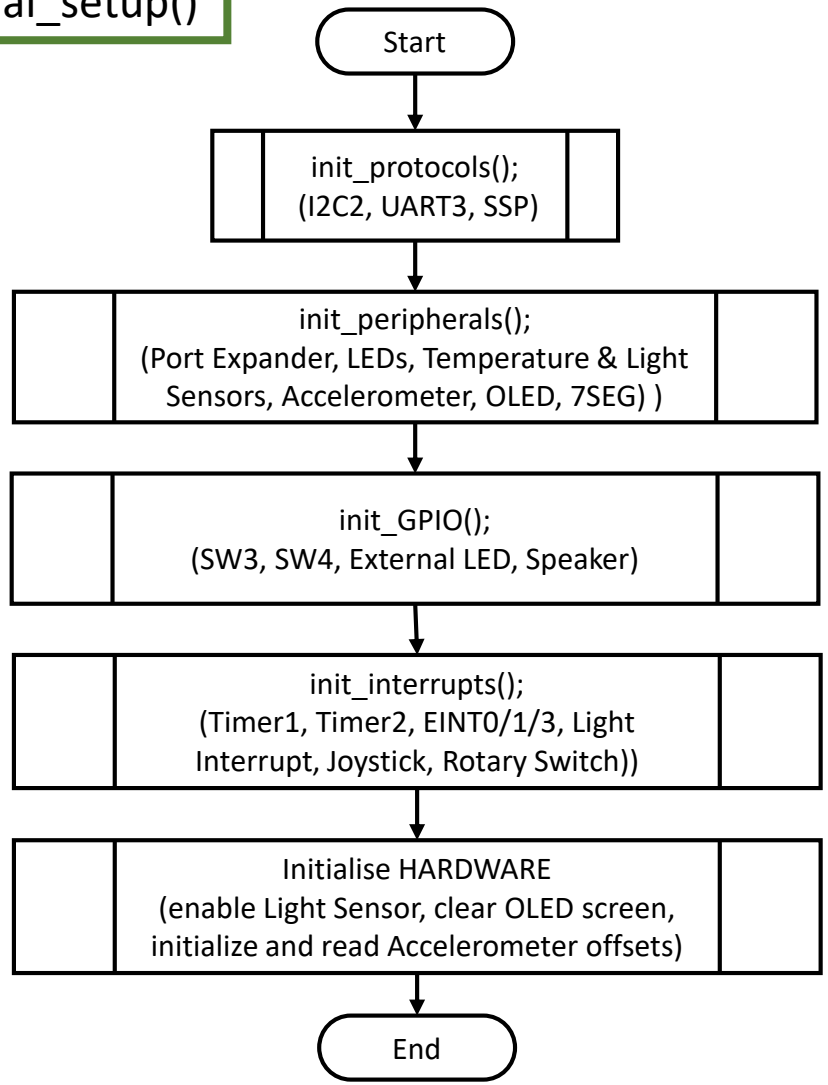
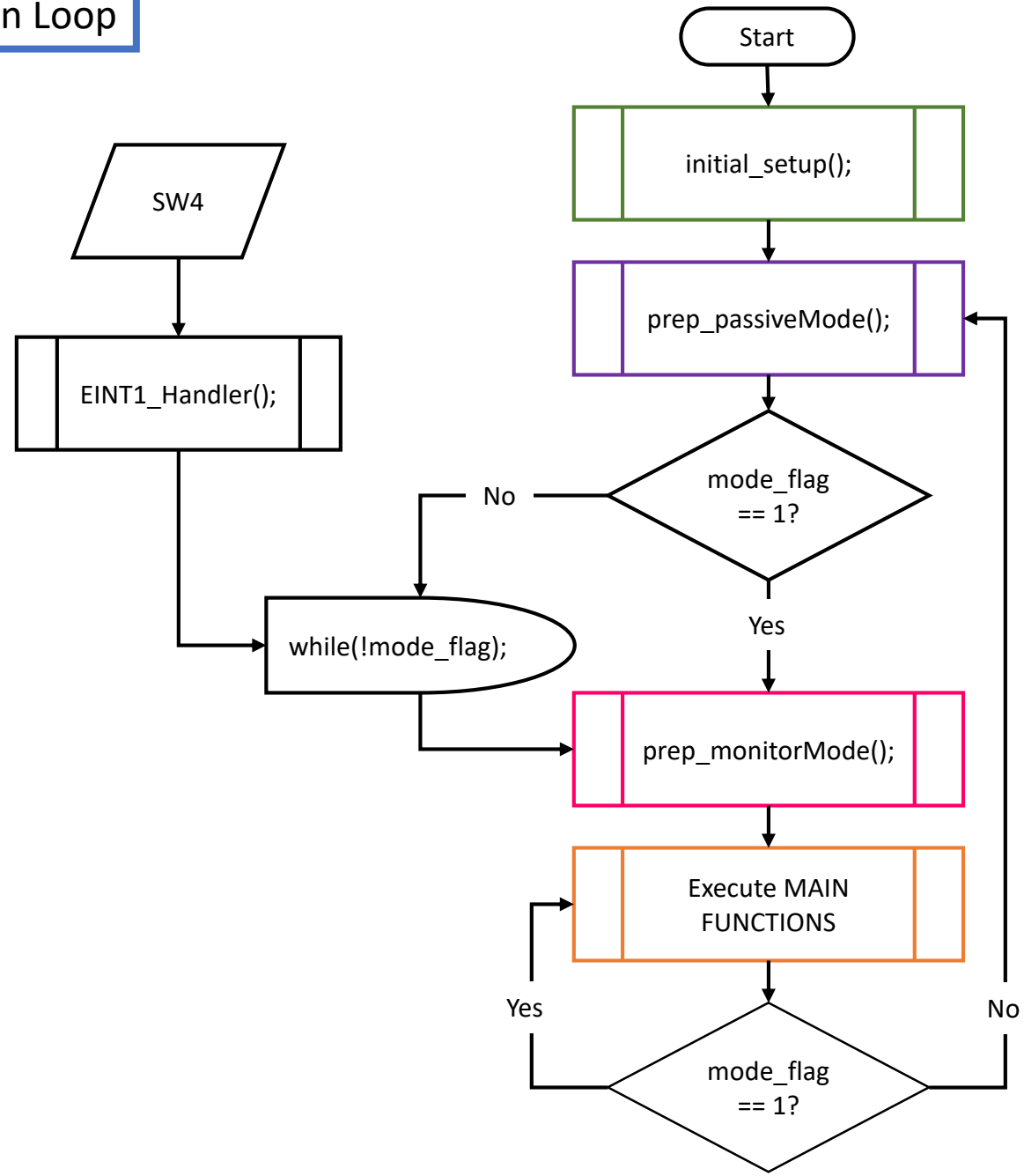
Initial Values

mode_flag = 0

initial_setup()



