Chapter 20. Pulse Shaping and Matched Filter

학번: 22012225 이름: 손보경

|  |
| --- |
| 1.C. Answer |
| 1. ISI가 발생하지 않는 펄스의 조건 : 심벌간격에서 샘플링을 통해 임펄스가 되도록 만듦. (심벌간격으로 샘플링된 ,.. h(-2T), h(-T), h(T), h(2T) ,..가 모두 0 ) 2. t=kT 에서 k=0일때에 1이고 나머지 정수 k에 대해서는 0이 됨. 심벌구간 Ts(=1)로 주어졌으므로 ,.. -2초, -1초, 1초, 2초, 3초,… 에서 0을 지나간다. 즉 0초를 중심으로 좌우 Ts의 정수배의 지점에선 zero crossing. 심벌 간의 간섭이 발생하지 않기 때문에 ISI free하다. |

|  |
| --- |
| 1.E. Answer |
| ISI가 발생한다. 시간축에서 SRRC 펄스는 Ts의 정수배에서 0을 지나지 않기 떄문. SRRC 펄스 그 자체로는 ISI free가 아님. |

|  |
| --- |
| 2.A4. Answer |
| 심벌구간 정수배의 각각의 펄스들이 더해져서 아래와 같이 하나의 스펙트럼으로 펄스성형이 되어야한다. |

|  |
| --- |
| 2.B1. Answer |
|  |

|  |
| --- |
| 2.B4. Answer |
| (a)     1. 한눈에 파악하기 어렵다. 비트 구간의 정수배 시점에서 펄스성형된 신호의 값이 무엇인지 한 눈에 파악하기 어렵기 때문이다. |

|  |
| --- |
| 2.C1. Answer |
|  |

|  |
| --- |
| 2.C2. Answer |
| (a)    (b)    (c) |

|  |
| --- |
| 2.C3. Answer |
| (a)    (b)일치한다.   1. 그렇다. 심벌구간의 정수배 지점에서 정확히 1또는 -1을 지난다. 2. ISI가 없다고 말할 수 있다. 정수배 지점에서는 1또는 -1을 지남으로써 수신 신호의 타이밍 오류에 덜 민감하게 된다. 따라서 이 지점들에서 오류 없이 샘플링이 가능하게 된다. |

|  |
| --- |
| 2.C4. Answer |
|  |

|  |
| --- |
| 2.C5. Answer |
| 일치한다. 정수배 지점에서 zero crossing 하며 심벌 간의 간섭 x. |

|  |
| --- |
| 2.C6. Answer |
| (a)      (b) 심벌 당 샘플 수를 L에서 L+1로 변경함으로써 본래 심벌구간의 정수배 지점에서 샘플링 되지 않기 때문 |

|  |
| --- |
| 2.D1. Answer |
| Command line에서 ‘>>2\*ceil(rand\*4)-5’를 여러 번 실행해본 결과 1,-1,3,-3 이 나옴. 4가지 값이 가능하다. |

|  |
| --- |
| 2.D2. Answer |
|  |

|  |
| --- |
| 2.D3. Answer |
|  |

|  |
| --- |
| 2.E1. Answer |
| (a)          (b) Roll-off factor이 커질수록(1에 가까울수록) eye가 크게 열린다. |

|  |
| --- |
| 3.A. Answer |
| p(t)신호가 Raised cosine pulse이면 실수신호이며 y축에 대하여 대칭성을 띠고 있기 때문에 p\*(-t)=p(t)이다. |

|  |
| --- |
| 3.B. Answer |
| Binary    4-ary |

|  |
| --- |
| 3.C. Answer |
| 1. binary신호의 경우, 심벌구간의 정수배 지점에서 ‘정확히’ 두개의 지점으로 모이지 않는다.. 4-ary신호의 경우도 4개의 지점으로 모이지않는다. 2. Matched filtering을 하는 경우, 수신단에서 제곱이 되기 때문에 송신기에서는 SRRC 펄스를 사용(squre root RC를 보내줘야)해야 ISI가 없는데 송신단에서 RC펄스를 그대로 사용했기 때문에 ISI가 발생했다. |

