

# Demo 2

## Team 4



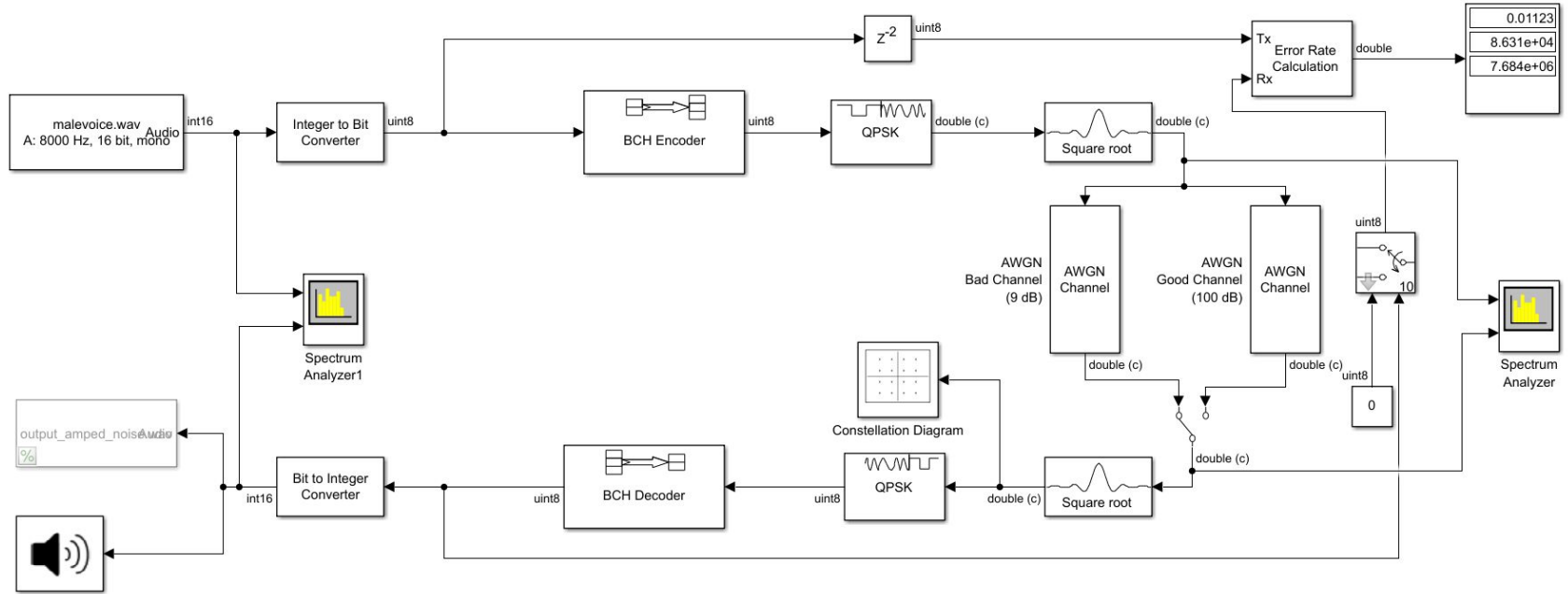
# Performance Table

Our assigned performance conditions:

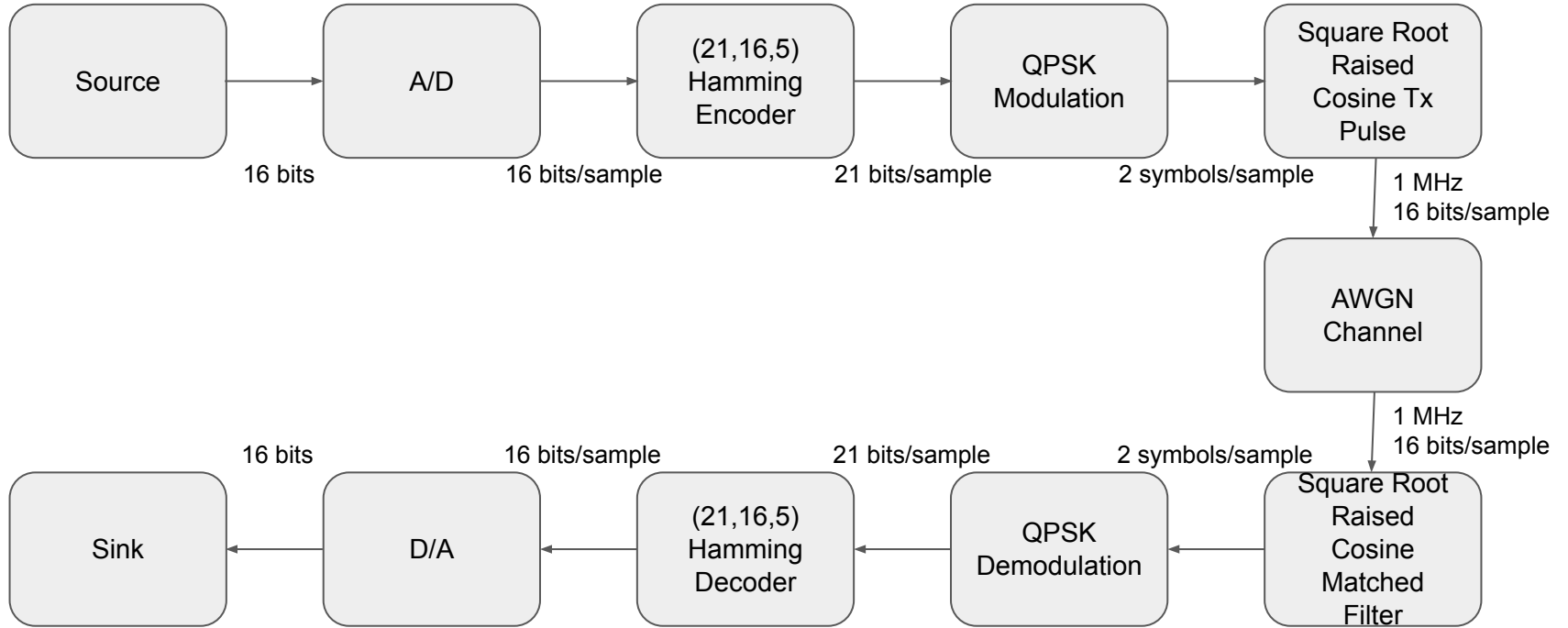
Audio BW (kHz)	Target BER	Spectral Mask (kHz)	Target Channel	Delay (ms)
4	$10^{-5}$	150	$\delta$	15