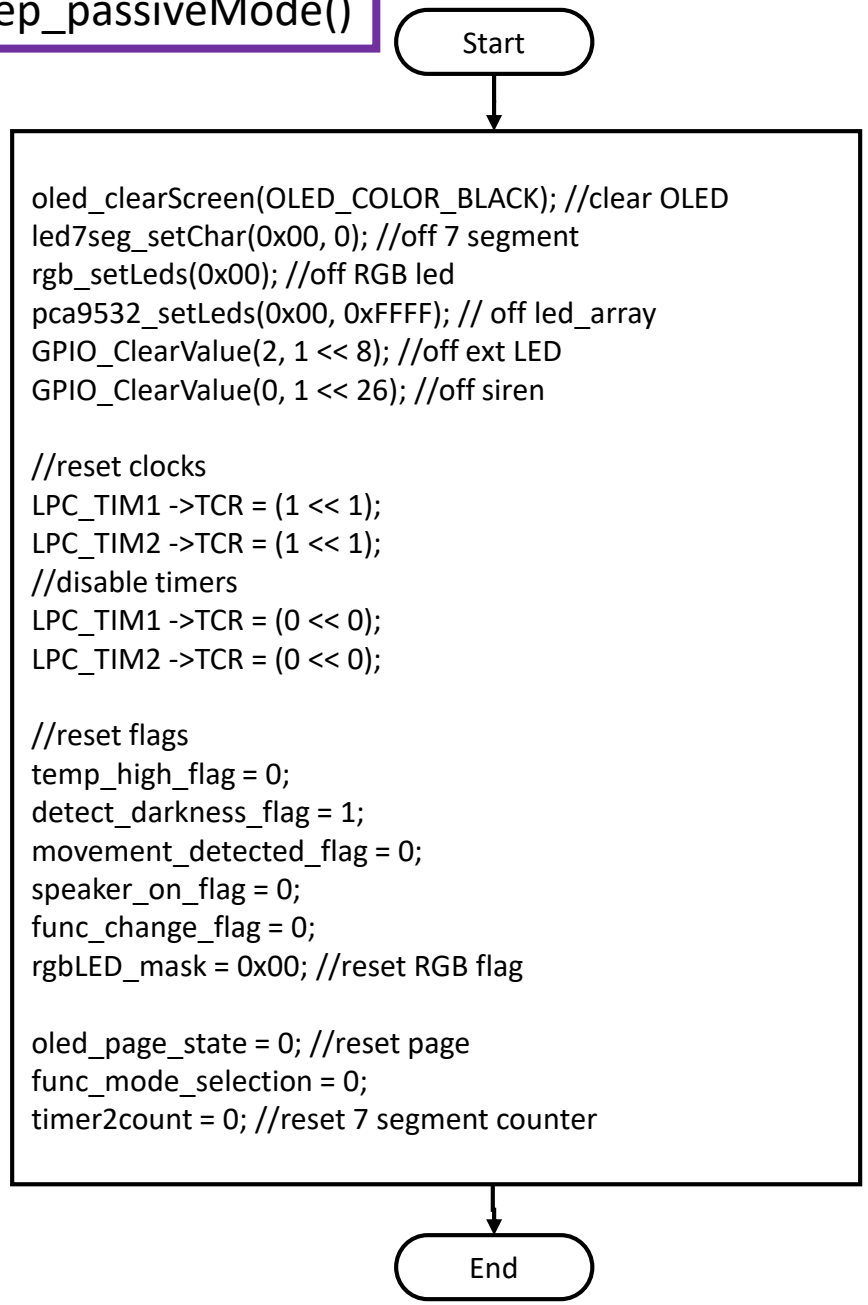
Main Loop



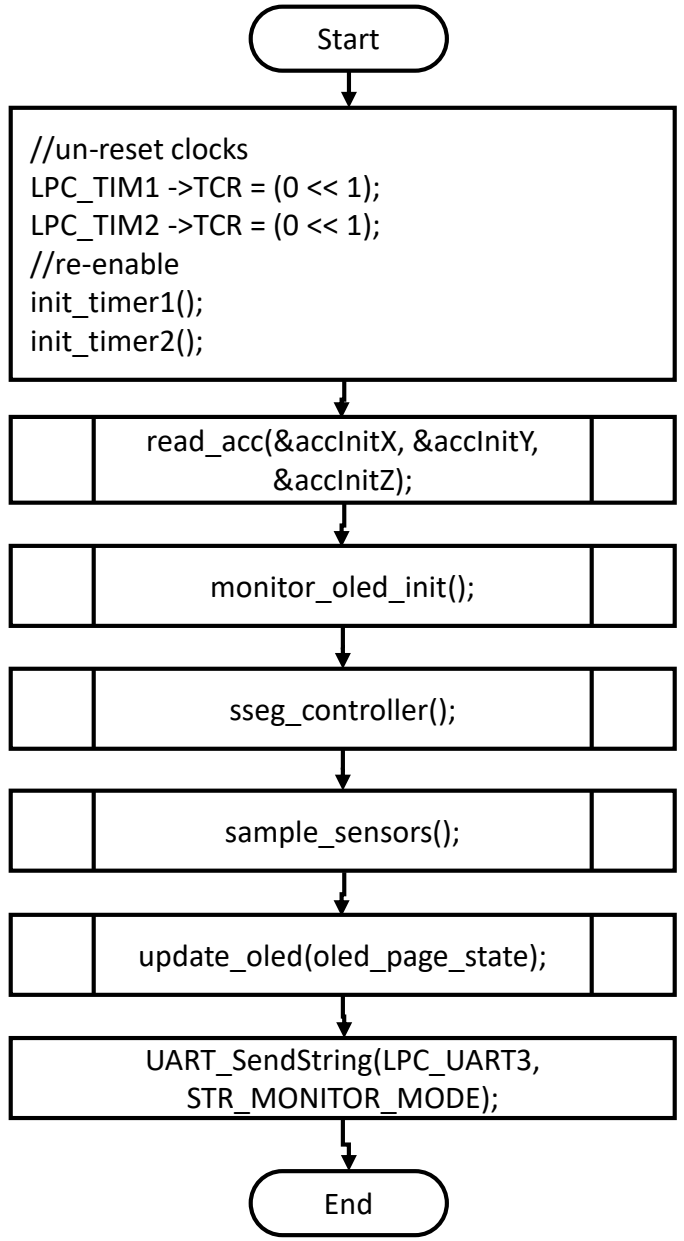
Initial Values

```
STR_MONITOR_MODE = "Entering
MONITOR Mode.\r\n";

temp_high_flag = 0;
detect_darkness_flag = 1;
movement_detected_flag = 0;
speaker_on_flag = 0;
func_change_flag = 0;
rgbLED_mask = 0x00;

oled_page_state = 0;
func_mode_selection = 0;
timer2count = 0
```

prep_passiveMode()



prep_monitorMode()



Initial Values

TEMP_HIGH_WARNING = 450;

sseg_flag = 0;

reinit_screen_flag = 0;

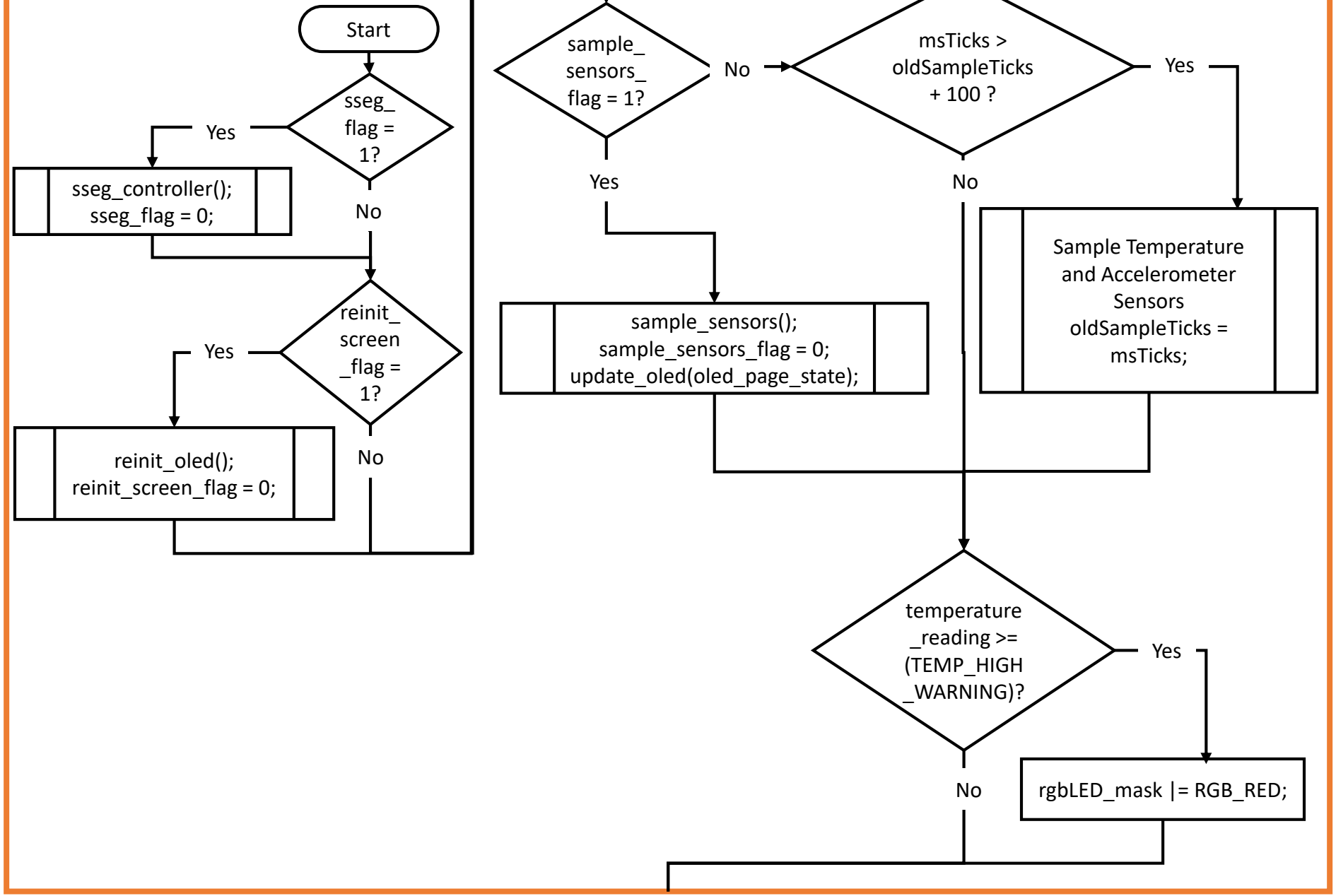
sample_sensors_flag = 0;

msTicks = 0;
oldSampleTicks = 0;

temp_high_flag = 0;
temperature_reading = 0;

rgbLED_mask = 0x00;

Execute MAIN FUNCTIONS (1)



Initial Values

movement_detected_flag = 0;

msTicks = 0;
lastMotionDetectedTicks = 0;

detect_darkness_flag = 1;

rgbLED_mask = 0x00;

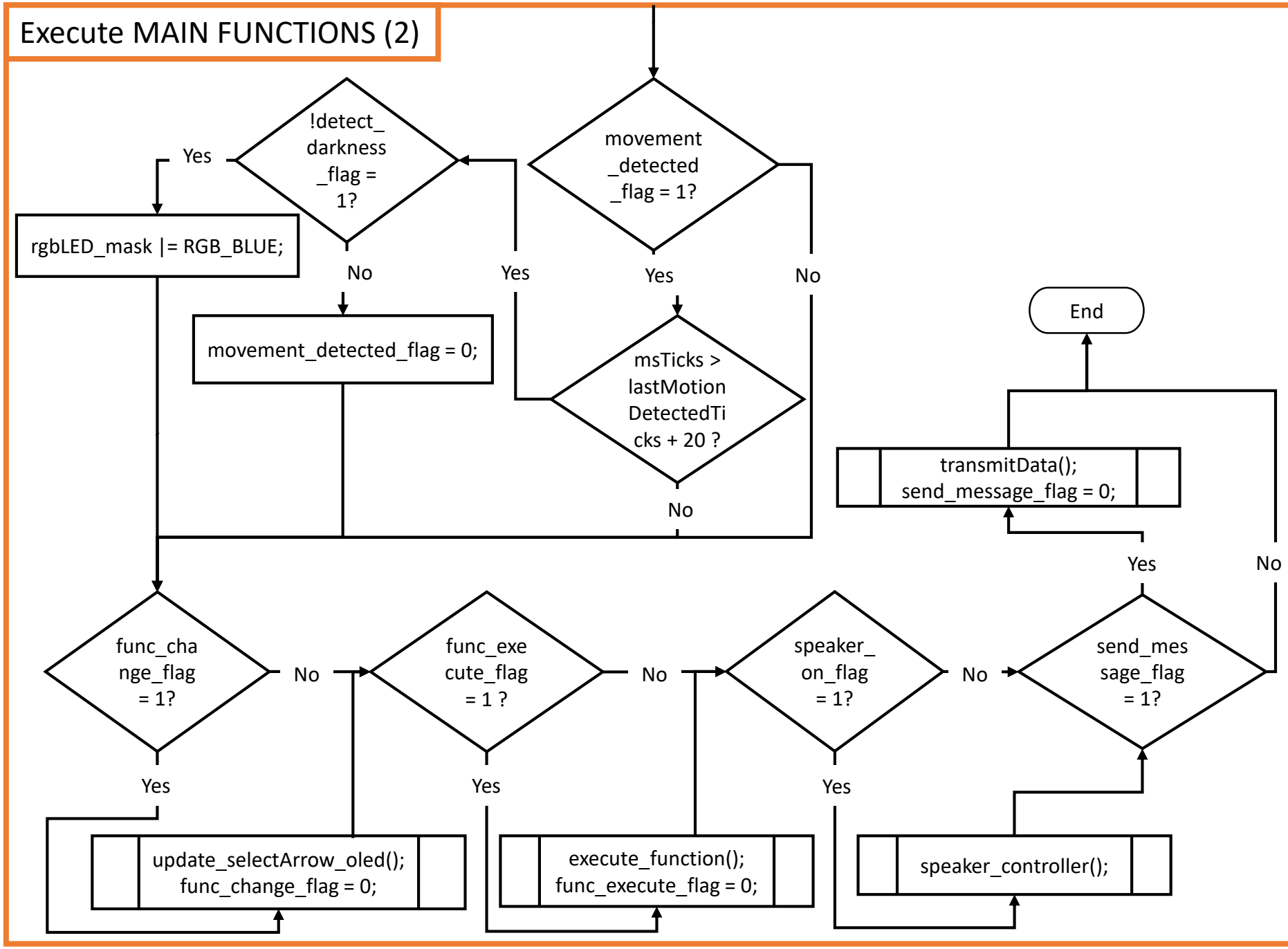
func_change_flag = 0;

func_execute_flag = 0;

speaker_on_flag = 0;

send_message_flag = 0;

