Overview

- Based on the classic kids game
- Goal to recreate the game in a digital format
The LM317 is required because of the large current draw the matrix is capable of >300mA. The HCS12 waits for input from the keyboard. If input is received the action assigned to that port will then trigger. Once triggered a software array is updated with the new information and set via SPI to the Led matrix. The matrix does not allow addressed data to be sent so every write must be 64 bytes long.
Hardware

- SSMI Board
- Keypad
- LED Matrix
Timing Diagram

PT 7,5,4

CS

SCLK

MISO

Led1  Led2  Led3  .....  .....  Led62  Led63  Led64
Pseudocode

- Init SPI
- Set Port T for 3 inputs and 1 output: left, right, drop, /cs
- Set PortM for SPI functions: MOSI, Sclk
- Start main loop
  - Check for input on port T from keyboard:
  - If input is true & action not valid return to main loop
  - Else: valid action:
    - Do action
    - Update board
    - Check for win or stalemate
    - Invert players
    - If win is true, highlight winning pieces in yellow
      - Wait for button press, clear board and return to main loop

- Return to main loop
Functions

- Waiting(): waits until key is pressed
- DelayuSec(int): waits # uSeconds
- InitializeSPI(void): set SPI parameters
- Putcspi(char): send char over SPI
- Checkwin(player): checks for win
- Updateboard(): updates all 64 Leds to correct color
- moveR(): move right
- moveL(): move left
- Droppiece(): drop piece
- Startgame(): set all variables to correct values
Main loop code

- `initializeSPI();`
- `DDRT=0x01;`
- `DDRM=0x34;`
- `while (play){`
  - `startgame();`
  - `while (!win){`
    - `if ((PTT & 0x60) !=0){`
      - `DelayuSec(1000);`
      - `if ((PTT & 0x60) !=0){`
        - `Moveright=TRUE;`
        - `DelayuSec(10000);`
        - `moveR();`
      - `}
    - `}
    - `if ((PTT & 0x80) !=0){`
      - `DelayuSec(1000);`
      - `if ((PTT & 0x80) !=0){`
        - `moveleft=TRUE;`
        - `DelayuSec(10000);`
        - `moveL();`
      - `}
    - `}
    - `if ((PTT & 0x50) !=0){`
      - `DelayuSec(1000);`
      - `if ((PTT & 0x50) !=0){`
        - `drop=TRUE;`
        - `DelayuSec(10000);`
        - `droppiece();`
      - `}
    - `}
  - `updateboard();`
  - `waiting();`
- `}`
Design Challenges

- Deciding how to control input to the HCS12
- To use interrupts or polling for inputs
- PT6 giving false reading (broken port)
Design Challenges

- Input and outputs labeled backwards
- Writing code to correctly check for a WIN condition
- Needing to provide two separate +5v sources (>3A draw)
Questions??