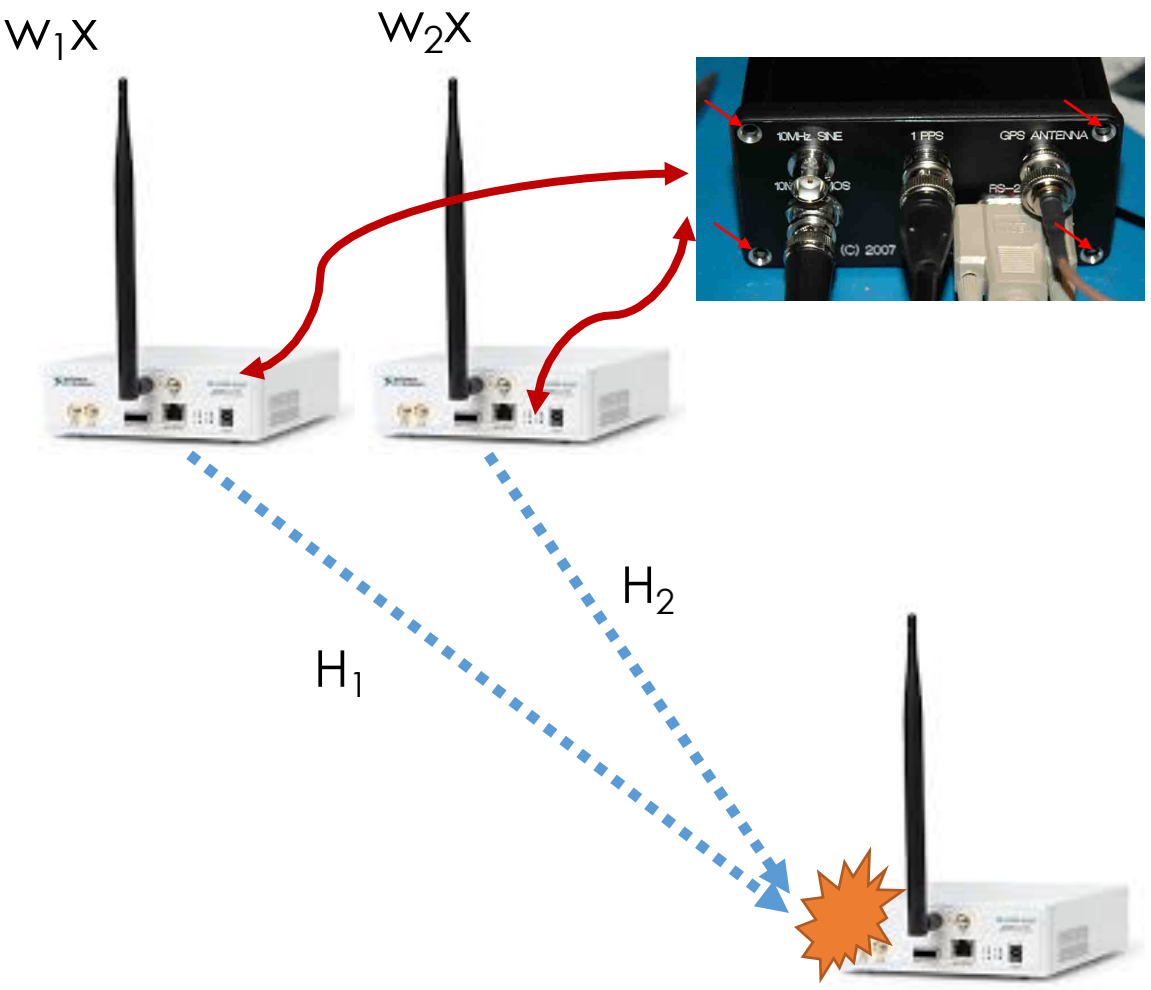
Wireless Communication Systems
@CS.NCTU

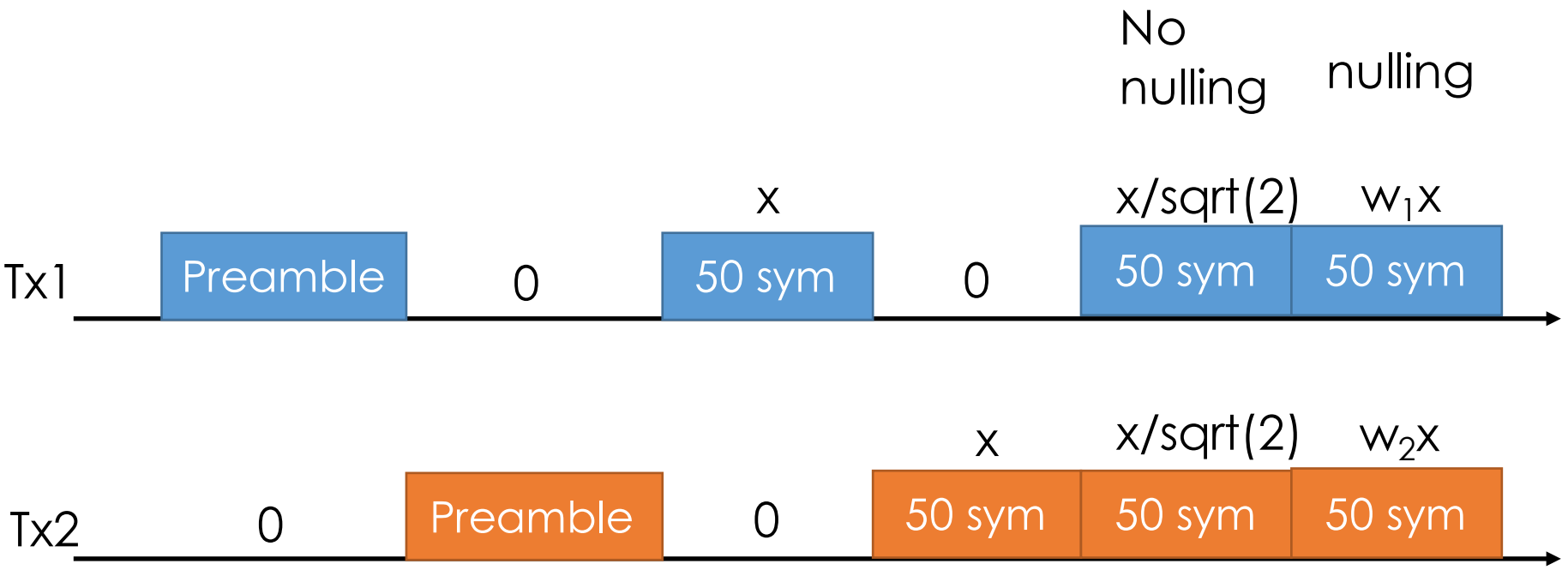
Lab3: Interference Nulling

Interference Nulling

- 2-anttena Tx nulls its signals at a 1-antenna Rx
  - Connect two USRPs to the same external clock as Tx



Packet Format



Modify Lab1 Code (Tx)

- Extend the code to tx1 (ant1) and tx2 (ant2)
  - A vector of digital bits (0 or 1)
  - Modulate them to two sequences of frequency-domain signals (X[k])
  - Generate random channels $H_1[k]$ and $H_2[k]$
    - In Matlab, $H_i[k]$ = (randn() + i * randn()) / sqrt(2)
    - One channel for each subcarrier across all symbols
  - Precode frequency-domain signals by **unit beamforming vector w**
    - ant1: $X_1'[k] = w_1[k]\ X[k]$, ant2: $X_2'[k] = w_2[k]\ X[k]$
    - $(w_1[k], w_2[k])$ is a unit vector, i.e., $w_1^2[k] + w_1^2[k] = 1$
  - Multiply the frequency-domain signals by the randomly-generated **unit frequency-domain channel**
    - $Y_1 = H_1[k] * X_1'[k]$, $Y_2 = H_2[k] * X_2'[k]$
  - Convert f.-domain signals to time-domain signals (y_1, y_2)