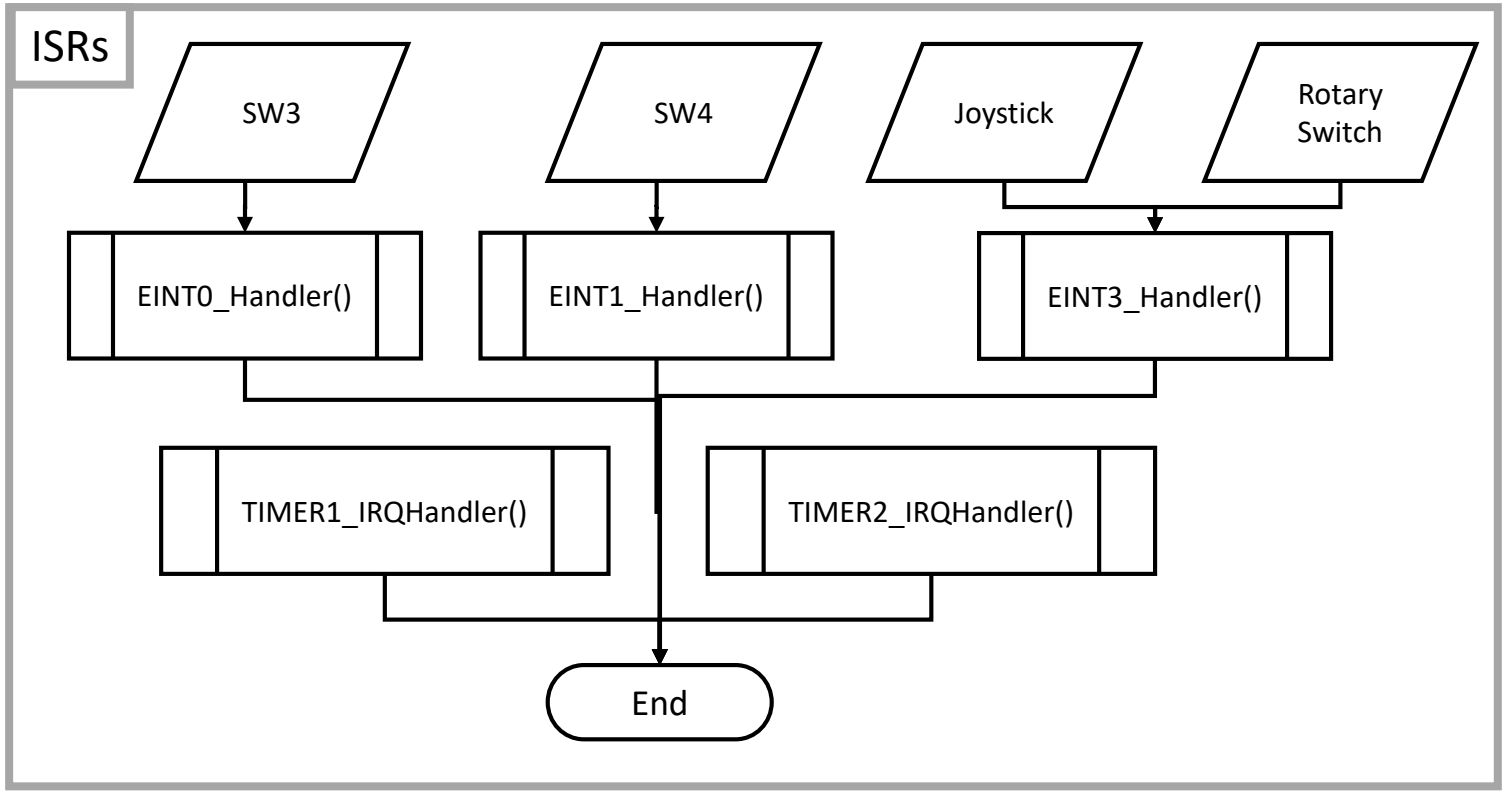
Execute MAIN FUNCTIONS (2)

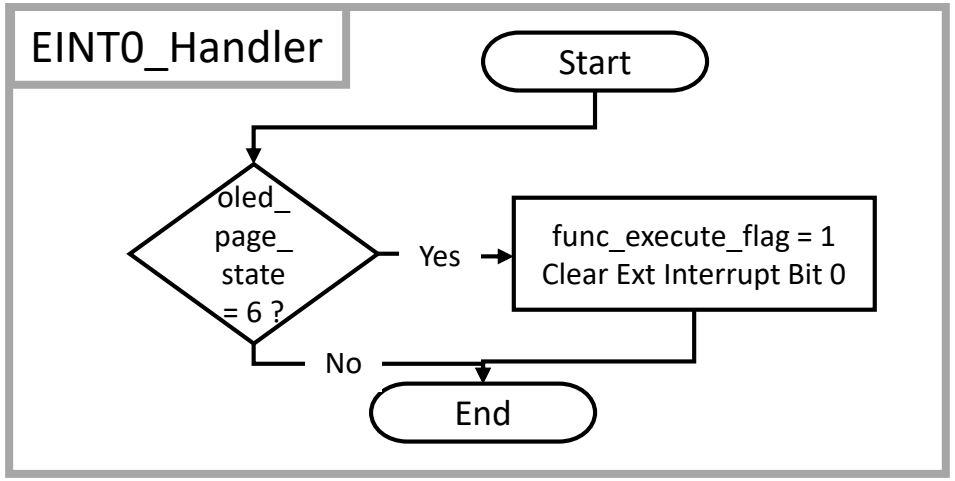


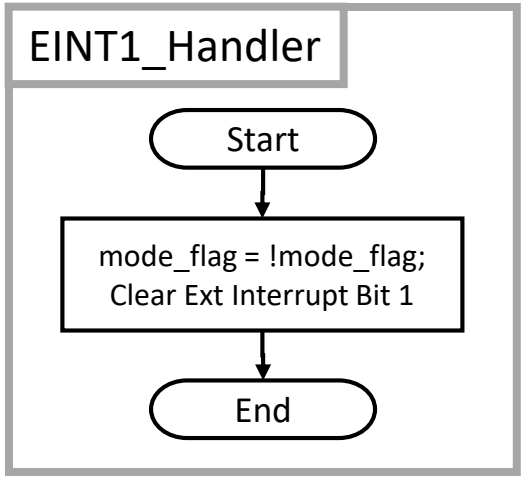
Initial Values

oled_page_state = 0;
func_execute_flag = 0;

mode_flag = 0;



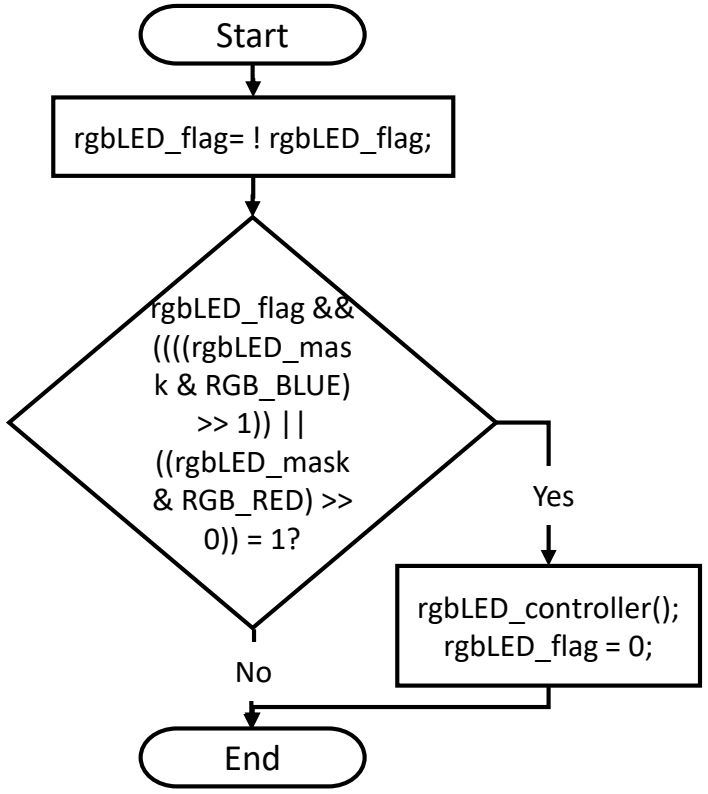




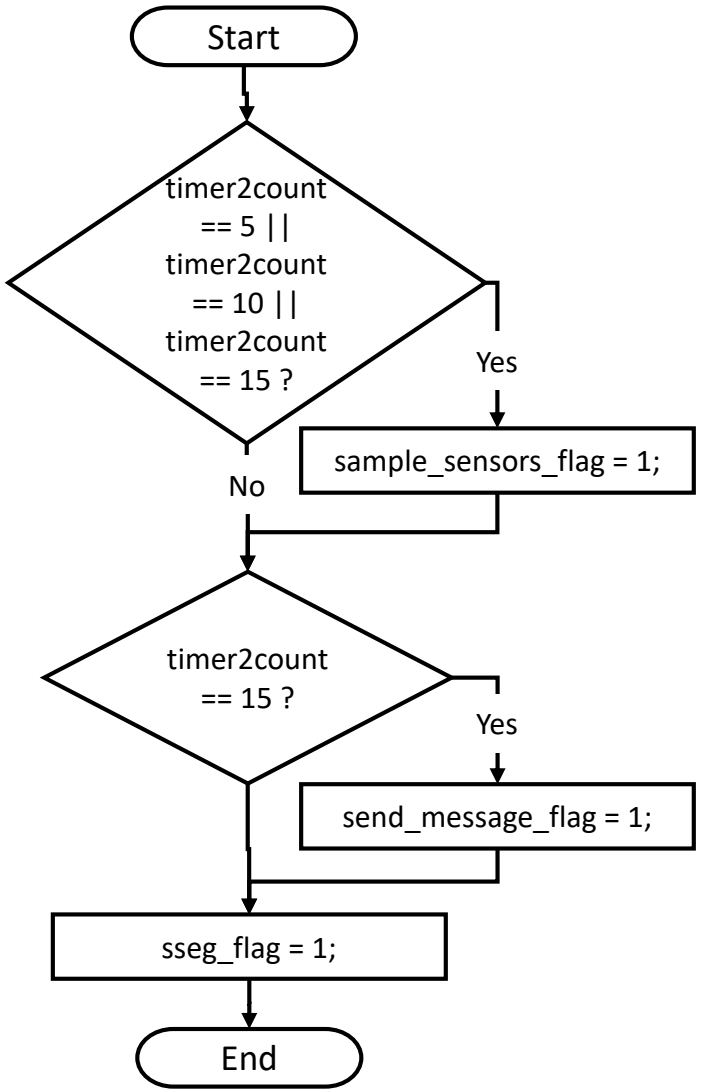
Initial Values

```
rgbLED_flag = 0;
rgbLED_mask = 0x00;

sample_sensors_flag = 0;
send_message_flag = 0;
sseg_flag = 0;

timer2count = 0;
```

