# Homework 4

1. Design a digital combination lock as a finite state machine that sets the output signal "unlock" to 1 after receiving a key, a specific serial binary pattern `101`. (Of course, for the lock to be practically useful, the binary pattern need to be much longer).

   Hint: This FSM should have one input to receive one bit in the binary sequence at a time, and one output which is 1 if the pattern 101 is detected. Use four states A, B, C and D to represent having 0, 1, 2 and 3 consecutive bits in the incoming bit sequence that match the key.

2. Design this FSM using D-FFs in the following steps:

   - Fill out the third row of the table below to show the state transition corresponding to the current input;

| Input | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Output | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| State | A | | | | | | | | | | | | | |

   - Draw the state diagram;
   - Complete the state transition table;
   - Implement the next-state decoder and the output decoder.

3. The advantage of Gray code over straight binary number is that Gray code changes by only 1 bit as it sequences from one number to the next. The 3-bit Gray-code representations for number 0 through 7 are listed below.

| Decimal | binary | Gray |
|---|---|---|
| 0 | 000 | 000 |
| 1 | 001 | 001 |
| 2 | 010 | 011 |
| 3 | 011 | 010 |
| 4 | 100 | 110 |
| 5 | 101 | 111 |
| 6 | 110 | 101 |
| 7 | 111 | 100 |

   Design a non-ripple, 3-bit, up-down, modulo-6, Gray-code counter using first D and then T flipflops. A modulo-6 counter counts from 0 to 5. (Compare this with a clock, for example, which is a modulo-12 counter for the hours and modulo-60 counter for the minutes and seconds.) In addition to the clock input, another input I is used to control whether the counter will count up (when I = `1`) or count down (when I = `0`). What you need to design is the combinational logic for the next-state decoder that generates the proper triggering signals for each flipflop when a new clock pulse comes. Complete your design by filling in the table on the next page.

| | Present State $Q_2$ $Q_1$ $Q_0$ | I | Next State $Q_2$ $Q_1$ $Q_0$ | $D_2$ | $D_1$ | $D_0$ | $T_2$ | $T_1$ | $T_0$ |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 0 0 | 0<br>1 | | | | | | | |
| 1 | 0 0 1 | 0<br>1 | | | | | | | |
| 2 | 0 1 1 | 0<br>1 | | | | | | | |
| 3 | 0 1 0 | 0<br>1 | | | | | | | |
| 4 | 1 1 0 | 0<br>1 | | | | | | | |
| 5 | 1 1 1 | 0<br>1 | | | | | | | |

4. Do the following number conversions:
   - Convert a base-7 number 123.45 into a decimal number.
   - Convert a decimal number 85.328125 into an octal number.
   - Convert a decimal number 204.75 into a hexadecimal number.
   - Convert a hexadecimal number AB.C into a decimal number.