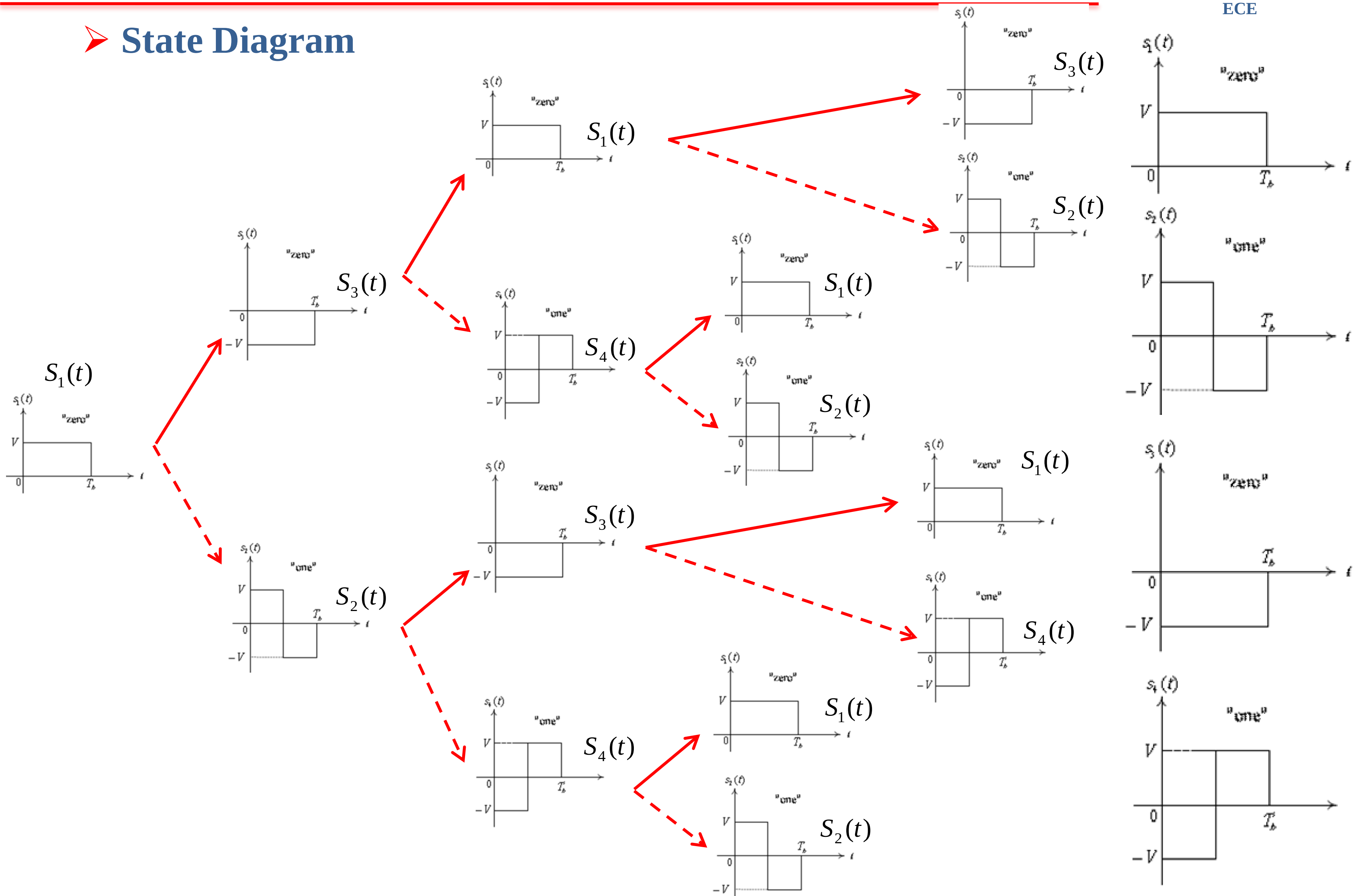
State Diagram



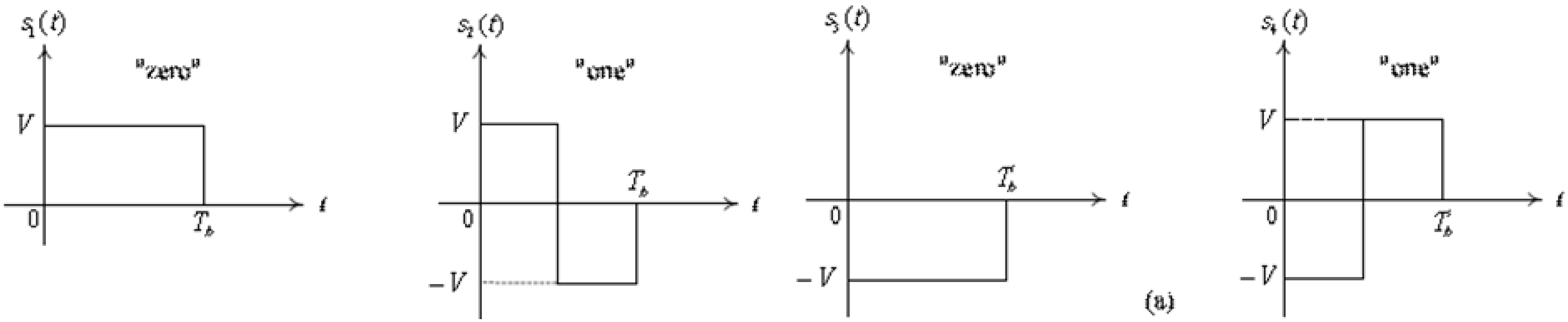


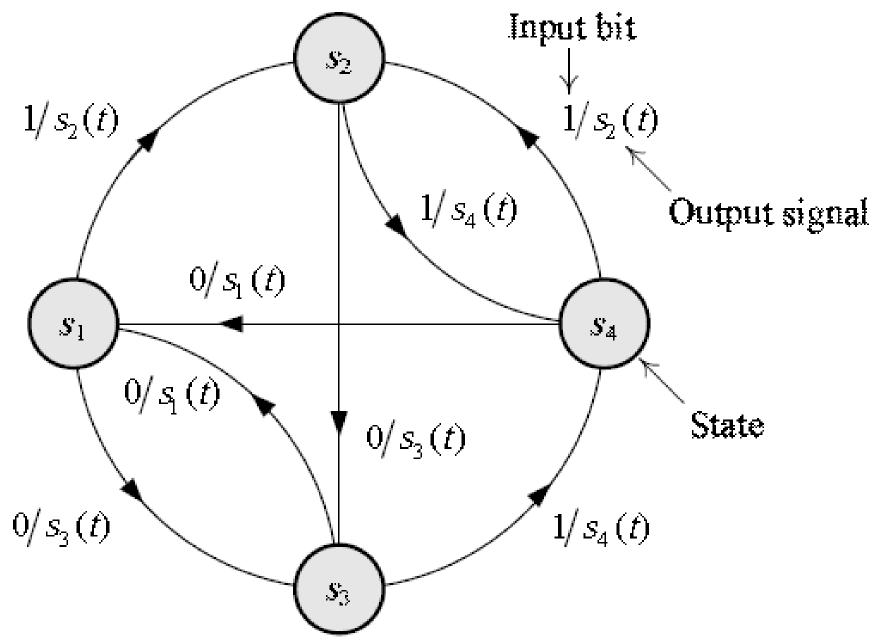
BIRZEIT UNIVERSITY
ECE

➢ State Diagram

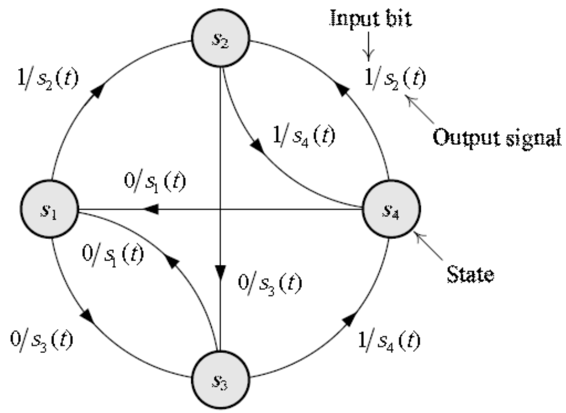


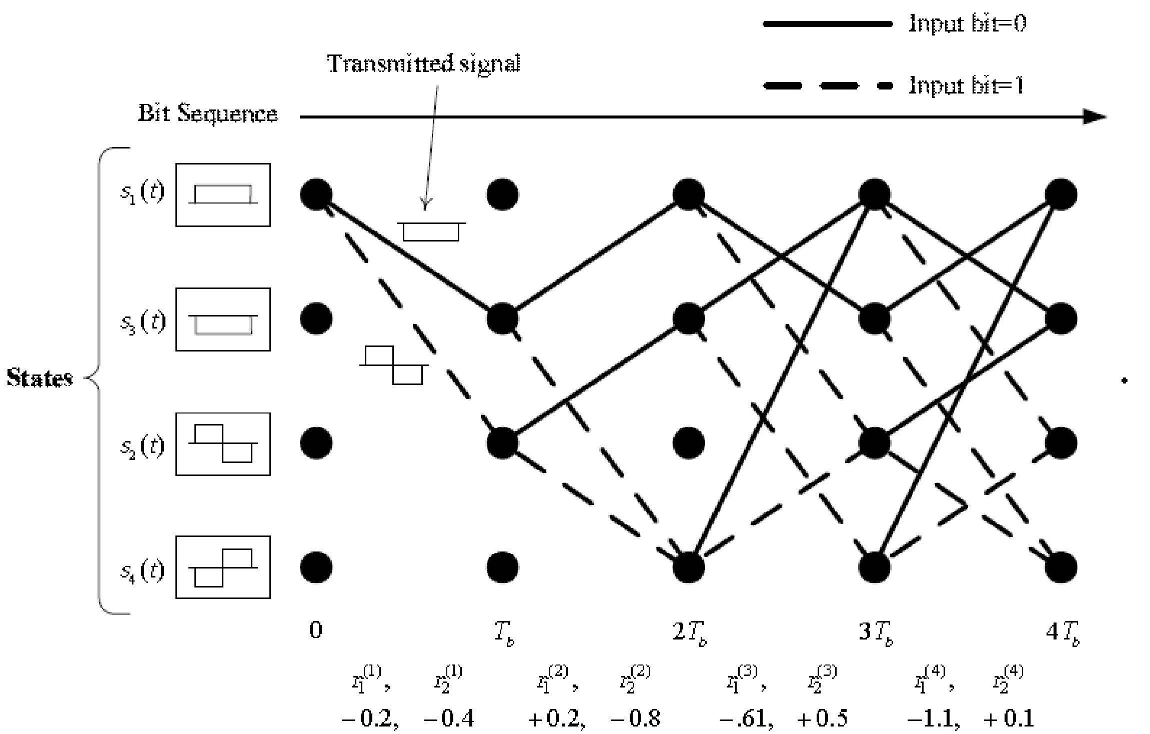
➢ State Diagram





➢ Trellis Diagram



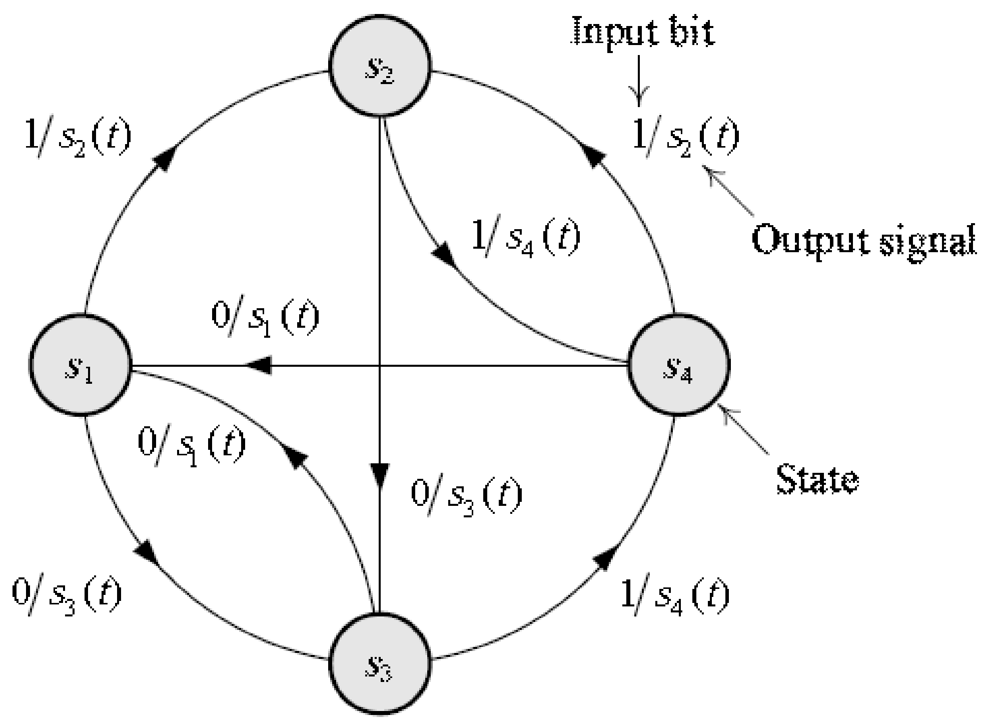


- **Viterbi Algorithm**
  - **Step 1:** Start from the initial state ($s_1(t)$ in our case).
  - **Step 2:** In each bit interval, calculate the branch metric, which is the distance squared between the received signal in that interval with the signal corresponding to each possible branch. Add this branch metric to the previous metrics to get the partial path metric for each partial path up to that bit interval.