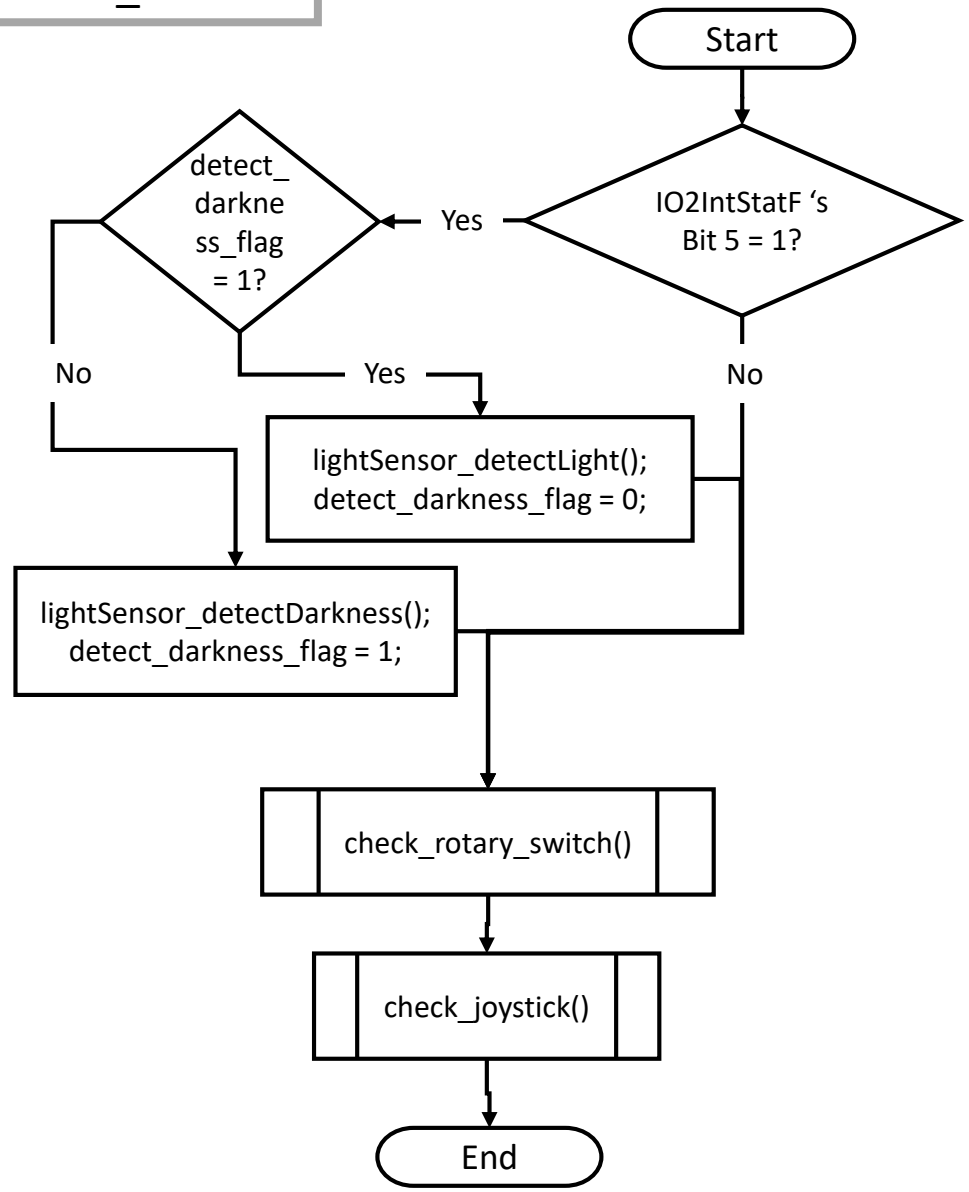
TIMER1_IRQHandler



TIMER2_IRQHandler



EINT3_Handler



Initial Values

SCREEN_CHG_DELAY = 100

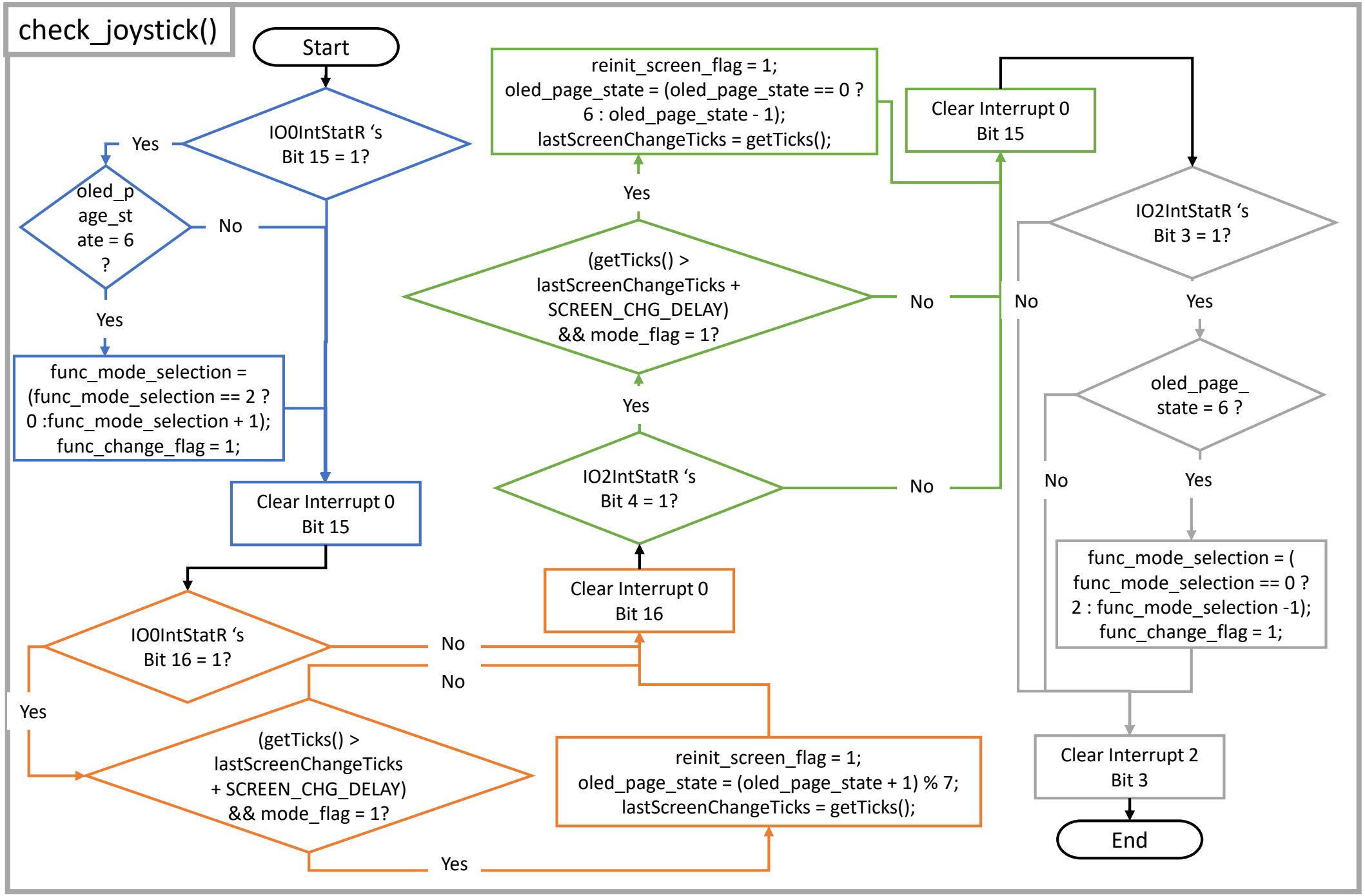
mode_flag = 0;

oled_page_state = 0;

func_mode_selection = 0;
func_change_flag = 0;

reinit_screen_flag = 0;

lastScreenChangeTicks = 0;



Initial Values

STR_FIRE_ALERT = "Fire was Detected.\r\n";

STR_DARK_ALERT = "Movement in darkness was Detected.\r\n";

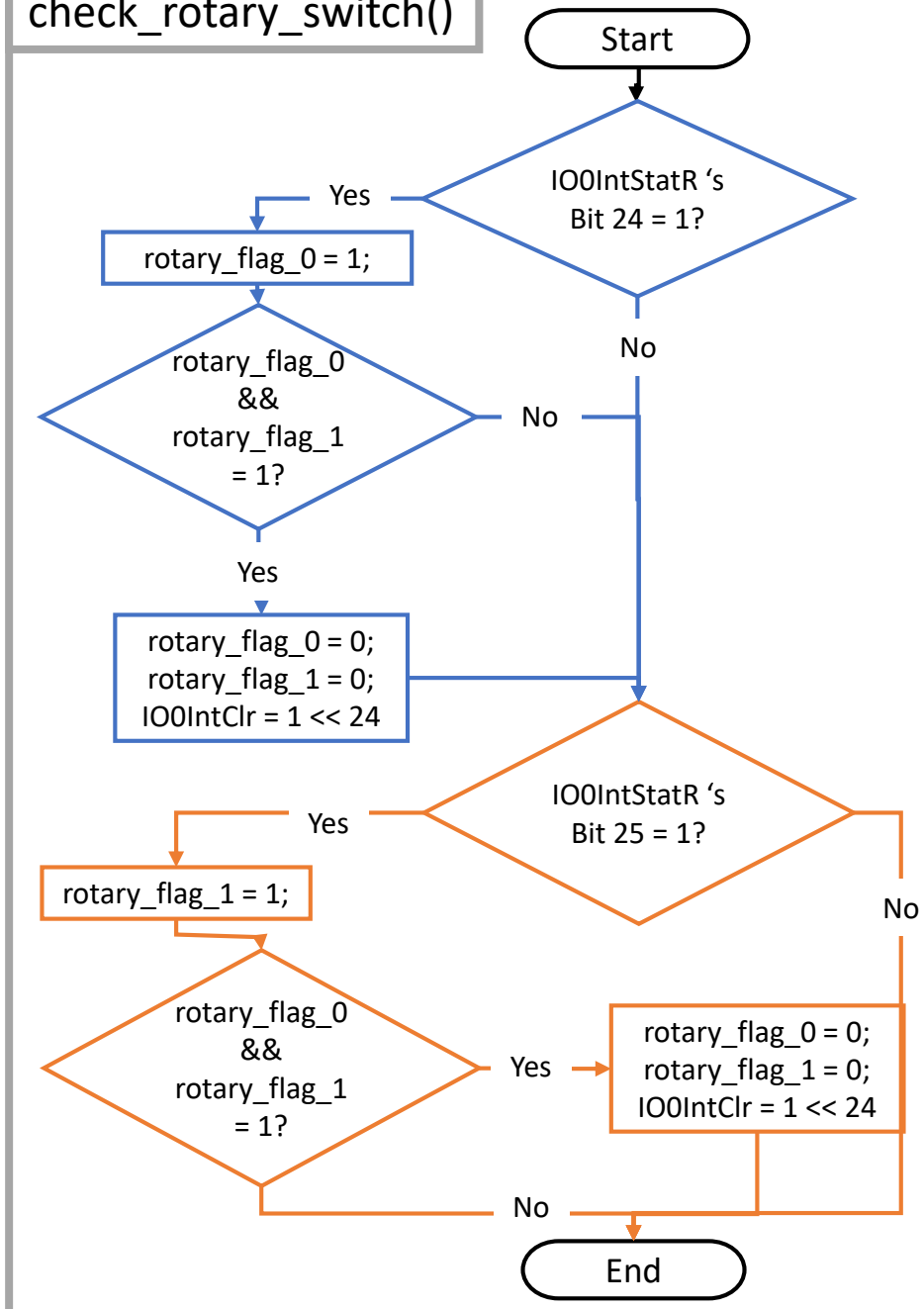
detect_darkness_flag = 1;

rotary_flag_0 = 0;
rotary_flag_ 1 = 0;

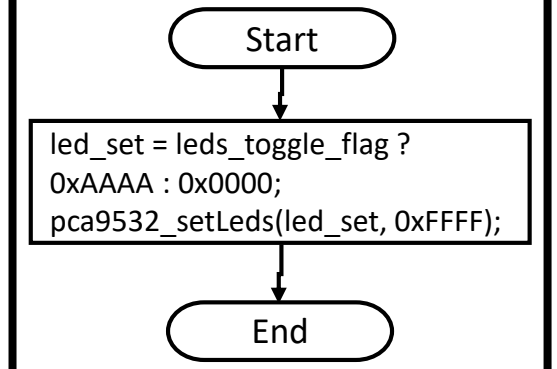
rgbLED_mask = 0x00;

led_set = 0;
leds_toggle_flag = 0;

