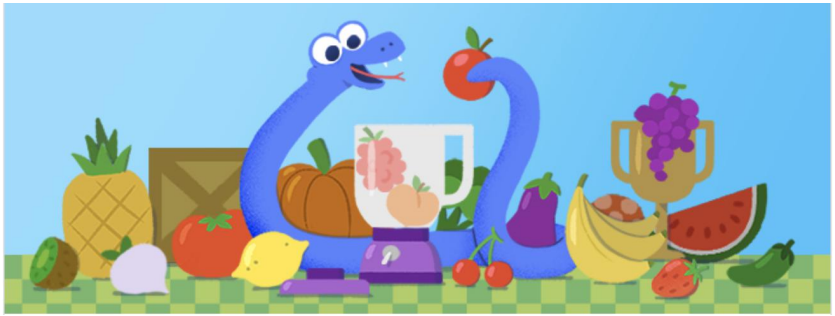
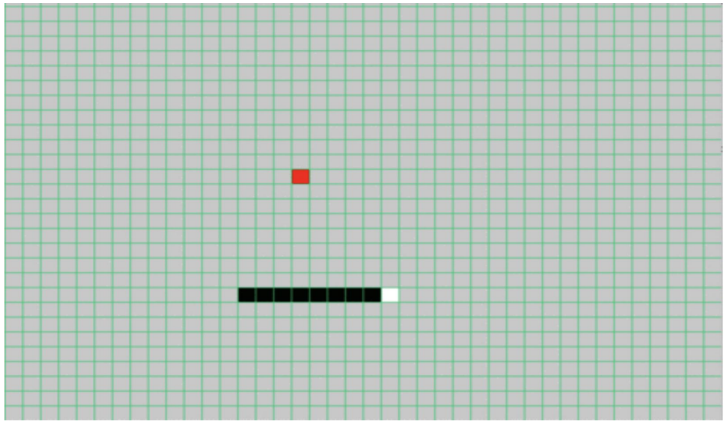
Introduction to Lab Snake Game

