Fundamentos de Telecomunicações 1 – MIEEC / FEUP

Trabalho sobre

DETECÇÃO DE SINAIS EM CANAIS AWGN

Conteúdo

1 OBJECTIVO

Neste trabalho será estudado o comportamento de vários tipos de detectores usados em receptores digitais cujo objectivo é a identificação de sinais imersos em ruído gaussiano branco aditivo. O estudo será acompanhado pela medição das probabilidades de erro e pela visualização dos sinais em diagramas de olho.

2 INTRODUÇÃO

Num sistema de comunicação binário, os dados binários consistindo numa sequência de 0s e 1s são transmitidos por meio de duas ondas de sinal, digamos, $s_0(t)$ e $s_1(t)$. Suponha que a taxa de dados seja especificada como R bits por segundo. Então, cada bit é mapeado numa das ondas de sinal correspondente de acordo com a regra:

$$0 \rightarrow s_0(t), 0 \leq t \leq T$$

$$1 \rightarrow s_1(t), 0 \leq t \leq T$$

onde T é o tempo de duração de bit. Assume-se que o canal através do qual o sinal é transmitido corrompe o sinal pela adição de ruído, denotado como $w(t)$, que é uma amostra de um processo gaussiano branco com uma densidade espectral de potência, $N_0/2$ Watt/Hertz. Esse ruído é chamado de ruído gaussiano branco aditivo (AWG) e o canal é denominado por canal AWGN. Consequentemente, a onda de sinal recebida é expressa como $r(t) = s_i(t) + w(t)$. A função do receptor é determinar, para cada bit recebido, se o bit transmitido é um 0 ou um 1, por observação do sinal recebido no intervalo $0 \leq t \leq T$. O receptor que minimiza a probabilidade de erro é designado por receptor óptimo.

A detecção de sinais binários envolve dois passos:

1. reduzir a forma de onda recebida, $r(t)$, (banda-base ou passa-banda) a um número, $z(t=T)$, (Filtro linear + Amostrador).
2. comparar a amostra, $z(T)$, com um nível limiar e decidir qual terá sido a forma de onda transmitida (Decisor):

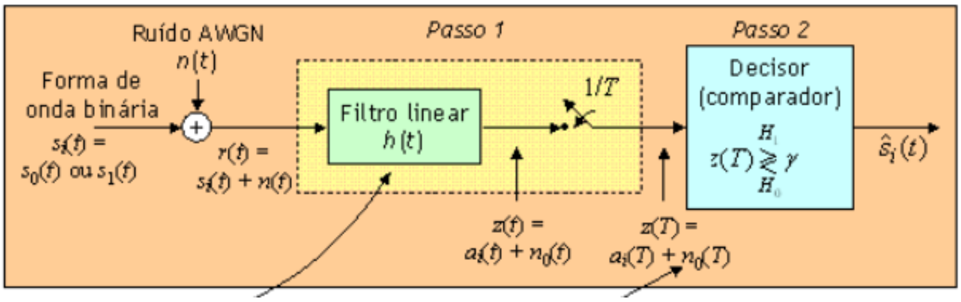

Figura 1: Diagrama de blocos do receptor.

3 TRANSMISSÃO E DETECÇÃO DE IMPULSOS BINÁRIOS COM RUÍDO AWGN

3.1 DETECÇÃO POR AMOSTRAGEM

3.1.1 INTRODUÇÃO TEÓRICA

Pretende-se transmitir uma sequência digital representada por impulsos em banda-base genericamente designados por $g(t), 0 \leq t \leq T$ através de um canal que apenas introduz ruído gaussiano branco $w(t)$ – o chamado canal AWGN (*Additive White Gaussian Noise*). A taxa de símbolos, ou número de símbolos transmitidos por segundo, é, pois, $1/T$. No receptor, o modo mais simples, mas não o mais eficiente, de recuperar a sequência binária é através da amostragem, de T em T segundos, da forma de onda, servindo os valores das amostras $x(T)$ para determinar quais os símbolos binários que lhes correspondem, consoante esses valores estão acima ou abaixo de um determinado limiar de decisão (veja-se a Figura 2).

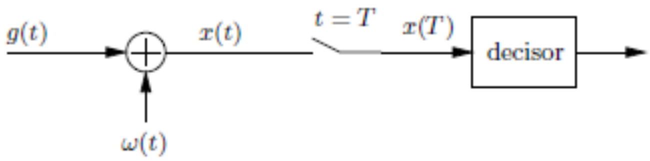

Figura 2: Sinal, ruído, amostrador e decisor.

