# QAM

**Category**: “Can’t Stop the Signal, Mal”

Can’t Stop the Signal, Mal, 64 points

Decode the QAM symbols to get the flag. The transmissions begins with 01 23 45 67 89 AB CD EF

## Description

We are provided with a symbols file, which contains a signal encoded as a series of 8 byte IQ samples and can be loaded with:

samples = []  
with open('symbols', 'rb') as fin:  
 data = fin.read()  
 for i in range(0, len(data), 8):  
 x,y = struct.unpack('ff', data[i:i+8])  
 samples.append((x,y))

The second provided file is a GNURadio file, which loads the signal and can be used to see its constellation. Running it shows a 4x4 grid, so we can safely assume the signal is 16-QAM modulated, with each symbol encoding 4 bits.

## Solution

Once the encoding is known, the solution is just a matter of decoding each symbol to get the 4 bits it represents, concatenating all the bits and printing out the flag.

## Solution script

import struct  
  
samples = []  
with open('symbols', 'rb') as fin:  
 data = fin.read()  
 for i in range(0, len(data), 8):  
 x,y = struct.unpack('ff', data[i:i+8])  
 samples.append((x,y))  
  
val = 0  
blen = 0  
flag = []  
  
def decode\_component(x):  
 # Threshold values come from the constellation plot  
 if x < -2: return 0  
 if x < 0: return 1  
 if x < 2: return 2  
 return 3  
  
for (x,y) in samples:  
 a = decode\_component(x)  
 b = decode\_component(y)  
 v = a \* 4 + b  
  
 val = (val << 4) | v  
 blen += 4  
  
 if blen == 8:  
 flag.append(val)  
 blen = 0  
 val = 0  
  
print(bytes(flag))