|  |
| --- |
| 3.D. Answer |
| >바이너리    >4-ary |

|  |
| --- |
| 3.E. Answer |
| 1. SRRC 펄스 성형 해놓고 수신단에서는 matched filtering이 아닌 tx\_signal(일반LPF)로 eye diagram을 확인했으니 ISI가 발생한다. 2. SRRC펄스성형된 송신신호이기 때문에 수신기에서 matched filterring을 해야한다. Sqare root RC 사용하기 때문에 수신단에서 제곱이 되어야 ISI문제가 발생하지 않기때문. (SRRC의 제곱 -> rasied cosine -> ISI-free) |

|  |
| --- |
| 3.F. Answer |
| >바이너리     * 4-ary |

|  |
| --- |
| 3.G. Answer |
| 1. ISI가 존재하지않는다. Binary 신호는 2개의 지점에서 정확히 모이고 있고, 4-ary PAM 신호는 4개의 지점에서 정확히 모이고 있기 때문에 . 2. 1)송신기에서 SRRC펄스성형 후 수신기에서 정합필터를 수행하는 경우, 정합필터를 수행함으로써 SRRC가 제곱되어 Raised cosine이 되므로 ISI-free가 된다.   2)송신기에서 Raised cosine(RC)펄스성형 후 수신기에서 이상적인 LPF를 수행하는 경우, 이미 ISI-free인 Raised cosine이기 때문에 수신기에서는 그냥 이상적인 LPF를 통과시켜 ISI-free이다.   1. SRRC펄스성형 후 수신기에서 정합필터를 수행하는 방법이 더 적합할 것 같다. 정합필터는 신호대잡음비가 최대가 되므로 신호 대비 노이즈 작아지기 때문에 노이즈 관점에서 더 적합한 방법이 될 것 같다.. |

Chapter 21. BPSK BER Simulation at the Waveform Level

학번: 22012225 이름: 손보경

|  |
| --- |
| 1.B. Answer |
|  |

|  |
| --- |
| 1.C3. Answer |
|  |

|  |
| --- |
| 1.C4. Answer |
| (a)      (b) |

|  |
| --- |
| 1.C5. Answer |
| 시간이고정->시간축평균아님. -> 앙상블평균이다.  랜덤 프로세스에 대한 ergodicity에 의해,  샘플 분산 =power <-> n[k]의 분산 =N0/(2t\_step)    (‘제곱해서 평균’ = ‘분산’= ‘power’ ) |

|  |
| --- |
| 1.E1. Answer |
|  |

|  |
| --- |
| 1.E2. Answer |
| 랜덤 프로세스에 대한 ergodicity에 의해, Power = PSD의 면적 = N0/(2\*t\_step) = v\_n = 샘플 분산  즉, 노이즈 샘플의 분산을 v\_n으로 만들어주기위해 sqrt(v\_n)을 randn ~N(0,1)에 곱해준다. |

|  |
| --- |
| 2.A. Answer |
|  |

|  |
| --- |
| 2.B. Answer |
|  |

|  |
| --- |
| 2.E. Answer |
| 1번째(n=1) 송신비트는 N만큼 지연이 되므로 n=1일 때 샘플인덱스는 N이어야함.  또한 심벌당 샘플수인 L씩 점프함.  따라서 z\_samples의 샘플 인덱스는 N+ (n-1)\*L 로 두어야한다. |

|  |
| --- |
| 2.F. Answer |
| <3. Unipolar to Bipolar (amplitude modulation)>에서 0-> -1로, 1->1로 전송.  따라서 (z\_samples(N+(n-1)\*L)>0) 이 참이었으면(즉, 양이면) 비트1, 거짓이었으면(음이면) 비트0이 왼쪽 변수(estimated\_data\_bit. 즉, 판별비트)로 저장되는 것임. |

|  |
| --- |
| 3.A2. Answer |
|  |

|  |
| --- |
| 3.A3. Answer |
| 1. N\_frame, Ns   N\_frame:프레임의 수, Ns:프레임 당 전송 비트 수이므로 .  전송된 전체 비트수 = 프레임수 \* 프레임당 전송비트수 하면 된다.  (b) |

