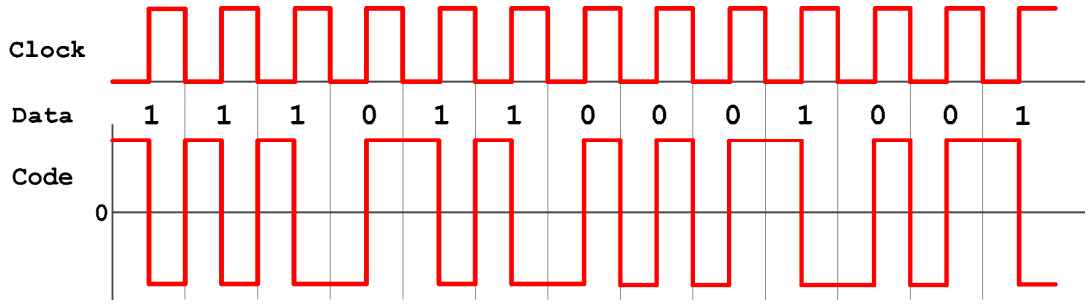
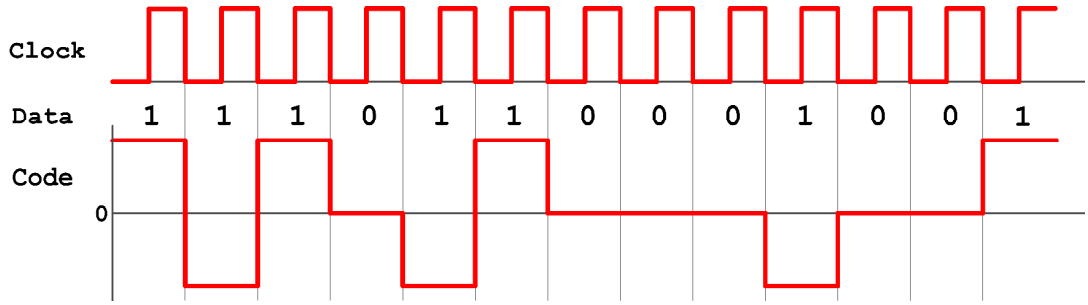
EE 528: Advanced Communications Lab
Prof. Mohammed Hawa

Project: Hardware for Line Coding Techniques

In this project, you are required to **understand, build and test** the following four circuits: Manchester code transmitter, Manchester code receiver, alternate mark inversion (AMI) line code transmitter and AMI code receiver.



Manchester Code

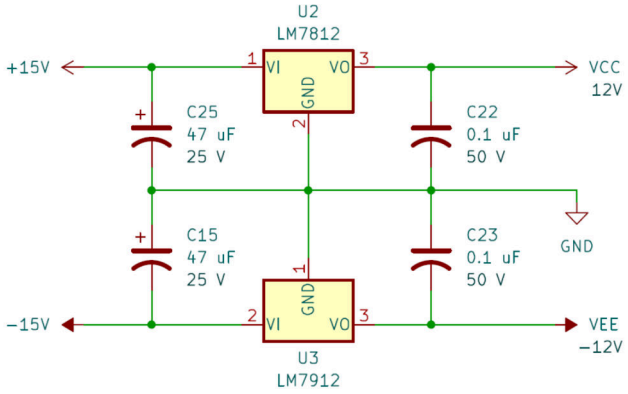


Bipolar (AMI) Code

For the Manchester transmitter/receiver circuits, use a single +5V DC supply, which sets the voltage to 0V and +5V. This is a simplification of the Manchester code, which is supposed to have positive and negative voltages, but this simplification allows you to use TTL-logic integrated circuits (ICs) to simplify your hardware.