  - **Step 3**: If there are two partial paths entering the same state, discard the one that has a larger partial path metric and call the remaining path the survivor.
  - **Step 4**: Extend only the survivor paths to the next interval. Repeat Steps 2 to 4 till the end of the sequence.

➤ Viterbi Algorithm

- Example: The four Miller signals have unit energy and the projections of the received signals on to $\phi_1(t)$ and $\phi_2(t)$ are

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$



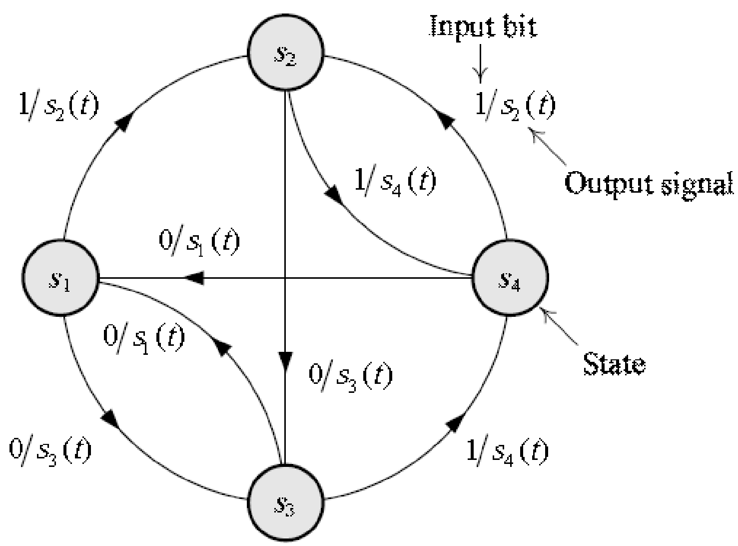
Viterbi Algorithm

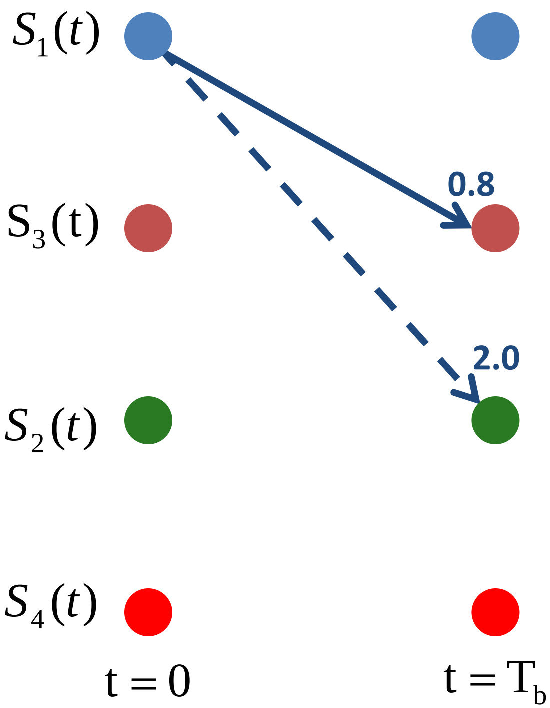
$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$

