

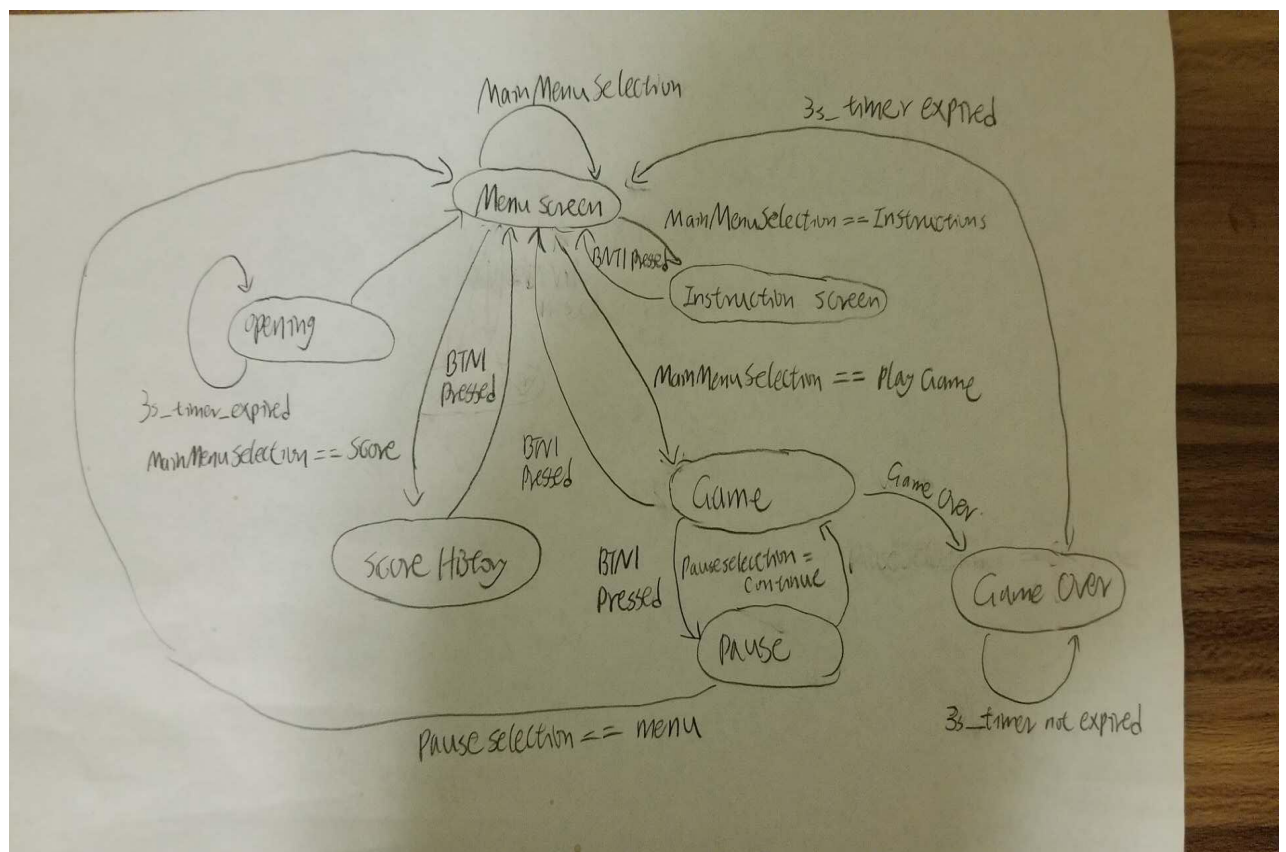
ECE2534 Lab3 Report

Section1:

In this Lab, each student is required to design a car game on the microcontroller. This car game has seven panel or screens. The first one is a opening screen and it will display for three seconds. Then the game goes to the main menu screen. There are three options on the menu panel which are "Play Game" , "How to Play", "Score History". When the user click the "Play Game" function. The game will start. On the game screen, the left side is the road and the car is at the bottom and the right side is the score information. When users click the upper left button, the game will be paused and user has two options, which are continue and menu. After the game ends, the screen will shows a game over message and display for three seconds, then return to the main menu. On the main menu, when the user clicks "How to play" option, an instruction for this game will be displayed and when the user push the upper button, the screen will return to the main menu. The "Score History" option is simply a panel to show the most three recent scores. My application could accomplish all the functions lists above.

