

Modul 11**Komunikasi Nirkabel antar ESP32 menggunakan modul WiFi
dan TFT LCD menggunakan LVGL****11.1 Tujuan Praktikum**

1. Praktikan dapat memahami cara kerja komunikasi nirkabel untuk TFT LCD.
2. Praktikan dapat memahami penggunaan LVGL untuk mengatur tampilan pada TFT LCD.
3. Praktikan dapat mengkonfigurasi TFT LCD dengan Touchscreen pada ESP32.
4. Praktikan dapat menampilkan gambar dan elemen grafis lain pada TFT LCD menggunakan LVGL.
5. Praktikan dapat membuat antarmuka interaktif dengan input sentuhan untuk mengontrol tampilan di layar TFT.

11.2 Alat dan Bahan

1. ESP32-WROVER
2. LCD TFT
3. Jumper
4. Project Board (Optional)
5. Kabel Micro USB Type B
6. Arduino IDE

11.3 Dasar Teori Modul 11**11.3.1 Komunikasi Nirkabel****11.3.1.1 Pengertian Komunikasi Nirkabel**

Komunikasi nirkabel adalah sistem komunikasi yang tidak menggunakan media fisik seperti kabel atau kawat untuk mengirimkan informasi. Sebagai gantinya, sistem ini memanfaatkan gelombang elektromagnetik, seperti gelombang radio, gelombang mikro, atau cahaya inframerah, sebagai media transmisinya. Komunikasi nirkabel juga sering disebut komunikasi radio, karena awalnya memanfaatkan gelombang radio sebagai media utama.

11.3.1.2 Cara Kerja Komunikasi Nirkabel

1. Pengirim (Transmitter): Mengonversi data, seperti suara, gambar, atau teks, menjadi sinyal elektromagnetik.
2. Media Transmisi: Sinyal tersebut dikirim melalui udara menggunakan gelombang elektromagnetik.
3. Penerima (Receiver): Menangkap sinyal dan mengubahnya kembali menjadi data yang dapat dimengerti.

11.3.2 LCD TFT



11.3.2.1 Pengertian TFT LCD

TFT LCD (Thin-Film Transistor Liquid Crystal Display) adalah jenis layar yang menggunakan teknologi transistor untuk mengatur piksel-piksel yang menghasilkan gambar. Layar ini menghasilkan tampilan warna yang cerah dan jelas, serta respons cepat, cocok untuk menampilkan gambar atau teks dengan detail. TFT LCD membutuhkan pencahayaan tambahan (backlight) untuk menghasilkan tampilan yang terang. Meskipun lebih hemat daya, TFT LCD memiliki kekurangan seperti sudut pandang yang terbatas dan kontras yang tidak setinggi layar jenis lain, seperti OLED.

11.3.2.2 Kelebihan dan Kekurangan TFT LCD

1. Kelebihan:

- Tampilan Warna yang Jelas dan Tajam:

TFT LCD memiliki kemampuan untuk menampilkan gambar dengan kualitas warna yang sangat baik, menjadikannya ideal untuk aplikasi yang memerlukan tampilan grafis yang jelas dan detail.

- Respon Cepat dan Efisien:

TFT LCD memiliki waktu respon yang cepat, sehingga cocok untuk aplikasi yang membutuhkan refresh rate tinggi seperti tampilan real-time.

- Konsumsi Daya yang Cukup Rendah:

Dibandingkan dengan jenis tampilan lain seperti OLED, TFT LCD cenderung lebih hemat daya, terutama pada model dengan backlight LED.

2. Kekurangan:

- Sudut Pandang Terbatas:

Meskipun kualitas gambar bagus, sudut pandang pada TFT LCD cenderung terbatas. Gambar bisa terlihat pudar atau terdistorsi jika dilihat dari sisi samping.

- Ukuran dan Ketebalan:

TFT LCD biasanya lebih tebal dan lebih besar dibandingkan dengan layar OLED atau teknologi layar terbaru lainnya, membuatnya kurang ideal untuk aplikasi yang mengutamakan ruang atau portabilitas.

- Kontras yang Kurang Tinggi:

Meskipun memiliki warna yang baik, kontras pada TFT LCD tidak setinggi OLED, yang dapat membuat tampilan terlihat kurang tajam dalam kondisi pencahayaan tertentu.