$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$$d^2_{S_3 \leftrightarrow r^{(1)}} = \left(S_{31} - r_1^{(1)}\right)^2 + \left(S_{32} - r_2^{(1)}\right)^2 = (-1+0.2)^2 + (0+0.4)^2 = 0.64 + 0.16 = 0.8$$

$$d^2_{S_2 \leftrightarrow r^{(1)}} = \left(S_{21} - r_1^{(1)}\right)^2 + \left(S_{22} - r_2^{(1)}\right)^2 = (0+0.2)^2 + (1+0.4)^2 = 0.04 + 1.96 = 2.0$$



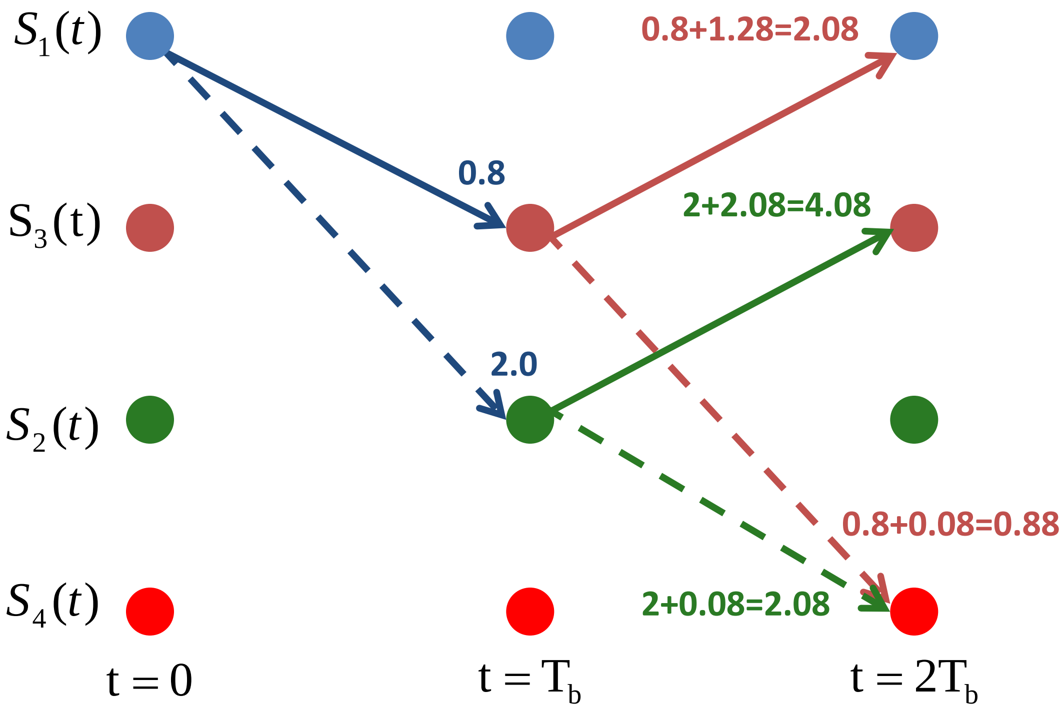


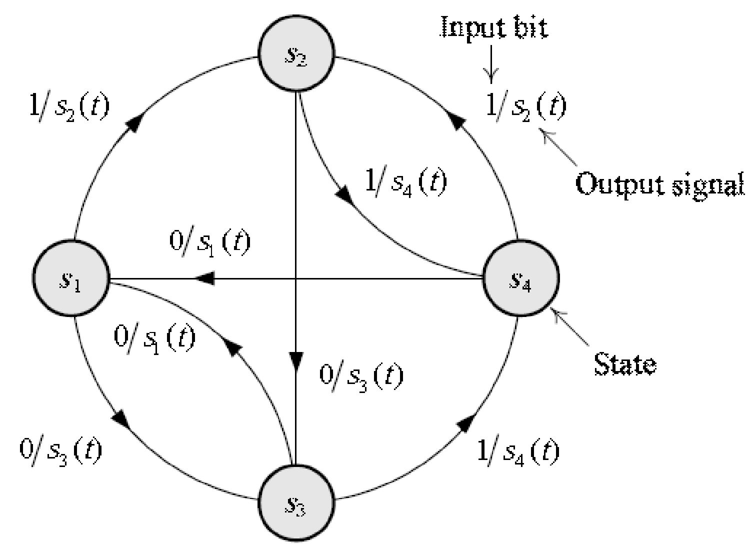
BIRZEIT UNIVERSITY
ECE

➤ Viterbi Algorithm

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

Transmitted signal	Distance squared			
	$0 \to T_b$	$T_b \to 2T_b$	$2T_b \to 3T_b$	$3T_b \to 4T_b$
$s_1(t)$	1.6	1.28	2.8421	4.42
$s_2(t)$	2.0	3.28	0.6221	2.02
$s_3(t)$	0.8	2.08	**0.4021**	**0.02**
$s_4(t)$	**0.4**	**0.08**	2.6221	2.42

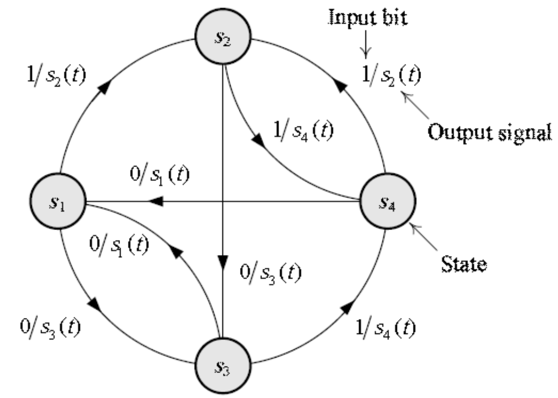


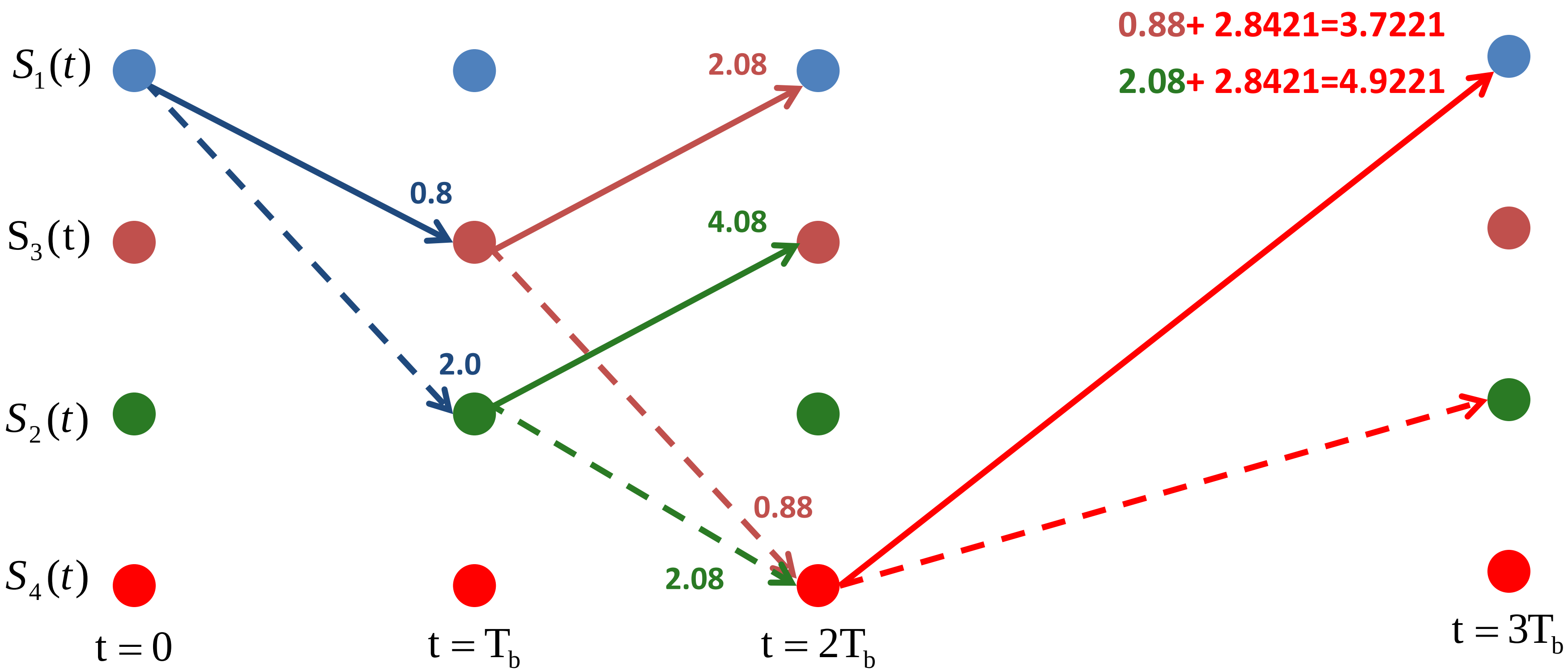


➢ Viterbi Algorithm

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$2T_b \rightarrow 3T_b$
2.8421
0.6221
0.4021
2.6221