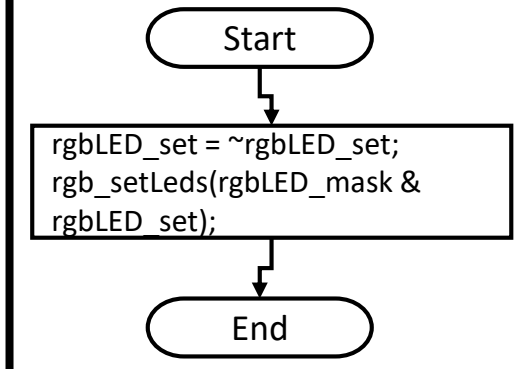
check_rotary_switch()



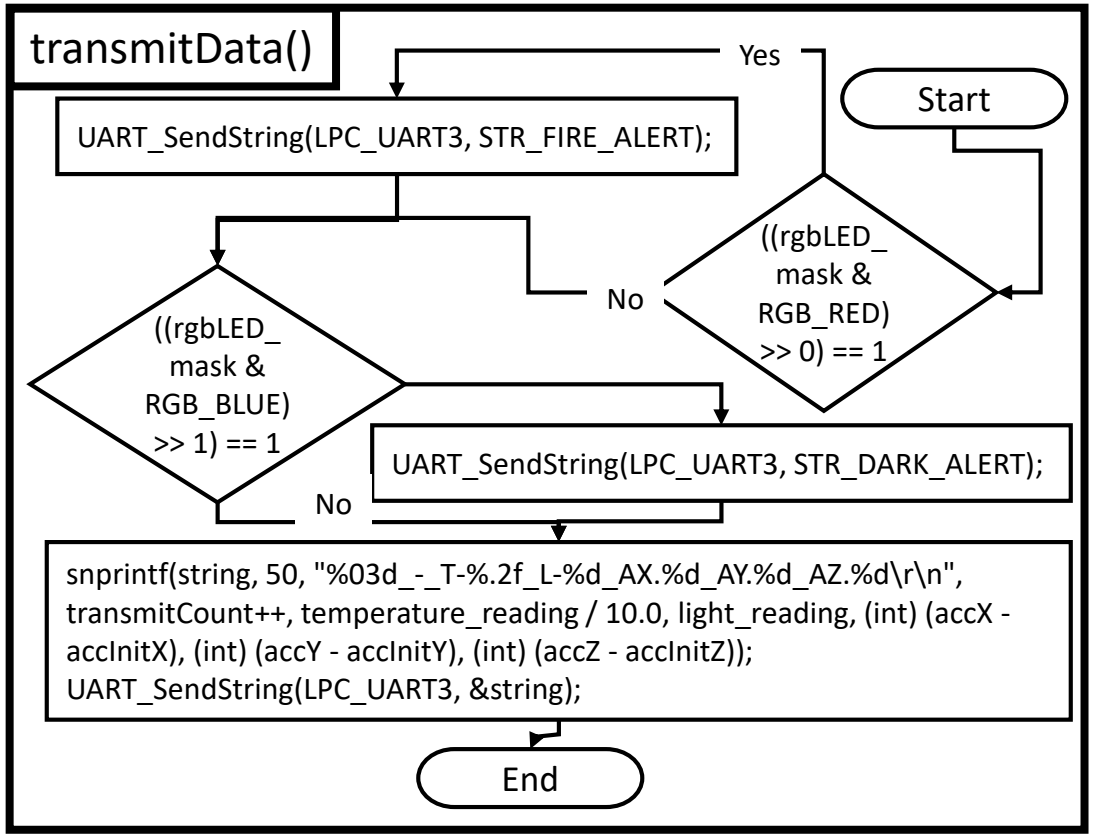
ledArray_controller()

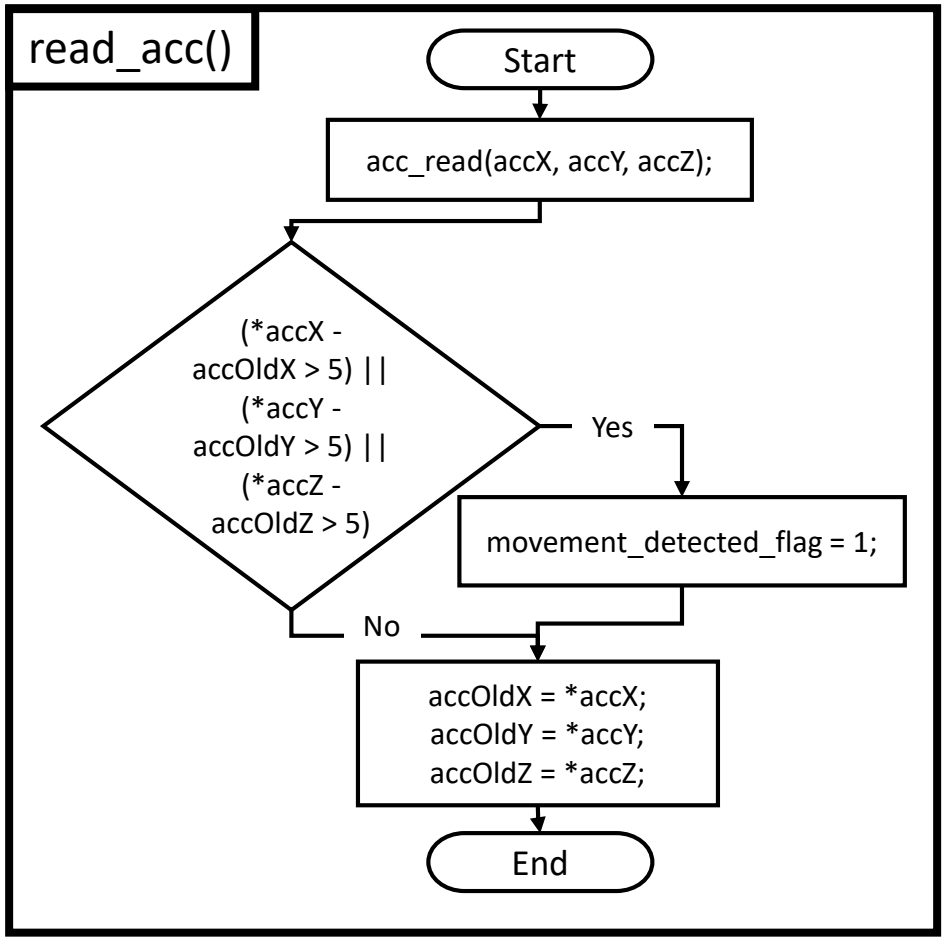


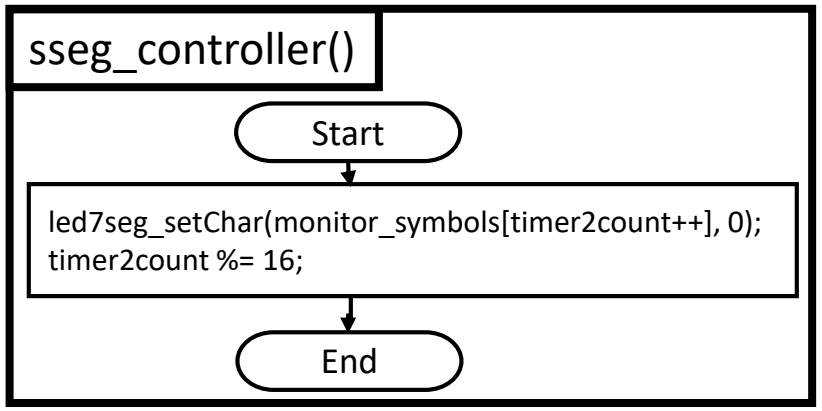
rgbLED_controller()

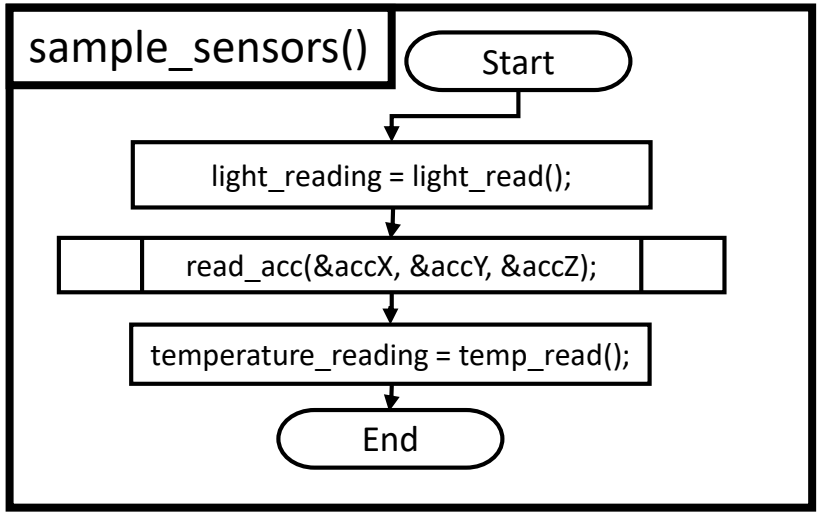


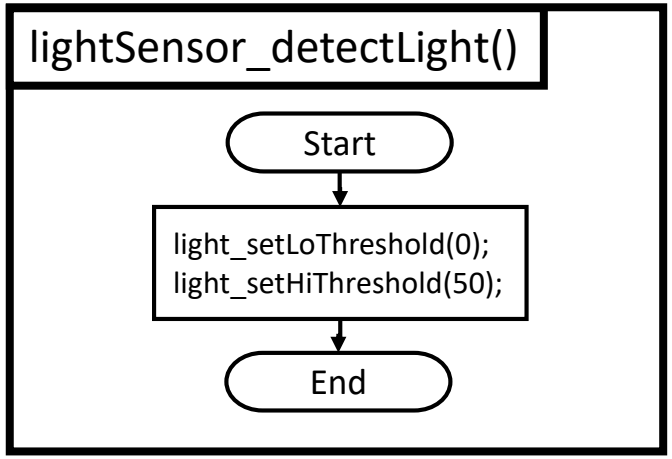
transmitData()