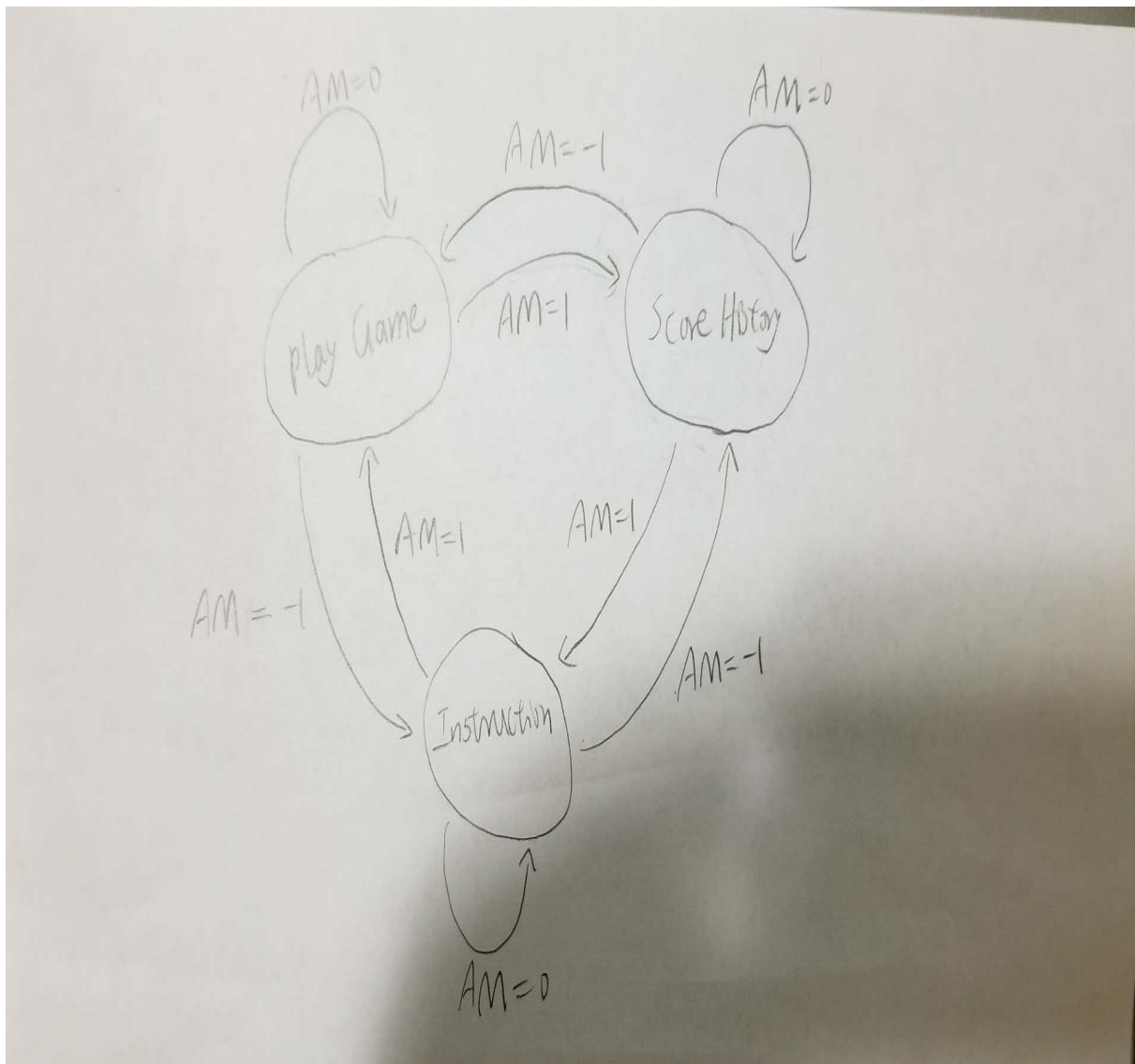
Section 2:

Main FSM:



FSM input	Description
3s_timer_expired	A boolean that is true when the 3s has passed
MainMenuSelection	An integer that represents the main menu selection
BTN1Pressed	A boolean that is true when the Button 1 is pressed
PauseSelection	An Integer that represents the pause selection, in which include menu and continue)

Menu FSM :



Second 3

HAL Function :

```
void LCDClearDisplay(int color)
//clear the LCD screen
void LCDDrawChar(unsigned row, unsigned col, int8_t c)
//draw the char on the LCD screen
void PrintString(char *str, int row, int col)
//print a string on the LCD screen
void InitGraphics()
//initial the graphics
bool Booster_Top_Button_Pushed()
//check if the top button is pushed
void InitButtons()
//initial all the buttons
void InitLEDs()
//initial all the LEDs
void InitOneShotSWTimer(OneShotSWTimer_t* OST,
                        uint32_t hwtimer,
                        uint32_t waitCycles);
//initial a one shot SW Timer
void StartOneShotSWTimer(OneShotSWTimer_t* OST);
//start the one shot sw timer
bool OneShotSWTimerExpired(OneShotSWTimer_t* OST);
//check if the one shot timer is expired
void initJoyStick()
//initial the joystick
void getSampleJoyStick(unsigned *X, unsigned *Y)
// get the position of the joy stick
```