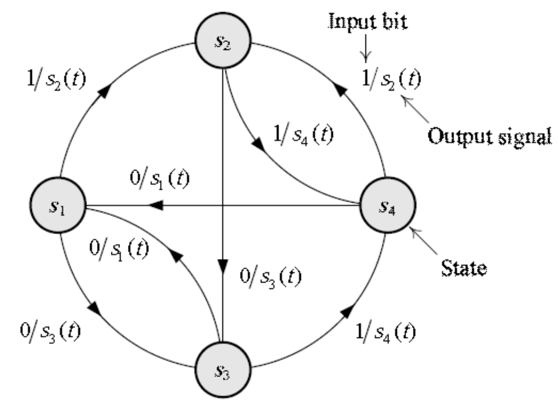


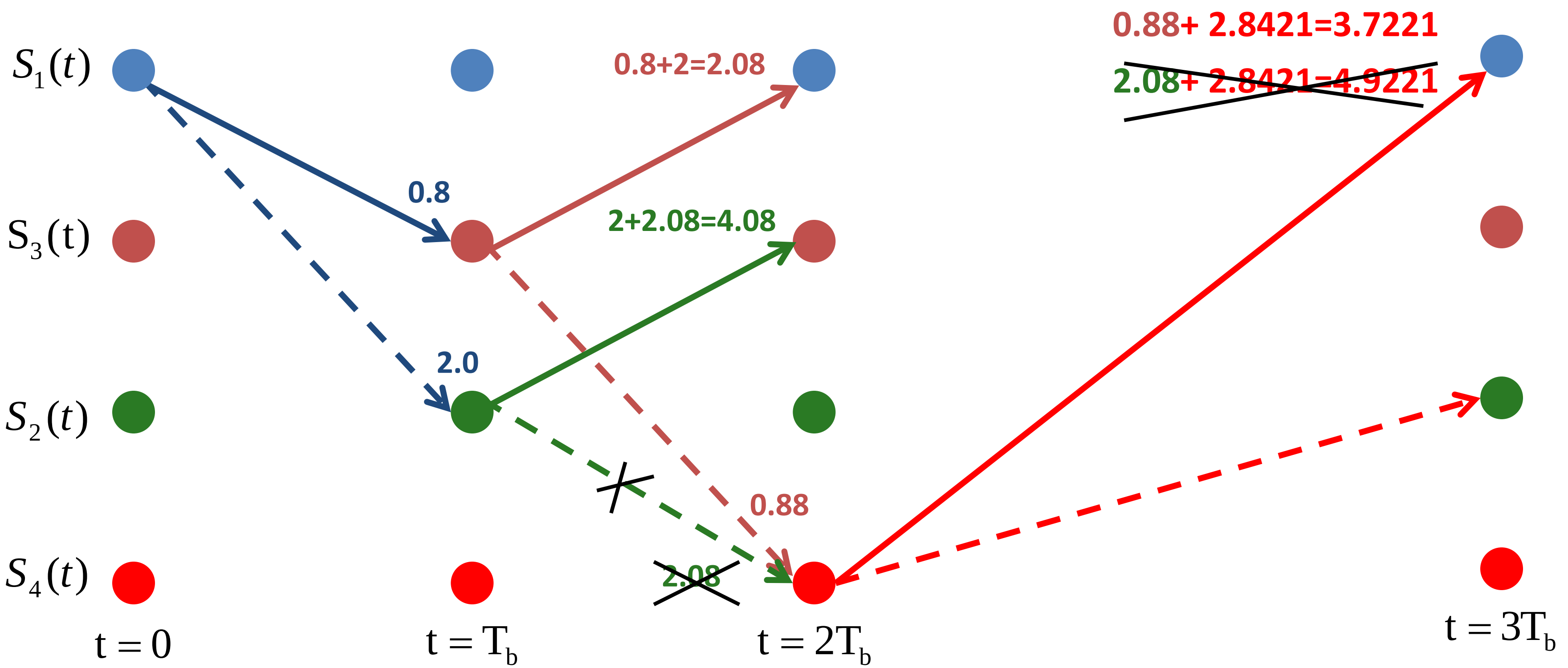
BIRZEIT UNIVERSITY
ECE

Viterbi Algorithm

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$2T_b \rightarrow 3T_b$
2.8421
0.6221
0.4021
2.6221