A probabilidade de o decisor cometer um erro – isto é, trocar um bit pelo outro – depende da diferença das amplitudes dos impulsos recebidos, d, e da potência do ruído, σ^2. Em concreto, é dada pela expressão

$$P_b = Q\left(\frac{d}{2\sigma}\right),$$

em que $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-\lambda^2} d\lambda$, que em Matlab é `qfunc(x)` (sendo `qfuncinv(x)` a respectiva função inversa).

3.1.2 DESCRIÇÃO DO PROBLEMA

Neste trabalho vai ser analisado o comportamento de um detector baseado num simples amostrador. O sistema a simular está representado na figura seguinte:

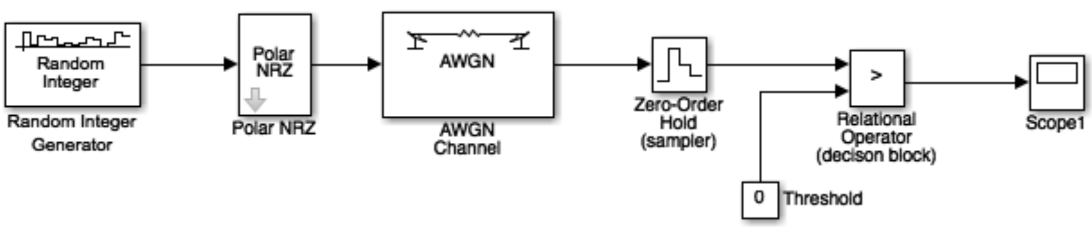

Figura 3: Diagrama do detector por amostragem.

- Random Integer: gerador de números aleatórios inteiros com distribuição uniforme entre 0 e 1.
- Polar NRZ: conversão dos bits gerados para a forma polar: ± 1
- AWGN: adição de ruído branco Gaussiano
- Zero-Order Hold: amostragem do sinal
- Decision block: circuito decisor por comparação do sinal recebido com o nível de decisão

Parâmetros de simulação:

- Taxa de transmissão bit/s
- Nº de amostras/bit = 100 (para a simulação numérica)
- Tempo de bit $T = 1/R = 1$ ms
- Tempo total de simulação 1000 ms (equivalente a simular o envio de 1000 bits)
- Nº total de amostras da sequência (1000 bits) x 100 amostras/bit

Tarefas:

- Simule o sistema durante 1000 ms e obtenha o diagrama de olho para uma potência de ruído $\sigma^2 = 0.02$.
- Do diagrama de olho identifique o melhor instante para fazer a amostragem do sinal.
- Complete o código MATLAB incluindo a amostragem do circuito decisor, com base no resultado do ponto anterior.
- Simule o sistema para vários valores de potência de ruído.
- Calcule a taxa de erros por simulação (probabilidade de erro estimada) e a probabilidade de erro teórica.
- Complete a tabela seguinte:

σ^2	E_b/N_0 (dB)	P_b (teórica)	P_b (estimada)
0,18			
0,36			
0,72			
1,44			

Implementação em MATLAB:

- Definir os parâmetros de simulação (como exemplo consideraremos um tempo de simulação de 1000 ms ou 1000 bits transmitidos)
- Gerar a sequência de bits aleatório de uma só vez, que será guardada num vector (1x1000): bitTx (bits transmitidos)
- Converter bits para a forma polar NRZ (± 1)
- Para cada bit em bitTx temos que gerar 100 amostras, obtendo-se o sinal para simulação: vector g
- Adicionar ruído ao sinal: vector x = g + ruído, a partir do conhecimento da potência (variância, σ^2) do ruído
- Observar o diagrama de olho obtido e identificar os instantes ideais de amostragem
- Calcular o vector iTs correspondente aos instantes de amostragem
- Obter as amostras do sinal x, nos instantes de amostragem xs = x(iTs)

- Definir o nível de decisão: para facilidade de computação geraremos um vector de zeros com um comprimento igual ao número de bits transmitido/recebido
- Comparar o nível de sinal recebido com o nível de decisão, obtendo-se os bits recebidos bitsRx
- Comparar a sequência de bits original com a sequência recebida e contar os erros
- Reportar o valor da taxa de erros ou probabilidade de erro obtida por simulação
- Calcular o valor teórico da probabilidade de erro
- Calcular o valor de E_b/N_0