Modify Lab1 Code (Rx)

- Sum up the two time-domain signals
  - $y = y_1 + y_2$
- Generate noise to simulate different SNR (as in Lab 1)
  - Make sure $(E[y_1^2 + y_2^2]) / N = SNR$
  - Not $E[(y_1+y_2)^2] / N = SNR$
- Learn the channel H_1 and H_2
- Decode the received signal via SISO decoding (as in lab1)

TODO

TODO: step1

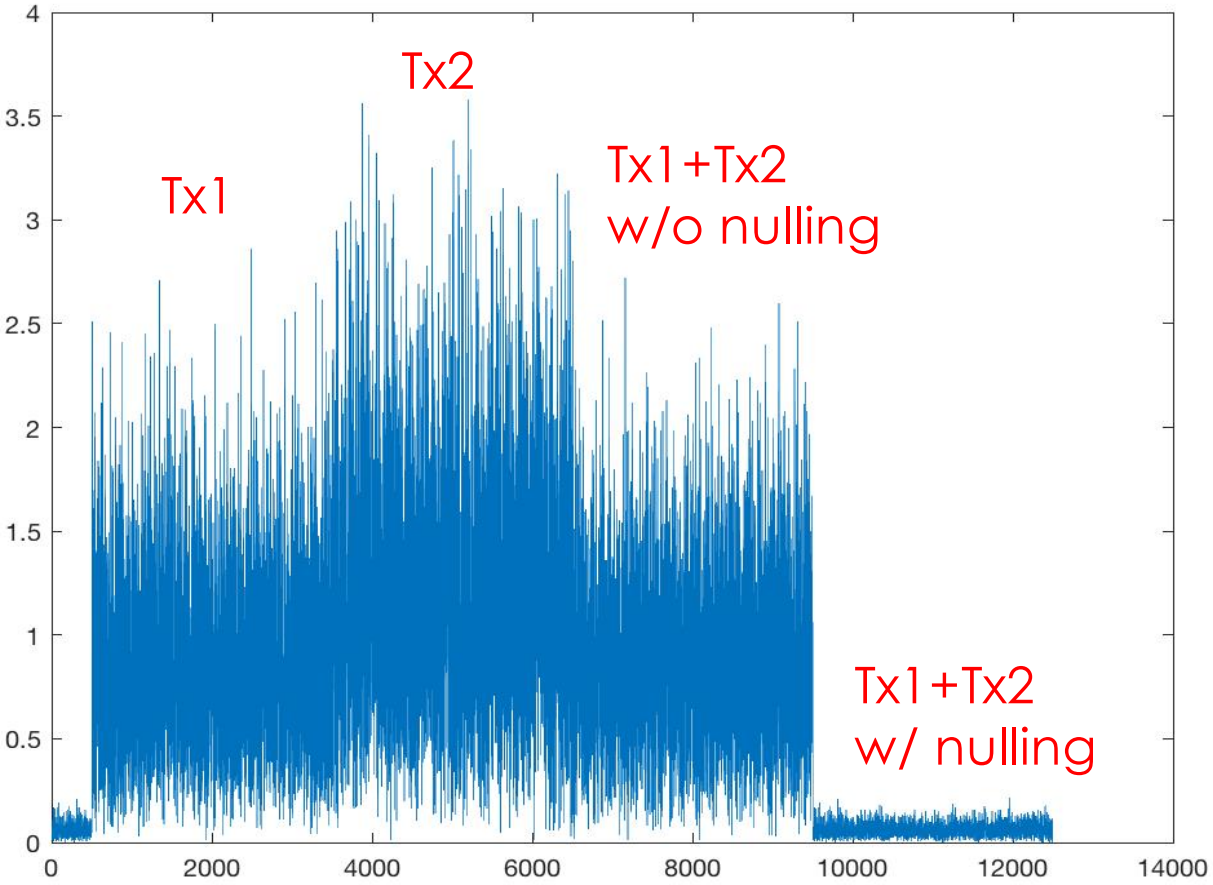
- Modify the gen_signal.m
  - Generate a stream of frequency-domain signals
  - randomize two frequency-domain random channels
  - Calculate the precoding coefficients w_1 and w_2
  - Perform precoding
  - Convert frequency-domain signals to time-domain signals

