**CS684 - Embedded Systems**

**Lab1 - FSM for Line Follower**

The bot has five white line sensors, thus 32 possible reading combinations. We will construct a Moore machine for the current problem. The output of the system is three-bit long, therefore 8 possible outputs and 8 possible states

Assumptions made:

* The end of the path gives a reading ‘11111’
* The state machine is run fast enough that readings between two samples cannot change drastically, say from ‘01000’ to ‘00010’
* On encountering a turn (reading of ‘00111’ or’11100’) taking the turn always leads to the end of the path (As in the example paths)

Some decisions:

* Whenever the reading is ‘00111’ or ‘11100’ we always turn right or left respectively, no matter if the turn is just a simple turn or a T-point
* With the assumptions made above, of the eight possible states, it is enough to use just four of them; which are forward (111), forward-right (110), forward-left (101), and stop (100)

State Transition Table

| State  Readings | 111  (forward) | 110  (forward-right) | 101  (forward-left) | 100  (stop) |
| --- | --- | --- | --- | --- |
| 00000 | 100 | 101 | 110 | 100 |
| 00001 | 110 | 110 | 110 | 100 |
| 00010 | 110 | 110 | 110 | 100 |
| 00011 | 110 | 110 | 110 | 100 |
| 00100 | 111 | 111 | 111 | 100 |
| 00101 | 111 | 110 | 101 | 100 |
| 00110 | 110 | 110 | 110 | 100 |
| 00111 | 110 | 110 | 101 | 100 |
| 01000 | 101 | 101 | 101 | 100 |
| 01001 | 111 | 110 | 101 | 100 |
| 01010 | 111 | 110 | 101 | 100 |
| 01011 | 111 | 110 | 101 | 100 |
| 01100 | 101 | 101 | 101 | 100 |
| 01101 | 111 | 110 | 101 | 100 |
| 01110 | 111 | 110 | 101 | 100 |
| 01111 | 110 | 110 | 101 | 100 |
| 10000 | 101 | 101 | 101 | 100 |
| 10001 | 111 | 110 | 101 | 100 |
| 10010 | 111 | 110 | 101 | 100 |

| 10011 | 111 | 110 | 101 | 100 |
| --- | --- | --- | --- | --- |
| 10100 | 111 | 110 | 101 | 100 |
| 10101 | 111 | 110 | 101 | 100 |
| 10110 | 111 | 110 | 101 | 100 |
| 10111 | 111 | 110 | 101 | 100 |
| 11000 | 101 | 101 | 101 | 100 |
| 11001 | 111 | 110 | 101 | 100 |
| 11010 | 111 | 110 | 101 | 100 |
| 11011 | 111 | 110 | 101 | 100 |
| 11100 | 101 | 110 | 101 | 100 |
| 11101 | 111 | 110 | 101 | 100 |
| 11110 | 101 | 110 | 101 | 100 |
| 11111 | 100 | 100 | 100 | 100 |

Some of the readings are not possible in reality(highlighted). For example, we can never read ‘00101’ in a normal setting. We might encounter them in a transition like we might read ‘00101’ while turning right from ‘00111’. Thus in such states, we will remain in the same state.

FSM for a General Path:

In the above FSM we made an assumption that whenever we encounter an intersection, taking the turn always is the right choice but this is not the case in a general setting.

In order to solve the issues of cross section we make following assumptions

* The end of the path gives a reading ‘10101’
* The state machine is run fast enough that readings between two samples cannot change drastically, say from ‘01000’ to ‘00010’
* The reading will be ‘00000’ only on reaching a dead-end

Some conventions

* On reaching a turn, always take the rightmost path(leftmost would also work). In means, if we get the reading ‘00111’ we take right turn(as usual), else if we ge a reading ‘11100’ we go straight.
* The second case is a little tricky as we might move forward even though it is just a turn and not an intersection. This can be solved by going to the state ‘backward’ as soon as we get reading ‘00000’
* When in ‘backward’ state, if the reading is ‘11111’, always take left. Since this situation arises only when we have taken a left at an intersection and it has turned out to be a wrong decision
* When in ‘backward’ state, if the reading is ‘00111’ or ‘11100’ take ‘backward-right’ and ‘backward-left’ turn respectively. Since this situation arises either when we went straight at an intersection and it turbned out to be wrong or when we have a turn in a path leading to a dead-end
* In ‘forward’ state, if the reading is ‘11111’ take right, since this can either as an intersection in which case we are taking the right most path or when the right turn taken turned out to be false but it itself had a turn before dead-end

With the above conventions we can build a FSM for a general line follower with seven states which are, forward, forward-right, forward-left, backward, backward-right, backward-left, stop

State Transition Table

| state  reading | 111  fwd | 110  fwd-right | 101  fwd-left | 011  bwd | 001  bwd-right | 010  bwd-left | 100  stop |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 00000 | 011 | 101 | 110 | 111 | 010 | 001 | 100 |
| 00001 | 110 | 110 | 110 | 001 | 001 | 001 | 100 |
| 00010 | 110 | 110 | 110 | 001 | 001 | 001 | 100 |
| 00011 | 110 | 110 | 110 | 001 | 001 | 001 | 100 |
| 00100 | 111 | 111 | 111 | 011 | 011 | 011 | 100 |
| 00101 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 00110 | 110 | 110 | 110 | 001 | 001 | 001 | 100 |
| 00111 | 110 | 110 | 101 | 001 | 001 | 001 | 100 |
| 01000 | 101 | 101 | 101 | 010 | 101 | 101 | 100 |
| 01001 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 01010 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 01011 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 01100 | 101 | 101 | 101 | 010 | 010 | 010 | 100 |
| 01101 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 01110 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 01111 | 110 | 110 | 101 | 001 | 001 | 010 | 100 |
| 10000 | 101 | 101 | 101 | 010 | 010 | 010 | 100 |
| 10001 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 10010 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |

| 10011 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 10100 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 10101 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| 10110 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 10111 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 11000 | 111 | 101 | 101 | 010 | 010 | 010 | 100 |
| 11001 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 11010 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 11011 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 11100 | 111 | 110 | 101 | 010 | 001 | 010 | 100 |
| 11101 | 111 | 110 | 101 | 011 | 001 | 010 | 100 |
| 11110 | 111 | 110 | 101 | 010 | 001 | 010 | 100 |
| 11111 | 110 | 110 | 110 | 101 | 101 | 101 | 100 |

Some of the readings are not possible in reality(highlighted). For example, we can never read ‘00101’ in a normal setting. We might encounter them in a transition like we might read ‘00101’ while turning right from ‘00111’. Thus in such states, we will remain in the same state