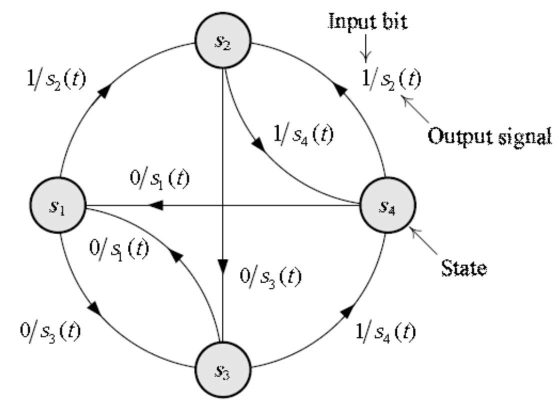


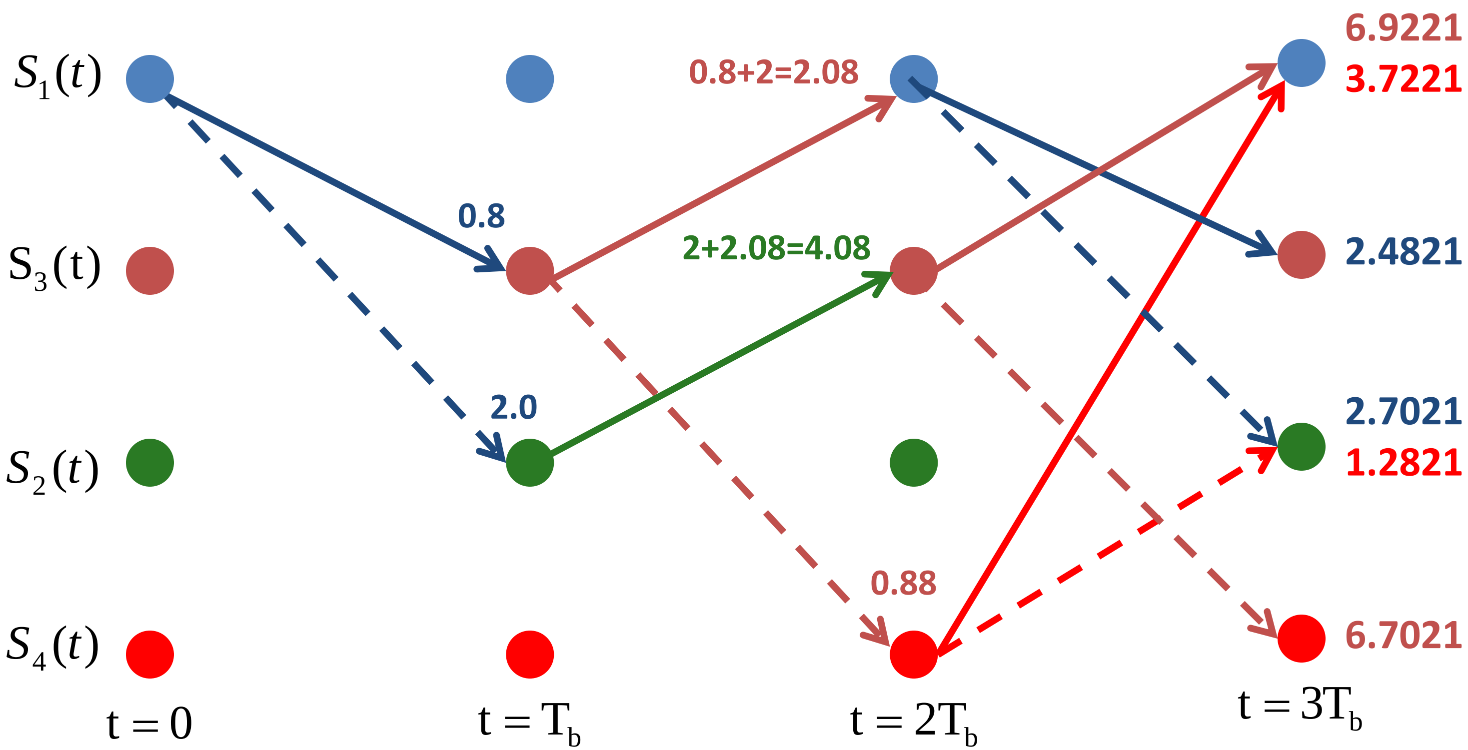


➢ Viterbi Algorithm

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$2T_b \rightarrow 3T_b$
2.8421
0.6221
0.4021
2.6221

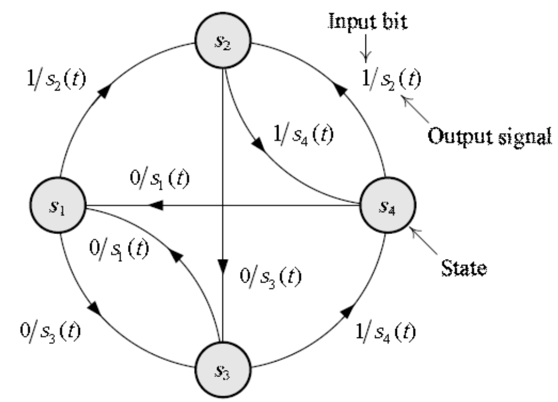


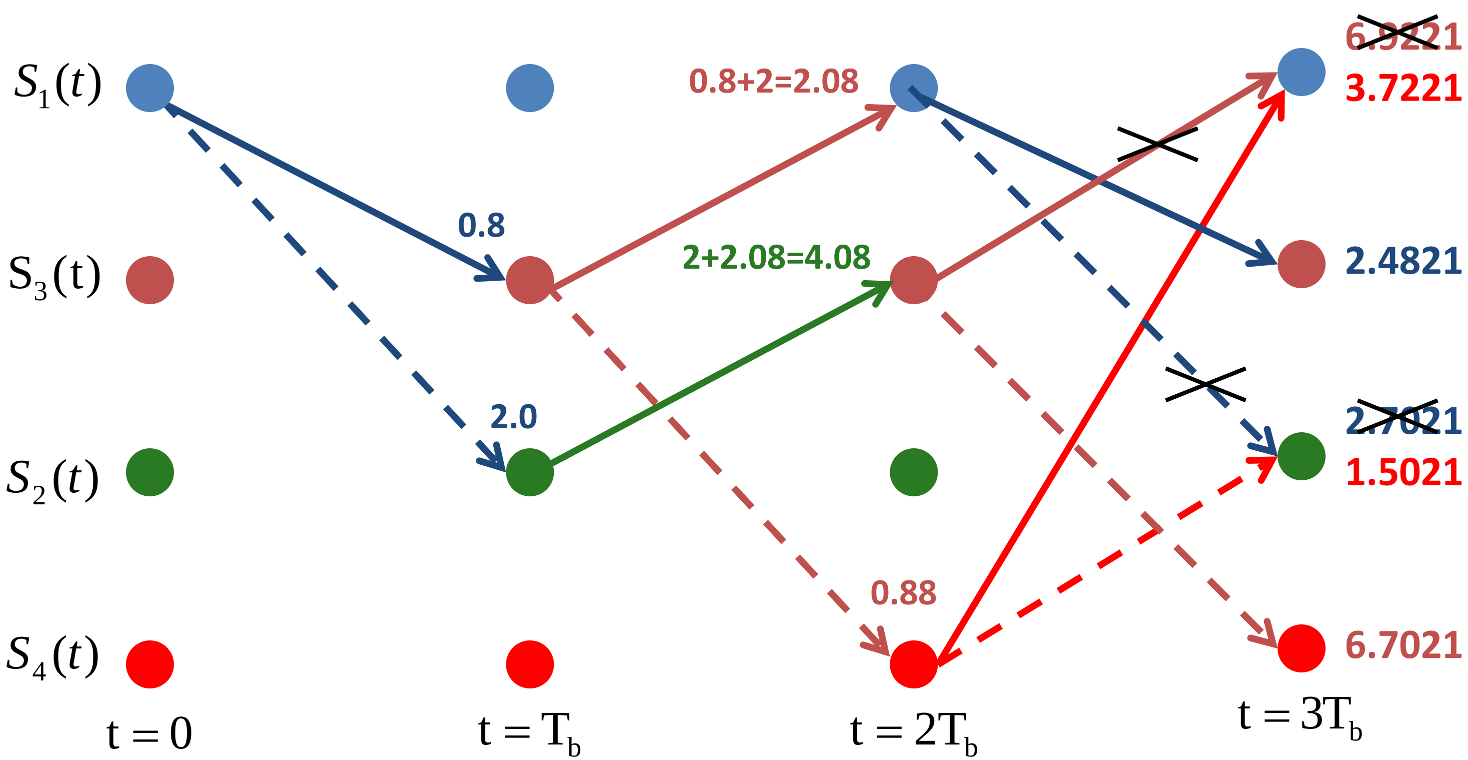


➢ Viterbi Algorithm

$$\left\{r_1^{(1)}=-0.2, r_2^{(1)}=-0.4\right\}, \left\{r_1^{(2)}=+0.2, r_2^{(2)}=-0.8\right\},$$
$$\left\{r_1^{(3)}=-0.61, r_2^{(3)}=+0.5\right\}, \left\{r_1^{(4)}=-1.1, r_2^{(4)}=+0.1\right\}.$$

