



---

# CUP & BALL GAME

---

Design Manual

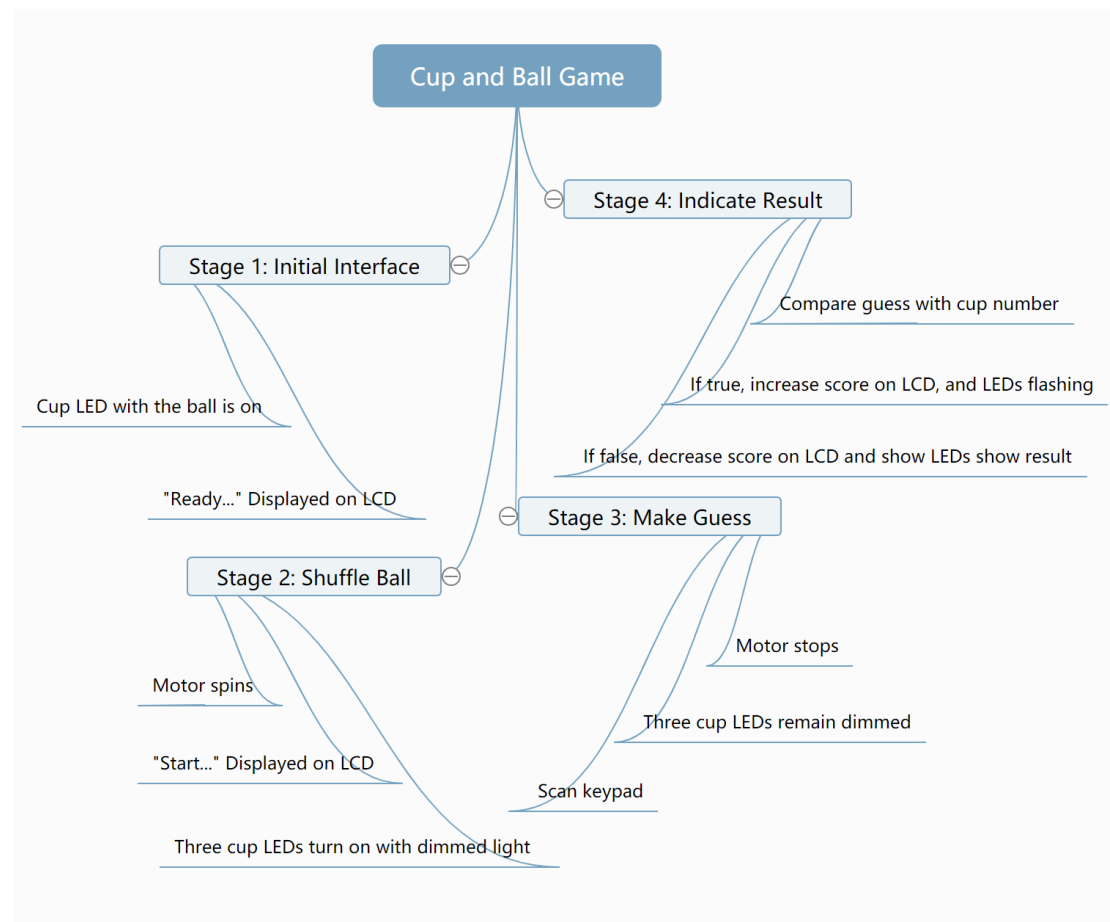


COMP 9032 FINAL PROJECT

YU HAN Z5219071

2018.10.23

## Flow



## Register & Address

All registers used in this project are defined as following:

Label	Register	Comment
Temp	R16	temporary register for various ops
Temp1	R17	temporary register for various ops
Cup	R18	Cup number
Lcd	R19	Lcd handle
Pattern	R20	Register for pattern display
Score	R21	The score display on LCD
Sign1	R22	Sign for control motor
Sign2	R23	Sign for control LED & LCD
Row	R24	Current row number
Col	R25	Current column number
Rmask	R29	Mask for current row during scan
Cmask	R30	Mask for current column during scan

The addresses used in this project were labelled as following:

Label	Address	Comment
PORTLDIR	0xF0	PL7-4: output, PL3-0, input
INITCOLMASK	0xEF	scan from the rightmost column
INITROWMASK	0x01	scan from the top row
ROWMASK	0x0F	for obtaining input from Port D
LCD_RS	7	RS pin for LCD
LCD_E	6	E pin for LCD
LCD_RW	5	RW pin for LCD
LCD_BE	4	BE pin for LCD

## Macros

All macros used in this project is for LCD commands and functions.