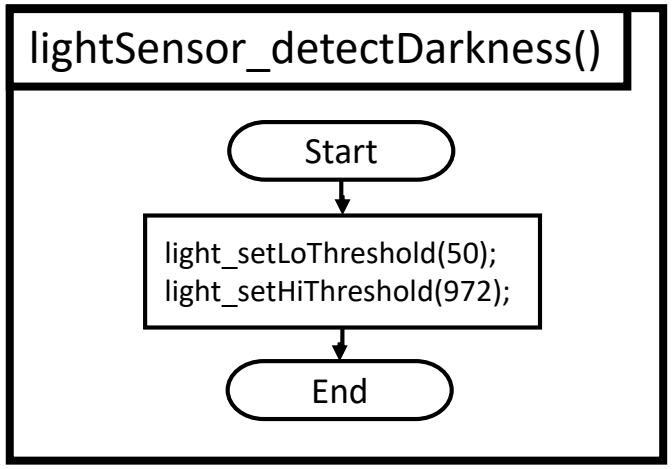


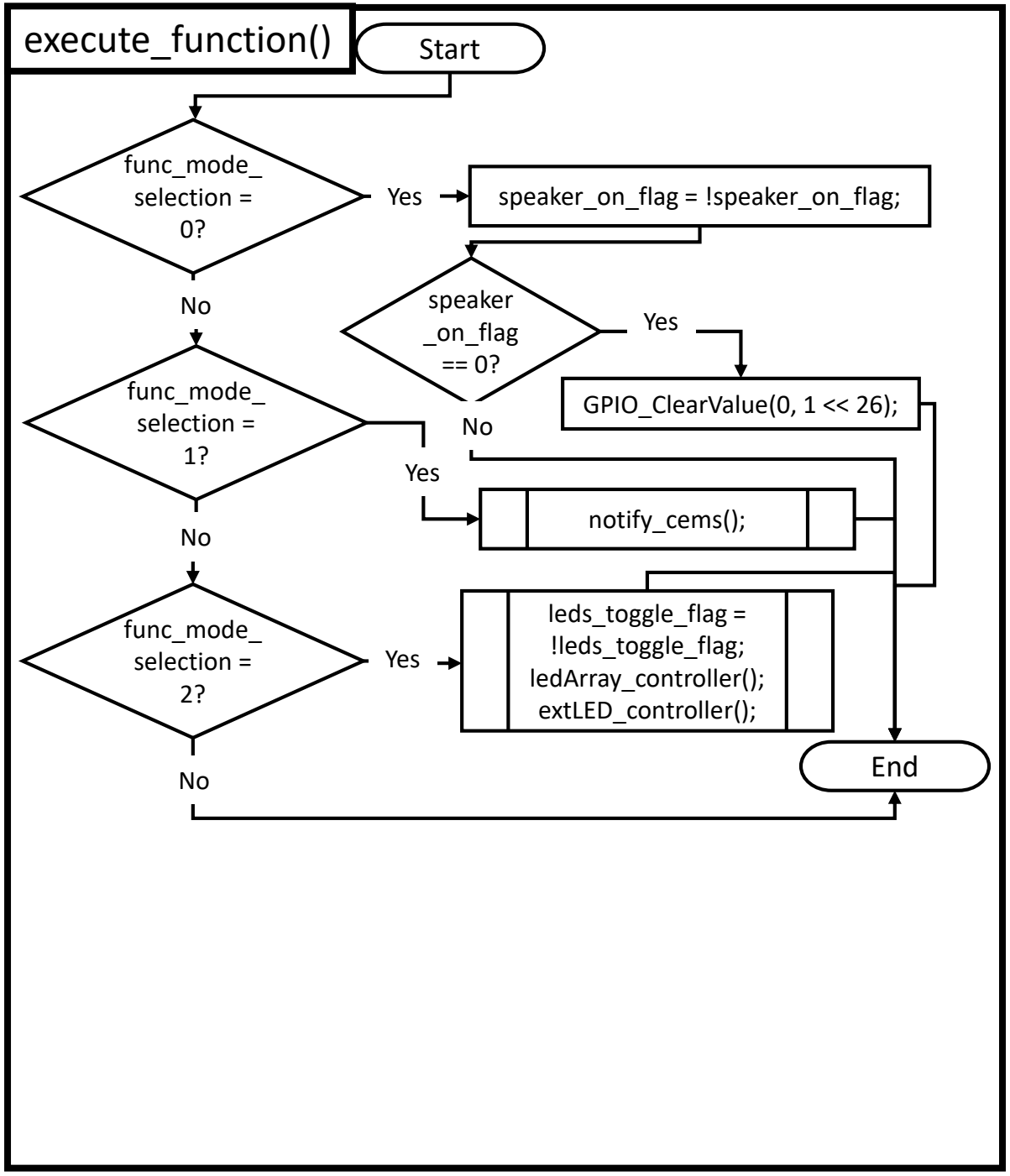

read_acc()
Start
acc_read(accX, accY, accZ);
(*accX - accOldX > 5) || (*accY - accOldY > 5) || (*accZ - accOldZ > 5)
Yes
movement_detected_flag = 1;
No
accOldX = *accX;
accOldY = *accY;
accOldZ = *accZ;
End

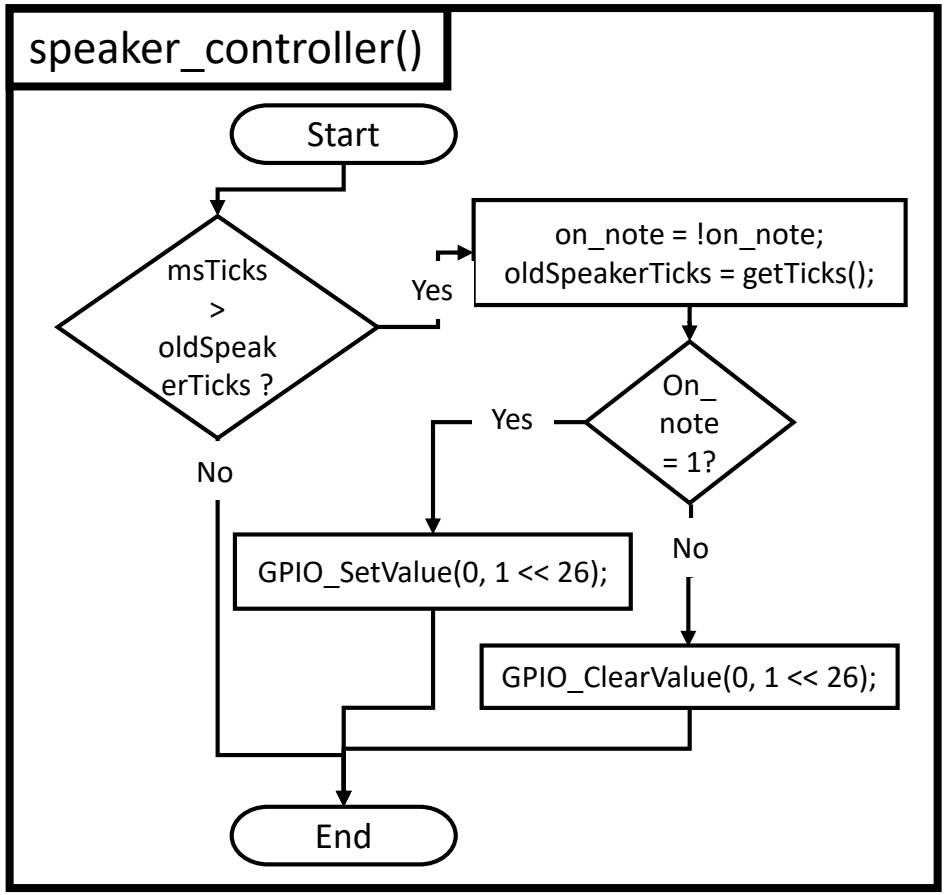

sseg_controller()
Start
led7seg_setChar(monitor_symbols[timer2count++], 0);
timer2count %= 16;
End

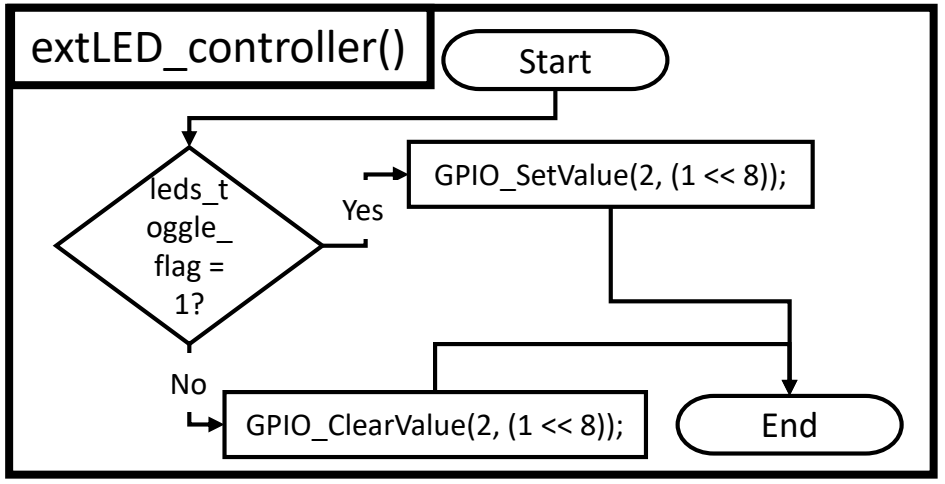

sample_sensors()
Start
light_reading = light_read();
read_acc(&accX, &accY, &accZ);
temperature_reading = temp_read();
End

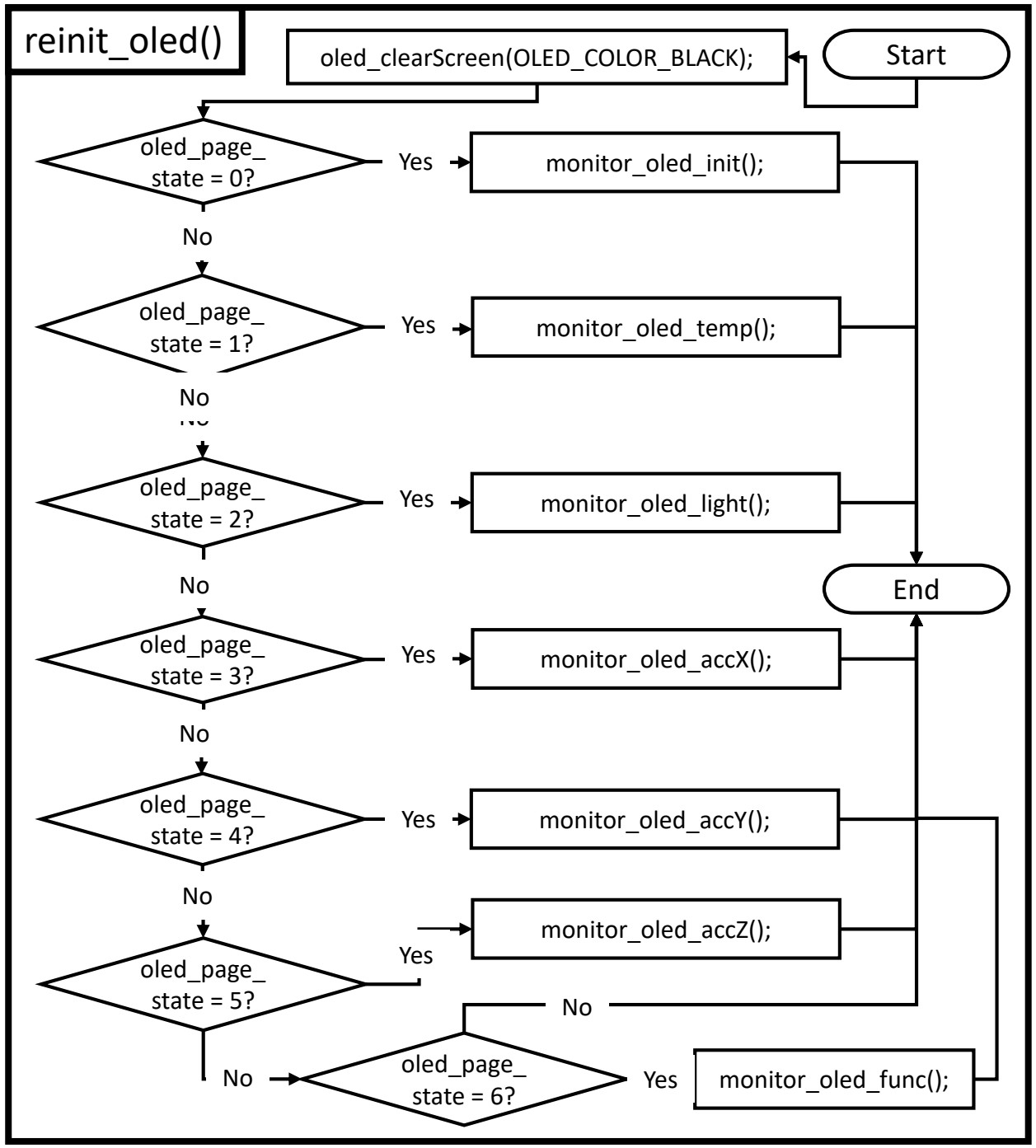

lightSensor_detectLight()
Start
light_setLoThreshold(0);
light_setHiThreshold(50);
End