TODO: step2

- Generate different levels of noise (as in Lab1)
- Compare the SNR w/ and w/o precoding
  - SNR_1:
    - Tx1 sends the **non-precoded signals y_1** ($y_1 = ifft(H_1X)$) along, and rx decodes y_1
  - SNR_2:
    - Tx2 sends the **non-precoded signals y_2** ($y_2 = ifft(H_2X)$) along, and rx decodes y_2
  - SNR_{orig}:
    - tx1 and tx2 send **the non-precoded signals** simultaneously ($y = ifft(H_1X_1 + H_2X_2)$)
    - rx decodes the combined original signals
  - SNR_{null}:
    - tx1 and tx2 send **precoded signals** simultaneously ($y = ifft(H_1W_1X_1 + H_2W_2X_2)$)
    - rx decodes the combined precoded signals

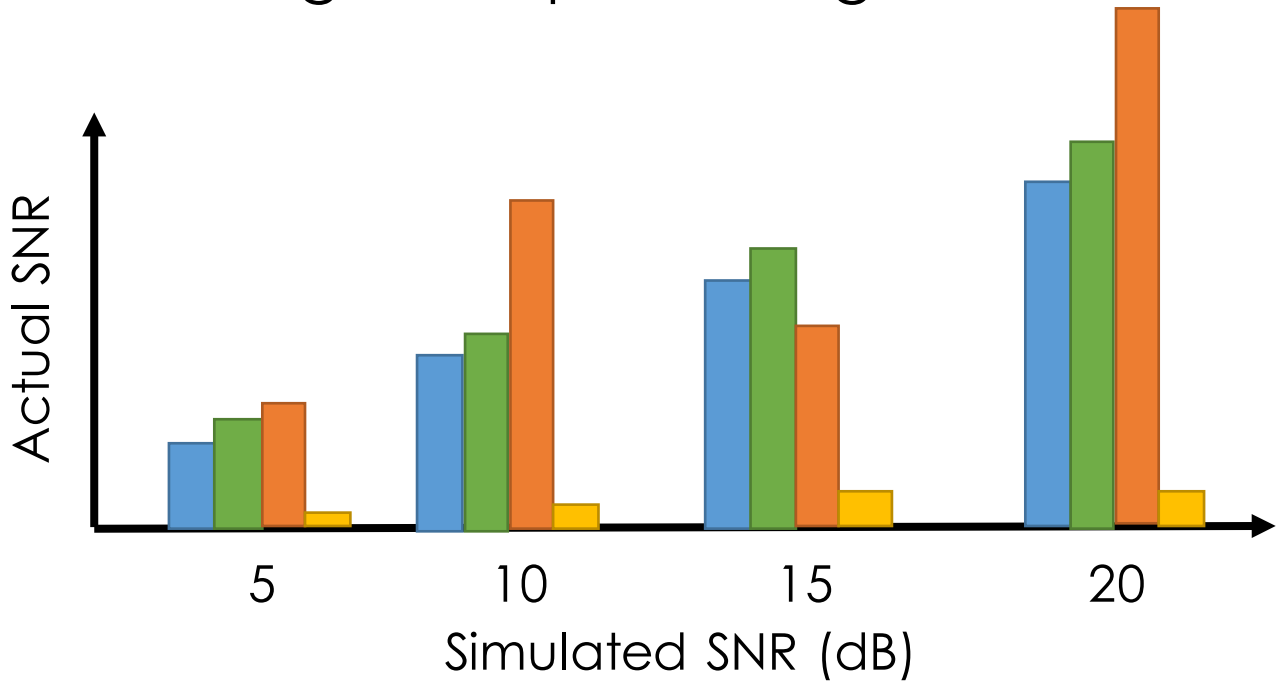
TODO: step3