|  |
| --- |
| 3.B2. Answer |
| BPSK는 Antipodal 시그널의 일종이므로 비트 에너지가 Eb인 Antipodal signal의 BER은 다음과 같다.  Q(sqrt(2\*EbN0)) -> 0.5\*erfc(sqrt(2\*EbN0/2)) -> 0.5\*erfc(sqrt(EbN0))    3.A3(b)를 여러 번 돌려본 결과 BER이 0.06XX부터 0.08XX도 나오는 것을 확인했는데, 이론치BER\_exact와는 조금 차이가 난다는 것을 알 수 있다. |

|  |
| --- |
| 3.C1. Answer |
| 조금 차이가 난다. |

|  |
| --- |
| 3.C2. Answer |
| 실험하는 프레임의 수가 충분히 크지 않아 이론적인 BER과는 차이가 발생한 것 같다. |

|  |
| --- |
| 3.D1. Answer |
| 1. sum\_Ne   (b)    거의 비슷하다. |

|  |
| --- |
| 3.D2. Answer |
| 프레임 개수를 0으로 초기화해놓고 while문을 한번 돌때마다 프레임 개수를 하나씩 카운트하도록하고 이 while문은 전체 비트 에러 발생 개수가 25보다 크면 반복이 멈추도록 변경했는데, 에러율이 낮기때문에 에러가 25번나기까지 while문이 많이 반복된다. 따라서 프레임 수가 많이 증가했다. |

|  |
| --- |
| 3.E. Answer |
|  |

|  |
| --- |
| 3.F. Answer |
| 거의 일치한다. |

Chapter 22. QPSK and OQPSK in Simulink

학번: 22012225 이름: 손보경

|  |
| --- |
| 1.A2. Answer |
| (b) |

|  |
| --- |
| 1.B. Answer |
|  |

|  |
| --- |
| 1.F. Answer |
|  |

|  |
| --- |
| 2.A1. Answer |
| -> Bernoulli Binary Gernerator  -> Bernoulli Binary Gernerator1  -> Sine Wave  -> Sine Wave1  x(t) -> Product  y(t) -> Product1  (t) -> Sum |

|  |
| --- |
| 2.A2. Answer |
| |  |  |  | | --- | --- | --- | | 블록 | 파라미터 | 이유 | | Bernoulli Binary Generator, Bernoulli Binary Generator1 | Sample time=1 | QPSK 심벌 구간(길이) =1이다. | | Output data type = Boolean | 데이터 비트는 1과 0만 존재하는 Boolean 형식의 값이다. | | Bernoulli Binary Generator | Initial seed = 25 | Bernoulli Binary Generator와 Bernoulli Binary Generator1의 Initial seed가 달라야하는 이유는 두 비트를 다른 값으로 생성하기 위해 | | Bernoulli Binary Generator1 | Initial seed = 디폴트 값(변경x) | | Unipolar to bipolar convertor, Unipolar to bipolar convertor1 | M-ary number = 2 | 2진 신호이기때문 | | Gain, Gain1 | Gain=4 | 비트에너지가 16이므로 식(22.4)에 의하여 비트에너지에 제곱근을 씌운 4를 sine wave와 곱해주는 것. | | Sine wave, Sine wave1 | Amplitude= sqrt(2) | Orthonormal 기저함수에서, sqrt(2/T)를 진폭으로 해줘야하는데 (  T(심벌주기)이 1이므로 sqrt(2)로 설정 | | Frequency = 60\*pi | 반송주파수=30Hz이므로 30\*2\*pi [rad/sec] 로 설정. | | Sample time= 3.90625e-3 | 생성된 통과대역 QPSK 파형의 심벌당 샘플 수는 256이므로 1/256 = 3.90625e-3으로 설정. | | Sine wave | phase= pi/2 | Sine wave에서 phase를 pi/2로 설정해서 cosine함수를 표현 | | Sine wave1 | phase= 0 | Sine 함수 | |

|  |
| --- |
| 2.B1. Answer |
| s\_i(t)를 20심벌동안 연속하여 생성하고자 하는데,  QPSK 심벌 주기가 1sec이므로 심벌 20개 주기에 해당하는 시간인 20sec으로 설정한다. |