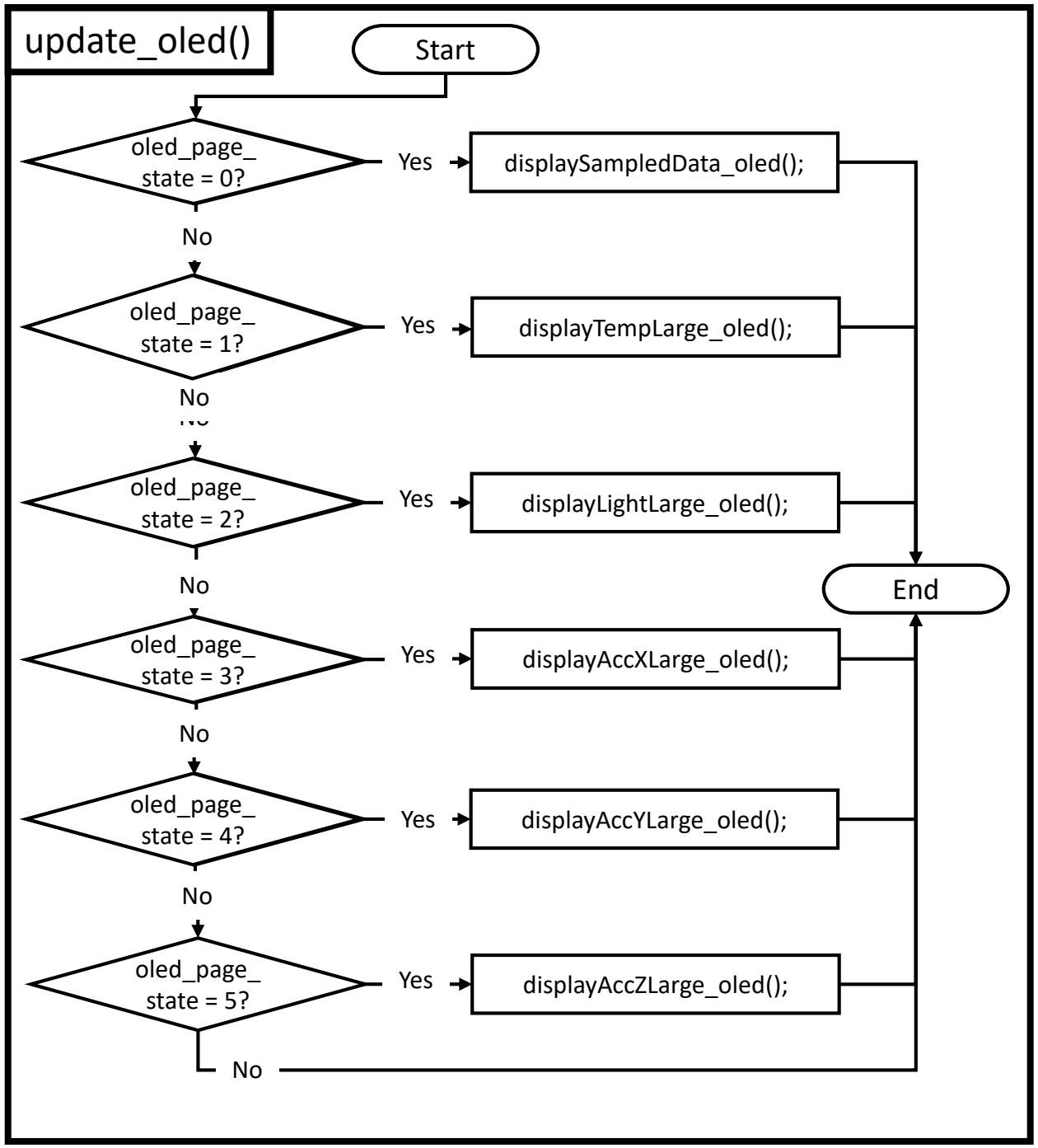

lightSensor_detectDarkness()
Start
light_setLoThreshold(50);
light_setHiThreshold(972);
End


execute_function()
Start
func_mode_ selection = 0?
Yes
speaker_on_flag = !speaker_on_flag;
No
speaker _on_flag == 0?
Yes
GPIO_ClearValue(0, 1 << 26);
No
func_mode_ selection = 1?
Yes
notify_cems();
No
func_mode_ selection = 2?
Yes
leds_toggle_flag = !leds_toggle_flag;
ledArray_controller();
extLED_controller();
No
End


speaker_controller()
Start
msTicks > oldSpeak erTicks ?
Yes
on_note = !on_note;
oldSpeakerTicks = getTicks();
On_ note = 1?
Yes
GPIO_SetValue(0, 1 << 26);
No
GPIO_ClearValue(0, 1 << 26);
No
End


extLED_controller()
Start
leds_t oggle_ flag = 1?
Yes
GPIO_SetValue(2, (1 << 8));
No
GPIO_ClearValue(2, (1 << 8));
End

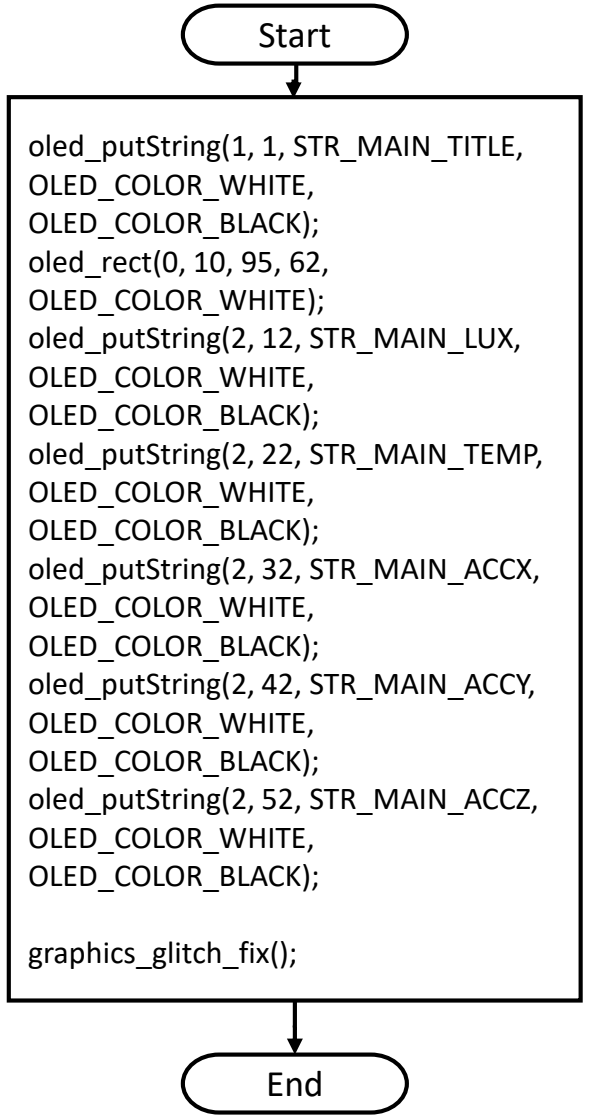

reinit_oled()
oled_clearScreen(OLED_COLOR_BLACK);
Start
oled_page_ state = 0?
Yes
monitor_oled_init();
No
oled_page_ state = 1?
Yes
monitor_oled_temp();
No
oled_page_ state = 2?
Yes
monitor_oled_light();
No
End
oled_page_ state = 3?
Yes
monitor_oled_accX();
No
oled_page_ state = 4?
Yes
monitor_oled_accY();
No
monitor_oled_accZ();
oled_page_ state = 5?
Yes
No
oled_page_ state = 6?
No
Yes
monitor_oled_func();


update_oled()
Start
oled_page_ state = 0?
Yes
displaySampledData_oled();
No
oled_page_ state = 1?
Yes
displayTempLarge_oled();
No
oled_page_ state = 2?
Yes
displayLightLarge_oled();
No
End
oled_page_ state = 3?
Yes
displayAccXLarge_oled();
No
oled_page_ state = 4?
Yes
displayAccYLarge_oled();
No
oled_page_ state = 5?
Yes
displayAccZLarge_oled();
No

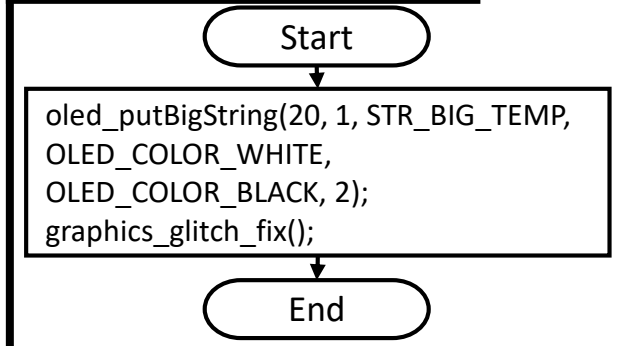
Initial Values

```
STR_CEMS_ALERT = "User %s has
requested for assistance.\r\n";

STR_ARROW_CHAR = ">";
STR_BLANK_CHAR = " ";

STR_INTVALUES_OUTPUT = "%d    ";
STR_UINTVALUES_OUTPUT = "%u    ";
STR_FLOATVALUES_OUTPUT = "%.2f   ";

STR_FUNC_TITLE = "Select Function:";
STR_FUNC_1 = "Siren      ";
STR_FUNC_2 = "SOS to CEMS ";
STR_FUNC_3 = "Lights     ";

STR_MAIN_LUX = "LUX : ";
STR_MAIN_TEMP = "TEMP: ";
STR_MAIN_ACCX = "ACCX: ";
STR_MAIN_ACCY = "ACCY: ";
STR_MAIN_ACCZ = "ACCZ: ";
STR_MAIN_TITLE = "MODE: MONITOR";

STR_BIG_TEMP = "TEMP   ";
STR_BIG_LIGHT = "LUX   ";
STR_BIG_ACCX = "ACC X  ";
STR_BIG_ACCY = "ACC Y  ";
STR_BIG_ACCZ = "ACC Z  ";
```

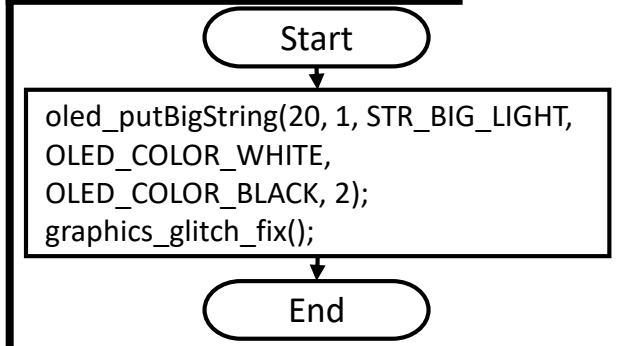
monitor_oled_init()