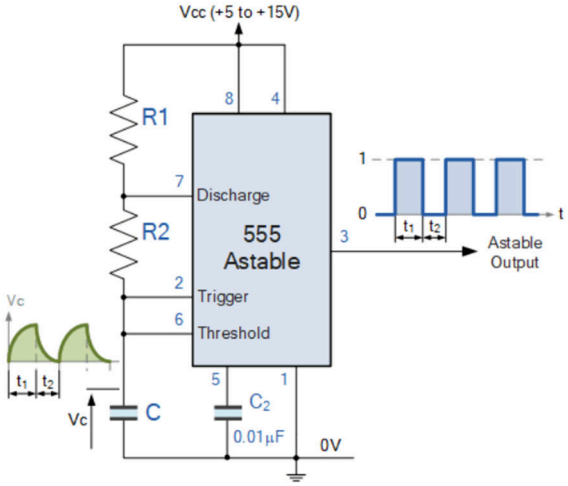
You can utilize a switching power supply unit from an old PC, which provides you with +12V, -12V, +5V, and 3.3V. If you do not have an old PC, you can find nowadays very cheap PC power supplies (their cost is about that of two 9V batteries). There are more expensive options, of course, but those are more efficient supplies that deliver high power suitable for gaming PCs. The circuits in this project require small power, and the cheapest power supply you can find should suffice. If you want to get both +5V and -5V from your supply, you can use the +12V and -12V output rails combined with two regulator circuits as shown below (use LM7805 and LM7905 voltage regulators). However, avoid using typical cheap power adopters in this project, because they are not well regulated.

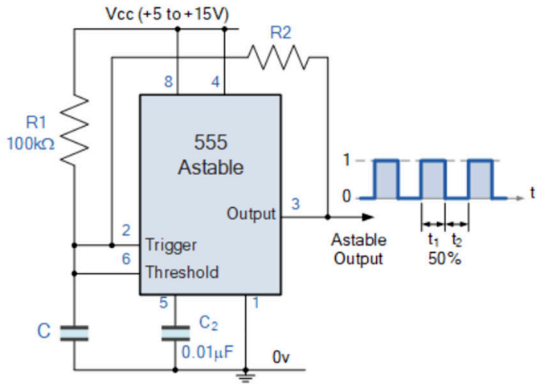
PC power supply unit



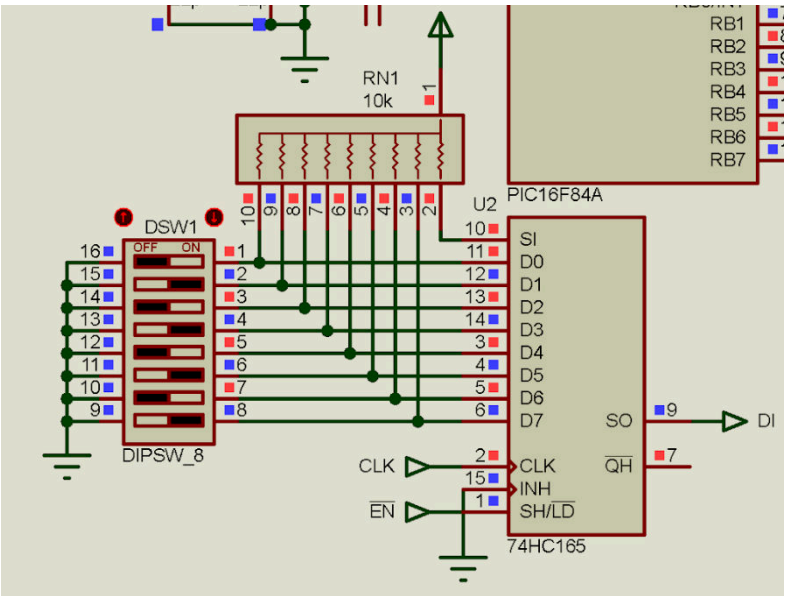
Voltage regulator circuit

The Manchester transmitter circuit should have a clock generator. You are required to use a 555 timer to generate a clock at f_0 = 10 kHz frequency, low voltage of 0 V and high voltage of 5 V, with 50% duty cycle. You will find different possible designs, such as the examples shown below. The first circuit shown provides a controllable duty cycle, while the second one delivers a fixed 50% cycle. You can use whichever of these designs, but make sure to calculate the proper R and C values to get 50% duty cycle and the desired frequency.

