Linnaeus University

1DT301 - Computer Technology 1
Assignment 2

Group number: Group I
Group member: Jiaxin Wang, Zejian Wang
Student ID: jw223jc@student.lnu.se
zw222bb@student.lnu.se

Task 1

```
@
@ Assembler program to print out “Hello World”
@ using the Pico SDK
@
@R0 – first parameter to printf
@R1 – second parameter to printer
@R7 – index counter
@

.thumb_func         @Necessary because sdk uses BLX
.global main        @Provide program starting address to linker

main:
          MOV     R7, #100            @initialize counter to 100
          BL      stdio_init_all      @initialize uart or usb

loop:
          LDR     R0, =helloworld
          MOV     R1, R7
          BL      printf     @ Call pico_printf
          SUB     R7, #1     @ Increment counter
          BGE     loop

          MOV     R7, #100
          B       loop

.data
                  .align     4          @necessary alignment
helloworld:  .asciz          “Hello World %d\n”
```

Task 2

```
@
@ Assembler program to flash the build-in LED on Raspberry Pi
@Pico GPIO port using the Pico SDK.
@
        .EQU    LED_PIN1, 25
        .EQU    GPIO_OUT, 1
        .EQU    sleep_time, 500

.thumb_func     @Necessary because sdk uses BLX
.global main    @Provide program starting address

main:
        MOV     R0, #LED_PIN1
        BL      gpio_init
        MOV     R0, #LED_PIN1
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir

loop:
        MOV     R0, #LED_PIN1
        MOV     R1, #1
        BL      link_gpio_put
        LDR     R0, = sleep_time
        BL      sleep_ms

        B       loop
```

Task 3

```
@
@Assembler program to make three LEDs to flask like a traffic
@ light through Raspberry Pi Pico GPIO port using the Pico SDK.
@

        .EQU   LED_PIN1, 0   @set GPIOPIN0 for Green LED
        .EQU   LED_PIN2, 1   @set GPIOPIN1 for Yellow LED
        .EQU   LED_PIN3, 2   @set GPIOPIN2 for Red  LED
        .EQU   GPIO_OUT, 1
        .EQU   sleep_time_green, 2000    @set green led
                                         sleeping time to 2000ms.
        .EQU   sleep_time_yellow, 500    @set yellow led
                                         sleeping time to 2000ms.
        .EQU   sleep_time_red, 1000      @set red led sleeping
                                         time to 2000ms.

.thumb_func @ Necessary because sdk uses BLX
.global main  @ Provide program starting address

main:
        MOV    R0,  #LED_PIN1
        BL     gpio_init
        MOV    R0,  #LED_PIN1
        MOV    R1,  #GPIO_OUT
        BL     link_gpio_set_dir

        MOV    R0,  #LED_PIN2
        BL     gpio_init
        MOV    R0,  #LED_PIN2
        MOV    R1,  #GPIO_OUT
        BL     link_gpio_set_dir

        MOV    R0,  #LED_PIN3
        BL      gpio_init
        MOV     R0,  #LED_PIN3
        MOV     R1,  #GPIO_OUT
        BL      link_gpio_set_dir

loop:
        MOV     R0,  #LED_PIN1        @Green LED
        MOV     R1,  #1
        BL      link_gpio_put
        LDR     R0,  =sleep_time_green
        BL      sleep_ms
        MOV     R0,  #LED_PIN1
        MOV     R1,  #0
        BL      link_gpio_put

        MOV     R0,  #LED_PIN2        @Yellow LED
        MOV     R1,  #1
        BL      link_gpio_put
        LDR     R0,  =sleep_time_yellow
        BL      sleep_ms
        MOV     R0,  #LED_PIN2
        MOV     R1,  #0
        BL      link_gpio_put

        MOV     R0,  #LED_PIN3         @Red LED
        MOV     R1,  #1
        BL      link_gpio_put
        LDR     R0,  =sleep_time_red
        BL      sleep_ms
        MOV     R0,  #LED_PIN3
        MOV     R1,  #0
        BL      link_gpio_put

        MOV     R0,  #LED_PIN2         @Yellow LED
        MOV     R1,  #1
```

```
BL      link_gpio_put
LDR     R0,  =sleep_time_yellow
BL      sleep_ms
MOV     R0,  #LED_PIN2
MOV     R1,  #0
BL      link_gpio_put

B     loop                    @loop infinitely
```

Task 4

```
@
@ Assembler program to flash the 7-segment display and
@ implement a counter that counts from 0 up to 9 and back to 0,
@ and repeats infinitely.
@

        .EQU    LED_PINA, 0
        .EQU    LED_PINB, 1
        .EQU    LED_PINC, 2
        .EQU    LED_PIND, 3
        .EQU    LED_PINE, 4
        .EQU    LED_PINF, 5
        .EQU    LED_PING, 6
        .EQU    LED_PINH, 7
        .EQU    GPIO_OUT, 1
        .EQU    sleep_time, 1000

.thumb_func     @Necessary because sdk uses BLX
.global main    @Provide program starting address

main:
        MOV     R0, #LED_PINA
        BL      gpio_init
        MOV     R0, #LED_PINA
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PINB
        BL      gpio_init
        MOV     R0, #LED_PINB
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PINC
        BL      gpio_init
        MOV     R0, #LED_PINC
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PIND
        BL      gpio_init
        MOV     R0, #LED_PIND
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PINE
        BL      gpio_init
        MOV     R0, #LED_PINE
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PINF
        BL      gpio_init
        MOV     R0, #LED_PINF
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PING
        BL      gpio_init
        MOV     R0, #LED_PING
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir
        MOV     R0, #LED_PINH
        BL      gpio_init
        MOV     R0, #LED_PINH
        MOV     R1, #GPIO_OUT
        BL      link_gpio_set_dir

loop:
        MOV     R0, #LED_PINA           @Zero
        MOV     R1, #1
        BL      link_gpio_put
        MOV     R0, #LED_PINB
        MOV     R1, #1
```

```
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA       @One
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #0
BL      link_gpio_put
```

```
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA          @Two
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #0
BL      link_gpio_put
```

```
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA           @Three
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
```

```
BL      sleep_ms


MOV     R0, #LED_PINA           @Four
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA           @Five
MOV     R1, #1
BL      link_gpio_put
```

```
MOV     R0, #LED_PINB
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA           @Six
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA         @Seven
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #0
```

```
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA           @Eight
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
```

```
LDR     R0, = sleep_time
BL      sleep_ms


MOV     R0, #LED_PINA         @Nine
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINB
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINC
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PIND
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINE
MOV     R1, #0
BL      link_gpio_put
MOV     R0, #LED_PINF
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PING
MOV     R1, #1
BL      link_gpio_put
MOV     R0, #LED_PINH
MOV     R1, #1
BL      link_gpio_put
LDR     R0, = sleep_time
BL      sleep_ms

B       loop
```