

CSE 379

Lab 7

Spring 2020

Presented by:

Dazhou Liu

Xiaoang Zhang

May 3rd 2020

Descriptions

Division of Work

Based on the requirements, Lab 7 is divided into portions and distributed to two team members. The distributions mainly consist of two parts. One part is for the 4 ghosts and the other part is for Pac-man. Using values from the counter of timer to generate random numbers for ghosts is accomplished by Dazhou Liu. The idea of storing address that contains pac-man's position comes from Xiaoang Zhang. This idea is also used to store the addresses of 4 ghosts.

As for the basic game board, the shape and elements inside are configured by Dazhou Liu. The codes to print out the whole board in main file is done by Dazhou Liu. The score portion of Lab 7 is written by Xiaoang Zhang.

One timer is used for controlling the Pac-man. The flag allowing movement of pac-man was set up by Xiaoang Zhang. The timer handler and interrupt rate are done by Xiaoang Zhang as well. 3 colors of RGB-LED indicating lives remaining is coded by Xiaoang Zhang and the blue color indicating power mode is coded by Xiaoang Zhang. Pac-man getting out from right side and wrap to left side in middle of board is written by Xiaoang Zhang. The up, down, left and right movements of Pac-man is written by Dazhou Liu. Pac-man getting out from left side then entering the board from right side in middle of board is written by Dazhou Liu.

Another timer is used to move ghosts randomly in the maze. This timer is set up by Dazhou Liu. Picking up a random direction if no pac-man presented within 4 grids is done by Dazhou Liu. It applies all 4 ghosts. Chase pac-man within 4 grids and move away from pac-man if power pellet is presented is divided into 2 parts. Dazhou Liu wrote code for ghosts 1 and 2. Xiaoang Zhang wrote code for ghosts 3 and 4. Codes dealing with collisions between Pac-man and ghosts are written by Xiaoang Zhang. Pac-man returning to original position and ghosts backing to center of box are done by Xiaoang Zhang.

Swapping speed after pac-man eating power pellets is done by Dazhou Liu. For reprinting the board, Dazhou Liu restored the flag. Xiaoang Zhang restored the memory contents of the board.

Receiving character p and prompt the pause menu is done by Xiaoang Zhang. The flag indicating pause is requested or not is set up by Xiaoang Zhang. Dazhou Liu modified the memory contents within the game in the center of board. Dazhou Liu put "PAUSED" in the box and restored the contents covered by PAUSED.

For colors, the user prompt and instruction before game starts is written by Xiaoang Zhang. The sentences for that was written by Xiaoang Zhang. The colors of pause menu are configured by Dazhou Liu. The instructions for pause menu was written by Dazhou Liu.

For hardwares, Xiaoang Zhang wrote codes for illuminating the RGB-LED. Xiaoang Zhang also verified the validity and debugged the initializations.

Purpose of the program

Objective

The objective of Lab 7 is to implement an alternative version of Pac-man game by using ARM assembly language in Code Composer Studio. The expectation of Lab 7 is a full version of Pac-man game. Requirements and rules specified in documentation on course website will be fulfilled. Memory handling, data arithmetic and utilization of subroutines with AAPCS are necessary contents towards completion of Lab 7. Upon completion of Lab 7, several functions are incorporated. Firstly, the user is able to control the Pac-man in up, down, left and right directions. The 4 ghosts can navigate randomly in the maze. Next, pellets can be eaten by Pac-man. The 4 power pellets can be eaten by Pac-man, which gives ability of eating the ghosts to pac-man. During game, the LED will indicate remaining lives of pac-man. Blue LED indicates that power mode is on. The user will be able to pause the game, and can choose to continue, restart or end the game. The user will be notified and prompted upon start and end of game.

Debugging steps

The debugging steps for Lab 7 involves two types of errors. One is syntax error and another is run time error. Towards completion of Lab 7, pieces of codes are written to satisfy certain functions. The first step to go through is building project. In this step, syntax errors will be fixed. The errors encountered are missing symbols, pointers and spelling errors. For example, dropping for hex decimal and writing 0x directly is one of the syntax errors.