|  |
| --- |
| 2.B2. Answer |
|  |

|  |
| --- |
| 2.C. Answer |
| s\_i(t), x(t), y(t)의 신호대역폭은 모두 같다. 그럼에도 s\_i(t)가 x(t)와 y(t)에 비해 비트 전송률이 두배 높은 이유는 x(t), y(t)에서의 두 직교기저함수는 동일한 주파수를 가지기 때문에 이 두 신호의 합인 s\_i(t)신호는 x(t), y(t)에서 쓰는것과 같이 동일한 신호대역폭을 쓸 수 있는 것이다. |

|  |
| --- |
| 2.D. Answer |
|  |

|  |
| --- |
| 3.A1. Answer |
| 생성된 통과대역 QPSK 파형의 심벌당 샘플수 =256이므로 적분 주기를 256으로 설정 |

|  |
| --- |
| 3.B2. Answer |
| ‘Integrate and dump’블록은 내부 파라미터인 integration period에 설정된 샘플 수 단위 즉, 256으로 입력 신호의 구간을 나누어, 매 구간의 샘플들이 다 들어오면 그 구간의 입력샘플 값을 모두 더하고, 그 다음 구간이 입력되는 동안 출력하는 기능을 하기 때문에 1sec씩 딜레이 되는 것을 확인할 수 있다. |

|  |
| --- |
| 3.C. Answer |
| 신호가 양수이냐 음수이냐에 따라서 비트를 판별하기 위해서.  (양수이면 1, 음수이면 -1로 판별) |

|  |
| --- |
| 3.D1. Answer |
|  |

|  |
| --- |
| 3.D2. Answer |
| 한 비트의 주기(Ts)인 1초의 시간차이가 발생한 이유는  QPCK\_TX의 sine wave블록에서 샘플 시간을 1/256으로 설정하여 초당 256개의 샘플을 생성했고 integrate and dump에서 integration period를 256으로 설정하여 256개의 샘플을 모두 더하고 다음 구간이 입력되는 동안 출력하기 때문에 1초의 시간차이가 발생하게 된다. |

|  |
| --- |
| 3.E. Answer |
|  |

|  |
| --- |
| 3.F2. Answer |
|  |

|  |
| --- |
| 4.C. Answer |
| EbN0dB -> N0= 16 / (10^( (EbN0dB) /10 )) -> Variance= N0 / (2/256)     1. Eb/N0 =1dB -> Variance = ( 16/(10^(1/10)) )/ (2/256) = 1.6268e+03      1. Eb/N0 =15dB -> Variance =(16/(10^(15/10)) )/(2/256)= 64.7634      1. Eb/N0 =30dB -> Variance = ( 16/(10^(30/10)) )/(2/256)= 2.0480     Eb/N0가 클수록 상대적으로 노이즈의 영향이 줄어들기 때문에 분산이 작아지고 그래프상으로도 점으로 모이며 분산이 작아지는 것을 확인할 수 있다. |

|  |
| --- |
| 4.D2. Answer |
| 1. Phase에 pi/12 더하기      1. Phase에 pi/6 더하기      1. Phase에 pi/3 더하기       원래 점에서의 위상오차는 각 사분면의 90도에 해당하는 위치 즉, pi/4, 3pi/4, -pi/4, -3pi/4인데  더한 위상오차만큼(예를들어 기존 phase에 pi/3을 더한 경우, (pi/4+pi/3), (3pi/4+pi/3), (-pi/4+pi/3), (-3pi/4+pi/3)으로) 점들이 이동하게 된다. |

Chapter 22. QPSK and OQPSK in Simulink

학번: 22012225 이름: 손보경

|  |
| --- |
| 5.C2. Answer |
| (b) 거의 일치한다. |

|  |
| --- |
| 5.C3. Answer |
| QPSK는 서로가 직교하는 두 개의 독립적인 BPSK신호의 합. 따라서 QPSK와 BPSK는 같은 BER을 가짐. |