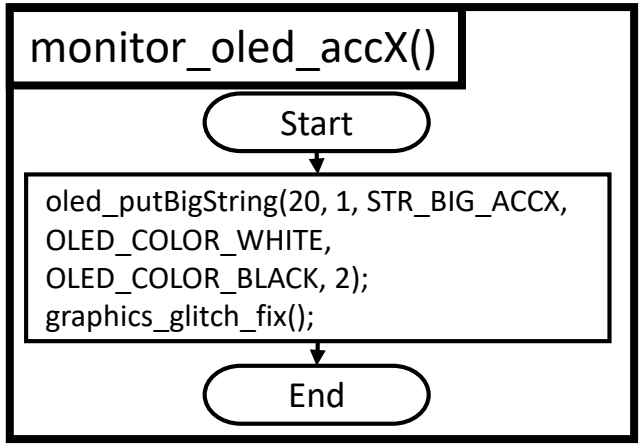
monitor_oled_temp()



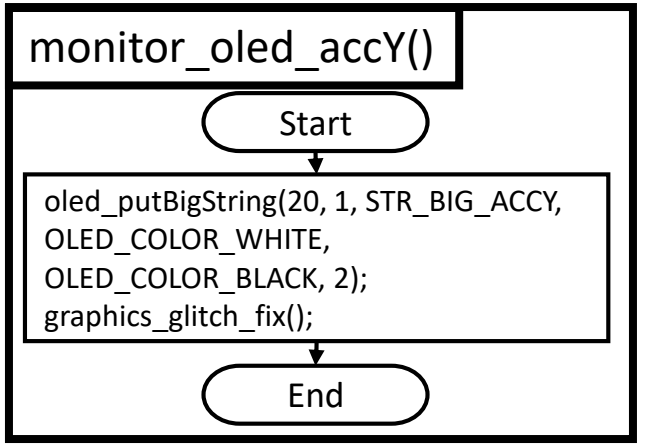
monitor_oled_light()



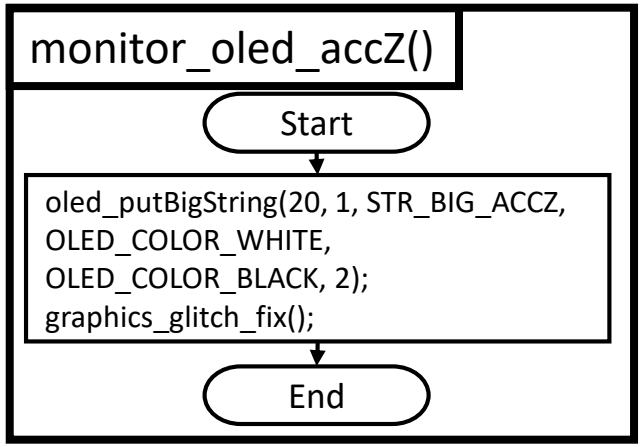
monitor_oled_accX()



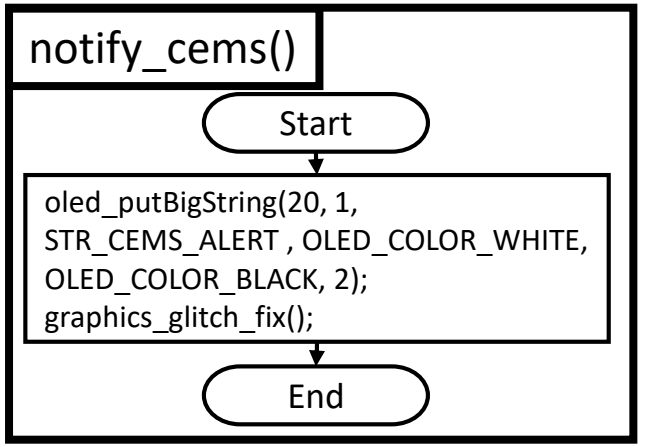
monitor_oled_accY()



monitor_oled_accZ()



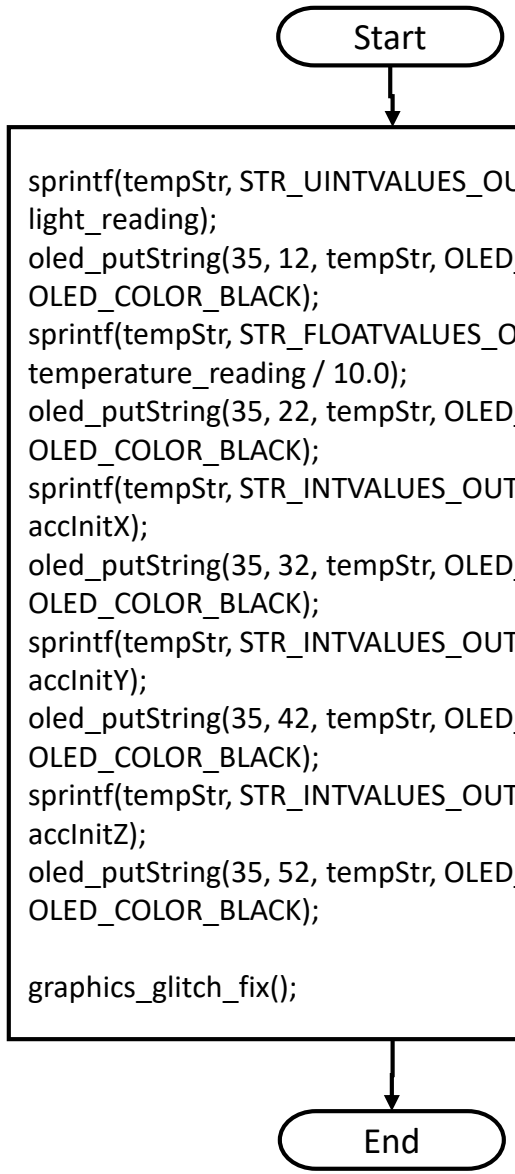
notify_cems()



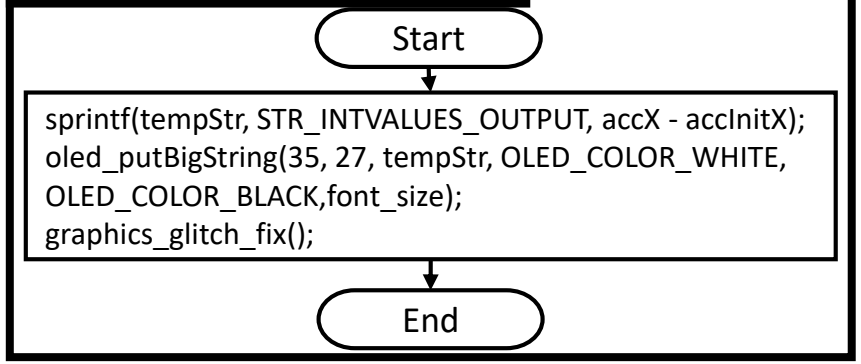
Initial Values

```
STR_CEMS_ALERT = "User %s has
requested for assistance.\r\n";

STR_INTVALUES_OUTPUT = "%d    ";
STR_UINTVALUES_OUTPUT = "%u    ";
STR_FLOATVALUES_OUTPUT = "%.2f   ";

STR_FUNC_TITLE = "Select Function:";
STR_FUNC_1 = "Siren       ";
STR_FUNC_2 = "SOS to CEMS ";
STR_FUNC_3 = "Lights      ";

STR_MAIN_LUX = "LUX : ";
STR_MAIN_TEMP = "TEMP: ";
STR_MAIN_ACCX = "ACCX: ";
STR_MAIN_ACCY = "ACCY: ";
STR_MAIN_ACCZ = "ACCZ: ";
STR_MAIN_TITLE = "MODE: MONITOR";

STR_BIG_TEMP = "TEMP   ";
STR_BIG_LIGHT = "LUX   ";
STR_BIG_ACCX = "ACC X  ";
STR_BIG_ACCY = "ACC Y  ";
STR_BIG_ACCZ = "ACC Z  ";
```

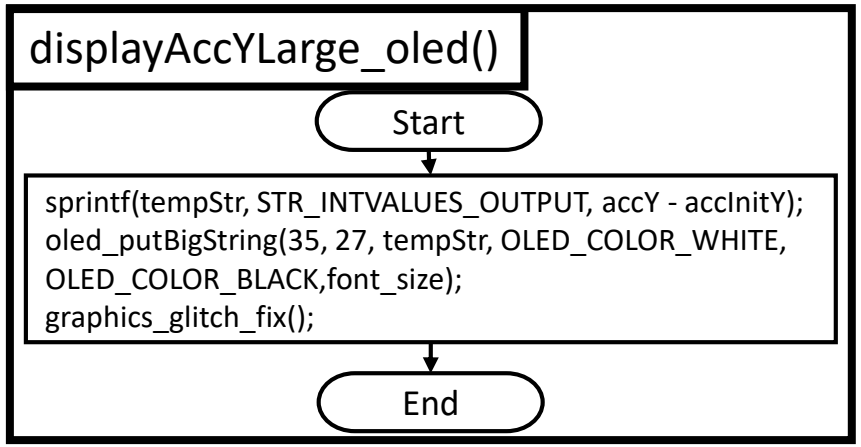
displaySampledData_oled()



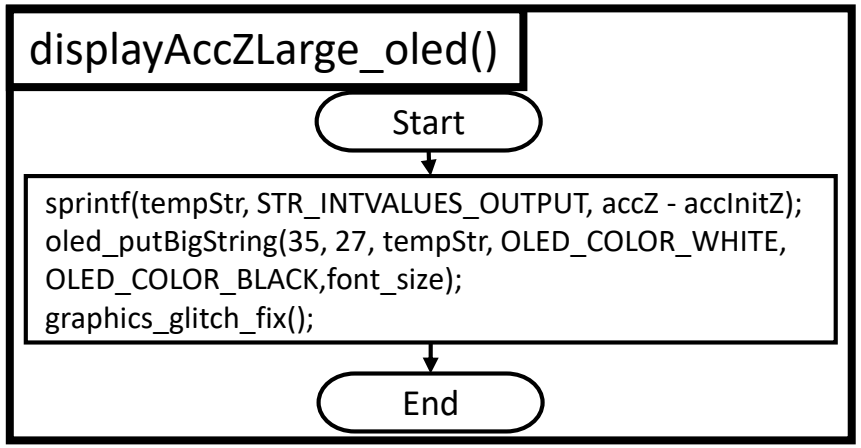
displayAccXLarge_oled()



displayAccYLarge_oled()



displayAccZLarge_oled()



Initial Values

```
STR_ARROW_CHAR = ">";
STR_BLANK_CHAR = " ";

STR_UINTVALUES_OUTPUT = "%u    ";
STR_FLOATVALUES_OUTPUT = "%.2f   ";
```

displayTempLarge_oled()

Start

```
sprintf(tempStr, STR_FLOATVALUES_OUTPUT,
temperature_reading / 10.0);
oled_putBigString(15, 27, tempStr, OLED_COLOR_WHITE,
OLED_COLOR_BLACK,font_size);
```

End

displayLightLarge_oled()

Start

```
sprintf(tempStr, STR_UINTVALUES_OUTPUT, light_reading);
oled_putBigString(25, 27, tempStr, OLED_COLOR_WHITE,
OLED_COLOR_BLACK, font_size);
```

End

update_selectArrow_oled()

Start

```
oled_putString(2, 13, STR_BLANK_CHAR,
OLED_COLOR_WHITE, OLED_COLOR_BLACK);
oled_putString(2, 26, STR_BLANK_CHAR,
OLED_COLOR_WHITE, OLED_COLOR_BLACK);
oled_putString(2, 39, STR_BLANK_CHAR,
OLED_COLOR_WHITE, OLED_COLOR_BLACK);
oled_putString(2, 13 * (1 + func_mode_selection),
STR_ARROW_CHAR,OLED_COLOR_WHITE,
OLED_COLOR_BLACK);

graphics_glitch_fix();
```

End