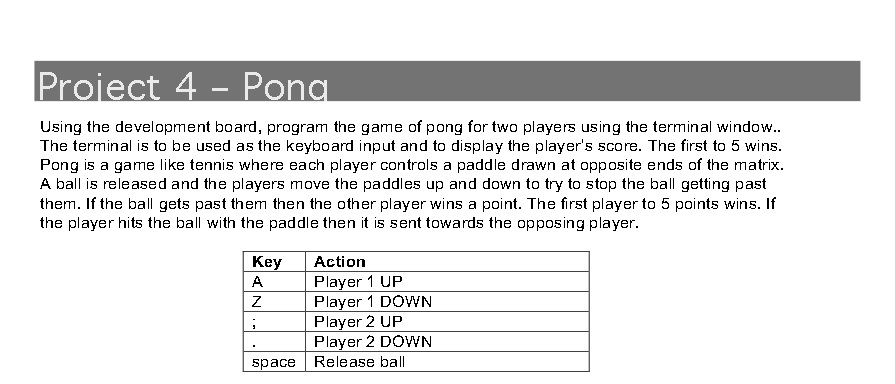
2303ENG MPT Project System Requirements



# Aim

The aim of this Design Project is to design, implement and demonstrate a complete embedded system, utilising programming skills and knowledge developed throughout the duration of the course 2303ENG Microprocessor techniques.

The task chosen for this project is the Pong game. The remainder of this document will specify the design and main components of the proposed project.

# System Requirements

## System design

The system intended for design is a remake of the earliest arcade video games Pong. Pong is a simple two dimensional graphics game, the aim is to defeat an opponent by earning a higher score. A ball will need to rebound off two opposing paddles and when one paddle misses a point will be rewarded to the opposing team. The paddles are a 2D representation of a small bat. The game will need to display the players score, this is to be displayed on the LCD terminal. Once a player has reached five games, the game is finished and the player with the highest score wins. To do this GPIO port A will be used in conjunction with UART 0 pins A0 and A1.

## Equipment

Tiva-C series TM4C123GH6PM board

LCD screen (TG12864H3-05A v1.0)

Connecting wires

## How it can be simplified

This project is simple as it is, as this project demonstrates simple understanding of Microprocessor programming, simplifying this project would defeat this purpose.

## How it can be extended

Once testing and basic implementation of this project is achieved, the system can be extended in several ways. For example, user names can be entered and displayed on top of the screen, more balls could be added into play, the ball speed could be increased as the game progresses or according to the difficulty selected, or perhaps a single player mode can be added to the options for practice, where the ball bounces off a wall. Another option is to add switches to the system instead of using the keyboard to move the paddles.

# Specifications

## Pseudo Code

### Tiva Setup:

Set up GPIOA

Set up UART0(uses GPIOA pins 0 and 1)

Set up Timer

Set up Interrupt

### Game Controls:

#### Paddle 1 (Player 1)

Define pixel coordinates (Center of y-coordinate)

Display to LCD

If ‘A’ pressed

move pixels up(in the y direction)

display to LCD

If ‘Z’ pressed

move pixels down(in the y direction)

Display to LCD

#### Paddle 2( Player 2)

Define pixel coordinates

Display to LCD

If ‘;’ pressed

move pixels up (in the y direction)

display to LCD

If ‘.’ pressed

move pixels down (in the y direction)

Display to LCD

#### Ball direction(interrupt)

Define pixel coordinate

Display to LCD

Move ball pixel right (x direction)

If ball pixel = top third of paddle2

Move diagonally up left( Y+, X-)

Else If ball pixel = middle third of paddle2

Move left( X-)

Else If ball pixel = Bottom third of paddle2

Move diagonally down left( Y-, X-)

If ball pixel = top third of paddle1

Move diagonally up right( Y+, X+)

Else If ball pixel = middle third of paddle1

Move right( X+)

Else If ball pixel = Bottom third of paddle1

Move diagonally down right( Y-, X+)

Display to LCD

#### Match point

Player 1 score

If ball pixel > paddle player 2

Player 1 score = score+1

If P1 score < 5

Board reset

Display score on putty

Else print to putty

“Player 1 wins”

Player 2 score

If ball pixel < paddle player 1

Player 2 score = score+1

If P1 score < 5

Board reset

Display score on putty

Else print to putty

“Player 2 wins”

#### Board Reset

Call

Move paddle for player 1

Move paddle for player 2

Ball Direction

# Simple View of the Game

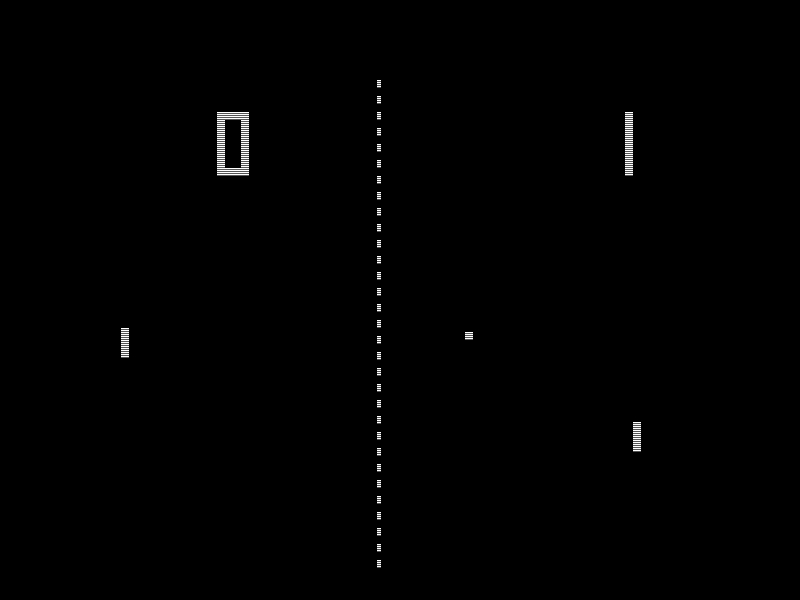


Figure 1 Simplest View of the proposed project(without extension)

# Flow Chart of Code (High level)