|  |
| --- |
| 6.A2. Answer |
| p(t)= |

|  |
| --- |
| 6.C2. Answer |
| (a)    (b) 신호의 진폭의 변화가 크다. |

|  |
| --- |
| 6.C3. Answer |
|  |

|  |
| --- |
| 6.C4. Answer |
| 1. 위의 신호 궤적도를 살펴보면 신호의 궤적이 원점을 지나는 시간이 있다. 이때 순간 진폭은 0이 된다. 즉 순간 진폭에 큰 변화가 있다는 것을 알 수 있다. 2. Eye가 최대로 열리는 이웃한 두 지점의 중간지점이다. |

|  |
| --- |
| 6.D1. Answer |
| 1사분면 <-> 3사분면,  2사분면 <-> 4사분면 |

|  |
| --- |
| 6.D2. Answer |
| (1,1) <-> (0,0),  (1,0)<->(0,1).  (0,1)<->(1,0)    (I(t)와 Q(t)의 극성이 동시에 변하는 경우) |

|  |
| --- |
| 7.A1. Answer |
| (b) 위 그래프를 보면 0을 지나지x. 따라서 순간적으로 크기가 0인 시간 없음. 따라서 I(t)와 Q(t)의 극성이 동시에 바뀌지 않음(즉, Zero crossing 없음.) 따라서 OQPSK가 순간 진폭 변화가 적을 것. |

|  |
| --- |
| 7.A2. Answer |
| (b) I(t)신호는 그대로 두고 Q(t)신호에 반심벌 딜레이 블락을 추가해서 한비트씩 딜레이를 시킴. 두 채널 중 한 채널에 반심벌씩 딜레이를 시켰기 때문에 I채널과 Q채널의 두 비트가 동시에 반전되는 것을 피할 수 있다. |

|  |
| --- |
| 7.B1. Answer |
| (b) OQPSK는 순간 진폭의 변화가 적을 것이라 예측했고 위와 같이 순간 진폭의 변화가 적은 것을 확인. |

|  |
| --- |
| 7.B2. Answer |
| 6.C2에서 캡쳐한 QPSK의 파형은 순간 진폭의 변화가 크고, 7.B1에서 캡처한 OQPSK파형은 순간 진폭의 변화가 작다. |

Chapter 23. QAM in Simulink

학번: 22012225 이름: 손보경

|  |
| --- |
| 1.B3. Answer |
| (a)    (b) |

|  |
| --- |
| 1.B4. Answer |
| [ 심벌 -> 심벌에너지 ]  0000 -> 18  0100 -> 10  1100 -> 10  1000 -> 18  0001 -> 10  0101 -> 2  1101 -> 2  1001 -> 10  0011 -> 10  0111 -> 2  1111 -> 2  1011 -> 10  0010 -> 18  0110 -> 10  1110 -> 10  1010 -> 18  평균심벌에너지 Es= 10  (b) 공식에 대입해보면, Es= 2\*(M-1)/3 = 2\*(16-1)/3 = 10. 위 (a)의 답과 일치한다. |

|  |
| --- |
| 1.B5. Answer |
| 1. Eb= Es / 2. Eb = 10/4= 2.5 |

|  |
| --- |
| 1.B6. Answer |
| 인접한 symbol과 비트 차이가 1인 형태를 gray mapping이라고 한다. 가장 가까운 곳은 에러율이 높으므로 맞는 비트 개수가 많았으면 하는데, 이 gray mapping의 경우 인접한 symbol과 틀린 비트개수가 1개이기 때문에 ‘비트’에러율을 줄일 수 있다는 장점이 있다. |

|  |
| --- |
| 1.C1. Answer |
| |  |  | | --- | --- | | 데이터비트 | 판별조건 | | b4 | If z1>0 ‘b4\_estimate’=1, ELSE ‘b4\_estimate’=0 | | b3 | If |z1|<2 ‘b3\_estimate’=1, ELSE ‘b3\_estimate’=0 | | b2 | If z2 < 0 ‘b2\_estimate’=1, ELSE ‘b2\_estimate’=0 | | b1 | If |z2| <2 ‘b1\_estimate’=1, ELSE ‘b1\_estimate’=0 | |

|  |
| --- |
| 1.C2. Answer |
| Distance 계산이 필요하지 않다. (z1, z2)에 따라서 비트 추정이 가능하기 때문이다. |

