Pulse Shaping

2016707044 김영웅

1) 기존 시뮬레이터에 추가했던 pulse shaping 부분을 Square root raised cosine filter를 사용하여 pulse shaping 하는 부분으로 대체하고 수신단에 Matched filter를 사용하는 부분을 추가하시오.

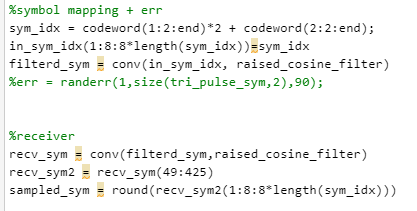
2) 수신단에서 최적의 sampling point를 어디에 잡는 것이 좋을지 고민해 보시오.

Linear block code와 cyclic redundancy code에서 코드의 차이가 없어서 제출하는 코드를 하나로 통합하였습니다.

해당 코드는 symbol mapping -> pulse shaping ->

* 수신부: matched filter -> re - sampling

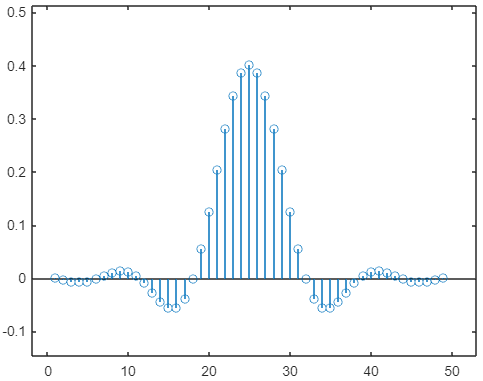
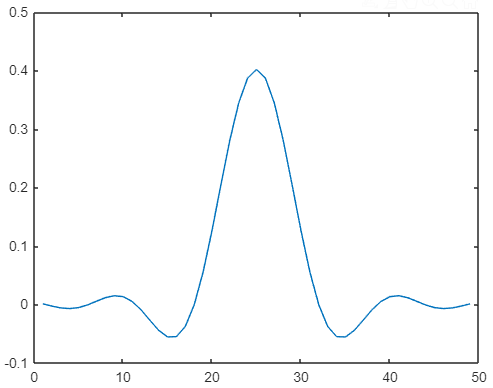
과정을 나타내고 있습니다.

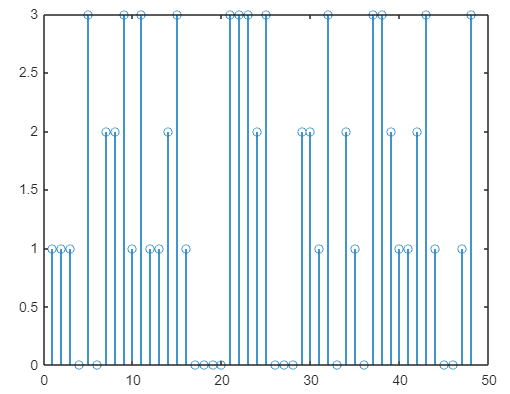
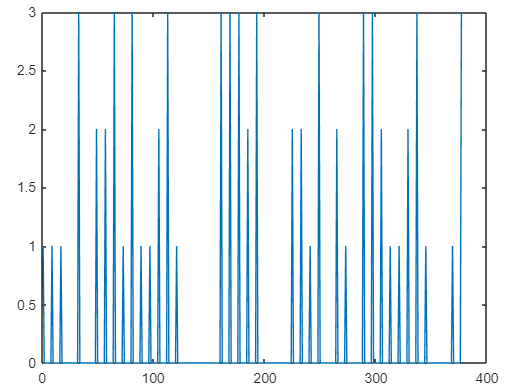


symbol mapping은 이전 과제에서 동일하게 QASK형식으로 하였다.