- Plot the amplitude of the combined signal, abs(y) (SNR = 10dB)



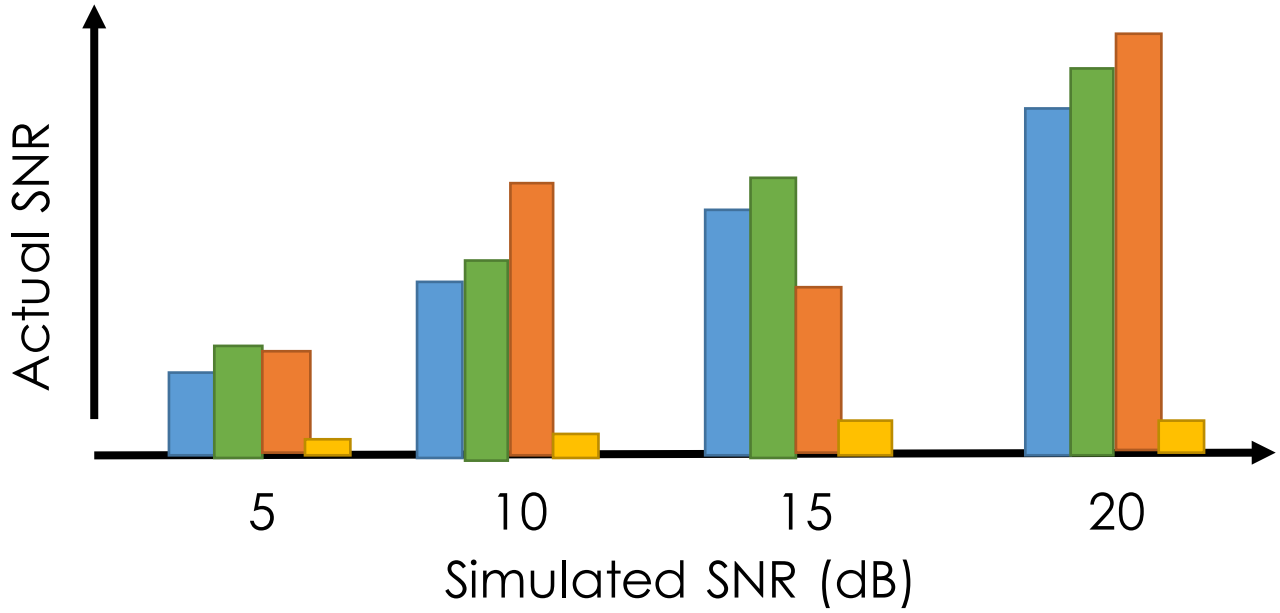
TODO: step4

- Plot the amplitude of the signal
  - Tx1 only
  - Tx2 only
  - Combined signal w/o precoding
  - Combined signal w/ precoding



TODO: step4

- Plot the SNR of the signal
  - Tx1 only
  - Tx2 only
  - Combined signal w/o precoding
  - Combined signal w/ precoding



Grading

- Generate two Tx streams: 20%
- Generate Channel and precoding: 30%
- Calculate decoded SNR (4 schemes): 20%
- Plot figures: 10%
- Report: 20%

Code Submission

- Deadline: May. 13 (Sun.) 23:59
- Submit to E3
    - source code: signal_gen.m, decode.m
    - Report (.pdf): include all figures along with your discussion