José Nelson Amaral

Snake Games

- Snake games available online
- Very famous games



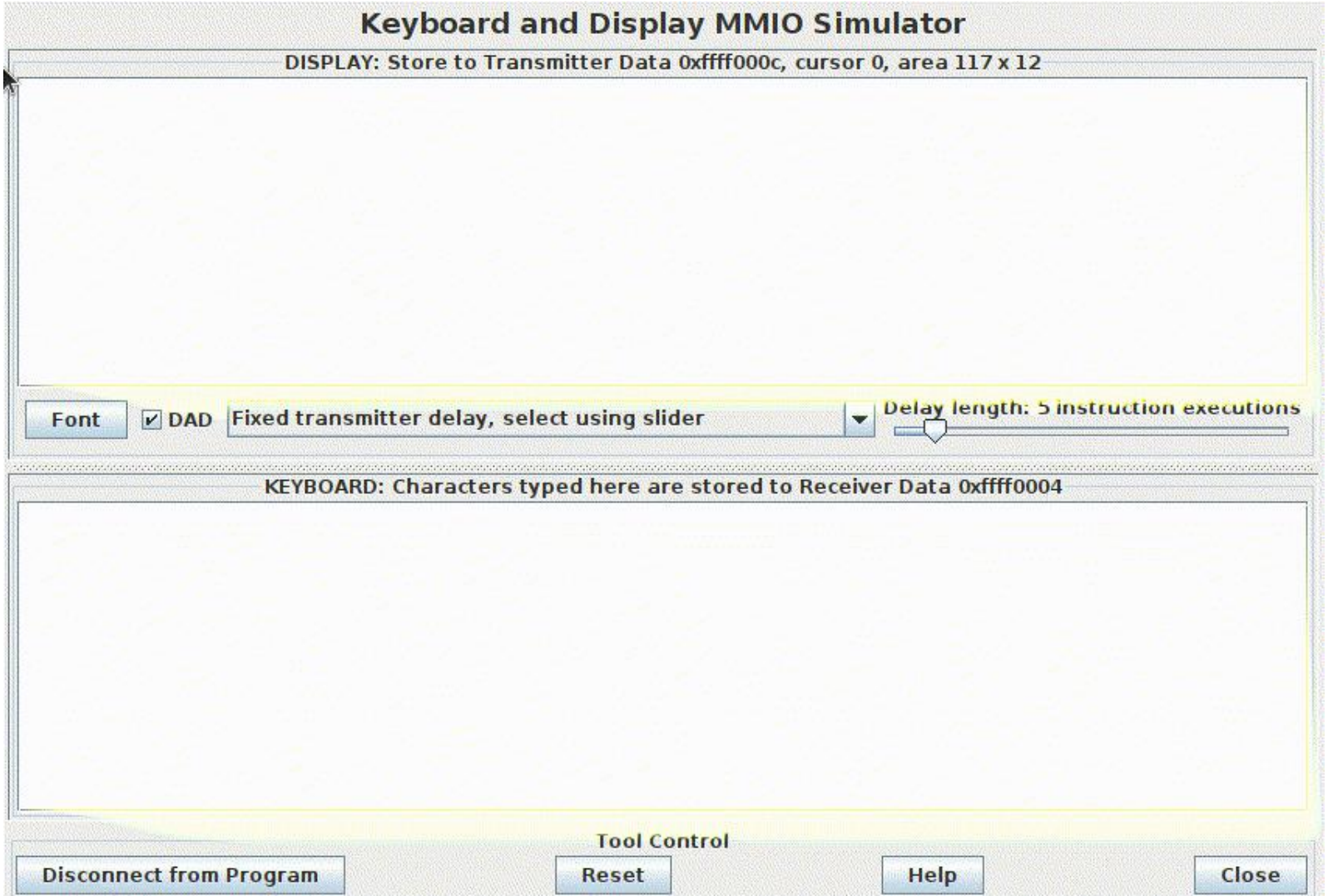


Our Snake Game is much simpler

- We don't consider the situation when the snake hits its own body. The players just need to avoid this situation while playing this game.
- After the snake eats the apple, the size of the snake body doesn't increase.

Two Screens

- Preparation Screen:
  - Reads the level of the game
- Game Screen:
  - Walls
  - Apple
  - Snake
  - Number of points
  - Time remaining


Keyboard and Display MMIO Simulator
DISPLAY: Store to Transmitter Data 0xffff000c, cursor 0, area 117 x 12
Font
DAD
Fixed transmitter delay, select using slider
Delay length: 5 instruction executions
KEYBOARD: Characters typed here are stored to Receiver Data 0xffff0004
Tool Control
Disconnect from Program
Reset
Help
Close

Random Coordinate

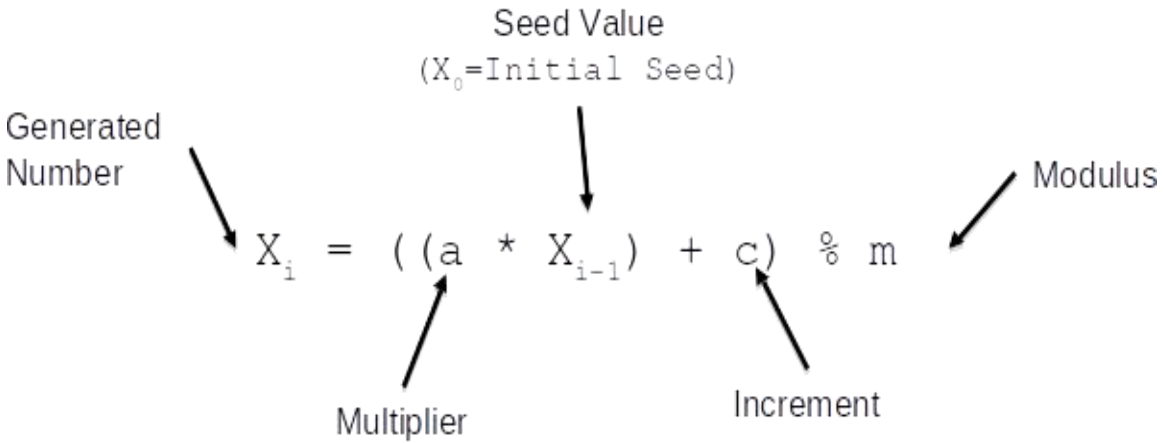
The coordinate (row, col) of the apple is randomly generated.

We need a random number generator.

Linear Congruential Generator

$$X_i = ((a * X_{i-1}) + c) \% m$$

Linear Congruential Generator Example



a = 5

X0 = 4

c = 13

m = 5

X1 = ((5 * 4) + 13) % 5 = 3

X2 = ((5 * 6) + 13) % 5 = ?

Exceptions/Interruptions

- Enable interrupts for both the timer and the keyboard
- Create an exception handler

Enable Interrupts

- Keyboard:
  - Keyboard Control - Bit 1 Must be 1 in order for the keyboard to be enabled
    - Must be reenabled after every keyboard interrupt
  - Keyboard Data: Contains the ASCII character after a key is pressed
- Timer:
  - Timer: Contains the current time
  - TimeCMP: User-specified value. When matched by the timer an interrupt is generated
- Interruption Control:
  - Ustatus register: bit 0 must be 1 in order for user interrupts to be allowed
  - Uie: Bits 4 and 8 must be 1 in order to enable keyboard and timer

Exception Handler

The snake.s already contains the Handler Terminate section

The common.s file will already have the iTrapData section

Saving Registers

An interrupt handler must save all the registers that it uses.

- The label iTrapData designates a section of memory allocated for saving registers in the handler.
- Outside of the handler, uscratch (CSR #64) should contain the address of the iTrapData section.
- Use the cssrw instruction to swap a register with the uscratch and save all the required registers.

Modified Forever Loop