After fixing the syntax errors, debugging for run time errors are needed. There were two types of run time errors. One is handling memory in invalid memory location or trying to use hardware that is yet to be initialized. This led the program to an exception handler. For wrong memory location, the actual memory location was applied to fix. For hardware, they will be initialized in proper location within codes. Another error is diverting from the requirements. One such case is printing the board incorrectly. Elements inside the board was missing or in correct. For example, dropping O or pellets. In this case, the memory location were checked. Incorrect elements were replaced with proper ones. Arithmetic correction is also performed. This involves calculating a specific address from a giving address. Another case is ghosts did not navigate as expected. This case is mainly due to storing wrong contents to specified locations. Forgetting to update memory locations with proper contents is also a cause to this type of error. The debugging process in this stage is going back to the location where the code handles that memory address. After debugging and fixing syntax and run time errors, one function can be satisfied. Another piece of code will be written to fulfill the next requirement.

Outside materials

1. Tiva™ TM4C123GH6PM Microcontroller

Chapter 11 General-Purpose Timers

Chapter 10 General-Purpose Inputs/Outputs (GPIOs)

Chapter 14 Universal Asynchronous Receiver-Transmitter (UARTs)

2. Lectures Provided on Course Website

Lectures introducing GPIOs, Data Processing, single/multiple register load/store instructions and

Control Flows

3. Online learning resources: consultation with TAs during Office Hours

Instructions on how to use the program

The program is the pac-man game with incorporated functions. For game rules, the pac-man is shown as a "<" and the 4 ghosts are shown as "M". Pac-man has 3 lives. Touching a ghost without power pellets results in loss of one life. Once the life remaining, game will end. The user will be prompted. "GAME OVER" is shown in the center box. Green LED indicates 3 lives remaining, yellow for 2 and red for 1. Blue color indicates power mode is on. It will last 8 seconds. During this time, pac-man can eat the ghost without losing one life and will earn scores. Alternatively, the user can pause the game by pressing lowercase p on keyboard. Upon pausing, "PAUSED" is shown in the center box. The user can choose to restart, end or resume the game. The first step is setting up required applications. The Putty is configured for the Tiva board with a board rate of 115200. Then open Code Composer Studio. Click the debugging button and run the program. Upon success, instructions introducing the rules and contents of the program is prompted. It is in green color. The user must choose to press decimal 1 or 2 on keyboard. Other inputs are ignored by the program. To start, press 1 on keyboard. To end game, press 2 on keyboard. After pressing 1 to start the game, the game starts. This leads to the actual game. The 4 ghosts start to get out from the center box one by one. The user then can control the Pac-man in 4 directions. If pac-man reaches borders, pac-man will stop. The user needs to pick up another direction without border. Without eating power pellets, the pac-man should avoid the ghosts and eat as many pellets as it can. 10 points are awarded for eating one pellet. If all pellets are eaten, a whole new board will be presented. Pac-man, 4 ghosts and all other elements will be restored. This means that a new level is entered. within this level, the moving speeds of ghosts and pac-man are increased by 25 percent. On the other hand, pac-man lost lives if it touches ghost without power pellet. 2 situations will end the game. One is pac-man used up all 3 lives. Another is the user choose to end the game in start prompt or pause menu. Once the game ends, the board will freeze with a "GAME OVER" in center box. The user is prompted that the game ended. The LED is also turned off.

Logic

Lab 7 is a comprehensive project. To finish certain functions of the pac-man game, it is divided into parts. The first step is to successfully print the board with required elements placed accurately. The number of pellets, power pellets, ghosts and pac-man are checked. Then the positions of these elements are also checked.

After printing out the board accurately, the movements are processed. The codes for movements of pac-man were written. The UART is utilized to get control data from user. Acquired data is stored in memory. The pac-man is able to move in up, down, left and right directions. Next comes pac-man eating pellets and power pellets and leave blank behind. After processing pac-man, the next step is to make 4 ghosts navigate randomly. Firstly, the code is written to make ghosts get out one by one. After exiting the box, real navigation begins. The ghosts are programmed to keep its direction unless a junction is reached. Once ghosts reach a junction, random number generator is used to pick up a direction. 4 numbers are obtained from random number generator. Each of them corresponds to a direction. The same rule applies to 4 ghosts,

which is that they cannot cross the border. If the ghosts encountered borders, they are directed to pick a direction that without borders ahead.