Criterion	Simulink Performance	FPGA Performance
Message Transmission (bit rate)	128 Kbps	16
Transmission Reliability (bit error probability)	0.01281	
Processing Delay	80.735 ms	
Channel Bandwidth	2 kHz	

# Simulink Model



# System Block Diagram



# Design Decision 1 - Sine Lookup Tables

Sine lookup tables with different phases: easy to implement and convenient approach to convert symbols into waveforms

Any loss? Not easy to restrict frequency spectrum for our channel without additional devices such as filters

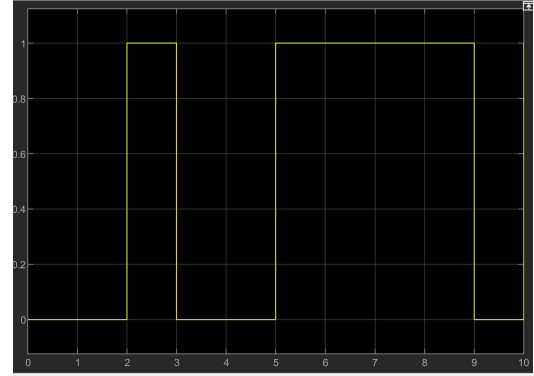
So, any alternatives?



# Transition to Raised Cosine Filters

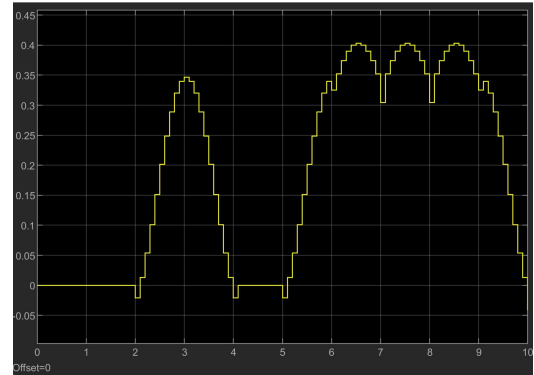
## Advantages:

- Minimal intersymbol interferences
- Allows easier manipulation of bandwidth



## Disadvantages

- Hard to implement on FPGA (look up tables !)
- Takes up more system resources



## 2nd Design Decision - Clock Domain Crossing

- Necessary for FIFO implementation
- Faster clock (50 MHz) stacks data
- Switch to slower clock (1 MHz)
- Slower clock necessary in order to meet the channel frequency requirements
- Read data using slower clock



Advantage:

More accuracy = Lower error rate!



Disadvantage:

Lower transmission rate = increased processing delay





# QPSK vs 16 QAM

Application: Communication Channel



QPSK	16 QAM
Less prone to error	More prone to error
Less symbols to represent	More symbols to represent
Slower to process	Faster to process

Our given channel has 9dB SNR which is smaller than 21dB SNR. To reduce the error from this, we decided to use QPSK with less potential error. In return, we had to bear with slower symbol rate.

# QPSK Mapping and Real & Imaginary Separation

Implications and trade-off from complex representation?

Yes, simpler method available when using a look-up table based on binary representation with each phase.

Yet, enables convolution through raised cosine filter. (Not an option but a must!)

Pulse shaping, why? But longer computation time and power consumed.