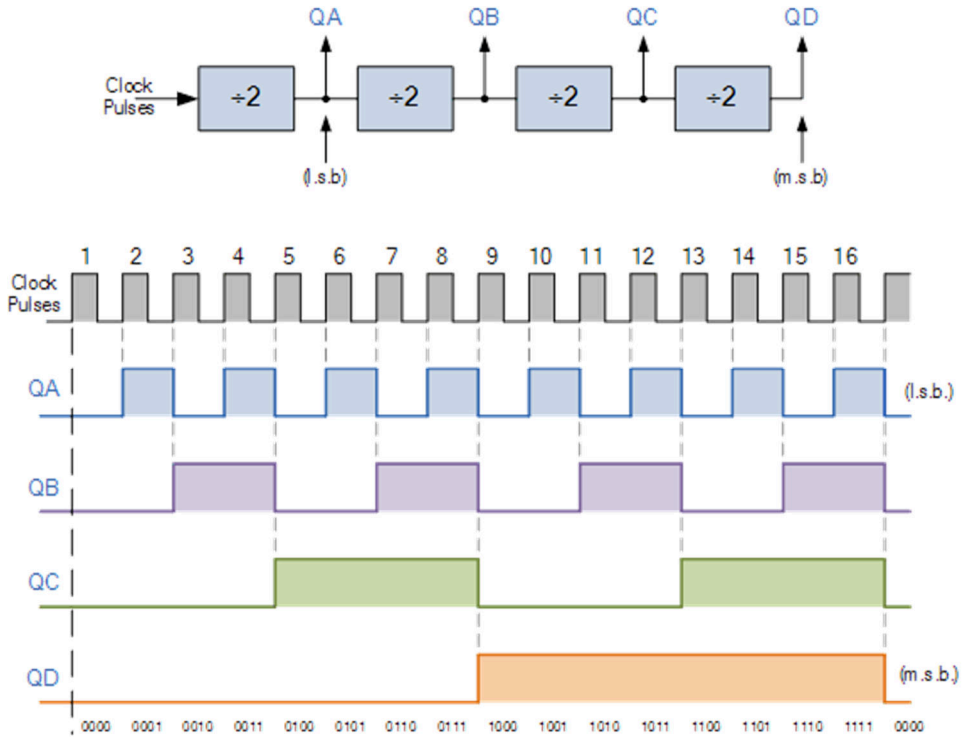




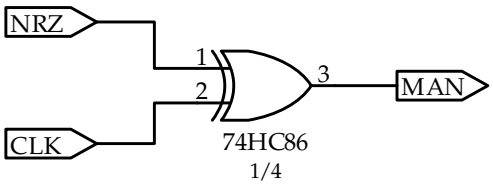
You also need to build a data bit generator, which produces a repeated 8-bit sequence, with one bit produced every one clock cycle. The sequence should be configurable using a DIP switch. You need to use the 74HC165 IC (8-bit parallel-load shift register). Here is an example showing how you can use one.