Funções MATLAB e detalhes de implementação numérica

- Geração de bits aleatórios e conversão de bits para forma polar: obtenha os valores de n1 e n2, tal que bitTx seja -1 ou 1

```
bitTx = ceil(n1+n2*rand(1,Nbits));


rand Uniformly distributed pseudorandom numbers.
   % R = rand(1,N) returns an 1-by-N matrix containing pseudorandom %
   values drawn
   % from the standard uniform distribution on the open
   % interval(0,1)

ceil   Round towards plus infinity.
   % ceil(X) rounds the elements of X to the nearest integers towards
   % infinity.
```

- Geração do sinal x com 100 amostras por bit:
  - Usar um ciclo que para cada valor de bitTx(i), gera 100 elementos todos iguais ao valor de bitTx(i); guardar os valores no vector x
  - Alternativa: (i) gerar uma matriz de amostras de ones (1s) de dimensão (N_amostras_bit x Nbits);(ii) multiplicar cada um dos elementos do vector linha bitTx, pela correspondente coluna da matriz; (iii) reshape matriz para vector linha

```
bsxfun  Binary Singleton Expansion Function
   % C = bsxfun(FUNC,A,B) applies the element-by-element binary
   % operation specified by the function handle FUNC to arrays A
   % and B, with singleton expansion enabled.
   % FUNC can be one of the following built-in functions:

               @plus          Plus
               @minus         Minus
               @times         Array multiply
               @rdivide       Right array divide

reshape Reshape array.
    % reshape(X,M,N) or reshape(X,[M,N]) returns the M-by-N matrix
    % whose elements are taken columnwise from X
```

- Geração de ruído

```
random Generate random arrays from a specified distribution
```

```
% R = random(NAME,A,B,M,N)
% returns an M-by-N array of random numbers chosen from the
% two-parameter probability distribution specified by NAME with
% parameter values A and B.

% Normal ou Gaussian distribution: NAME = 'Normal'
% A = average, B = standard deviation
```

- Cálculo dos instantes de amostragem
  - Os instantes de amostragem têm período *T*, basta conhecer o primeiro instante de amostragem para obter os restantes.
  - Obtenha o instante de amostragem, iTs, do índice do vector x, correspondente aos instantes de amostragem: os índices dos instantes de amostragem estão espaçados de N_amostras_bit.
  - O sinal amostrado é xs = x(iTs)

```
find   Find indices of nonzero elements.

  % I = find(X) returns the linear indices corresponding to
  % the nonzero entries of the array X.  X may be a logical expression.
```

- Decisão dos bits recebidos, bitRx
  - Sendo a variável *nivel_decisão* um vector de zeros com um comprimento igual a Nbits, determine as constantes a e b, por forma que bitRx seja ± 1.

```
bitRx = a*(xs > nivel_decisao)-b;
```

- Cálculo de E_b/N_0: relação entre a densidade espectral de ruído e a potência de ruído .
  - Na simulação a máxima frequência do sinal que é possível resolver é $f_s/2$, onde f_s é a frequência de amostragem, o que é equivalente a usar um filtro ideal de largura de banda $B = f_s, [-f_s/2\,, f_s/2]$, . Se à entrada desse filtro tivermos ruído branco com uma densidade espectral $\sigma^2 = N_0/2$, à saída temos uma potência de ruído dada por $\sigma^2 = N_0/2 \cdot f_s$ou de forma equivalente $N_0 = 2\sigma^2/f_s$.
  - E_b é a energia do impulso, $E_b = \int_0^{T_b} g^2(t)dt$

3.2 DETECÇÃO COM FILTRO ADAPTADO: RESPOSTA DO FILTRO

3.2.1 INTRODUÇÃO TEÓRICA

O modo de detecção de impulsos do problema anterior (Figura 2) não é o melhor. De facto, se se pretender minimizar a probabilidade de erro ou, dito de outra forma, maximizar a relação sinal-ruído no instante de decisão, dever-se-á anteceder o amostrador de um filtro adequado, o denominado *filtro adaptado* (*matched filter*), como se mostra na Figura 4.

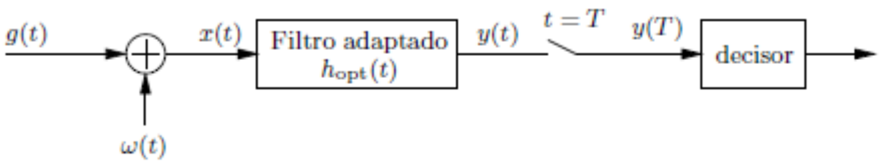

Figura 4: O filtro adaptado.