11.3.2.3 Jenis – Jenis TFT LCD

1. Matriks Aktif

Matriks Aktif adalah teknologi layar LCD dengan transistor tipis (TFT) untuk mengontrol piksel secara individu, menghasilkan gambar tajam dan respons cepat. Teknologi ini cocok untuk perangkat modern seperti smartphone, televisi, dan monitor.

2. Matriks Pasif

Matriks Pasif adalah teknologi layar LCD yang mengontrol piksel melalui elektroda kaca tanpa transistor tambahan, sehingga lebih sederhana namun menghasilkan kualitas gambar lebih rendah dibanding matriks aktif. Teknologi ini cocok untuk perangkat sederhana seperti kalkulator dan jam digital.

11.3.2.4 Cara Kerja TFT LCD

- Transistor Tipis: TFT menggunakan transistor untuk mengendalikan cairan kristal yang terletak di antara dua lapisan kaca.
- Backlight: Karena LCD tidak memancarkan cahaya sendiri, diperlukan backlight (pencahayaan latar) untuk membuat layar terlihat.
- Piksel: Setiap piksel pada layar TFT terdiri dari sub-piksel merah, hijau, dan biru yang dapat diatur intensitas cahayanya untuk menghasilkan berbagai warna.

11.3.3 LVGL

11.3.3.1 Pengertian LVGL

LVGL adalah pustaka (library) grafis open-source yang dirancang untuk aplikasi embedded system, seperti mikrokontroler atau papan pengembangan. LVGL memungkinkan pengembang untuk membuat antarmuka grafis yang efisien dengan sedikit sumber daya, cocok untuk aplikasi pada perangkat dengan kapasitas memori terbatas.

11.3.3.2 Fitur utama LVGL

- Desain UI yang Mudah: Menyediakan berbagai elemen antarmuka pengguna seperti tombol, slider, grafik, dan sebagainya, untuk membangun tampilan grafis yang interaktif.
- Efisien dalam Penggunaan Memori: LVGL dirancang agar efisien dalam penggunaan memori dan dapat berjalan pada perangkat dengan sumber daya terbatas.
- Dukungan untuk Berbagai Layar: LVGL mendukung banyak jenis layar, termasuk TFT LCD, dengan fitur-fitur seperti animasi, transisi, dan efek grafis.

11.3.3.3 Cara Kerja LVGL dengan TFT LCD

- Driver TFT LCD: LVGL menggunakan driver untuk mengendalikan tampilan pada layar TFT. Driver ini berfungsi untuk mentransmisikan data grafis dari LVGL ke layar TFT.

- Rendering Grafis: LVGL menyediakan API untuk menggambar elemen grafis pada layar, seperti gambar, teks, atau grafik lainnya.
- Manajemen Tampilan: LVGL memungkinkan pembaruan tampilan secara efisien, mengelola layar dengan sistem buffering untuk menghindari flicker atau gangguan gambar saat pembaruan.

11.3.3.4 Kelebihan dan Kekurangan LVGL

1. Kelebihan LVGL:

- Portabilitas: LVGL mendukung berbagai platform hardware dan sistem operasi, sehingga dapat digunakan pada berbagai jenis perangkat embedded.
- Skalabilitas: Cocok untuk proyek dari aplikasi kecil hingga yang lebih kompleks dengan banyak elemen grafis.
- Komunitas dan Dukungan: Sebagai proyek open-source, LVGL memiliki komunitas aktif yang dapat memberikan dukungan dan kontribusi.

2. Kekurangan LVGL:

- Kurva Pembelajaran: Meskipun sangat efisien, pengaturan LVGL mungkin memerlukan sedikit waktu untuk dipahami oleh pemula dalam pengembangan sistem embedded.
- Ketergantungan pada Driver Layar: Penggunaan LVGL dengan layar TFT memerlukan pengaturan driver TFT yang kompatibel, yang bisa sedikit menantang tergantung pada perangkat keras yang digunakan.

