Lectures 6&7 Modulation

Eytan Modiano

AA Dept.

Modulation

- Representing digital signals as analog waveforms
- Baseband signals
  - Signals whose frequency components are concentrated around zero
- Passband signals
  - Signals whose frequency components are centered at some frequency fc away from zero
- Baseband signals can be converted to passband signals through modulation
  - Multiplication by a sinusoid with frequency fc

Baseband signals

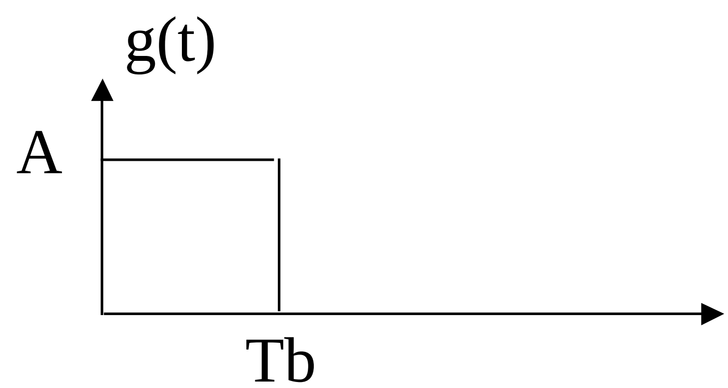
- The simplest signaling scheme is pulse amplitude modulation (PAM)
  - With binary PAM a pulse of amplitude A is used to represent a “1” and a pulse with amplitude -A to represent a “0”

- The simplest pulse is a rectangular pulse, but in practice other type of pulses are used
  - For our discussion we will generally assume a rectangular pulse

- If we let g(t) be the basic pulse shape, than with PAM we transmit g(t) to represent a “1” and -g(t) to represent a “0”

$$1 \Rightarrow S(t) = g(t)$$

$$0 \Rightarrow S(t) = -g(t)$$



M-ary PAM

- Use M signal levels, $A_1 \ldots A_M$
  - Each level can be used to represent $Log_2(M)$ bits

- E.g., M = 4 => $A_1 = -3$, $A_2 = -1$, $A_3 = 1$, $A_4 = 3$
  - $S_i(t) = A_i\, g(t)$

- Mapping of bits to signals

Si	b1b2
S_1	00
S_2	01
S_3	11
S_4	10

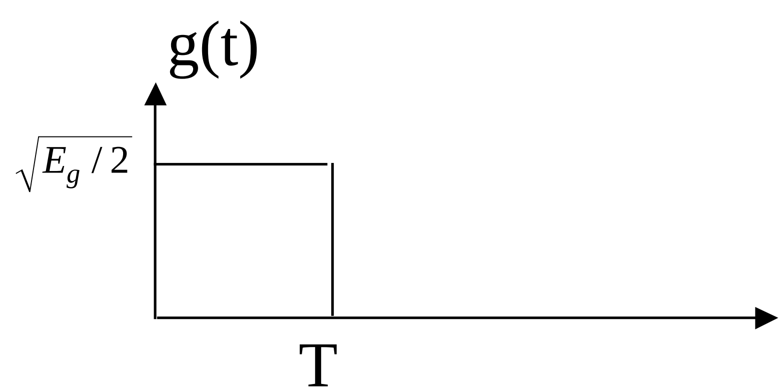
Signal Energy

$$E_m = \int_0^T (S_m(t))^2 dt = (A_m)^2 \int_0^T (g_t)^2 dt = (A_m)^2 E_g$$

- **The signal energy depends on the amplitude**
- **E_g is the energy of the signal pulse g(t)**
- **For rectangular pulse with energy E_g =>**

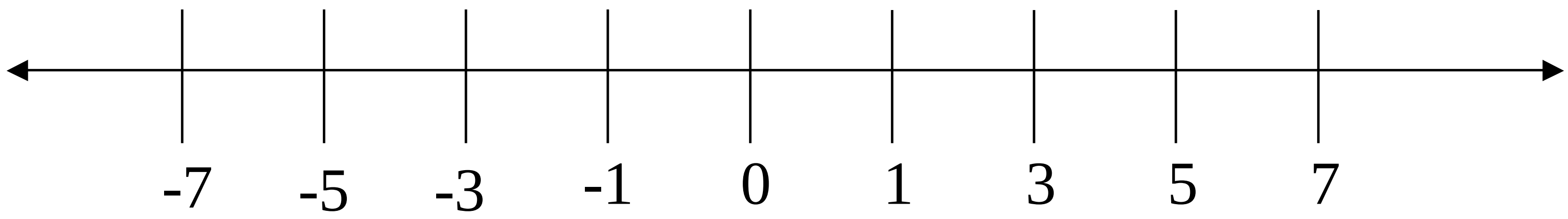
$$E_g = \int_0^T A^2 dt = TA^2 \Rightarrow A = \sqrt{E_g / 2}$$

$$g(t) = \begin{cases} \sqrt{E_g / 2} & 0 \le t \le T \\ 0 & otherwise \end{cases}$$



