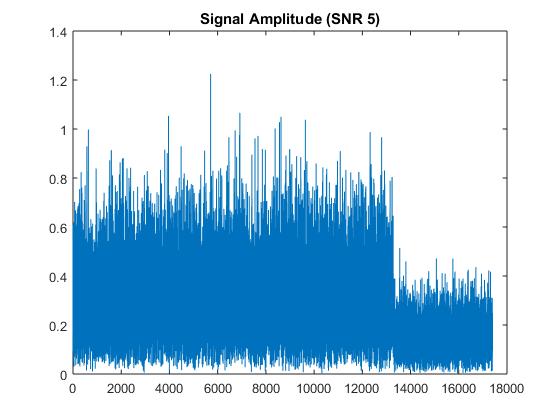
**Lab3 Interference Nulling**

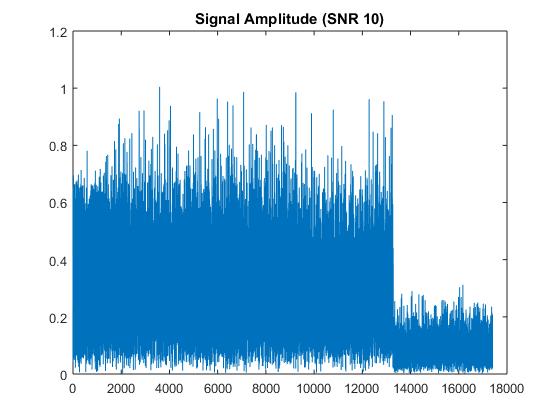
0656511 黃誠發

* Figures

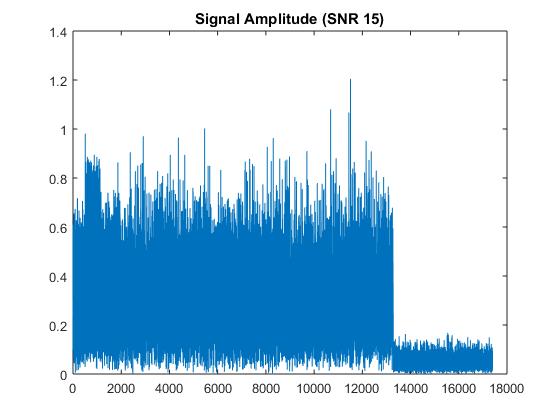
1. Amplitude of the signal
   1. SNR 5



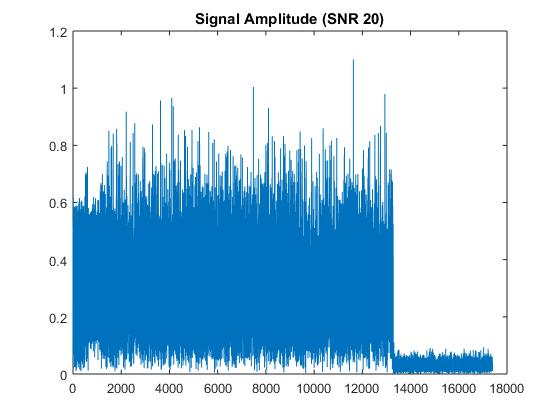
* 1. SNR 10



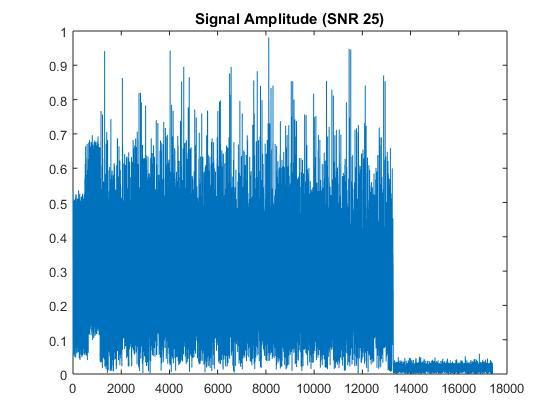
* 1. SNR 15



* 1. SNR 20



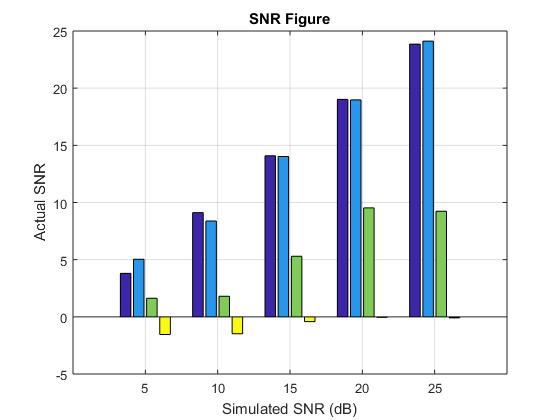
* 1. SNR 25



1. SNR of the signal
   1. The Actual SNR

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Simulated SNR | Tx1 only | Tx2 only | Combined w/o | Combined w/ |
| 5 | 3.8046 | 5.0348 | 1.6301 | -1.5457 |
| 10 | 9.1129 | 8.3794 | 1.8018 | -1.4786 |
| 15 | 14.0860 | 14.0277 | 5.3036 | -0.4151 |
| 20 | 19.0163 | 18.9786 | 9.5300 | -0.0429 |
| 25 | 23.8513 | 24.1189 | 9.2376 | -0.0890 |

* 1. Figure



* Code specification
  + signal\_gen.m

1. Generate two random channels:
   1. line 113 to 117
2. Generate w1 and w2 with normalization:
   1. line 121 to 126
3. Modified the sts and lts before the ifft:
   1. line 146 to 150
   2. line 157 to 161
4. Generate preamble1 and preamble2
   1. line 167 and 168
5. Generate the tx1 and tx2 and the transmitted signal
   1. line 223 to 319
   2. do the same as the lab 1 twice
6. Generate the signal with noise
   1. line 420 to 429
   * decode.m
7. Turn off the CFO and SFO parameters for simulation
8. Load the bin file we want to decode
   1. line 131
9. Channel estimation:
   1. line 181 to 241
   2. done the procedure twice for estimation for H1 and H2
   3. Note that the rx\_lts1\_f1 - rx\_lts2\_f2 must be 0
10. Payload processing
    1. line 247 to 324
    2. remove CP and take fft for the payload
    3. separate the four parts of the data according to the index we designed
11. Calculate the SNR for each part
    1. line 338 to 352
    * sim.m
12. call the signal generation with the parameter SNR
13. call the decode function assign the actual SNR back to the sim.m
14. plot all the amplitude of signal and the actual SNR for all situation and segments