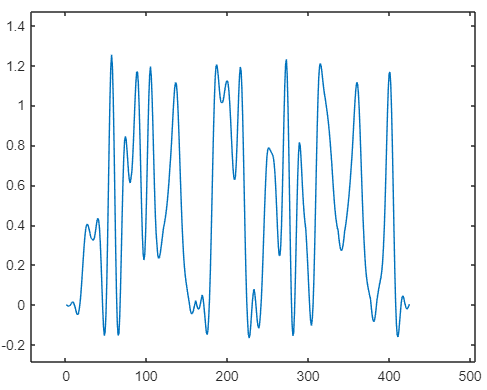
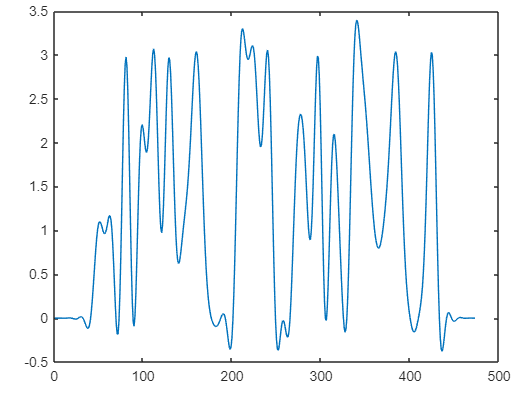
Raised cosine filter는 rcosfir를 통해 만들었다. Roll – off factor는 0.5로 주었고, range는 -3부터 3까지로 하였고, rate는 8, T는 1초로 하였다.

Symbol over sampled symbol

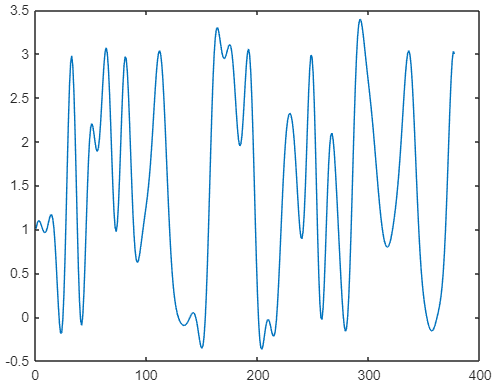
Raised cosine filter를 적용하기 위해서 filter의 rate가 8이기 때문에 symbol을 8로 over sampling해주었다.

송신부에서 raised\_cosine\_filter를 적용하여 pulse\_shaping을 했다. (좌측 사진)

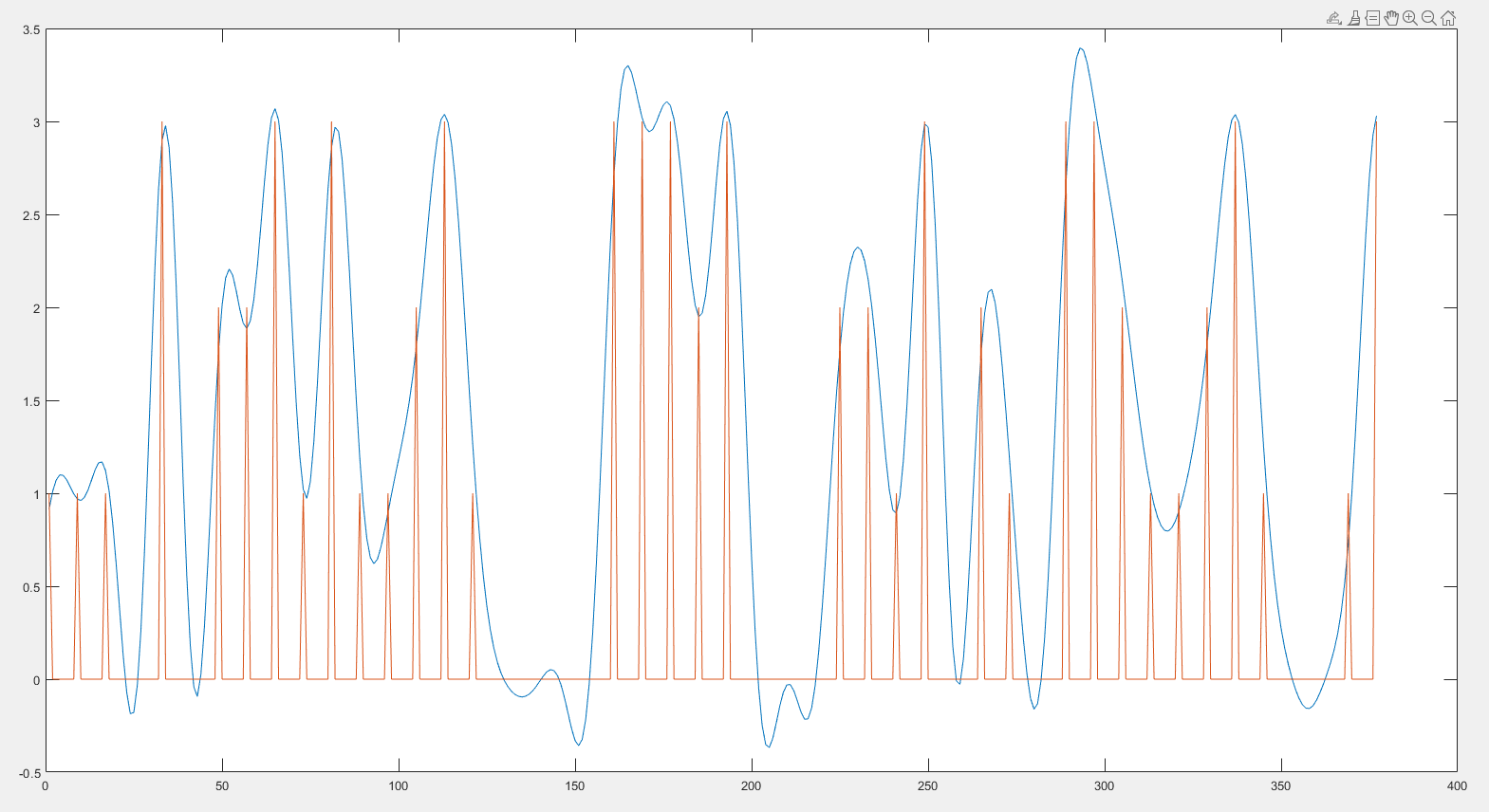
수신 부에서는 matched filter를 적용하기 위해서 다시 제작했던 raised cosine filter와 pulse shaping한 신호를 convolution하여 필터를 적용했다. (우측 사진)