Symmetric PAM

- **Signal amplitudes are equally distant and symmetric about zero**



A_m = (2m-1-M), m=1…M

$$E_{ave} = \frac{E_g}{M}\sum_{m=1}^{M}(2m-1-M)^2 = E_g(M^2-1)/3$$

Gray Coding

- **Mechanism for assigning bits to symbols so that the number of bit errors is minimized**
  - **Most likely symbol errors are between adjacent levels**
  - **Want to MAP bits to symbols so that the number of bits that differ between adjacent levels is mimimized**

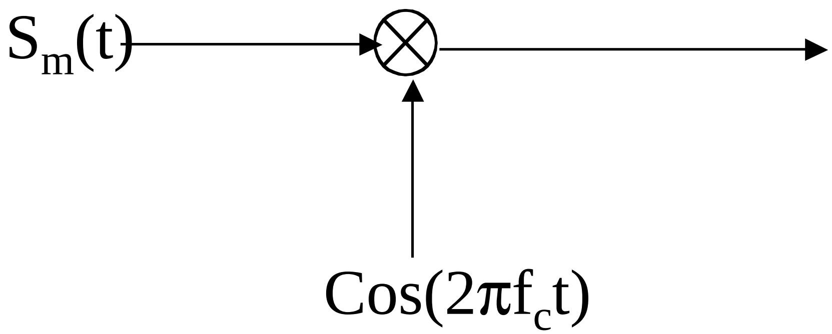
- **Gray coding achieves 1 bit difference between adjacent levels**

- **Example M= 8 (can be generalized)**

A_1	000
A_2	001
A_3	011
A_4	010
A_5	110
A_6	111
A_7	101
A_8	100

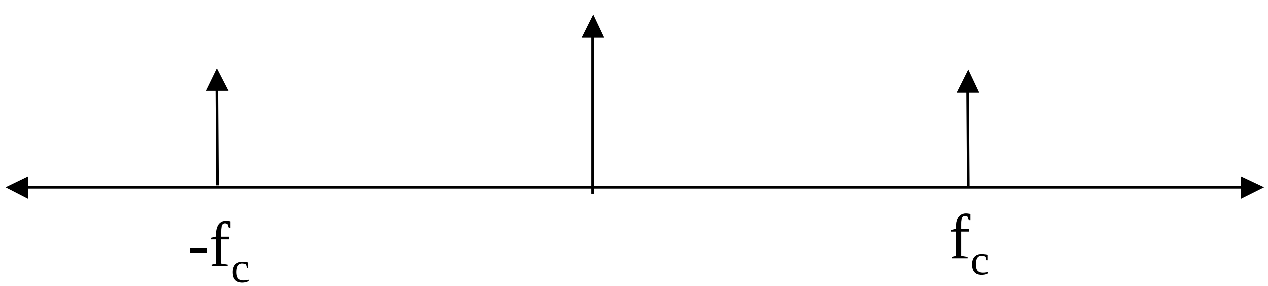
Bandpass signals

- **To transmit a baseband signal S(t) through a bandpass channel at some center frequency f_c, we multiply S(t) by a sinusoid with that frequency**



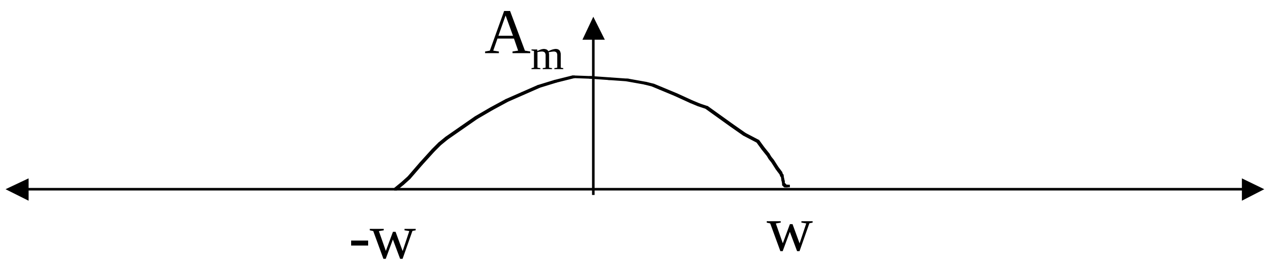
$$U_m(t) = S_m(t) Cos(2\pi f_c t)$$
$$= A_m g(t)\ Cos(2\pi f_c t)$$

$$F[Cos(2\pi f_c t)] = (\delta(f-f_c)+\delta(f+f_c))/2$$

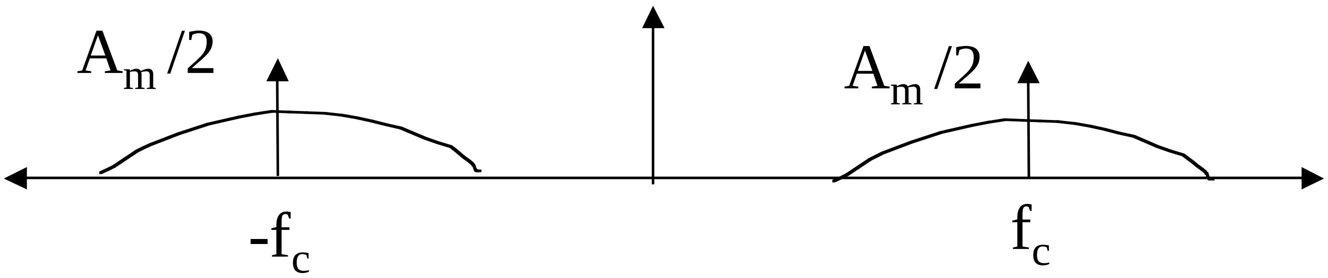


Passband signals, cont.