$2T_b \rightarrow 3T_b$
2.8421
0.6221
0.4021
2.6221

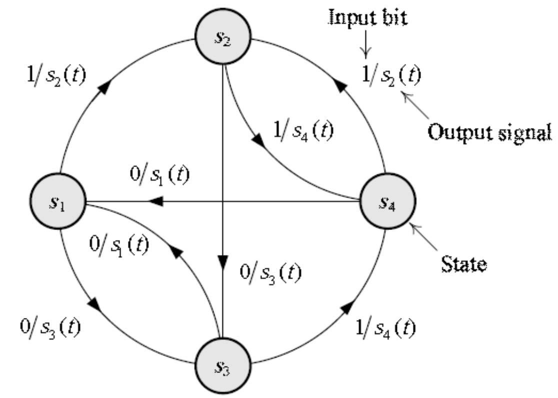


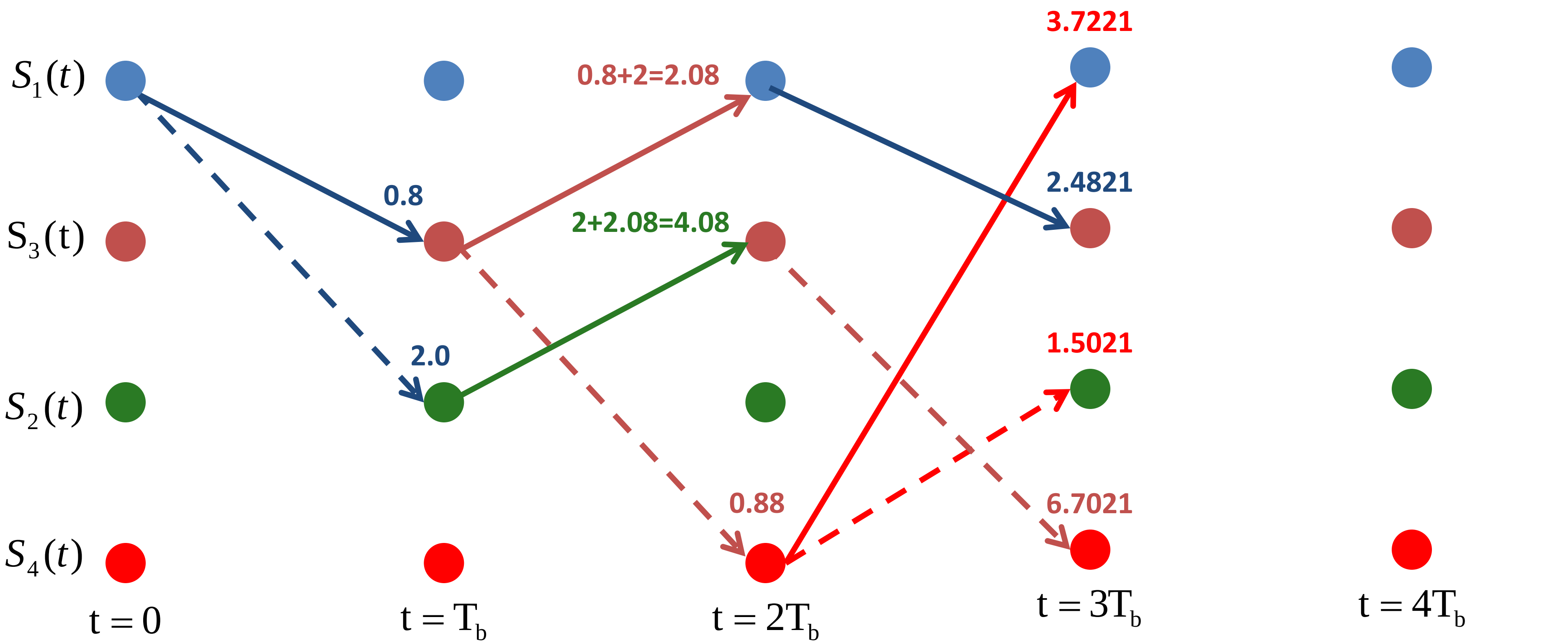


Viterbi Algorithm

$$\left\{r_1^{(1)}=-0.2, r_2^{(1)}=-0.4\right\}, \left\{r_1^{(2)}=+0.2, r_2^{(2)}=-0.8\right\},$$
$$\left\{r_1^{(3)}=-0.61, r_2^{(3)}=+0.5\right\}, \left\{r_1^{(4)}=-1.1, r_2^{(4)}=+0.1\right\}.$$

$3T_b \to 4T_b$
4.42
2.02
0.02
2.42



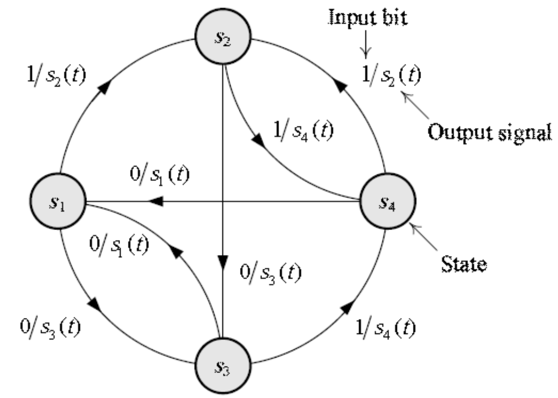


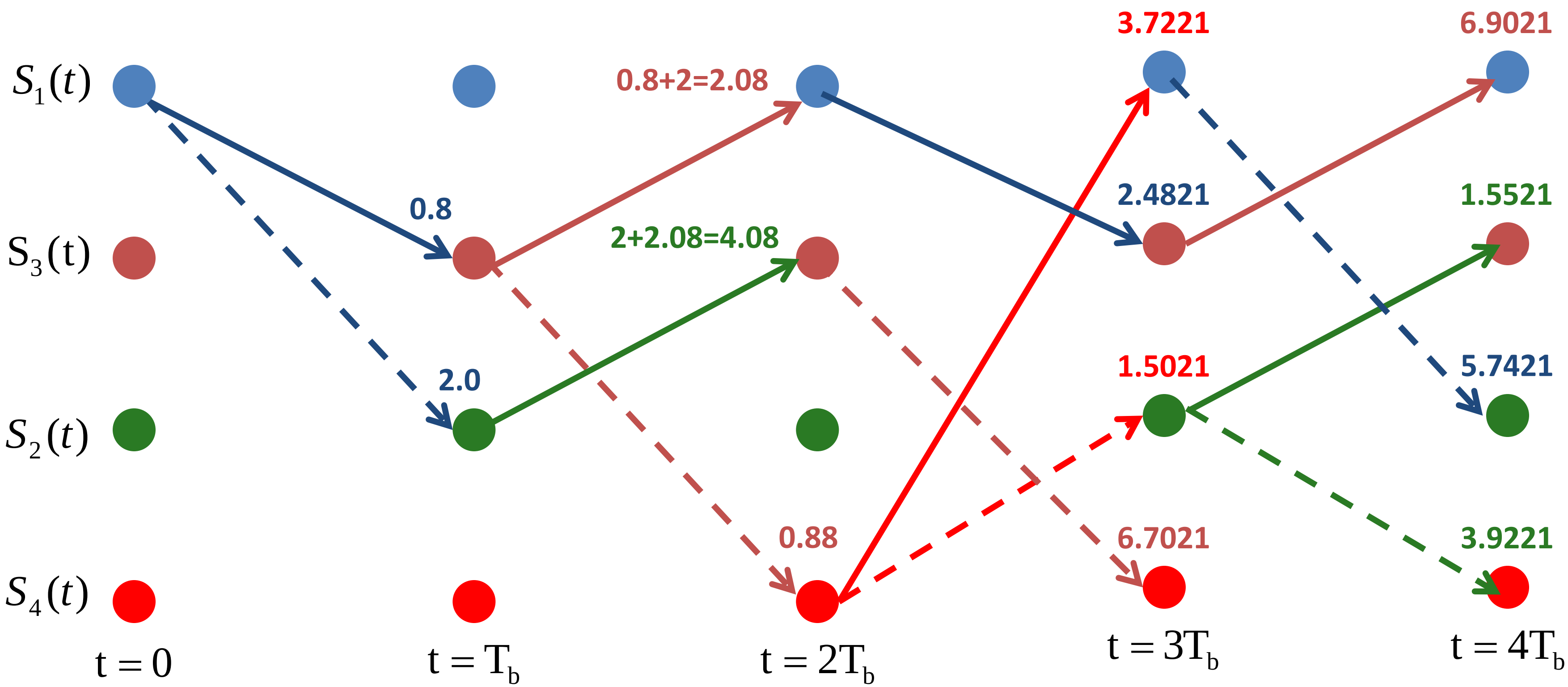
BIRZEIT UNIVERSITY
ECE

➤ Viterbi Algorithm

$$\left\{r_1^{(1)}=-0.2, r_2^{(1)}=-0.4\right\},\left\{r_1^{(2)}=+0.2, r_2^{(2)}=-0.8\right\},$$
$$\left\{r_1^{(3)}=-0.61, r_2^{(3)}=+0.5\right\},\left\{r_1^{(4)}=-1.1, r_2^{(4)}=+0.1\right\}.$$