11.3.3.5 Langkah-Langkah Praktikum

1. Wiring diagram PIN

TFT LCD PIN	ESP32-CAM PIN
VCC	3V3
GND	GND
CS	5
D/C	4
RESET	13
SDI	23
LED	2

2. Open software Arduino Ide kalian, setelah itu ketikkan sketch berikut ini :

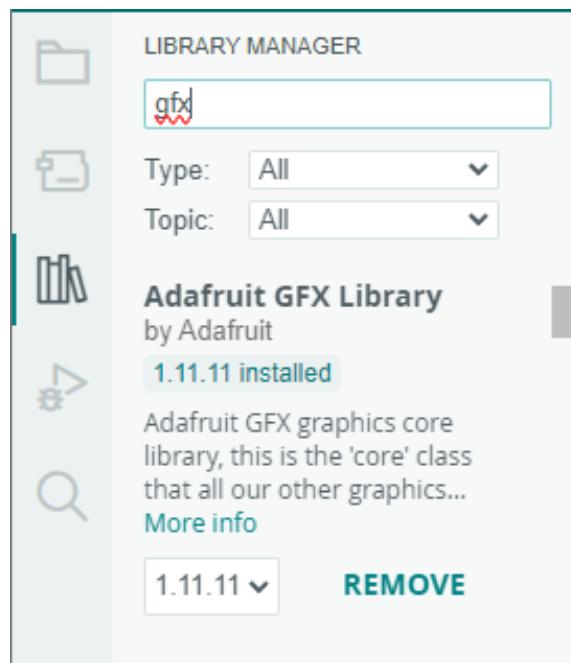
```
1  ****
2  * Exercise of Arduino_GFX, color testing.
3  * Run on Xiao ESP32C3 + 480x320 ILI9488 SPI TFT
4  ****
5  #include <Arduino_GFX_Library.h>
6
7  #define TFT_CS    5    //GPIO5
8  #define TFT_RESET 13   //GPIO3
9  #define TFT_DC    4    //GPIO4
10 #define TFT_MOSI  23   //GPIO10/MOSI
11 #define TFT_SCK   18   //GPIO8/SCK
12 #define TFT_LED   2    //GPIO2
13 #define TFT_MISO  -1   // not used for TFT
14
15 #define GFX_BL  TFT_LED // backlight pin
16
17 /* More data bus class: https://github.com/moononournation/Arduino\_GFX/wiki/Data-Bus-Class */
18 Arduino_DataBus *bus = new Arduino_HWSPI(TFT_DC, TFT_CS);
19
20 /* More display class: https://github.com/moononournation/Arduino\_GFX/wiki/Display-Class */
21 Arduino_GFX *gfx = new Arduino_ILI9488_18bit(bus, TFT_RESET, 3 /* rotation */, false /* IPS */);
22
23 ****
24 * End of Arduino_GFX setting
25 ****
26
27 void setup(void)
28 {
29     gfx->begin();
30     gfx->fillScreen(BLACK);
31
32     #ifdef GFX_BL
33         pinMode(GFX_BL, OUTPUT);
34         digitalWrite(GFX_BL, HIGH);
35     #endif
36
37     gfx->setTextColor(WHITE);
38     gfx->setTextSize(2, 2, 2);
39
40     gfx->setCursor(10, 10);
41     gfx->println("XIAO ESP32C3 (in Arduino framework)");
42
43     gfx->setCursor(10, 30);
44     gfx->println("+ ILI9488 SPI TFT");
45
46     gfx->setCursor(10, 50);
47     gfx->println("using Arduino_GFX Library");
48
49     int w = gfx->width();
50     int h = gfx->height();
51
52     gfx->setCursor(10, 70);
53     gfx->printf("%i x %d", w, h);
54     gfx->drawRect(0, 0, w, h, WHITE);
```

