University of Calgary

Fall 2019

CPSC 359

Assignment 1:

Combibot: a sumobot strategy in combinatinal logic

Minh Hang Chu

30074056

1. Identify all the inputs and outputs for your circuit. This step is fairly obvious, but it is a good habbit to write these down anyway.

Inputs:

- M_0 senses if the edge of the arena is in the front of the combibot. Returns 0 is the combibot is inside the arena, returns 1 is the front of the combibot is at the edge of the arena.
- M_1 senses if the edge of the arena is in the back of the combibot. Returns 0 is the combibot is inside the arena, returns 1 is the back of the combibot is at the edge of the arena.
- E_0 senses if the opponent of the arena is to the left of the combibot. Returns 0 is no opponent detected, returns 1 is the opponent is on the left side.
- E_1 senses if the opponent of the arena is to the right of the combibot. Returns 0 is no opponent detected, returns 1 is the opponent is on the right side.

Outputs:

- L_0 receives input, the left wheel moves forward.
- L_1 receives input, the left wheel moves backward.
- R_0 receives input, the right wheel moves forward.
- R_1 receives input, the right wheel moves backward.

If L_0 and L_1 (R_0 and R_1) don't receive any inputs the combibot stops. If both receives inputs, then this is not allowed.

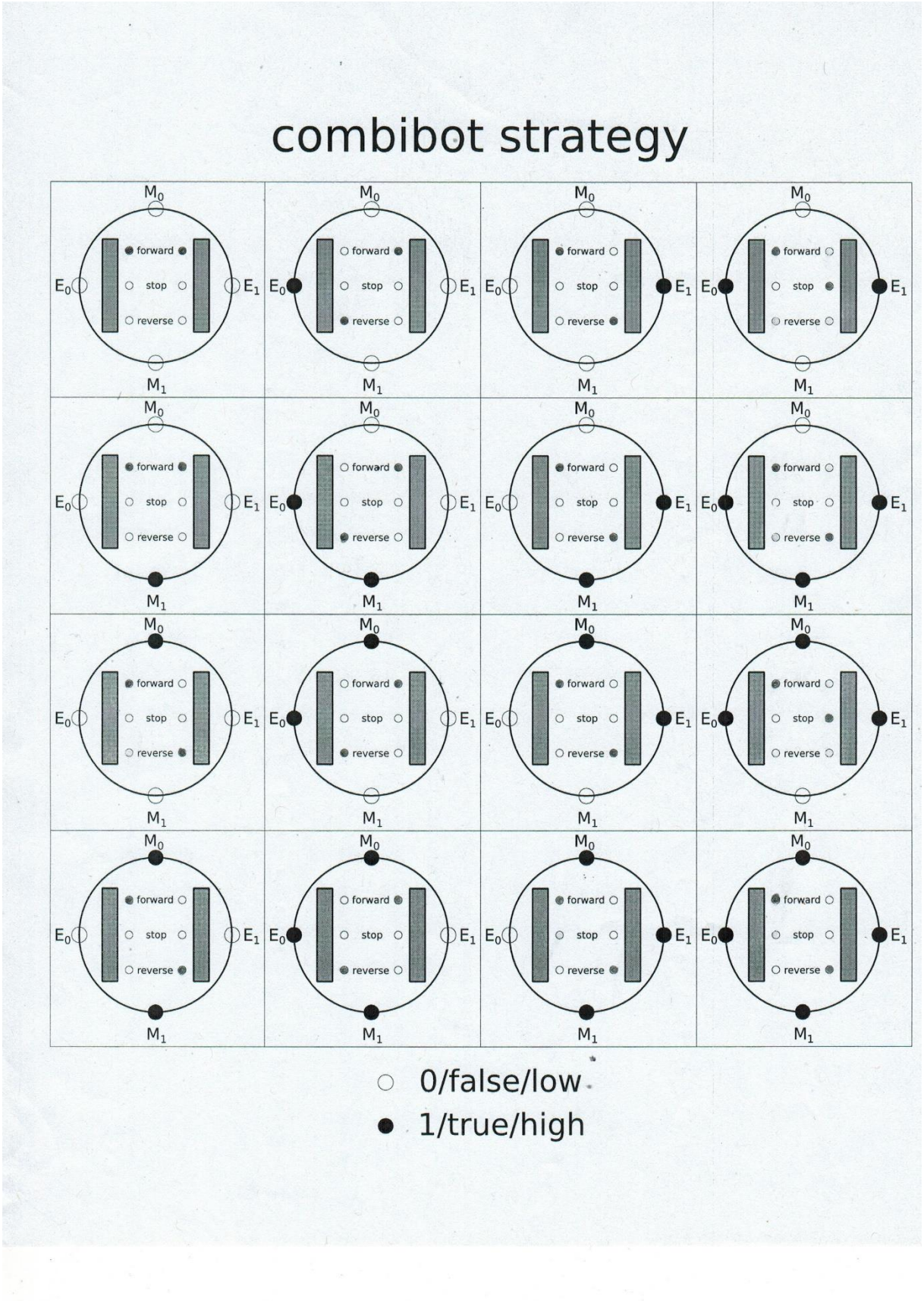
2. Use the combibot strategy sheet provided to develop your strategy for combinational control of combibot. You are free to create your own strategy, but it must be reasonable. I.e., your strategy should look like it could possibly win a Robo-sumo competition against a weak opponent. Strategies that the marker deems overly simple (in order to trivialize the circuit design) will receive low grades.