After the movements, the times were adjusted. The time such as 8 seconds are calculated from the number of times interrupts occur. For example, 8 seconds are considered with 16 interrupts with an interruption rate of 0.5 second. The moving speed pac-man and ghosts are copied to register. Then that speed is passed to initialization functions through that register following AAPCS.

Reprinting board may occur during game. One situation is that the user choose to restart from pause menu. Another situation is that all pellets are eaten by pac-man. Ahead of reprinting board, memory contents are restored to the first start of game. Besides the flags are also reset to original values. Once resetting is completed, the program restarts.

Summary of flowcharts

There will be nine flowcharts presenting 9 subroutines respectively.

lab7.s describes flows of the main function of lab 7.

timer_init describes initialization of timer 0.

timer1_init describes initialization of timer 1.

timer_handler describes interruption servicing triggered by timer0.

timer1_handler describes interruption servicing triggered by timer1.

randomghostmove demonstrates the flow of the movements of 4 ghosts.

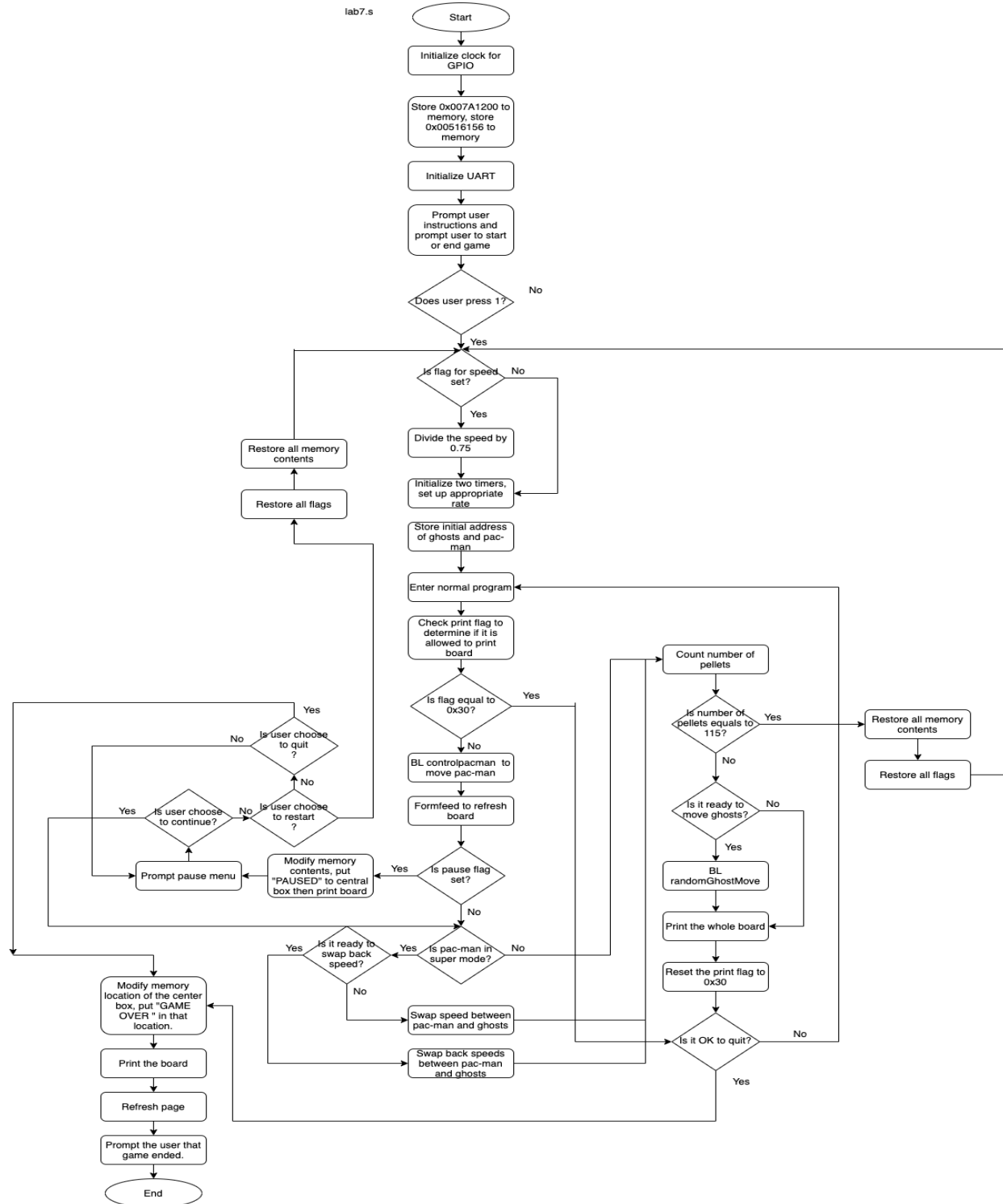
controlpacman demonstrates the flow of the movements of pac-man

