Moore Machines

Motivation

DFA's which are augmented with a output function which is a function of the current state are known as Moore Machines.

These have an initial state (also known as a power on or reset state) and may or may not have a final state (or acceptance) state.

Moore Machines may have additional ways of being combined (**composed**), with the output being fed to additional more machines as input (**sequential composition**), (when the output and inputs are compatible). It is also possible to construct systems of machines which have **feedback**, where part or all of the input of the systems is a function of the system's output.

Moore machines, **synchronized** to a clock, are the basis of many instruction set computers, as well as dedicated discrete processing systems.

Practical Example

One application of a Moore Machine is in telecommunications, and is used to modify the stream of symbols on the fly. Manchester encoding can be done simply with an XOR (***exclusive or***) logic gate Manchester decoding can be done using a clocked at a multiple (at least twice) the transmitter clock rate. On the receiver side, a low to high transition mid-cycle is recovered as a one, a high to low transition is recovered as a zero. We start with the line (data) idle, which means no mid-clock transition.

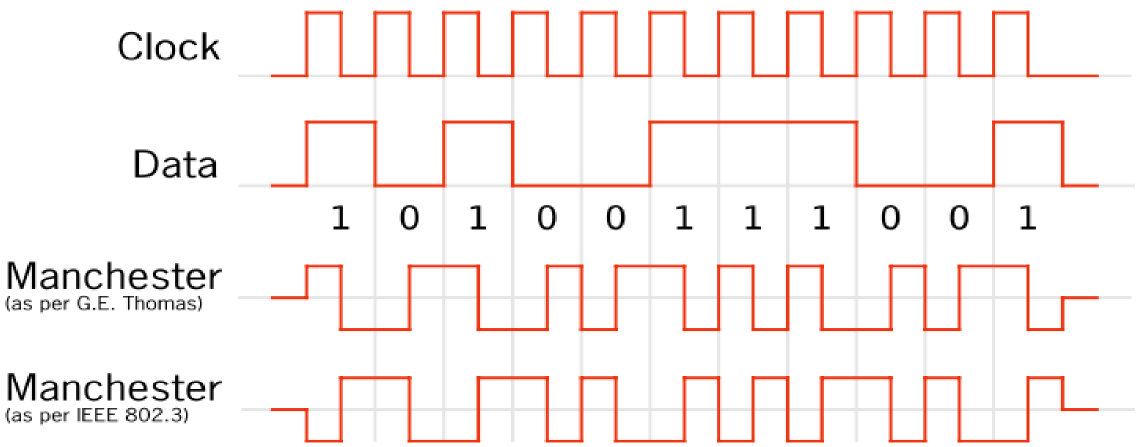

Illustration 1: http://en.wikipedia.org/wiki/Manchester_code

The figure shows the construction of a decoder which is clocked at twice the transmission clock rate. State ***q0*** is the reset, and idle state. Depending on the value of sensed on the line (either a 0 or zero), we transition to another idle state, where the system remains until there is a change of the input value. Remember, the system is being clocked at twice the transmission clocked rate, so any transition must occur at the mid point.

Questions

1) Examine state ***q1*** and the values used to exit the state. What form of Manchester encoding is being used?
2) What changes to the machine would need to be done to use the other form of Manchester encoding?

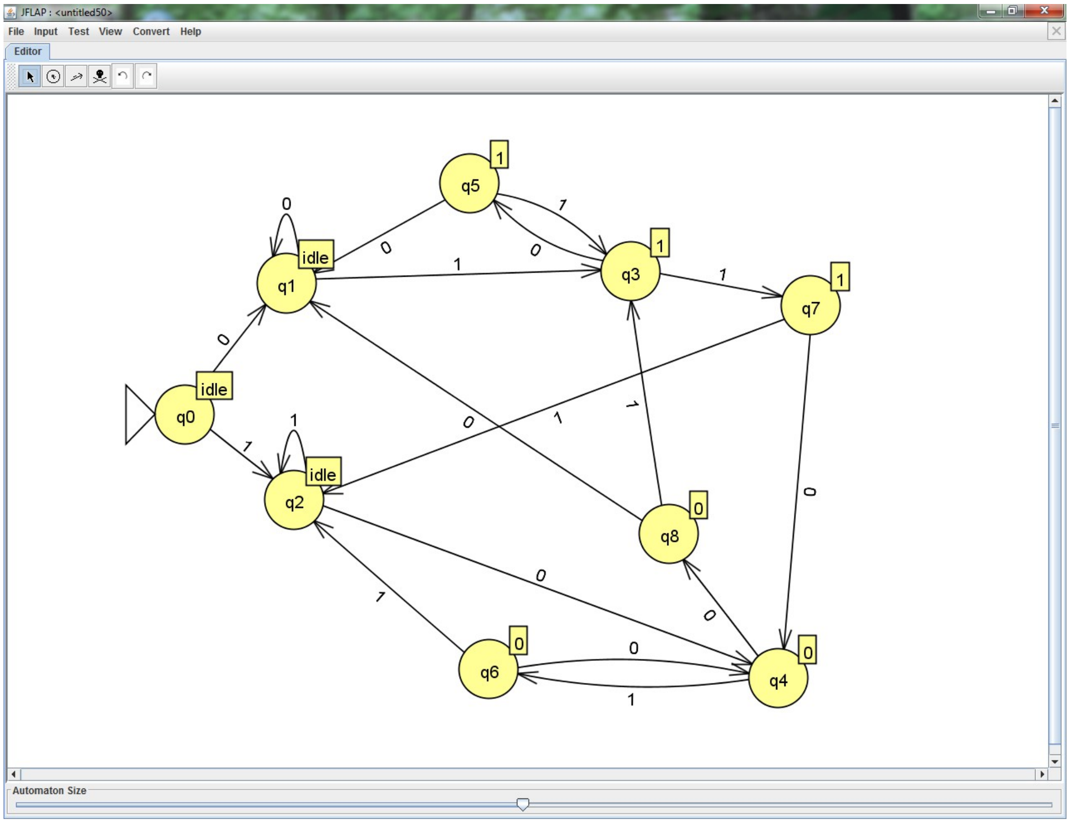

Illustration 2: A Manchester Decoder, Moore Machine

From state ***q1*** no change in the input means we are still idle. A change from a zero to a 1 means we have detected a 1 and we move to state ***q3*** which outputs a 1. If the input returns to a 0, the system moves to state ***q5*** since either the input indicates multiple ones being received or the line has returned to idle. If the input remains at 1, the system transitions to ***q7***, since either the input returned or a input indicates a zero following a 1.

The implementation assumes the transmitter brings the input to the receiver to the idle value of the first bit to be sent for 2 cycles.

Questions or Tasks

1) Test the machine on the input 000001011010011001? What is the output?
2) Test the machine on the input 1111100101011010? What it the output?

3) Bring the line to either a 0 or a 1 for two clock cycles, seems like a bit of a artificial requirement. Can you modify the design of the machine to eliminate this requirement? What if you increase the multiple of the clock rate to 4 instead of 2?