$F[A_m g(t)]$ = depends on g()



$F[A_m g(t) \, Cos(2\pi f_c t)]$



Recall: Multiplication in time = convolution in frequency

Energy content of modulated signals

$$E_m = \int_{-\infty}^{\infty} U_m^2(t)dt = \int_{-\infty}^{\infty} A_m^2 g^2(t) Cos^2(2\pi f_c t)dt$$

$$Cos^2(\alpha) = \frac{1+\cos(2\alpha)}{2}$$

$$E_m = \frac{A_m^2}{2}\int_{-\infty}^{\infty} g^2(t) + \frac{A_m^2}{2}\int_{-\infty}^{\infty} g^2(t) Cos^2(4\pi f_c t)dt$$

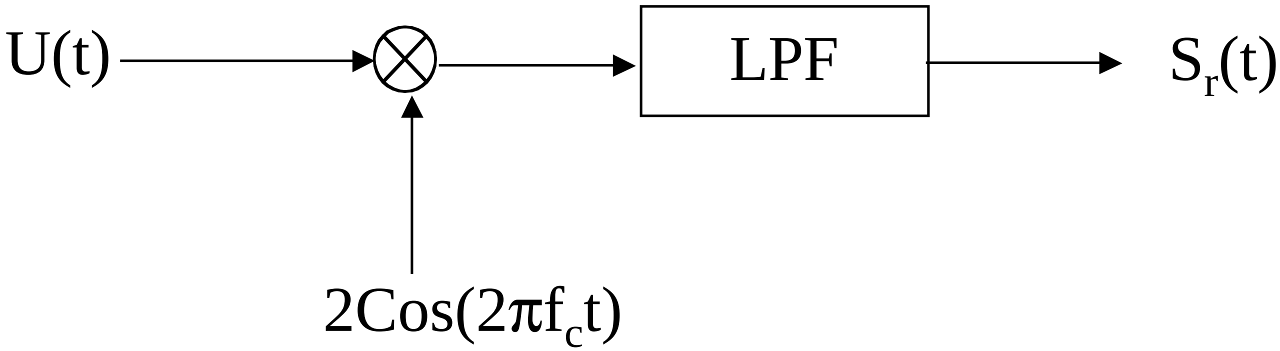
$$E_m = \frac{A_m^2}{2}E_g + \approx 0$$

- **The cosine part is fast varying and integrates to 0**
- **Modulated signal has 1/2 the energy as the baseband signal**

Demodulation

- **How do we recover the baseband signal?**

$$U_m(t) = S_m(t)\mathrm{Cos}(2\pi f_c t)$$
$$= A_m g(t)\ \mathrm{Cos}(2\pi f_c t)$$

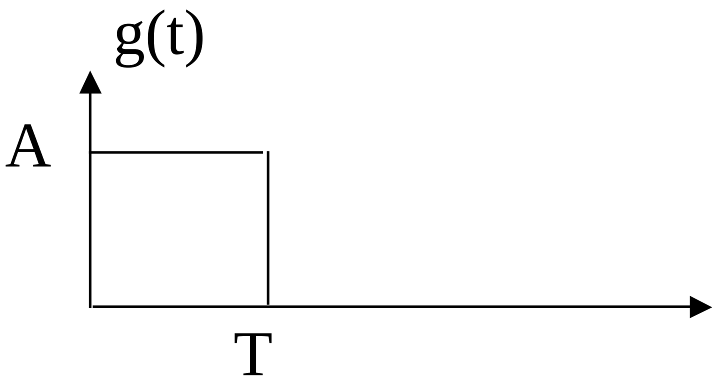


$$U(t)2\mathrm{Cos}(2\pi f_c t) = 2S(t)\mathrm{Cos}^2(2\pi f_c t) = S(t) + S(t)\mathrm{Cos}(4\pi f_c t)$$

The high frequency component is rejected by the LPF and we are left with S(t).