Modul Praktikum

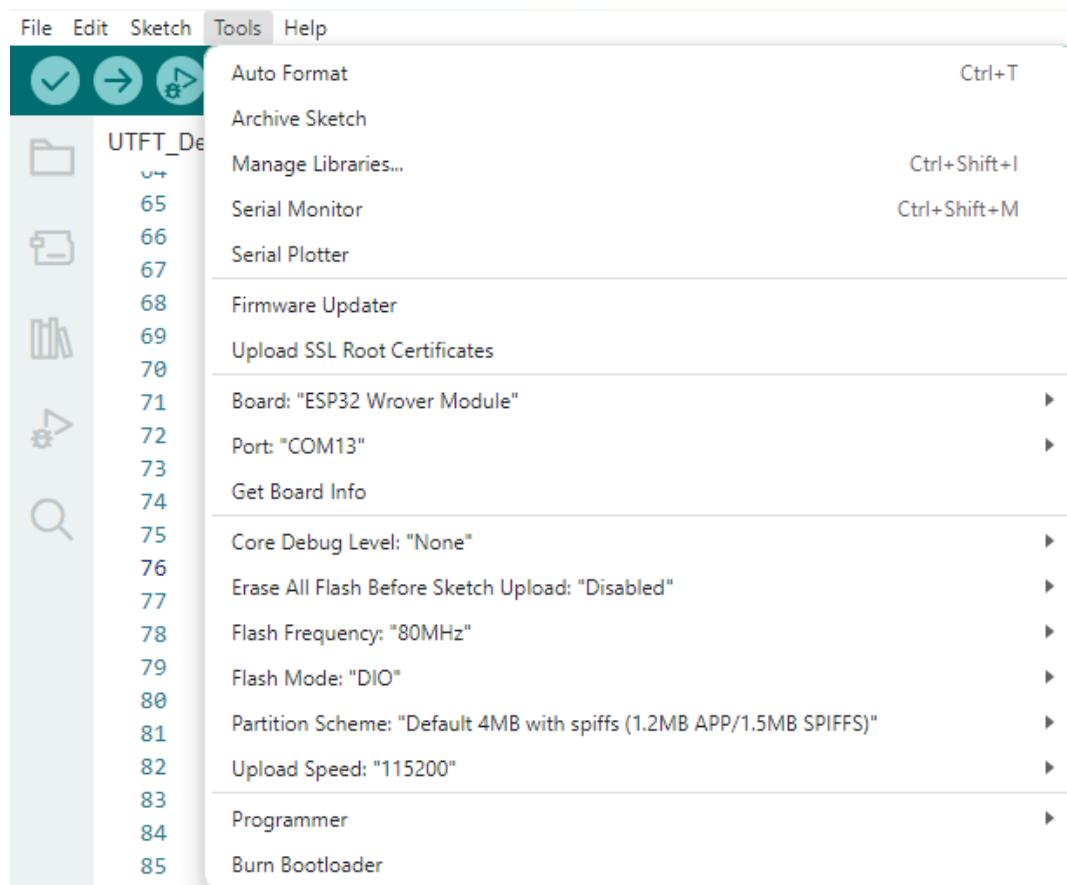
```
54     delay(3000);
55
56
57     for(int i=0; i<w; i++){
58         int d = (int)(255 * i/w);
59         gfx->drawLine(i, 0, i, w, RGB565(d, 0, 0));
60         delay(10);
61     }
62     for(int i=0; i<w; i++){
63         int d = (int)(255 * i/w);
64         gfx->drawLine(w-i, 0, w-i, w, RGB565(0, d, 0));
65         delay(10);
66     }
67     for(int i=0; i<w; i++){
68         int d = (int)(255 * i/w);
69         gfx->drawLine(i, 0, i, w, RGB565(0, 0, d));
70         delay(10);
71     }
72 }
73
74 void loop()
75 {
76     gfx->setTextColor(WHITE);
77     gfx->setTextSize(6, 6, 2);
78
79     gfx->fillScreen(RED);
80     gfx->setCursor(100, 100);
81     gfx->printf("RED");
82     delay(2000);
83
84     gfx->fillScreen(GREEN);
85     gfx->setCursor(100, 100);
86     gfx->printf("GREEN");
87     delay(2000);
88
89     gfx->fillScreen(BLUE);
90     gfx->setCursor(100, 100);
91     gfx->printf("BLUE");
92     delay(2000);
93 }
```

Modul Praktikum

3. Lalu Install Library berikut ini :



4. Lalu aturlah bagian Toolsnya seperti pada gambar di bawah ini dan untuk bagian Port sesuaikan dengan laptop masing-masing :



Modul Praktikum

5. Kemudian kalian Verify setelah itu klik Upload, tunggu hingga selesai maka akan muncul tampilan pada TFT LCD yang kalian gunakan seperti gambar di bawah ini :