UART0_Handler describes the flow of how the interruptions from keyboard are serviced.

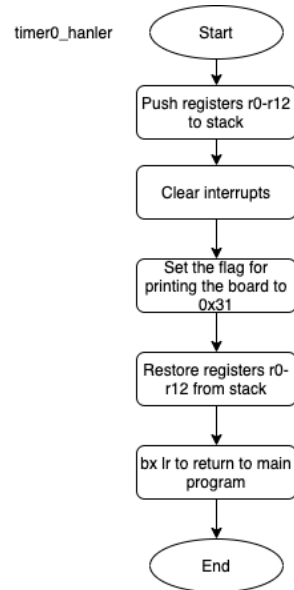
chaseorawayM describes the basic flow of the 4 ghost determining when to approach pac-man or move away from pac-man.

Flowcharts

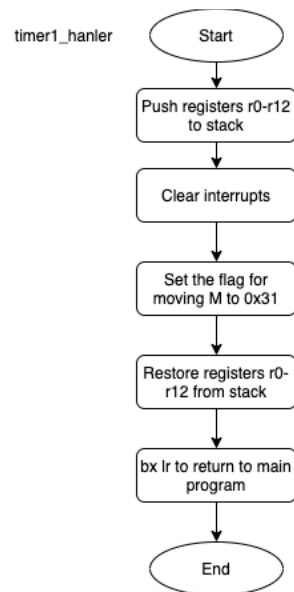
lab7.s

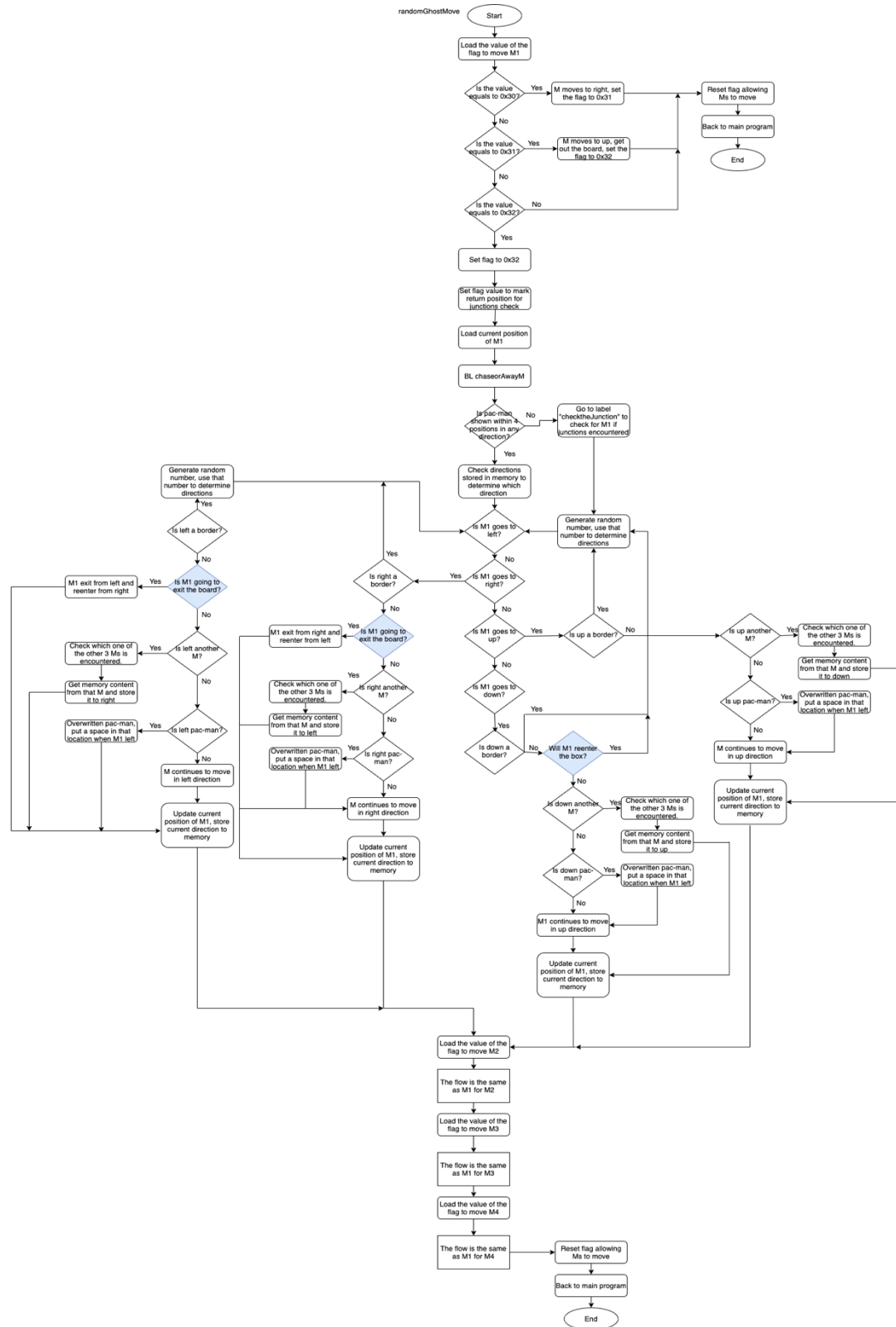


timer0_handler



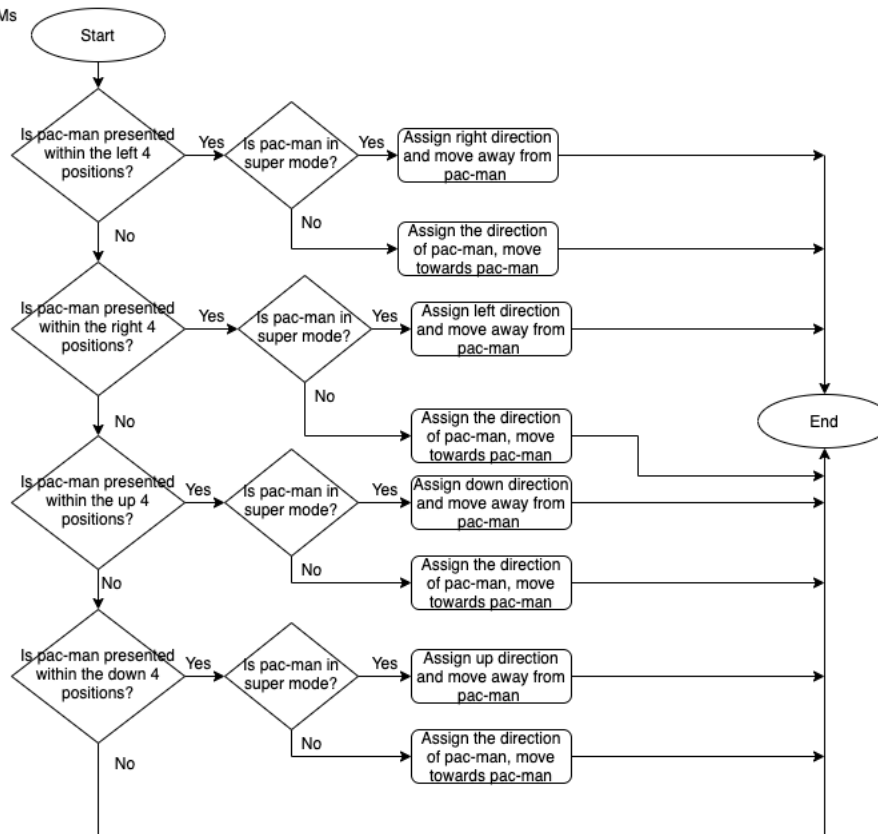
timer1_handler



randomGhostMove

chaseorAwayM

chaseorAwayMs



checktheJunctions

checktheJunctions