송신 수신부에서 convolution을 거치면서, 신호의 앞 부분과 뒷부분에 딜레이가 포함된 것을 볼 수 있었다. 따라서 수신부 신호에서 유의미한 값이 나오기 이전과 유 의미한 값이 더 이상 나오지 않는 부분은 모두 제거하여 신호 처리를 하였다.

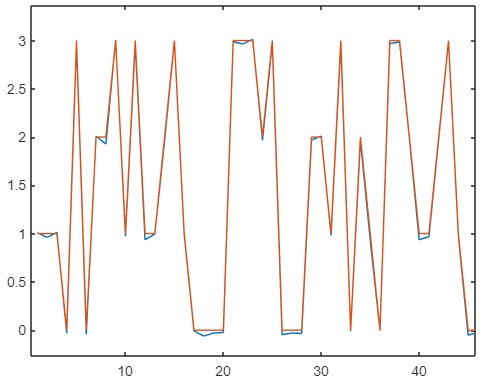


앞 뒤로 불필요한 부분을 보정한 수신 신호

Sampling point를 over sampling rate와 같은 8로 한 결과, symbol sequence와 가장 흡사한 결과를 얻을 수 있었다.



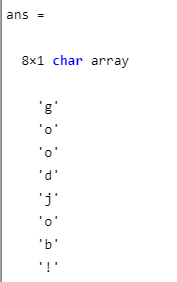
수신 신호에 대해서 sampling rate를 8로 하여 sampling한 신호 사진



수신 신호를 8로 sampling한 신호와 원래 symbol을 비교한 사진이다.

두 신호가 매우 흡사하지만, ISI로 인해서 조금씩 오차가 있는 것을 확인할 수 있다.

Round 함수를 활용하여 원 신호를 복원하였고, 복원된 신호를 토대로 수신 신호를 처리한 결과 송신했던 goodjob! 그대로 출력 되는 것을 확인할 수 있었다.



과제를 하면서 적당한 sampling point를 찾기 위해서 몇번의 다른 시도들을 했었는데 sampling 값이 조금만 달라져도 전혀 원래 값을 복원하지 못했다. Over sampling rate가 그대로 수신부에서 신호를 복원 할 때 에도 그대로 적용이 되는 것 같다.