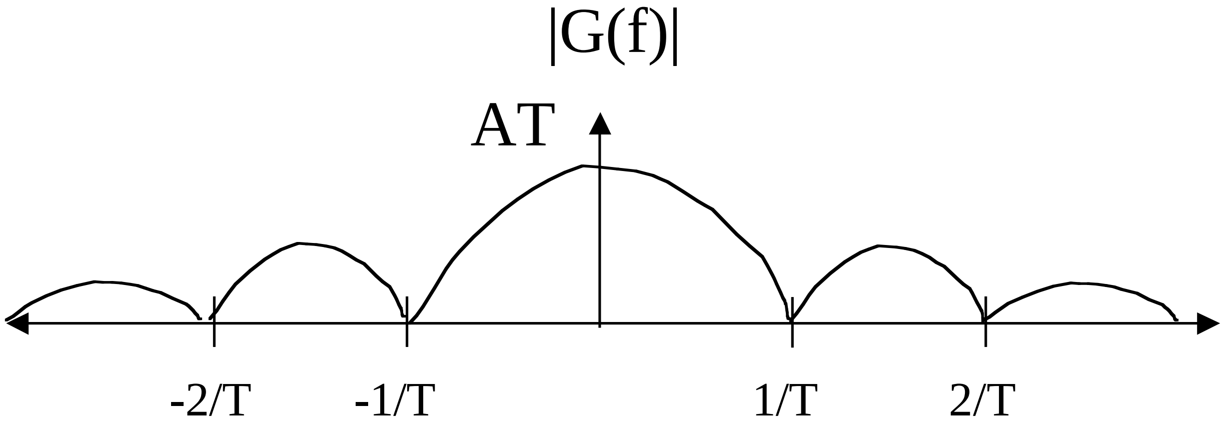
Bandwidth occupancy

G(f) = F[g(t)]



$$G(f) = \int_{-\infty}^{\infty} g(t)e^{-j2\pi ft}dt = \int_0^T Ae^{-j2\pi ft}dt$$

$$G(f) = (AT)Sinc(\pi fT)e^{-j\pi fT}$$



- **First “null” bandwidth = 2(1/T) = 2/T**

Bandwidth efficiency

- Rs = symbol rate = 1/T
  - $Log_2(M)$ bits per symbol => Rb = bit rate = $log_2(M)/T$ bits per second
- BW = 2/T = 2Rs
  - Bandwidth efficiency = Rb/BW = $log_2(M)/T * (T/2) = log_2(M)/2$ BPS/Hz

- Example:
  - M=2 => bandwidth efficiency = 1/2
  - M=4 => bandwidth efficiency = 1
  - M=8 => bandwidth efficiency = 3/2

- Increased BW efficiency with increasing M
- However, as M increase we are more prone to errors as symbols are closer together (for a given energy level)
  - Need to increase symbol energy level in order to overcome errors
  - Tradeoff between BW efficiency and energy efficiency

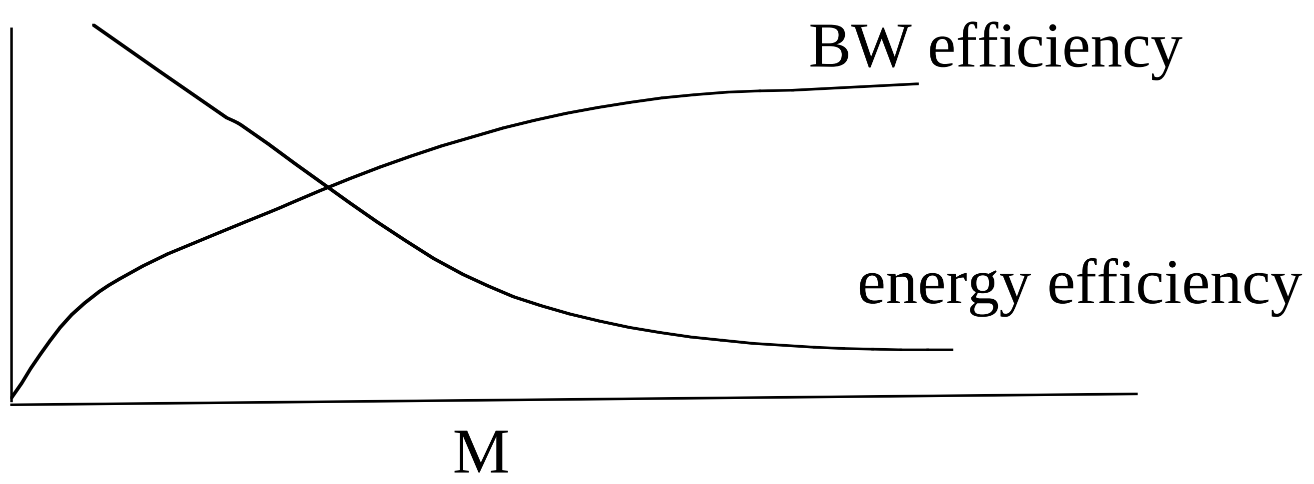
Energy utilization

$$E_{ave} = \frac{E_g}{M}\sum_{m=1}^{M}(2m-1-M)^2 = E_g(M^2-1)/3, \ E_g = \text{basic pulse energy}$$

After modulation $E_u = \frac{E_s}{2} = E_g(M^2-1)/6$

$E_b =$ average energy per bit $= \dfrac{(M^2-1)}{6Log_2(M)}E_g$

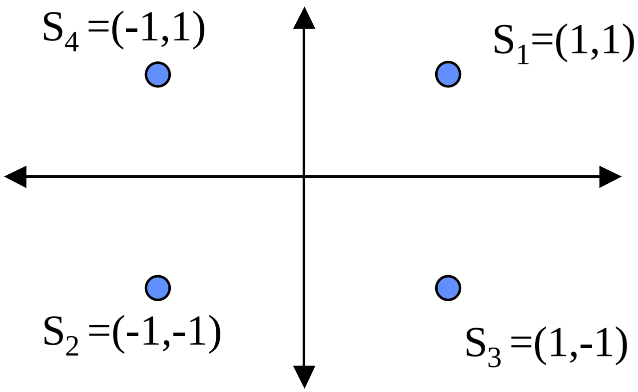
- **Average energy per bit increases as M increases**



