Mini-Golf Game Plan

Overview: Mini-golf is a game modeled after real golf. A player (or players) will hit a stationary ball across a game field to make it into the hole. The game counts the number of strokes a player takes per level/map; the lower the strokes, the better the score.

Basic Rules:

1. Players must use the slide pot(s) to control the direction and power of the ball. Once they get their desired direction and power, they will push a button to swing.
2. Players take a maximum of 10 strokes to get the ball into the hole.
3. If the ball goes into water or out of bounds, the player gets a +1 stroke penalty and starts his next shot at the same place.
4. If the ball goes into sand, the ball moves slower while in the sand. Once it gets out, it changes back to the normal velocity.

Basic Features:

1. Levels – Each hole is a level. For sake of brevity, we will only design a maximum of 9 holes. Levels will increase in difficulty: more bunkers, more water, different obstacles, narrower maps, weird/crazy maps
2. Score Display – The current state of the game will be displayed on the screen
   1. Level #, stroke #, par #, options and menu buttons
   2. Start screen
   3. Scorecard displayed after every hole
3. Music/Sound effects control
4. Multiplayer – Up to 2 players can play

Potential Additional Features:

1. Power-ups/power-downs – Ball changes size; stroke penalty not assessed if ball hit out of bounds; speed/power boost; “cheat actions” for holes-in-one on each level
2. Players will be able to customize characters or golf balls or clubs
3. Another knob potentiometer for controlling direction

Difficulty:

1. Physics engine: Must be able to react to players’ shots and account for water and sand traps
2. Control screen speed and frame rate

Basic Hardware:

1. Microcontroller
2. LCD Display
3. Slide potentiometer – Controls swing power
4. Knob potentiometer – Controls swing direction
5. Button 1 – Main button for gameplay. “Select” on menus, “Swing” during gameplay
6. Button 2 – Secondary button. “Page up” on menus, “Options” in gameplay
7. Button 3 – Secondary button. “Page down” on menus, “Quit” in gameplay
8. DAC and audio jack – Headphones/music

Advanced Hardware:

1. Distance/Proximity Sensor (HC-SR04) – Used to put system in “sleep mode” until user picks up board.
2. Another set of Basic Hardware – Implement multi-player

Basic Software:

1. Game engine – main loop
   1. Giant FSM
   2. On start screen, should have sprite flashing to signal “select”
2. Physics engine – Calculates trajectories of ball
3. ADC for potentiometers
4. Several timers for sound effects – SysTick to generate them
5. ST7735 files for LCD
6. GPIO-A for LCD, GPIO-B for DAC, GPIO-D for button inputs, GPIO-E for ADC inputs, GPIO-F for heartbeat
7. Edge-triggered interrupts (with debouncing) on buttons

Advanced Software:

1. UART communication – GPIO-C
   1. Distributed computing solution – Send “your turn” code to microcontroller and poll to receive “your turn”.
   2. Full-duplex system – Try for 1 screen if possible
2. Ultrasonic Sensor – GPIO-E
   1. Requires GPIO ISR to measure echo time
   2. Calculates distance

Game Engine:

FSM control: Start screen, game play, level select, help, high scores, scoreboard

1. Initialize the game modules
2. Initialize FSM to start screen output
3. Go through FSM – switch statement has code for each state

Physics Engine:

1. Ball hits wall: Ricochet like light beam (reflect ball path over normal plane)
   1. Maintain same speed and trajectory (basically just updates direction of ball)
2. Player hits ball: Calculate trajectory based on power and direction
   1. Kinematics equations – Power meter should map to acceleration; x=vt+at^2/2
3. Direction of ball given by array of line paths
   1. Determine function that maps slide pot input to a line on screen