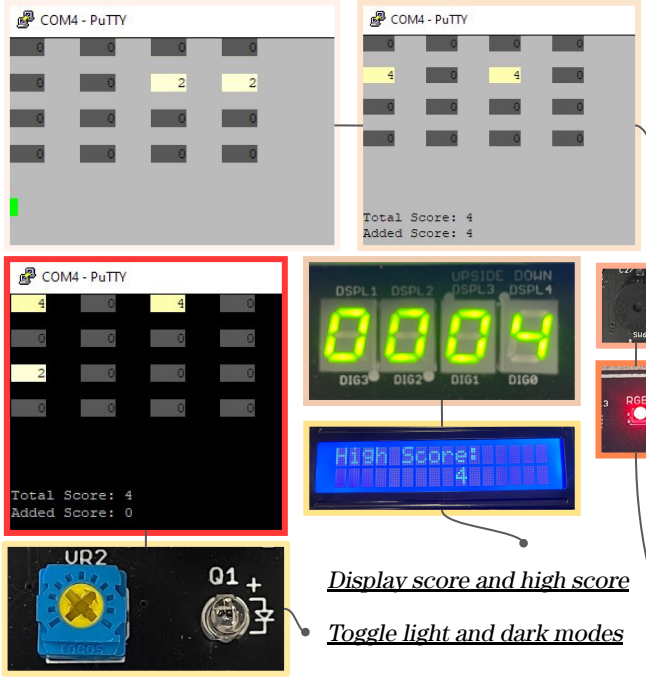
Rules of 2048

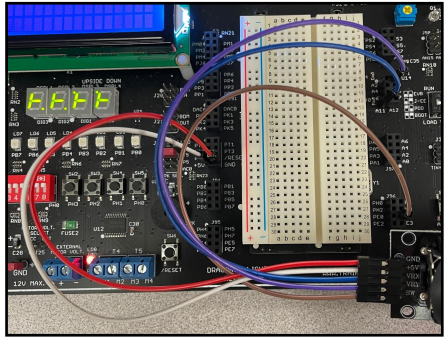
- Played on a 4x4 grid of tiles, each with their own value
- The player can slide the tiles in a cardinal direction
- A new tile will appear after each move (a 2 or a 4)
- If two tiles with the same value collide, they merge
- The primary objective is to reach the 2048 tile
- The game ends when there are no valid moves left

Included Features

- Toggleable sound effects that play after scoring or losing
- Joystick controls to move tiles + the z-axis for other input
- 7-Segment and LCD displays for the score and high score
- Automatically updated light and dark modes
- Game Over indicator with flashing LEDs
- PuTTY terminal display

2048 in PuTTY





- GND → GND
- +5V → +5V (ADC)
- VRX → V11 (AD1 CHANNEL 3)
- VRY → A11 (AD0 CHANNEL 3)
- SW → PE1 (Int. #25, PTH 0x08)

Display score and high score

Toggle light and dark modes

Slide tiles with the Joystick

Toggle audio/sound effects

Code Snippets

The X and Y axes of the joystick are read from channel 3 of AD1 and AD0 respectively

```
int joystick_get_x_axis () {
  return adlconv(AD_CHANNEL_JOYSTICK_X) ;
}
```

The game is played with a 4x4 grid of ints

```
  int** grid = (int**)malloc(GRID_LENGTH * sizeof(int*));
  for (i = 0; i < GRID_LENGTH ; i++) {
    grid[i] = (int*)malloc(GRID_LENGTH * sizeof(int));
  }
```

The dark/light modes are selected based on the values of the light sensor (or potentiometer for a manual override)

```
//Force Light Mode
else if (valPotentiometer > AD_POTENTIOMETER_THRESHOLD_LIGHT)
{
use_light_mode = _TRUE_ ; }
//Use light sensor value
else { use_light_mode = ( ad0conv(AD_CHANNEL_LIGHT_SENSOR) >
AD_LIGHT_SENSOR_THRESHOLD); }
```