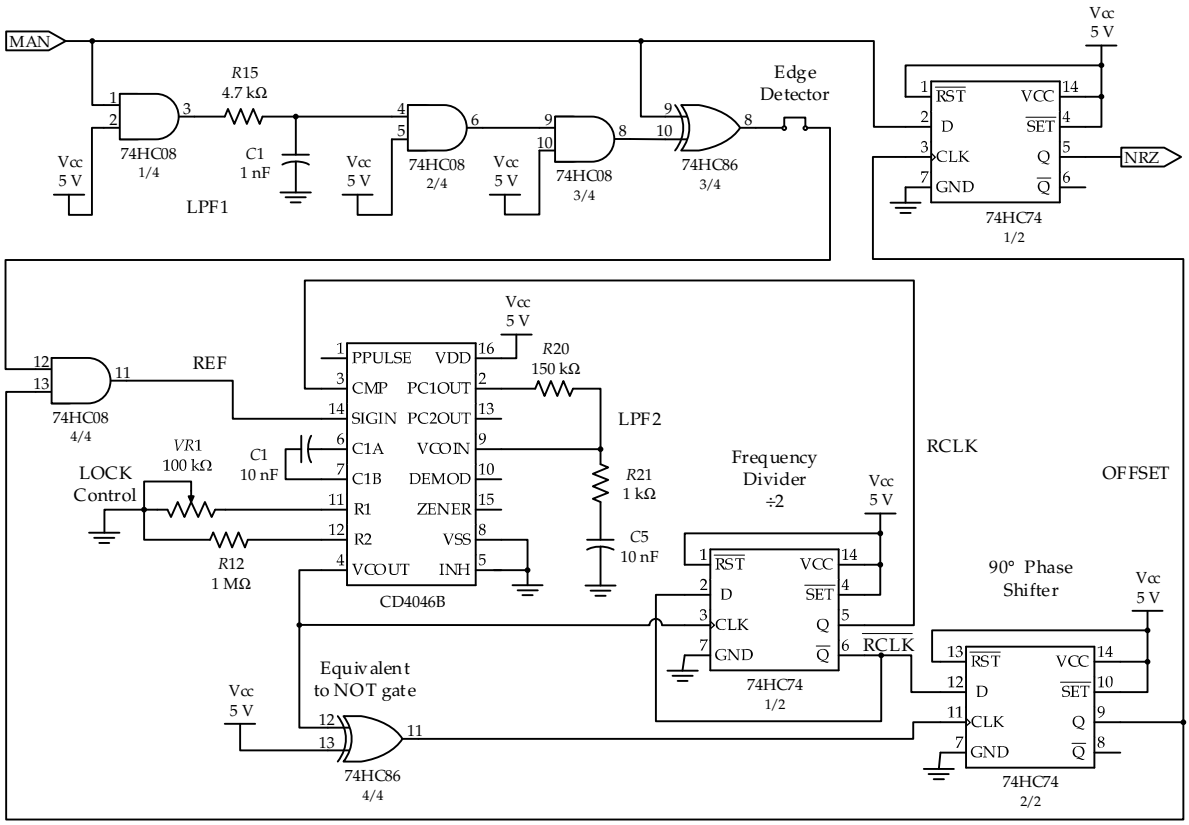
You will also need a SYNC signal to use as a TRIGGER source for the oscilloscope, in order to be able to see multiple bits on the oscilloscope screen at once. You can use the 74HC93 (4-bit binary ripple counter), which can divide the clock frequency by 8 or 16. An illustration of the concept is shown below.



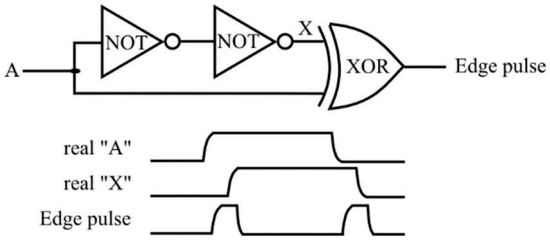
For the Manchester transmitter circuit, once you have the clock signal (CLK) and the data signal encoded as Unipolar NRZ, a simple XOR gate can be used to generate the Manchester signal. You can use the 74HC86, which includes four XOR gates.



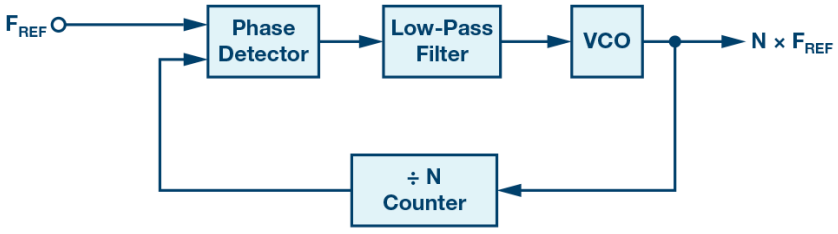
For the Manchester decoder, you need to use a PLL to first recover the clock from the received Manchester signal, then convert the Manchester code into a Unipolar NRZ code. Use the circuit shown below, which utilizes the CD4046B PLL IC, the 74HC08 (Quad 2-input AND gates), 74HC74 (Dual D flip-flops), and 74HC86 (Quad 2-input XOR gates). Do not forget to connect the power supply to energize each of these ICs.