Two-dimensional signals

- $S_i = (S_{i1}, S_{i2})$

- Set of signal points is called a constellation



- 2-D constellations are commonly used

- Large constellations can be used to transmit many bits per symbol
  - More bandwidth efficient
  - More error prone
- The "shape" of the constellation can be used to minimize error probability by keeping symbols as far apart as possible

- Common constellations
  - QAM: Quadrature Amplitude Modulation
    PAM in two dimensions
  - PSK: Phase Shift Keying
    Special constellation where all symbols have equal power

Symmetric M-QAM

$$S_m = (A_m^x, A_m^y),\ A_m^x, A_m^y \in \left\{+/-1, +/-3, \ldots, +/-(\sqrt{M}-1)\right\}$$

M is the total number of signal points (symbols)

$\sqrt{M}$ signal levels on each axis

Constellation is symmetric

$\Rightarrow \mathrm{M} = \mathrm{K}^2,$ for some K

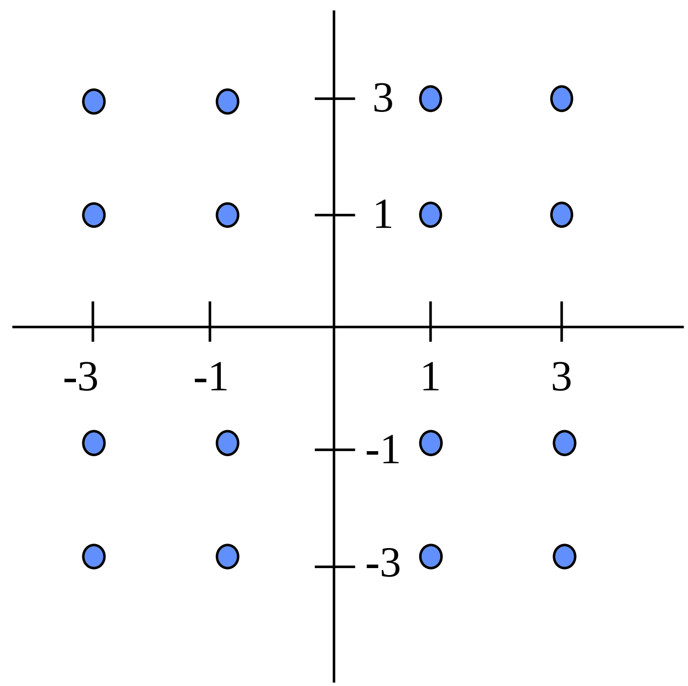
Signal levels on each axis are

the same as for PAM

$$E.g., 4-QAM \Rightarrow A_m^x, A_m^y \in \{+/-1\}$$

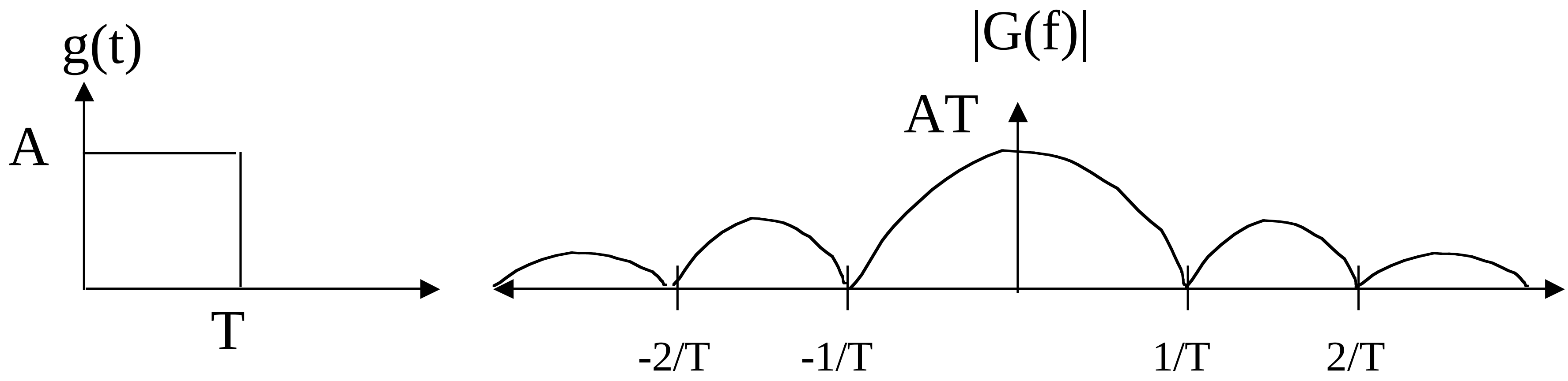
$$16-QAM \Rightarrow A_m^x, A_m^y \in \{+/-1, +/-3\}$$