|  |
| --- |
| 2.B2. Answer |
|  |

|  |
| --- |
| 2.C1. Answer |
|  |

|  |
| --- |
| 2.C2. Answer |
| 2\*pi\*20 (rad/s) |

|  |
| --- |
| 2.C3. Answer |
| 캐리어 주파수가 2\*pi\*20 (rad/s) (즉, 20Hz)이고  신호들 간의 직교성을 사인, 코사인으로 나타낼 수 있기 때문. |

|  |
| --- |
| 2.C4. Answer |
|  |

|  |
| --- |
| 3.A. Answer |
|  |

|  |
| --- |
| 3.B2. Answer |
|  |

|  |
| --- |
| 3.B3. Answer |
| 좌표점이 올바르게 찍혀 있는 것으로 보아 올바르게 설계됐다. |

|  |
| --- |
| 3.B4. Answer |
|  |

|  |
| --- |
| 3.B5. Answer |
|  |



|  |
| --- |
| 3.B6. Answer |
| 주파수오차가 BER 성능에 더 치명적이다. 노이즈 Variance를 0에서 10으로 바꿨을때는 기존의 좌표점을 중심으로 분산이 생기는 정도이지만 주파수오차가 있을 경우 이에 따라 좌표점이 전체적으로 회전하는 것을 볼 수 있기 때문이다 |



|  |
| --- |
| 3.B7. Answer |
| 위상과 주파수 오차에 오류가 발생할 수 있으므로 수신 신호로부터 반송파의 주파수와 위상 정보를 검출하여 이 위상 정보를 이용해 복조하는 (송수신 간에 위상을 동기) 동기 검파 방식이 필요하다. |

|  |
| --- |
| 3.E. Answer |
|  |

|  |
| --- |
| 3.F. Answer |
| 일치한다. |

|  |
| --- |
| 4.B2. Answer |
|  |

|  |
| --- |
| 4.C. Answer |
| 1.D6)    거의 일치한다. |

Chapter 24. Convolutional Coding

학번: 22012225 이름: 손보경

|  |
| --- |
| 1.B1. Answer |
| 1/2 |

|  |
| --- |
| 1.B2. Answer |
| 인코딩에 사용된 가장 먼 비트의 거리. -> 3 |

|  |
| --- |
| 1.B3. Answer |
|  |



|  |
| --- |
| 1.B4. Answer |
|  |

|  |
| --- |
| 1.C1. Answer |
| 위의 코드 순서가 바뀌어  D(2)=D(1)  D(3)=D(2)  가 된다면, D(2)와 D(3) 모두 D(1) 값을 가지게 되어 의도한 갱신동작을 하지 못한다. |

|  |
| --- |
| 1.C2. Answer |
| 동일하다. |

|  |
| --- |
| 3.A1. Answer |
| c=01  d=11 |



|  |
| --- |
| 3.A2. Answer |
|  |

|  |
| --- |
| 3.A3. Answer |
|  |

|  |
| --- |
| 3.B1. Answer |
|  |

|  |
| --- |
| 3.B2. Answer |
| 1, 2번째 데이터 |

|  |
| --- |
| 3.B3. Answer |
| (a)      (b)  바로잡았다. |

|  |
| --- |
| 3.C1. Answer |
| a는 b벡터에 원소 1를 추가한 벡터가 된다. |

|  |
| --- |
| 3.C2. Answer |
| new\_output1 -> 1번째 원소로 가기 위한 최단경로로, trellis decoding를 하며 경로를 기록하기 위해. |

|  |
| --- |
| 3.C3. Answer |
|  |

|  |
| --- |
| 3.C4. Answer |
| 3.C3에 첨부. |

|  |
| --- |
| 3.C5. Answer |
| 1. ML\_state\_index는 최소 branch metric를 가진 인덱스를 저장하는 변수. 즉, 벡터 d에서 가장 작은 원소의 인덱스를 반환한다. (여기서 d=[d1 d2 d3 d4]) 2. result는 output벡터의 최소 branch metric를 가진 인덱스에 있는 원소를 저장. 즉, 최소 branch metric를 갖는 첫번째 path를 저장하기 위함. |