## Initialization

1. **LCD:** Set Port F and Port A for LCD. Do an ASCII change to decimal command to display the score.
2. **Keypad:** Set port L for keypad. Only scan the first three column of first row in this project, since there are only three cups. Once the guess pressed one of the three buttons, stop scan keypad and jump to next stage.
3. **Motor:** Set Port E for motor. Start spinning when sign 1 is 1 and stop when sign 1 is 0.
4. **LED:** Set Port C as output to display patterns on LED. Used to indicate the ball location and result of player's guess.
5. **Interrupt:** Set INT 0 and INT 1 as falling edge sensed interrupts. Set sign 2 to change stages. The sign flag will change once raise the interrupt. Sign 2 was cleared in the beginning, which means that motor is stationary and game start from stage 1. Once raise the interrupt 0, the sign 2 will changed to 1 and the game will start stage 2. In stage 2, there is a compare command to monitor

sign 2. Once sign 2 changed to 0, game will jump to stage 3.

6. **Debounce:** Used delay and change the interrupt status to debounce.

### Stage 1: Initial Interface

1. Set LCD commands at first. Display word string "Ready... Score:" on LCD and display the current score number on LCD.
2. Set LED patter equals to cup number, which means if ball was in first cup in last game, the LED is going to turn on the first light to show that the ball is here.
3. Compare sign 2 with 1, once sign 2 equals 1, jump to next stage.

### Stage 2: Shuffle Ball

1. Load 1 to sign 1 on Port E, which means turn on the motor and let it spin.
2. Change the word on LCD to "Start... Score:" and display the current score.
3. Increase the cup number from 1 to 3 to simulate shuffling the ball. Once the cup number is equal to 3, jump back to 1 and increase again.
4. Used 'nop' to delay the time to turn on LEDs with dimmed light. Turn on the LEDs and turn off immediately and jump back this loop. Normally we see the LEDs stay lit because it is too fast to see turn off for us. But if we add some 'nop' to delay the turn off period, then we can see the LEDs turn on with dimmed light.
5. Compare sign 2 with 0, once sign 2 equals 0, jump to next stage.

### Stage 3: Scan Keypad, Make a Guess

1. Load 0 to sign 1 on Port E, which means turn off the motor and let it stop.
2. The ball stops shuffle, so there is no more increase cup number command in this stage.
3. Since there is nothing changed for LCD part, we do nothing about it.
4. Similar like stage 2, also used same method to show the dimmed light. But there are too many gaps between each loop, we will see the LED turned off if there is only one command for turn on the LEDs. Therefore, we need to add more turn on and turn off commands during this long loop.
5. Scan the first three buttons in first row of the keypad since we only need 1-3 to represent the cup numbers. Once we scanned the input from player, stop scan and jump to next stage.

### Stage 4: Compare and Indicate Result

1. Record the column number and increase by 1 to indicate the guess number and stored in temp 1.
2. Compare the guess number (temp 1) with the result (cup) and refresh new score of LCD based on the result. If the score is less or equal than zero after decrease, the game will automatically jump to restart.
3. If the guess is correct, the LEDs will be flashing and show the result; if the guess is wrong, the LEDs won't flashing but will show the result. Used sleep function to delay in order to achieve flashing purpose.
4. Compare sign 2 with 2 in this result indicate loop, once the player pressed PB1

button, sign 2 will change to 2 and the game will jump to stage 1 to start a new game, and the score will be remained.