16-QAM



Bandwidth occupancy of QAM

- When using a rectangular pulse, the Fourier transform is a Sinc



- First null BW is still 2/T
  - K = $Log_2(M)$ bits per symbol
  - Rb = $Log_2(M)/T$
  - Bandwidth Efficiency = Rb/BW = $Log_2(M)/2$
  - => “Same as for PAM”

Energy efficiency

$$E_{sm} = [(A_m^x)^2 + (A_m^y)^2]E_g$$

$$E[(A_m^x)^2] = E[(A_m^y)^2] = \frac{K^2 - 1}{3} = \frac{M-1}{3}, \quad K = \sqrt{M}$$

$$\overline{E}_s = \frac{2(M-1)}{3} E_g$$

$$Transmitted\ energy = \frac{\overline{E}_s}{2} = \frac{(M-1)}{3} E_g$$

$$E_b(QAM) = Energy/bit = \frac{(M-1)}{3Log_2(M)} E_g$$

- **Compare to PAM: Eb increases with M, but not nearly as fast as PAM**

$$E_b(PAM) = \frac{(M^2 - 1)}{6Log_2(M)} E_g$$

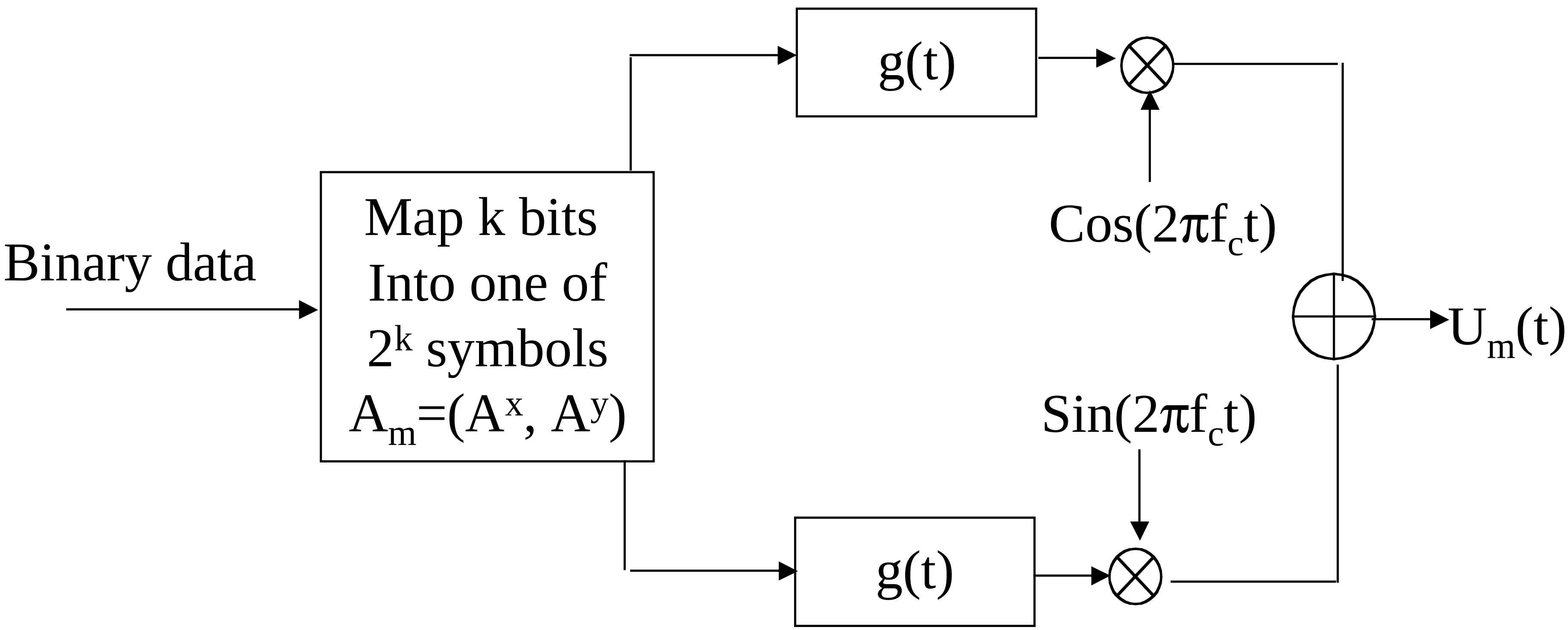
Bandpass QAM

- **Modulate the two dimensional signal by multiplication by orthogonal carriers (sinusoids): Sin & Cos**
  - **This is accomplished by multiplying the A^x component by Cos and the A^y component by sin**
  - **Typically, people do not refer to these components as x,y but rather A^c or A^s for cos and sin or sometimes as A^Q, and A^I for quadrature or in-phase components**