The above circuit works as follows: The top three AND gates (connected as buffers), along with LPF 1 (first-order RC circuit) act as a delay line, slightly delaying the input Manchester signal. The delayed version is compared to the original Manchester signal via an XOR gate, which produces narrow pulses at the transition points of the code. Hence, the circuit acts as an *edge detector*. You can further tune the R and C values of LPF 1 to get the desired narrow pulse width.



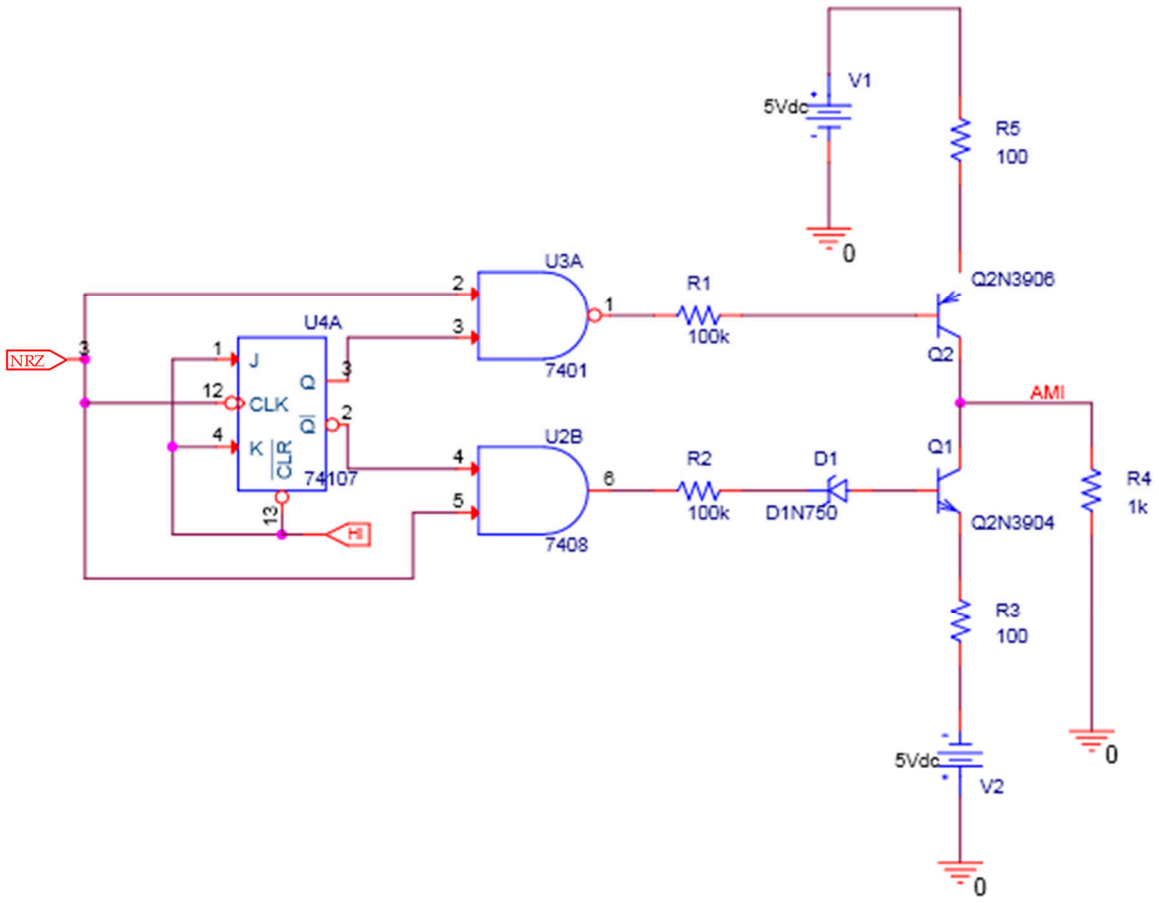
The next AND gate eliminates any pulses at the beginning of each bit time, and keeps those at the center of each bit time, producing the REF signal. This signal is fed to the 4046 phase-locked loop (PLL) circuit to read the clock information embedded within the Manchester code. Remember that a PLL consists of: phase detector, LPF, VCO and frequency divider. The 4046 chip provides the phase detector and VCO. You connect LPF 2 to complete the feedback loop, plus the design introduces the first D-type flip-flop as a frequency divider (dividing the frequency by 2).