|  |
| --- |
| 3.C6. Answer |
| 일치한다. |

|  |
| --- |
| 3.E1. Answer |
| 0과 1로 판정하기 전의 수신신호열과 1과 -1로 이루어진 부호어와 거리 계산 -> 연판정 디코딩 |

|  |
| --- |
| 5.A. Answer |
|  |

|  |
| --- |
| 5.B. Answer |
| 한글이 깨져 있는 것으로 보아 노이즈가 심각하게 큰 상황 같다. |

|  |
| --- |
| 5.C1. Answer |
|  |

|  |
| --- |
| 5.C2. Answer |
|  |

|  |
| --- |
| 5.C3. Answer |
| 수신 신호열 r에서 양수는 1, 음수는 0으로 하여 수신 비트를 0과 1로 판정하여 예상과 일치한다. |

|  |
| --- |
| 5.C4. Answer |
| 노이즈가 많은 환경에서 수신 신호열(BPSK복조 전) r에 대하여 인코딩한 후, 연판정을 하기위해 위와 같이 수정하였다. |

|  |
| --- |
| 5.C5. Answer |
|  |

|  |
| --- |
| 5.C6. Answer |
| 1. X      * 해당 라인은 송신 비트열 EncodedBits를 BPSK 변조하여 AWGN 채널을 통과시킨 후 수신한 신호 벡터 r을 생성한 것이다. 즉 r은 컨벌루션 부호화된 비트열의 원소들을 -1과 1로 변환한 후 노이즈를 더한 값이므로 해당 라인을 수정하지 않으면 노이즈를 그대로 두는 것이다. |

Chapter 25. QAM in Simulink

학번: 21511824 이름:김소정

|  |
| --- |
| 1.A. Answer |
| h = sqrt(0.5)\*(randn + j \* randn) |

|  |
| --- |
| 1.B1. Answer |
|  |
| 1.B2. Answer |
|  |

|  |
| --- |
| 1.C1. Answer |
|  |
| 1.C2. Answer |
|  |
| 1.C3. Answer |
|  |
| 1.C4. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명  텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 1.C5. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 1.C6. Answer |
| 페이딩이 1보다 클 때 즉 >1 은 순간적으로 BER이 감소하고,  페이딩이 1보다 작을 때 즉 <1은 순간적으로 BER이 증가한다. |
| 1.C7. Answer |
| 원래 n=n1+j\*n2인데 이 n은 랜덤변수로서 원래 분포 자체가 circular symetry하다. 따라서 위상을 바꾸더라도 회전만 할 뿐 원점거리와 원 위에 유지된다. |

|  |
| --- |
| 2.A1. Answer |
| (a)    → h=z+jy이다. Z와 y는 평균이 0이고 분산이 1/2인 가우시안 랜덤변수이기 떄문에 sqrt(1/2)\*(randn+j\*randn)으로 나타낼 수 있다.    → n~N(0, N0/2) 분포를 따르는 노이즈가 실수부, n~N(0, N0/2) 분포를 따르는 노이즈가 허수부    →h의 phase 만큼 신호성분이 돌아갔기 때문에 h의 phase반대로 위상을 돌려준다.  (b)  텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 2.A2. Answer |
| 레일레이 fading channel에서의 BER은 같은 Eb/N0를 가지더라도 fading factor에 의해서 비트에너지가 순간적으로 변하기 때문에 그러한 상황을 시뮬레이션하기 위해선 while문 안에 h=sqrt(0.5)\*(randn+j\*randn) 구문을 위치시켜야 한다. |
| 2.A3. Answer |
| 같은 Eb/N0에서 fading factor의 영향에 따라 BER이 변화하는 것을 관찰 할 수 없다. |