- **The transmitted signal, corresponding to the m^th symbol is:**

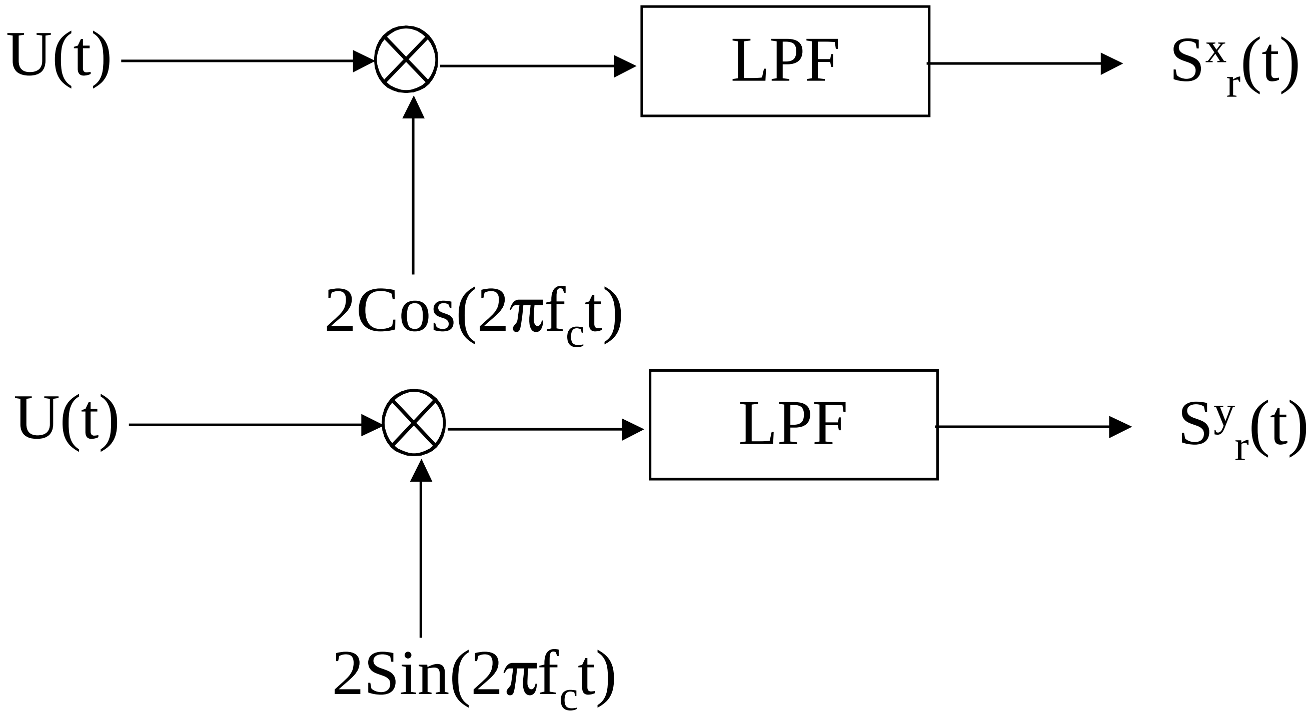
$$U_m(t) = A_m^x g(t) Cos(2\pi f_c t) + A_m^y g(t) Sin(2\pi f_c t),\ m = 1...M$$

Modulator



Binary data
Map k bits
Into one of
2^k symbols
$A_m=(A^x, A^y)$
g(t)
g(t)
$Cos(2\pi f_c t)$
$Sin(2\pi f_c t)$
$U_m(t)$

Demodulation: Recovering the baseband signals



- Over a symbol duration, $Sin(2\pi f_c t)$ and $Cos(2\pi f_c t)$ are orthogonal
  - As long as the symbol duration is an integer number of cycles of the carrier wave (fc = n/T) for some n
- When multiplied by a sin, the cos component of U(t) disappears and similarly the sin component disappears when multiplied by cos

Demodulation, cont.

$$U(t)2Cos(2\pi f_c t) = 2A^x g(t)Cos^2(2\pi f_c t) + 2A^y g(t)\cos(2\pi f_c t)\sin\left(2\pi f_c t\right)$$

$$Cos^2(\alpha) = \frac{1+\cos(2\alpha)}{2}$$

$$=> U(t)2Cos(2\pi f_c t) = S^x(t) + S^x(t)\cos(4\pi f_c t) \approx S^x(t) = A^x g(t)$$

Similarly,

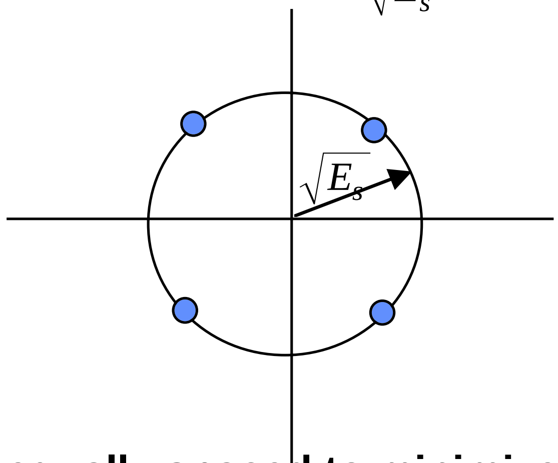
$$U(t)2Sin(2\pi f_c t) = 2A^x g(t)Cos(2\pi f_c t)Sin\left(2\pi f_c t\right) + 2A^y g(t)\sin^2\left(2\pi f_c t\right)$$

$$Sin^2(\alpha) = \frac{1-\cos(2\alpha)}{2}$$

$$=> U(t)2Sin(2\pi f_c t) = S^y(t) - S^y(t)\cos(4\pi f_c t) \approx S^y(t) = A^y g(t)$$

