

آزمایشگاه ریزپردازنده

گزارش کار آزمایش دهم

اشکان مرادی – ۸۱۰۱۹۵۴۷۵

محمد مرآتی – ۸۱۰۱۹۵۴۷۳

زهرا افتخاری – ۸۱۰۱۹۵۳۵۴

برای طراحی محیط گرافیکی بازی مورد نظر از LCD و برای نمایش امتیازات از GLCD استفاده شد.

```
int main() {
    SystemInit();
    LCD_SetUp(P0_0,P0_1,P0_2,P_NC,P_NC,P_NC,P_NC,P0_4,P0_5,P0_6,P0_7);
    LCD_Init(2,16);
    LCD_Clear();
    LCD_SetCursor(0, 0);
    GLCD_Initialize();
    GLCD_ClearScreen();
    for (int i = 0; i < 16; i ++)
    {
        line1[i] = ' ';
        line2[i] = ' ';
    }

    LCD_Printf("Faraar Az Mane");
    LCD_SetCursor(1,3);
    LCD_Printf("By: Group 1");
    DELAY_ms(10000);
    LCD_Clear();
    LCD_SetCursor(0, 0);
    LCD_Printf("Press Enter");
    LCD_SetCursor(1, 7);
    LCD_Printf("to Start");
    UART2_Init(9600);
    char data = UART_RxChar(2);
    if (data == '\n')
    {
        xTaskCreate(screen, "screen", 128, NULL, 1, &screen_handler);
        xTaskCreate(movement, "movement", 128, NULL, 1, &movement_handler);
        vTaskStartScheduler();
    }
    while(1);
}
```

ابتدا منتظر اعمال ورودی \n از سوی UART در برنامه ی Hercules می شویم. پس از دریافت این داده ۲ تسک اصلی ساخته می شود که یکی مسئول نمایش صفحه ی بازی و دیگری مسئول دریافت ورودی های مربوط به جا به جایی بالا و پایین بازیکن می باشد.

```

char player_line = 0;
char line1[16];
char line2[16];

TaskHandle_t screen_handler;
TaskHandle_t movement_handler;

void movement(void *pvParameters){
    char data;
    while(1){
        data = UART_RxChar(2);
        if(data == 'w' && player_line == 1){
            player_line = 0;
        }
        else if (data == 's' && player_line == 0)
        {
            player_line = 1;
        }
        vTaskDelay(100/portTICK_PERIOD_MS);
    }
}

```

در تسک movement، حرکات بازیکن توسط UART به صورت real time (با تاخیر ۱۰۰ میلی ثانیه برای انتقال صحیح به صفحه ی بازی) خوانده می شود و متغیر گلوبالی که نشان دهنده ی خط کنونی بازیکن است مقداردهی می شود.

```

void screen(void *pvParameters){
    int cntr = 0;
    int point = 0;
    while(1){

        int random_num1 = rand();
        for (int i = 0; i < 15; i++)
        {
            line1[i] = line1[i+1];
            line2[i] = line2[i+1];
        }

        if (cntr % 3 == 0)
        {
            if (rand() % 2 == 0)
            {
                line1[15] = '#';
                line2[15] = ' ';
            }
            else
            {
                line1[15] = ' ';
                line2[15] = '#';
            }
        }
        else
        {
            line1[15] = ' ';
            line2[15] = ' ';
        }

        if (player_line == 0)
        {
            if (line1[0] == '#')
                line1[0] = 'X';
            else
                line1[0] = '$';
        }
        else if (player_line == 1)
        {
            if (line2[0] == '#')
                line2[0] = 'X';
            else
                line2[0] = '$';
        }

        LCD_Clear();
        LCD_SetCursor(0, 0);
        LCD_Printf(line1);
        LCD_SetCursor(1, 0);
        LCD_Printf(line2);
        if ((line1[0] == '#' && line2[0] == '$') || (line2[0] == '#' && line1[0] == '$'))
        {
            point ++;
        }
        if ((line1[0] == 'X') || (line2[0] == 'X'))
        {
            GLCD_GoTo(30, 5);
            GLCD_WriteString("Game Over!");
            vTaskDelete(screen_handler);
            vTaskDelete(movement_handler);
        }

        GLCD_ClearScreen();
        GLCD_GoTo(20, 1);
        GLCD_WriteString("Your Score:");
        GLCD_GoTo(86, 3);
        GLCD_WriteChar((char) point/100 + 48);
        GLCD_GoTo(92, 3);
        GLCD_WriteChar((char) (point % 100) / 10 + 48);
        GLCD_GoTo(98, 3);
        GLCD_WriteChar((char) (point % 100) % 10 + 48);
        cntr ++;
        vTaskDelay(1000/portTICK_PERIOD_MS);
    }
}

```

در تسک Screen ابتدا هربار محتویات دو متغیر گلوبال line1 و line2 که محتویات خطوط قابل نمایش (مانع ها) را در خود دارند یکی به سمت چپ شیفت می خورد که حرکت مانع ها شبیه سازی شود. سپس بر اساس الگوریتمی، که طبق آن مقدار یک شمارنده در هر بار اجرای حلقه بررسی می شود و به ازای تولید هر ۲ اسپیس (به منظور ایجاد فضا برای بازیکن برای فرار از مانع) یک مانع به صورت رندوم در خط اول یا دوم به انتهای آرایه ی مورد نظر اضافه میشود، مانع ها طراحی می شوند. سپس در خانه ی ۰ ام آرایه های مربوط به خطوط نمایش، بر اساس موقعیت بازیکن و بررسی برخورد آن با مانع یا عدم برخورد (X یا \$)، کاراکتر های مناسبی داده می شود.

سپس آرایه ها روی LCD چاپ می شوند و با عبور هر یک مانع از صفحه یک امتیاز به بازیکن داده می شود و همچنین در صورت برخورد با مانع پیغام Game Over! روی نمایشگر گرافیکی چاپ می شود و هر دو تسک screen و movement پاک می شوند. در پایان این تسک نیز امتیاز بازیکن به صورت زنده روی نمایشگر گرافیکی چاپ می شود و تاخیر ۱ ثانیه ای به این تسک داده می شود تا موانع با سرعت ۱ واحد در ثانیه حرکت کنند.

فیلمی از نحوه ی اجرای بازی به پیوست ارسال گردیده است.