Sozib Al Mamun

Embedded System Engineer

Print icon or logo : use some of function to print icon and logo

```c
void iconPrint(int x_offset, int y_offset, uint8_t w, uint8_t h,char*
logobuff,uint16_t color ,camera_fb_t *buff) {
    // Ensure logo fits within the buffer dimensions
    if (x_offset + w > buff->width || y_offset + h > buff->height) {
        printf("Logo position out of bounds\n");

        return;
    }
    for (int y = 0; y < h; y++) {
        for (int x = 0; x < w; x++) {
            int logo_index = y * w + x;
            int buff_index = ((y + y_offset) * buff->width + (x +
x_offset)) * 2; // 2 bytes per pixel
            // Copy the logo pixel to the buffer
            uint8_t pixel = logobuff[logo_index];
            if(pixel==0){// white color
            buff->buf[buff_index] = color>>8;
            buff->buf[buff_index + 1] = color&0xff;
            }
        }
    }
}
```

Generate uniq id:

```c
uint64_t generate_unique_id(void)
{
    uint8_t mac[6];
    esp_read_mac(mac, ESP_MAC_WIFI_STA);
    uint64_t unique_id = ((uint64_t)mac[0] << 40) | ((uint64_t)mac[1] << 32) | ((uint64_t)mac[2] << 24) |
                          ((uint64_t)mac[3] << 16) | ((uint64_t)mac[4] << 8) | mac[5];
    return unique_id;
}
```

Print serial no of device:

```c
void writeSn(camera_fb_t *buff){

    char tempFrame[13] ;
    snprintf(tempFrame, sizeof(tempFrame), "SN-%09llu", generate_unique_id());

    uint16_t len = (buff->width-(strlen(tempFrame)*LETTER_WIDTH))-3;//x start poss

    WriteString(len, buff->height-(LETTER_HEIGHT+3),tempFrame,buff);
}
```

Display 7 segment time: for display 7 segment time in sleep mode use some of function

```c
void sleepTimeDate(camera_fb_t *buff);
void wrighSingle7segment(int x_offset, int y_offset, char c, camera_fb_t *buff);
void WriteMulti7segment(int x_offset, int y_offset, const char *str, camera_fb_t *buff);
void timeDisplay(uint8_t x, uint8_t y, uint8_t value,camera_fb_t *buff);
```

1

Make time library

To calculate year to second it needs a time library. Here is some function for that.

```c
static const uint8_t days_in_month[] = {31, 28, 31, 30, 31, 30, 31, 31,
30, 31, 30, 31};
const char* day_names[] = {
    "Sun", "Mon", "Tue", "Wed", "Thu", "Fri","Sat"
};
// Function to check if a year is a leap year
static bool is_leap_year(uint16_t year) {
    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
}
// Time structure
typedef struct {
    uint16_t year;
    uint8_t month;
    uint8_t day;
    uint8_t hour;
    uint8_t minute;
    uint8_t second;
} time_library_time_t;

// Function to add seconds to a time structure
static void add_seconds_to_time(time_library_time_t *time, uint32_t
seconds);
// Function to initialize the time library
void time_library_init(time_library_time_t *initial_time);
// Function to set the current time manually
void time_library_set_time(time_library_time_t *time);
// Function to get the current time
void time_library_get_time(time_library_time_t *time);
void get_time(time_library_time_t *time, bool is_pm);
// Function to calculate the day of the week
uint8_t calculate_day_of_week(uint16_t year, uint8_t month, uint8_t day);
// Function to calculate the elapsed time in milliseconds
uint32_t time_library_elapsed_time_ms(uint32_t start_time);
// Function to get the current time in milliseconds
```

```
uint32_t time_library_get_time_ms(void);
```

Time library not yet complete. Next week i will with this library