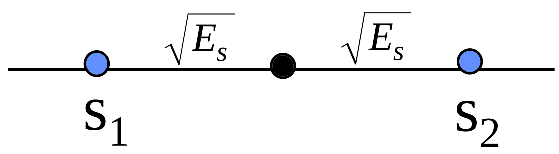
Phase Shift Keying (PSK)

- Two Dimensional signals where all symbols have equal energy levels
  - I.e., they lie on a circle or radius $\sqrt{E_s}$



- Symbols can be equally spaced to minimize likelihood of errors

- E.g., Binary PSK



- 4-PSK (above) same as 4-QAM

M-PSK

$$A_i^x = Cos(2\pi i / M),\ A_i^y = Sin(2\pi i / M),\ \ m = 0, \ldots, M-1$$

$$U_m(t) = g(t) A_m^x Cos(2\pi f_c t) - g(t) A_m^y Sin(2\pi f_c t)$$

$$Notice:\ Cos(\alpha)Cos(\beta) = \frac{Cos(\alpha - \beta) + Cos(\alpha + \beta)}{2}$$

$$Sin(\alpha)Sin(\beta) = \frac{Cos(\alpha - \beta) - Cos(\alpha + \beta)}{2}$$

$$Hence, \quad U_m(t) = g(t) Cos(2\pi f_c t + 2\pi m / M)$$

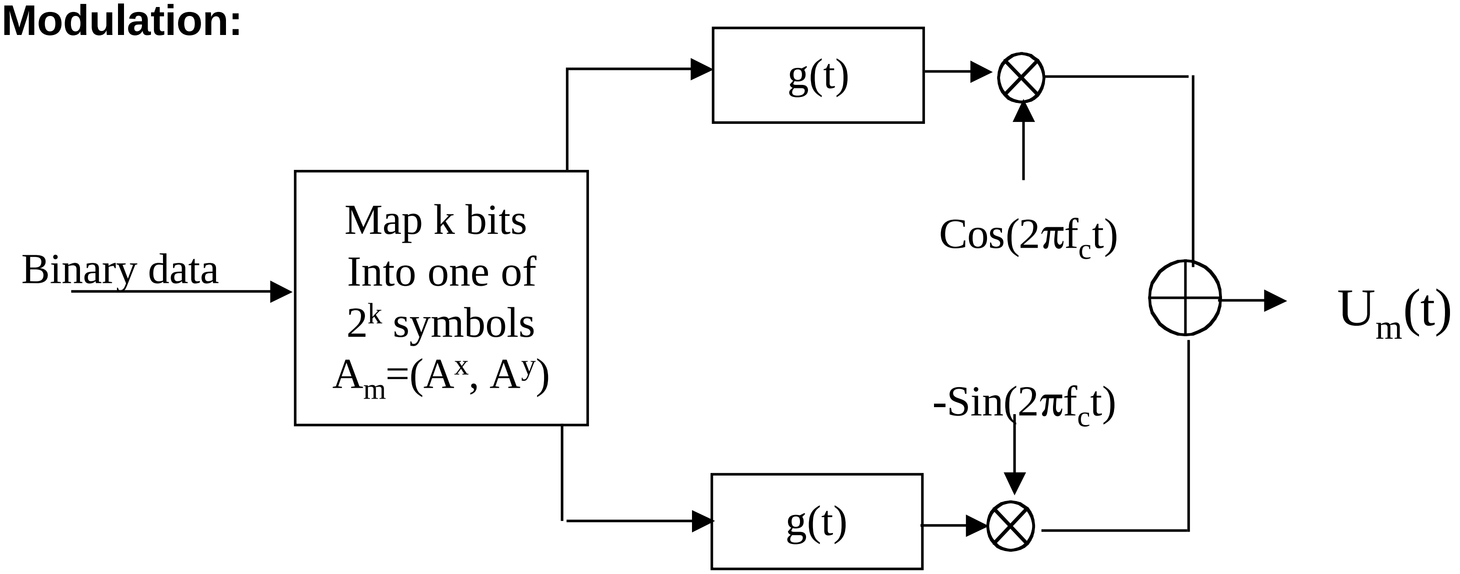
$$\phi_m = 2\pi m / M = phases\ shift\ of\ m^{th}\ symbol$$

$$U_m(t) = g(t) Cos(2\pi f_c t + \phi_m),\ m = 0 \ldots M-1$$

M-PSK Summary

- **Constellation of M Phase shifted symbols**
  - **All have equal energy levels**
  - **K = Log_2(M) bits per symbol**

- **Modulation:**



- **Notice that for PSK we subtract the sin component from the cos component**
  - **For convenience of notation only. If we added, the phase shift would have been negative but the end result is the same**

- **Demodulation is the same as for QAM**