"Adaptado" ao impulso $g(t)$, este filtro óptimo tem a resposta impulsional

$$h_{opt}(t) = g(T - t), 0 \leq t \leq T,$$

A forma de onda de saída do filtro adaptado, *y*(*t*), em resposta ao sinal *x*(*t*), é dada pelo integral de convolução

$$y(t) = \int_0^t x(\tau)h_{opt}(t - \tau)d\tau = \int_0^t x(\tau)x(T - t + \tau)d\tau.$$

Se amostrarmos a forma de onda em *t*=*T*, resulta

$$y(T) = \int_0^T x^2(t)dt = E,$$

onde *E* é a energia do sinal.

Veja-se o seguinte exemplo. Seja $g(t)$ um impulso unipolar RZ de amplitude A, com *duty-cycle* de 50% e duração T (correspondente ao bit 1, por exemplo), como o representado na Figura 5.

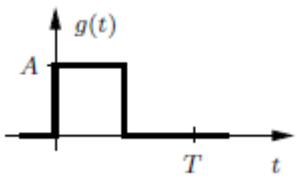

Figura 5: Impulso transmitido *g(t)*.

O filtro receptor que está adaptado a esta forma de onda tem, portanto, a resposta impulsional da Figura 6:

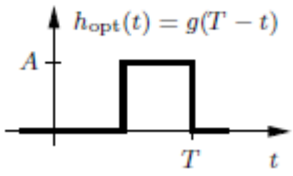

Figura 6: Resposta impulsional óptima, $h_{opt}(t)$.

A aplicação de $g(t)$ a este filtro produz a forma de onda na saída representada na Figura 7:

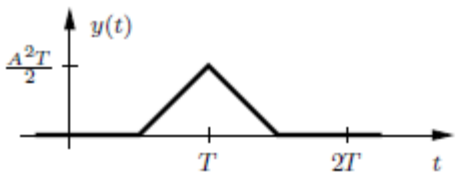

Figura 7: Resposta do filtro adaptado.

3.2.2 DESCRIÇÃO DO PROBLEMA

Considere a transmissão de impulsos polares NRZ com a seguinte forma de onda, para o impulso positivo (o impulso negativo será -$g(t)$) e calcule a resposta do filtro adaptado :

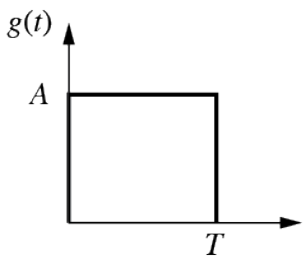

Figura 8: Resposta do filtro adaptado.

Tarefas:

- Amostre a forma de onda de sinal a uma taxa $f_s = 20/T$
- Obtenha o filtro, $h_{opt}(t)$, adaptado a estes impulsos.
- Calcule numericamente a resposta do filtro adaptado do sinal recebido $x(t)$
- Repita o problema anterior quando as amostras do sinal recebido, $x(kT_s)$, estão corrompidas por amostras de ruído AWG, $w(kT_s)$, onde $1 \leq k \leq 20$, com média zero e variância $\sigma^2 = 0.2$ e $\sigma^2 = 1$.

Questões:

- Qual o instante ideal de amostragem?
- Sem ruído adicionado qual o valor da forma de onda no instante de amostragem, para os símbolos "0" e "1"?
- Admitindo que o ruído nos níveis "0" e "1" são iguais qual o valor óptimo para o nível de decisão?
- Se o desvio padrão do ruído no nível "1" for 2x superior ao desvio padrão do ruído no nível "0", $\sigma_1 = 2\sigma_0$, calcule o valor óptimo do nível de decisão, para o caso em que $\sigma_0 = 0.25$.