$3T_b \rightarrow 4T_b$
4.42
2.02
0.02
2.42



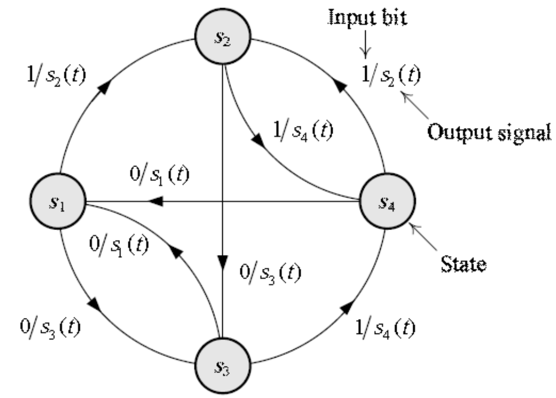


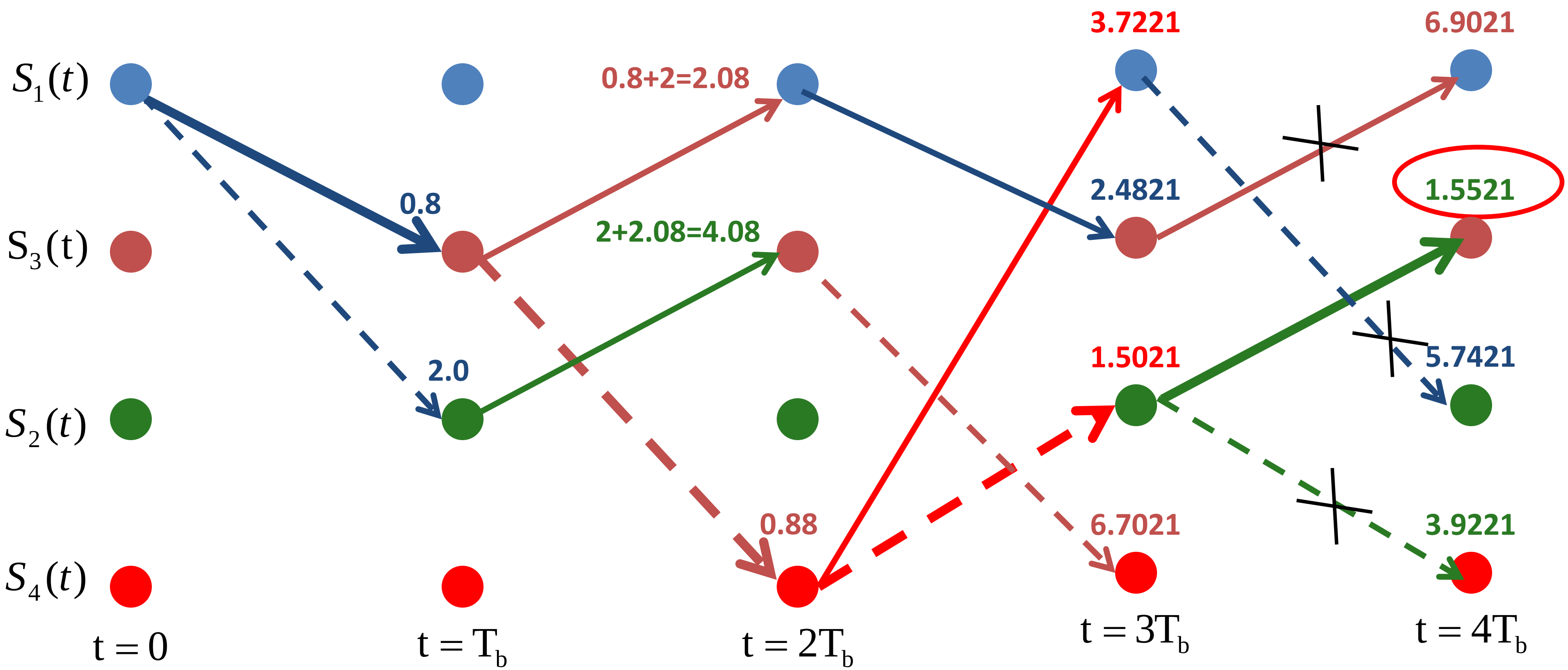
BIRZEIT UNIVERSITY
ECE

➢ Viterbi Algorithm

$$\left\{r_1^{(1)} = -0.2, r_2^{(1)} = -0.4\right\}, \left\{r_1^{(2)} = +0.2, r_2^{(2)} = -0.8\right\},$$
$$\left\{r_1^{(3)} = -0.61, r_2^{(3)} = +0.5\right\}, \left\{r_1^{(4)} = -1.1, r_2^{(4)} = +0.1\right\}.$$