Strategy: The strategy to win sumo match is identifying enemy, turning to their direction. At that time, the enemy will be in the front, and the combibot can push the enemy outside the arena.

There are some cases:

- When there is no edges or enemies around, the combibot moves forward.
- When the combibot senses enemies on the left side, it should turn left. When the combibot senses enemies on the right side, it should turn right. Turning helps the combibot facing the enemies and can push enemies outside.
- When the combibot senses the edges of arena, it should change direction so it can keep moving and facing enemies.

See Combibot Strategy next page.


combibot strategy
M0
M1
E0
E1
forward
stop
reverse
0/false/low
1/true/high

3. Design the combinational circuits to implement your strategy

Truth table

M_0	M_1	E_0	E_1	L_0	L_1	R_0	R_1
0	0	0	0	1	0	1	0
0	0	0	1	1	0	0	1
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	1	0	1	0
0	1	0	1	1	0	0	1
0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	1
1	0	0	1	1	0	0	1
1	0	1	0	0	1	1	0
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1
1	1	0	1	1	0	0	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

Karnaugh maps and functions in minimized disjunctive normal form

$F_{L_0} = E_0' + E_0E_1$

M_oM_1 \ E_oE_1	00	01	11	10
00	1	1	1	0
01	1	1	1	0
11	1	1	1	0
10	1	1	1	0

$F_{L_1} = E_0E_1'$

M_oM_1 \ E_oE_1	00	01	11	10
00	0	0	0	1
01	0	0	0	1
11	0	0	0	1
10	0	0	0	1

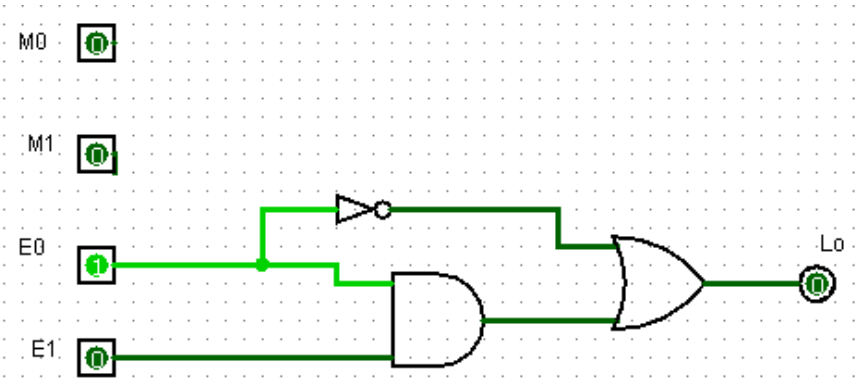
$F_{R_0} = E_0E_1' + E_1M_0'$

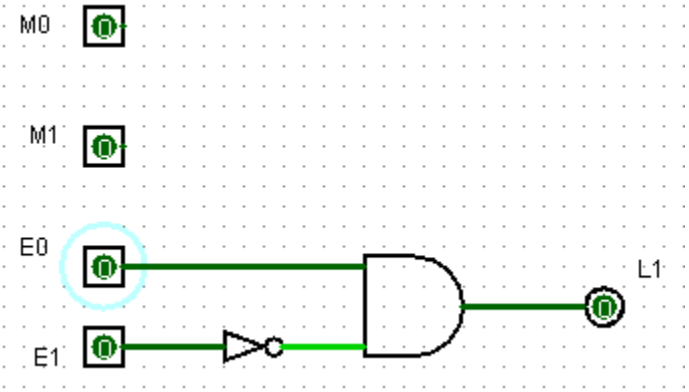
M_oM_1 \ E_oE_1	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	0	0	1
10	0	0	0	1

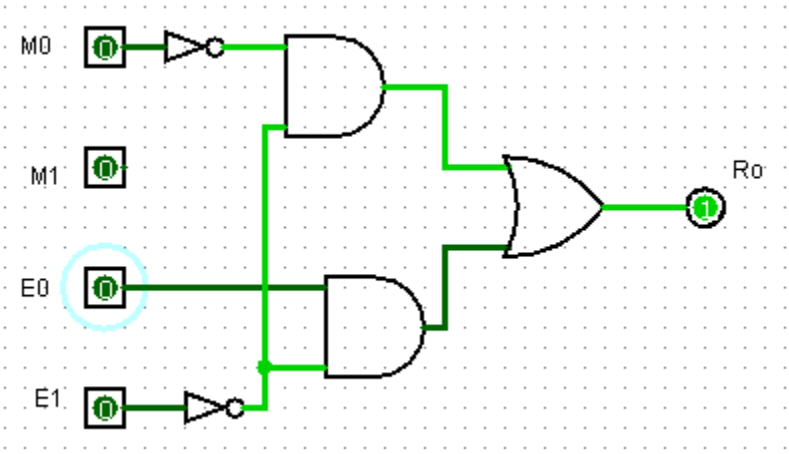
$$F_{R_1} = E_0'E_1 + E_1M_1 + E_0'M_0$$

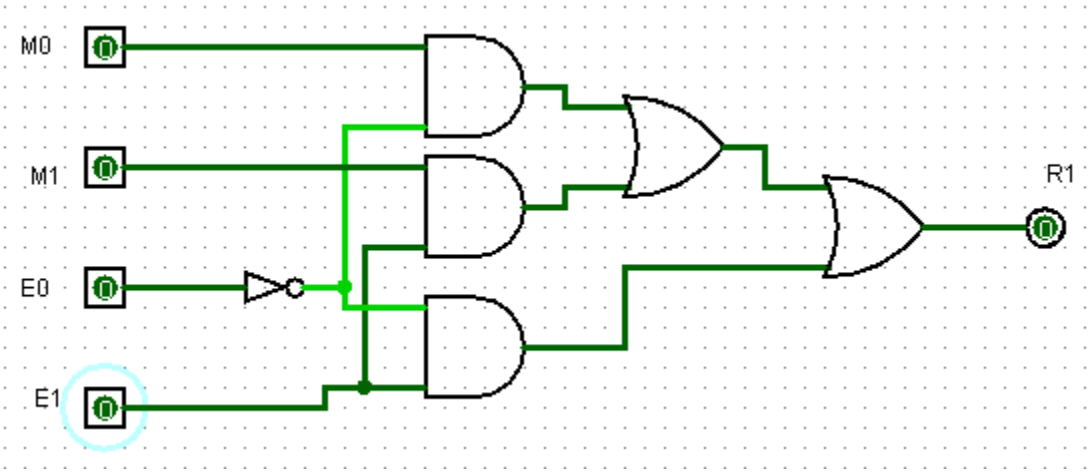
M_oM_1 \ E_oE_1	00	01	11	10
00	0	1	0	0
01	0	1	1	0
11	1	1	1	0
10	1	1	0	0

Logic Diagrams









4. Implement your design in Logisim using the provided template

See a1_template.circ attached.