Funções MATLAB:

```
conv Convolution and polynomial multiplication.
   % C = conv(A, B)
   % convolves vectors A and B.  The resulting vector is
   % length MAX([LENGTH(A)+LENGTH(B)-1,LENGTH(A),LENGTH(B)]).
   % If A and B are vectors of polynomial coefficients, convolving them
   % is equivalent to multiplying the two polynomials.
```

3.3 DETECÇÃO COM FILTRO ADAPTADO: SIMULAÇÃO MONTE CARLO

3.3.1 INTRODUÇÃO TEÓRICA

O instante de amostragem é o instante de tempo em que se toma uma decisão sobre qual o bit, 0 ou 1, que terá sido transmitido. A escolha correcta desse instante conduz ao valor máximo possível da relação sinal-ruído, na presença de ruído gaussiano branco. Neste exemplo o instante de amostragem deve ser tomado em $t = T$, pois é nesse momento que a saída do filtro adaptado atinge o valor máximo. Esse valor corresponde à energia do bit, E_b. No instante $t = T$, a saída do filtro é dada por

$$y(t) = E_b + n(t)$$

onde, E_b é a energia do sinal e n é a componente aditiva do ruído. Usaremos este resultado para simular o detector com filtro adaptado, gerando um número elevado de eventos (simulação Monte Carlo), sequência de bits, aos quais é adicionado uma variável aleatória que obedece à distribuição de probabilidade Gaussiana.

A probabilidade de bit errado associada ao detector óptimo no instante de amostragem T, para impulsos polares com qualquer forma de onda, vale

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right),$$

em que N_0 é a densidade espectral de potência unilateral do ruído branco $w(t)$.

- Algumas relações úteis:
  - $SNR = \left(\frac{S}{N}\right) = \frac{2E_b}{N_0}$
  - $\sigma^2 = E[{n_0}^2] = \frac{N_0 E_b}{2}$

3.3.2 DESCRIÇÃO DO PROBLEMA

Simule a transmissão de impulsos polares NRZ e receptor óptimo constituído por filtro adaptado, cujo diagrama de blocos se encontra representado na figura seguinte:

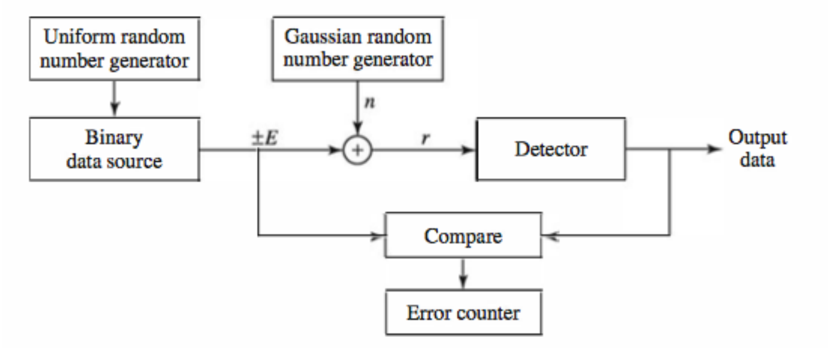

Figura 9: Diagrama de blocos da simulação Monte Carlo.

- Faça a simulação para 10000 bits e para diferentes níveis de ruído, definido pela razão sinal-ruído.
- Dado a SNR, obtenha o valor da potência de ruído, σ^2, a adicionar ao sinal, sabendo que os impulsos transmitidos são impulsos NRZ polares de largura *T* e amplitude *A* , de tal forma que a resposta do filtro a cada dum impulsos é *E* e -*E*, aos quais é adicionado ruído (variável aleatória Gaussiana com média zero e variância, σ^2).
- Calcule a probabilidade de erro estimada e calcule os correspondentes valores teóricos.
- Faça o gráfico da probabilidade de erro, P_e, em função de SNR em dB.
- Compare a probabilidade de erro obtida com filtro adaptado do detector por amostragem sem filtro. O que pode concluir?

3.4 DETECÇÃO POR FILTRO "INTEGRATE & DUMP"

3.4.1 INTRODUÇÃO TEÓRICA

Uma forma alternativa de efectuar a detecção óptima é recorrer ao circuito correlacionador, cuja estrutura está representada na Figura 10.

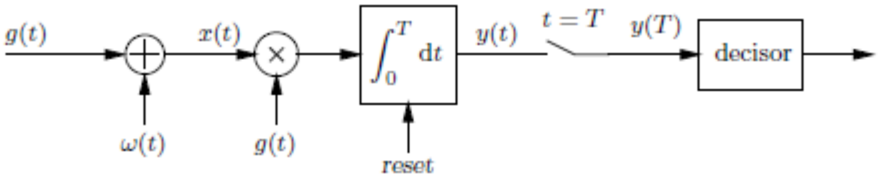

Figura 10: O correlacionador.