|  |
| --- |
| 2.B. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명    테이블이(가) 표시된 사진  자동 생성된 설명 |
| 2.C. Answer |
| 허수부분을 날리기 위해서 판별변수의 실수부를 취한 후 부호를 취한다.  fading을 해도 노이즈에 영향을 주지 않는다. 그래서 노이즈에는 여전히 허수부분이 존재하기 때문에 real을 해야한다. |
| 2.D. Answer |
|  |
| 2.E. Answer |
|  |
| 2.F. Answer |
| 이론치와 일치한다. |
| 2.G1. Answer |
| ) |
| 2.G2. Answer |
|  |
| 2.H. Answer |
| fading 채널의 곡선모양은 거의 직선모양으로 감소하고 AWGN채널의 곡선모양은 폭포수처럼 급격하게 감소하는 곡선모양이다. 이러한 차이점으로 인해 AWGN 채널은 SNR이 증가할수록 BER성능이 좋아지고 fading 채널은 SNR이 증가한다해도 BER성능이 그닥 좋아지지 않는다. |
| 2.I. Answer |
| 일치한다. |
| 2.J1. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 2.J2. Answer |
|  |
| 2.J3. Answer |
| 일치한다. 비트 에너지가 동일하기 때문이다. |

|  |
| --- |
| 4.A1. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 4.A2. Answer |
|  |
| 4.A3. Answer |
|  |
| 4.A4. Answer |
| 1. L이 증가함에 따라 BER이 작아지는 경향을 보이고, 커브의 기울기도 증가한다. 2. L은 에너지를 L만큼 나눠서 보내겠다는 것을 의미한다.   낮은 SNR영역에서 L만큼 에너지를 나눠서 보내면 SNR이 작아지게 되고 SNR이 작은 신호에서 BER성능을 비교하는 것은 큰 의미가 없다.  그래서 BER 성능이 약간씩 감소할 것 같다.  높은 SNR영역에서 L만큼 에너지를 나눠서 보내면 각각 가질 수 있는 신호가 노이즈 비율보다 신호의 비율(SNR)이 더 크고 그 중 제일 큰 값을 고르는 것이 SDC방식이기 때문에 BER성능이 좋아지는 것을 볼 수 있다.  -----------------------------------------------------------------------------------------------------  신호를 L만큼 나눠서 보낸다. 낮은 SNR영역에서는 신호의 에너지를 L만큼 나눠서 보내면 신호성분중에 가장 좋은 것을 골라도 노이즈가 너무 많게 되는것이다. 즉, 신호성분이 노이즈에 묻힌다. 그래서 BER이 증가한다. |

|  |
| --- |
| 4.B1. Answer |
| (a)    (b)  텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 4.B2. Answer |
|  |
| 4.B3. Answer |
| L이 증가함에 따라 BER은 감소한다. |
| 4.B4. Answer |
| SDC BER은 L값에 따라서 변화가 크지 않지만, EDC는 L값이 증가함에 따라 BER이 확연히 감소하는 것을 볼 수 있다. 그리고 L값이 ㅋㅓ질수록 bER의 감소기울기가 크다. |

|  |
| --- |
| 4.C1. Answer |
| 텍스트이(가) 표시된 사진  자동 생성된 설명 |
| 4.C2. Answer |
|  |
| 4.C3. Answer |
|  |
| 4.C4. Answer |
| L이 증가할수록 BER은 더욱 낮아지고 기울기는 증가하였다. |
| 4.C5. Answer |
| SRC, EGC보다 BER이 더 낮게 나왔다. |

|  |
| --- |
| 4.D1. Answer |
| 1. BER성능은 MRC가 제일 좋다. (SDC<EGC<MRC) 2. SDC는 L이 3에서 5로 증가해도 BER성능의 변화가 거의 없다.   EGC는 L값이 증가하면 할수록, BER성능이 좋아지는 경향을 보이고,  MRC가 L값의 증가에 따라 BER성능이 더 많이 좋아지는 것을 볼 수 있다. |
| 4.D2. Answer |
| 구현 복잡도가 제일 낮은 방식 : SDC  branch 중 |H(k)|가 가장 큰 brach의 r값을 선택하여 판별변수 D를 생성한다.  따라서 계산 량이 많지 않기 때문에 복잡도가 가장 낮다고 볼 수 있다.  구현 복잡도가 제일 높은 방식 : MRC  branch 들의 r값을 |H(k)|의 값이 크면 가중치를 많이 줘서 더하고, |H(k)|의 값이 작으면 가중치를 적게 줘서 더하는 방식이기 때문에 계산 량이 가장 많아 복잡하다고 볼 수 있다. |