ECE 405 - BASEBAND TRANSMISSION - INVESTIGATION 19
INTRODUCTION TO BASEBAND SIGNALS

FALL 2005 **A.P. FELZER**

To do "well" on this investigation you must not only get the right answers but must also do neat, complete and concise writeups that make obvious what each problem is, how you're solving the problem and what your answer is. You also need to include drawings of all circuits as well as appropriate graphs and tables.

Once we've sampled, quantized and compressed a message signal m(t) we need to decide how we're going to transmit the resulting PCM code over the communication **channel**. There are basically two alternatives. One way is to transmit pulses like the following

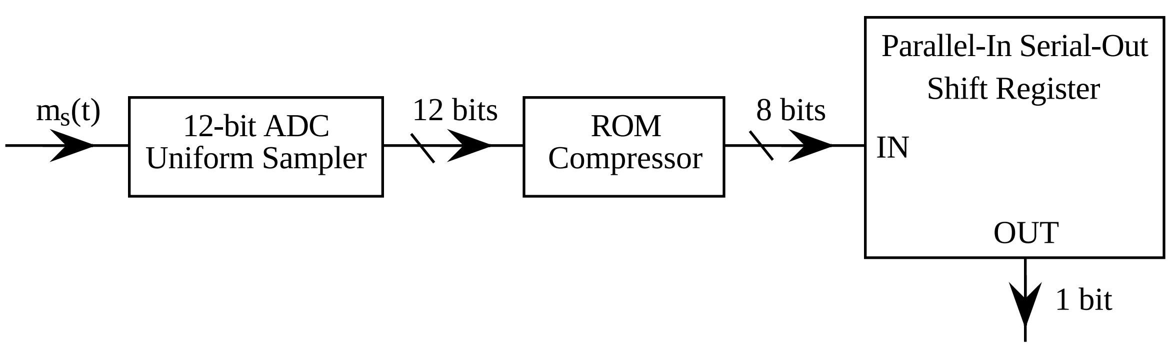
We call this **baseband transmission**. The alternative is to use the 1's and 0's to modulate the amplitude, frequency or phase of a sinusoidal carrier just like in analog communications as follows

We call this **passband transmission**.

The focus of this class is baseband transmission. The objective of this Investigation is to introduce some common ways to code 1's and 0's with pulses. Note that we will make use of the following terms in this Investigation:

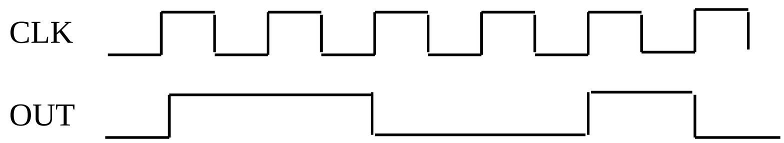
Mark = 1
Space = 0
Unipolar means the signal is always positive or always negative
Polar means the signal is both positive and negative

1. Continuing on from where we left off in the last Investigation let's suppose we load 8 bits from the output of the compression ROM into a parallel-in serial-out shift register as follows

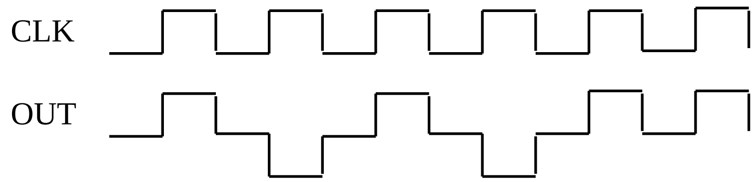


Draw a timing diagram for the serial output signal if the 8-bit signal loaded into the shift register is IN=1011101 and there's one shift per clock pulse. Draw your timing diagram with the MSB on the left. Assume all flip-flops are positive-edge triggered.

2. Baseband signals like those in Problem (1) as follows



are great but there are a number of other ways we can encode the 1's and 0's like the following polar code



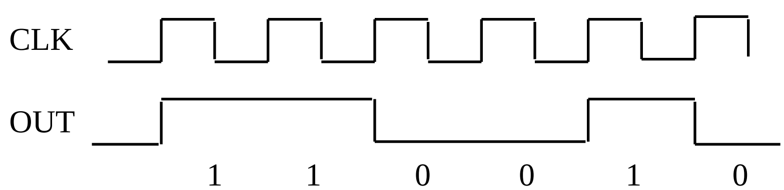
where a positive pulse is sent when the message bit is a 1 and a negative pulse when it's a 0.

a. Which of these two message signals has a smaller bandwidth
b. Which of these two signals is easier to keep synchronized. Why
c. What about the average powers of these two signals

3. The objective in this and the next several problems is to introduce some common encoding schemes. This problem introduces the **Nonreturn-to-Zero-Level (NRZ-L)** code as follows

Bit[n]	NRZ-L[n]
0	0
1	1

as illustrated in the following example for OUT=110010



Note that "nonreturn-to-zero" means that when the signal is one it stays one until the positive edge of the next clock pulse. Draw the timing diagram for the NRZ-L signal for OUT=10101101. Draw your timing diagram with the MSB on the left.