No instante de amostragem $t = T$ o valor da saída do correlacionador tem o mesmo valor que a saída do filtro adaptado. Após o instante de amostragem é feito o *reset* da saída do integrador de modo que a integração em cada período de símbolo comece sempre a partir de zero.

No caso dos impulsos transmitidos serem rectangulares (situação mais frequente) o correlacionador anterior pode ser simplificado para o circuito que habitualmente se designa por *Integrate&Dump* (I&D), representado na Figura 11.

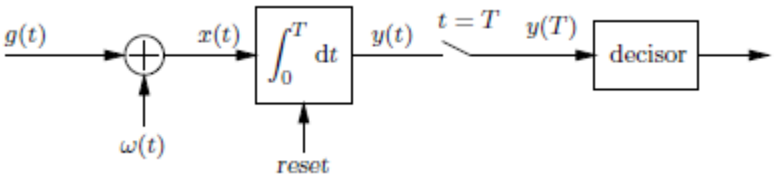

Figura 11: O filtro *Integrate & Dump*.

A resposta do filtro I&D ao impulso $g(t)$ da Figura 5 está ilustrada na Figura 12.

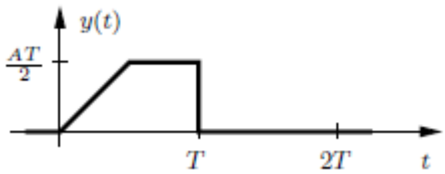

Figura 12: Resposta do filtro *Integrate & Dump* ao impulso *g(t) da* Figura 5.

Esta é a forma mais simples que um detector óptimo pode apresentar e, como tal, de mais fácil realização física.

3.4.1 DESCRIÇÃO DO PROBLEMA

Como última experiência e voltando a usar codificação polar NRZ, o filtro adaptado irá ser substituído por um circuito Integrate & Dump (I&D) e o desempenho deste será comparado com aquele. Note que, como os impulsos transmitidos são rectangulares, o filtro I&D é equivalente a um correlacionador. Neste caso a simulação será realizada no Simulink, usando o modelo fornecido "AWG_detect_ID.mdl".

- Abra o modelo *AWGN_detect_ID*, que inclui um bloco *Integrate and Dump*.
- Para um valor reduzido de ruído, $\sigma^2 = 10^{-5}$, observe as formas de onda e verifique se estão de acordo com o esperado
- Simule o sistema durante 1 s (1000 bits enviados), para os níveis de ruído indicados e registe as diversas probabilidades de erro estimadas, copiando os valores obtidos para o script MATLAB, no Cody Coursework
- Complete a tabela com os valores correspondentes da probabilidade de erro teórica
- Ajuste no bloco AWGN os níveis de potência de ruído desejados.
- Compare o desempenho deste detector com o detector com filtro adaptado

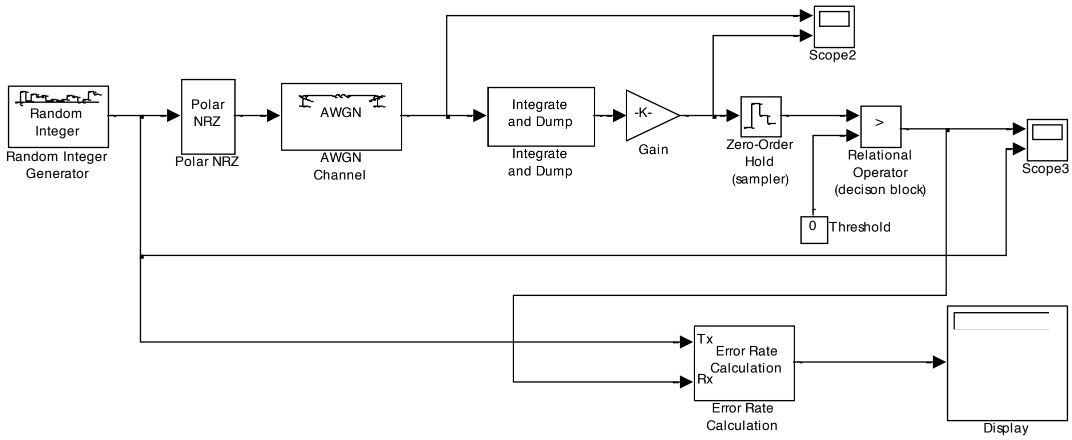

Figura 13: O modelo *AWGN_detect_ID*.

σ^2	E_b/N_0 (dB)	P_b (teórica)	P_b (estimada)
17			
35			
75			
155			