$3T_b \to 4T_b$
4.42
2.02
0.02
2.42

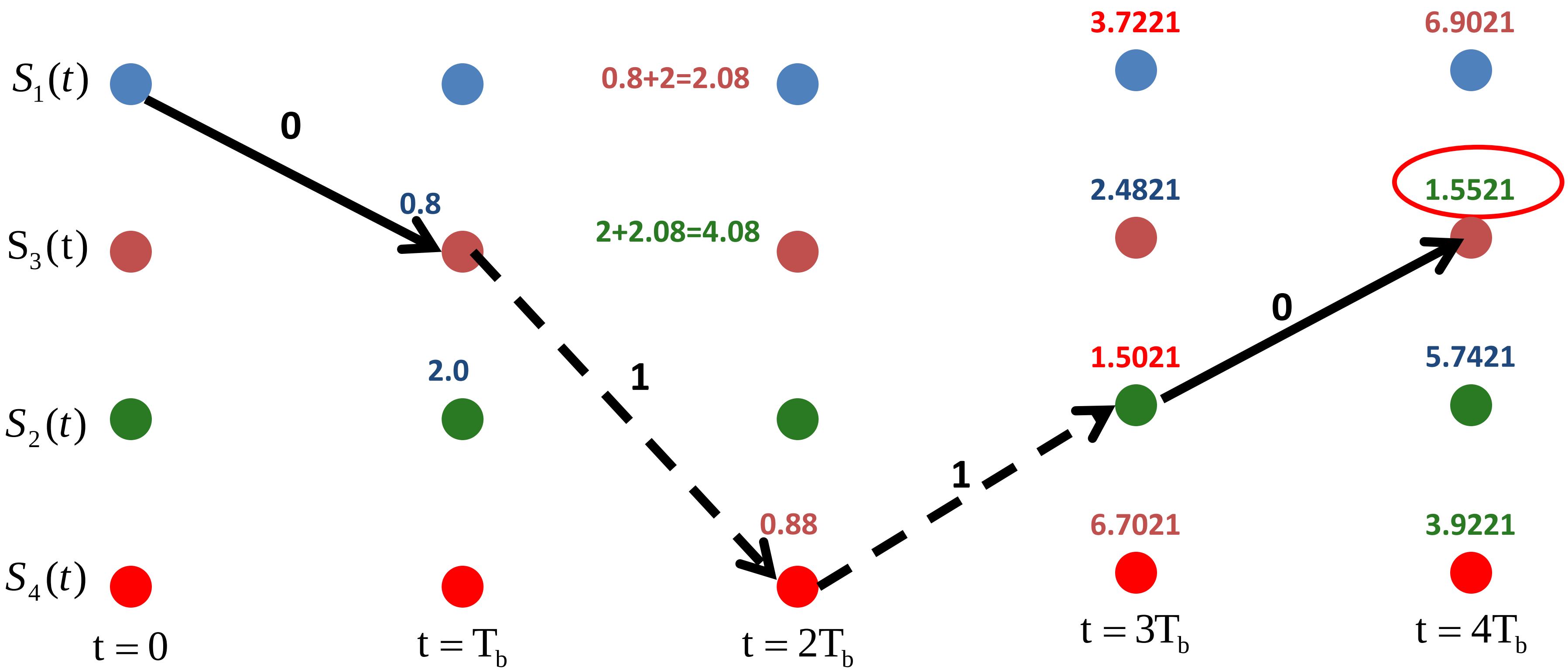




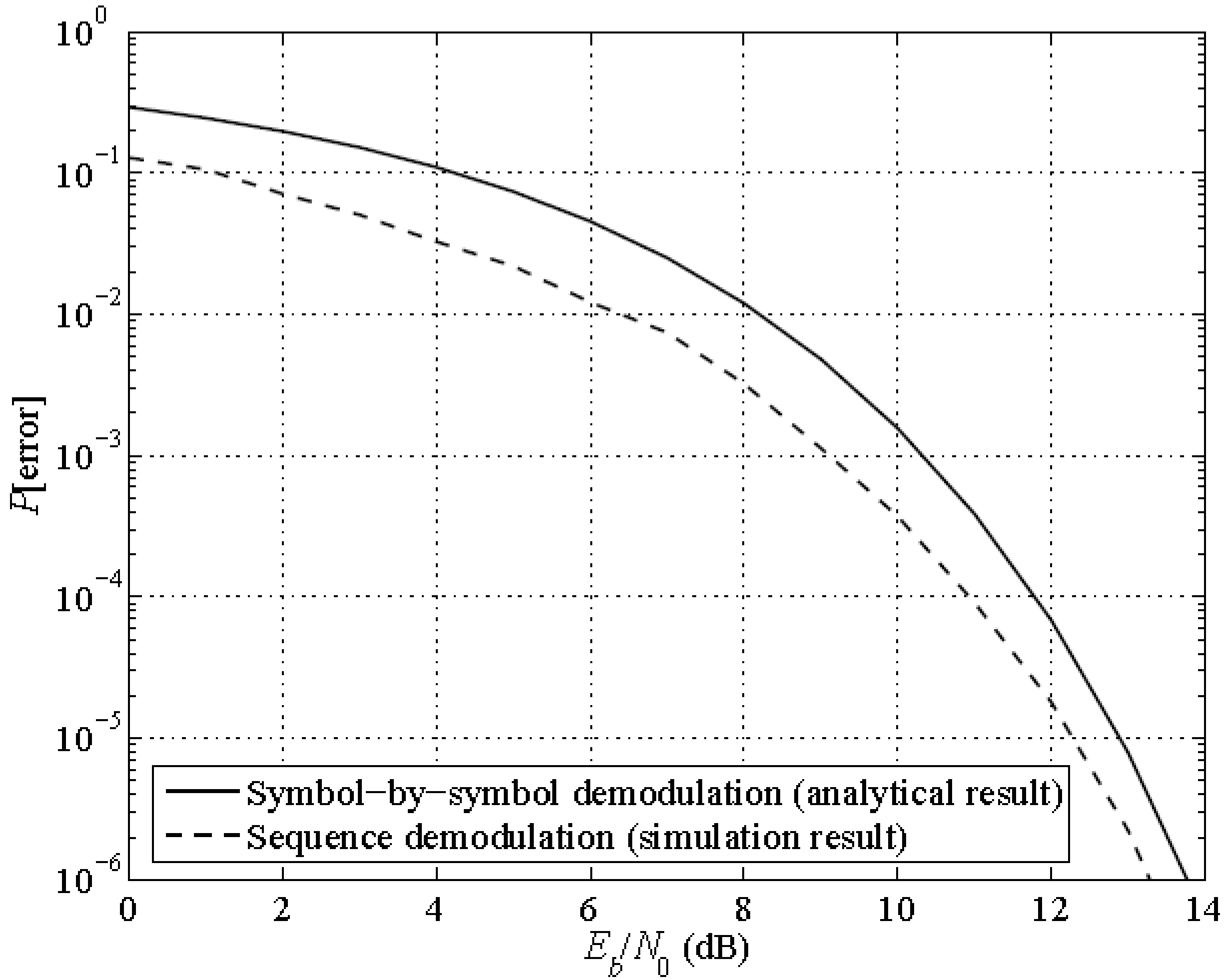
BIRZEIT UNIVERSITY
ECE

➢ Viterbi Algorithm

Solution: the transmitted signal with the minimum probability of error according to viterbi decoding algorithm is (**Survival Path**) 0110 ➔ ($S_3(t)$, $S_4(t)$, $S_2(t)$, $S_3(t)$)

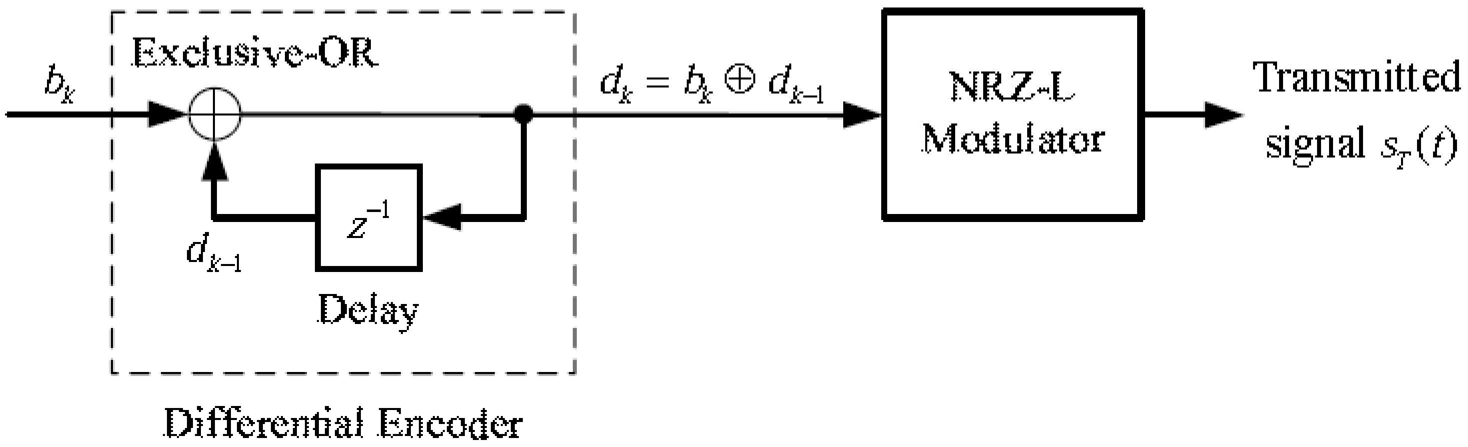


➢ Symbol-by-Symbol vs. Sequence Demodulation



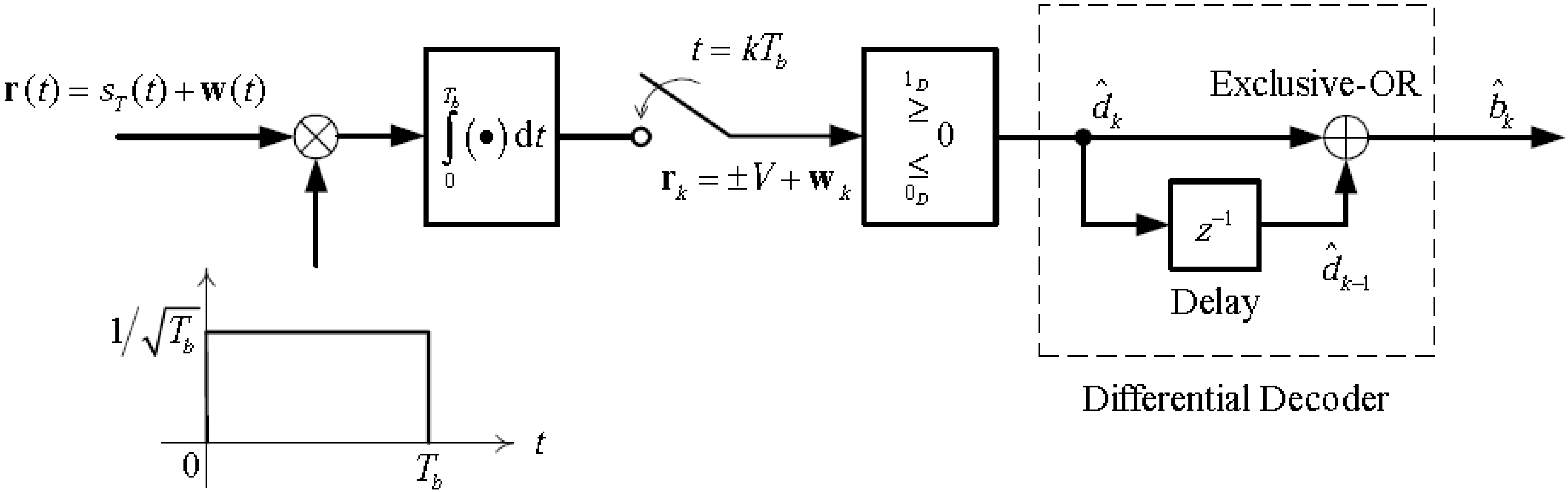
2 dB gain at the error probability of 10^{-2} and 0.5 dB at 10^{-6}.

- In differential modulation, the signal transmitted in one bit interval is relative to the one transmitted in the previous interval.
  - If the present bit is a "1", then transmit a level that is opposite to that of the previous interval.
  - If the present bit is a "0", then stay at the same level.



- If $b_k = 1$ then $d_k = b_k \oplus d_{k-1} = 1 \oplus d_{k-1} = \overline{d_{k-1}}$ implying a level change and if $b_k = 0$ then $d_k = b_k \oplus d_{k-1} = 0 \oplus d_{k-1} = d_{k-1}$ which means no level change.

Baseband data transmission - Differential NRZ-L Modulation



- First determine d_k and call this estimate $\hat{d}_k$.
- To recover b_k, note that $d_k \oplus d_{k-1} = b_k \oplus d_{k-1} \oplus d_{k-1} = b_k$
- The receiver uses $\hat{d}_{k-1}$ instead of d_{k-1}.
- If $\hat{d}_k$ is in error, there will be two errors in the sequence $\{\hat{b}_k\}$.