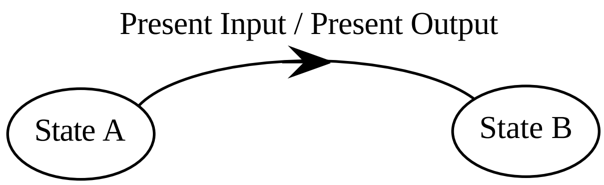
4. This problem introduces the **Nonreturn-to-Zero-Mark (NRZ-M)** code as follows

Bit[n]	NRZ-M[n]
0	(NRZ-M[n-1])
1	(NRZ-M[n-1])′

a. Describe in words the relationship between NRZ-M[n] and NRZ-M[n-1]
b. Find the NRZ-M signal corresponding to the NRZ-L signal 10101101. Assume all registers are initially cleared to 0
c. Draw the timing diagram for the NRZ-M signal in part (b). Assume that the output is initially cleared to 0 before the first clock pulse. Draw your timing diagram with the MSB on the left.
d. What's an advantage of NRZ-M over NRZ-L when there are a large number of 1's in

succession

e. Draw the state diagram for NRZ-M using the following format



Hint - make the state equal to the value of NRZ-M[n-1]

5. This problem introduces the **Nonreturn-to-Zero-Space (NRZ-S)** code as follows

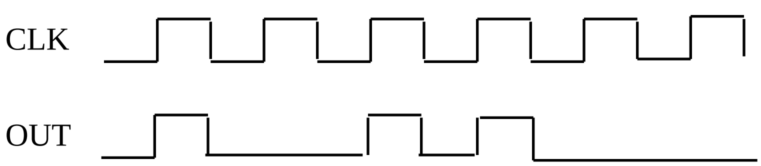
Bit[n]	NRZ-S [n]
0	(NRZ-S [n-1])′
1	(NRZ-S [n-1])

a. Describe in words the relationship between NRZ-S[n] and NRZ-S[n-1]
b. Find the NRZ-S signal corresponding to the NRZ-L signal 10101101
c. Draw the timing diagram for the NRZ-S signal in part (b). Assume that the output is initially cleared to 0 before the first clock pulse. Draw your timing diagram with the MSB on the left.
d. Draw the state diagram for NRZ-S with the state equal to the value of NRZ-S[n-1]

6. This problem introduces the **Return-to-Zero (RZ)** code as follows

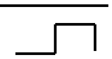
Bit[n]	RZ [n]
0	0
1	

with outputs like the following

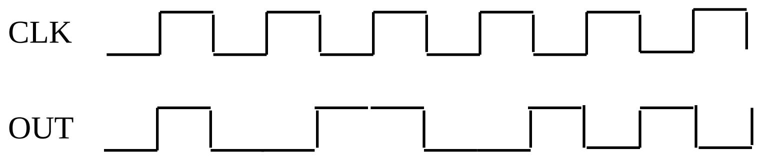


where the output for 1 returns to 0 in the middle of the clock pulse. Draw the timing diagram for the RZ signal for OUT=10101101. Draw your timing diagram with the MSB on the left.

7. This problem introduces the **Manchester** code as follows

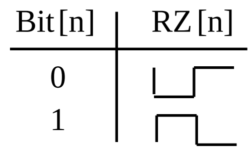
Bit[n]	RZ [n]
0	
1	

with outputs like the following

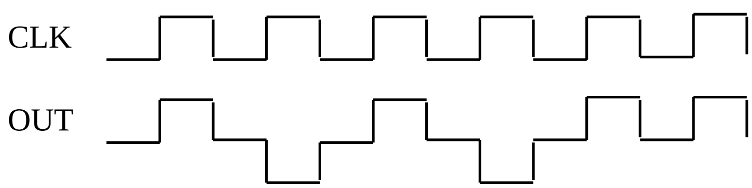


where the output for 1 returns to 0 in the middle of the clock pulse. Draw the timing diagram for the Manchester signal for OUT=10101101. Draw your timing diagram with the MSB on the left.

8. This problem introduces the **Bipolar** code as follows

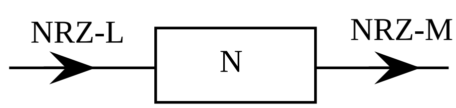


with outputs like the following



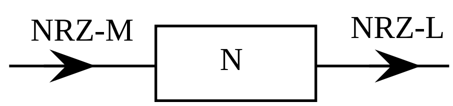
where the outputs return to 0 in the middle of the clock pulse. Draw the timing diagram for the Bipolar signal for the PCM code OUT=10101101. Draw your timing diagram with the MSB on the left.

9. The objective of this problem is to design a sequential circuit to generate NRZ-M from NRZ-L as follows



   a. Draw a state diagram for converting NRZ-L to NRZ-M. Hint - the state is NRZ-M[n-1]
   b. Make use of your result in part (a) to design a sequential circuit with D flip-flops for converting NRZ-L to NRZ-M
   c. Test your circuit in part (b) with the NRZ-L signal 11001101

10. The objective of this problem is to design a sequential circuit for recovering NRZ-L from NRZ-M as follows



   a. Draw a state diagram for converting NRZ-M to NRZ-L. Hint - the state is NRZ-M[n-1]
   b. Make use of your result in part (a) to design a sequential circuit with D flip-flops for converting NRZ-M to NRZ-L
   c. Make use of your results in Problem (9) to test your circuit