Hence, VCO_OUT from the 4046 chip is double the frequency of the transmitter CLK, while the output of the frequency divider, called recovered clock (RCLK), is the same as the frequency of the transmitter. The LOCK control potentiometer adjusts the frequency range of the VCO, allowing the PLL to achieve lock even if you face some component tolerance issues.

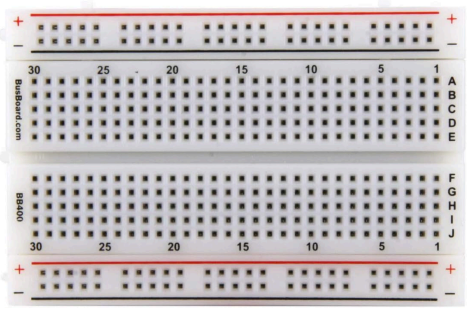
Notice that the recovered NRZ signal is produced by the top D-type flip-flop, with two inputs: The received Manchester code and the shifted recovered clock (OFFSET), which results in a phase shift in the recovered NRZ signal.

For the AMI transmitter circuit, we will use three voltage levels, +5V, 0V and -5V. Use the circuit shown below to get the AMI signal you need.

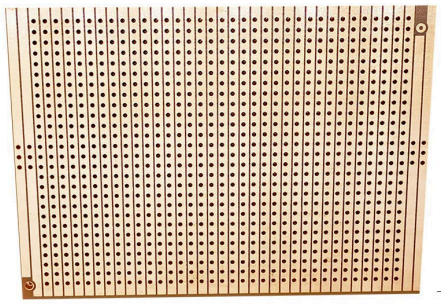


For the AMI receiver circuit, think about the block diagram that you require, then design and build the circuit.

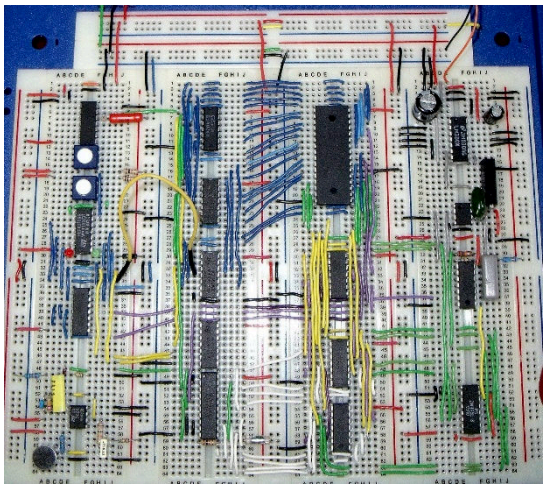
You need to build your circuits on top of a breadboard, so that you can re-use the hardware for both Manchester and AMI codes. You are not allowed to use a stripboard nor a PCB for this project. Make sure to build a neat circuit to simplify debugging of any issue that can arise during your work.



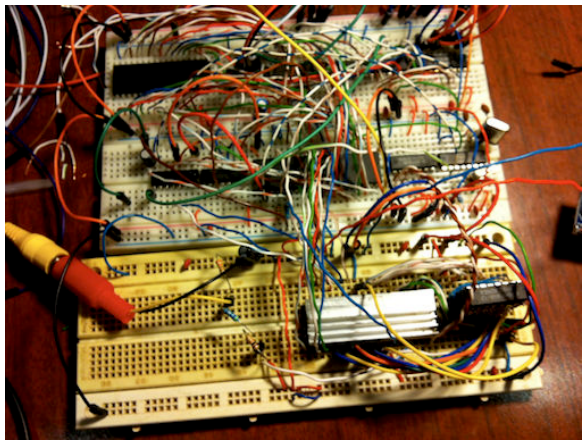
Breadboard



Stripboard



Good design (easy to debug)



Bad design