Pour connecter la matrice LED RGB haute résolution 64x64 d'Elecrow à une ESP32, il faut suivre ces étapes :

Matériel requis :

1. Matrice LED RGB 64x64
2. ESP32 : microcontrôleur.
3. Câbles d'alimentation : Pour connecter l'alimentation à la matrice LED et à l'ESP32.
4. Résistances de limitation de courant : pour limiter le courant traversant les LED, selon les spécifications de la matrice LED.

Étapes :

1. Identifiez l'interface utilisée par la matrice LED Elecrow. Elle utilise une interface de type HUB75, qui est couramment utilisée pour les matrices LED RVB.
2. Connectez les broches d'alimentation de la matrice LED à une source d'alimentation. Connectez les broches VCC de la matrice LED à une tension d'alimentation adéquat (généralement 5V) et connectez les broches GND à la masse (GND) de l'alimentation.
3. Connectez les broches de données de la matrice LED aux broches de sortie de l'ESP32. Pour l'interface HUB75, on a besoin de plusieurs broches de données pour les signaux de contrôle et les données RVB. Généralement, on a besoin de connecter les signaux suivants :

* Broche R1, G1, B1 : Ces broches contrôlent la première colonne de LEDs rouges, vertes et bleues.
* Broche R2, G2, B2 : Ces broches contrôlent la deuxième colonne de LEDs rouges, vertes et bleues.
* Broche A, B, C, D : Ces broches sont les signaux de contrôle de la ligne et déterminent quelle ligne est activée à un moment donné.

1. Connectez les broches de sélection (par exemple, OE, LAT, CLK) de la matrice LED aux broches de sortie de l'ESP32. Ces broches permettent de contrôler le fonctionnement de la matrice LED. Généralement, on a besoin de connecter les signaux suivants :

* Broche OE (Output Enable) : Permet d'activer ou de désactiver toutes les sorties de la matrice LED. Reliez cette broche à une broche de sortie de l'ESP32.
* Broche LAT (Latch) : Permet de transférer les données des broches de données à la matrice LED. Reliez cette broche à une broche de sortie de l'ESP32.
* Broche CLK (Clock) : Permet de générer une horloge pour synchroniser le transfert des données. Reliez cette broche à une broche de sortie de l'ESP32.

1. Connectez les résistances de limitation de courant. On a besoin de résistances pour limiter le courant traversant les LED. Placez les résistances en série avec les broches de données RGB (R1, G1, B1, R2, G2, B2) pour chaque colonne de LED.
2. Une fois les connexions physiques effectuées, On doit écrire du code pour contrôler la matrice LED à partir de l'ESP32. On peut utiliser la bibliothèque FastLED ou la bibliothèque Adafruit\_GFX pour faciliter le contrôle des matrices LED.
3. S’assurer d'installer la bibliothèque appropriée dans votre environnement de développement Arduino. On peut le faire en accédant à "Sketch > Include Library > Manage Libraries" et en recherchant la bibliothèque souhaitée.
4. Importez la bibliothèque dans votre code en ajoutant une directive #include au début de votre sketch.
5. Initialisez la matrice LED dans votre code en spécifiant le nombre de colonnes, de lignes et les broches de données qu’on utilise. Par exemple :

#include <FastLED.h>

#define NUM\_ROWS 64

#define NUM\_COLS 64

#define DATA\_PIN\_R1 2

#define DATA\_PIN\_G1 3

#define DATA\_PIN\_B1 4

// Définissez les broches de données pour les autres colonnes (R2, G2, B2) de manière similaire.

CRGB leds[NUM\_ROWS \* NUM\_COLS];

void setup() {

FastLED.addLeds<NEOPIXEL, DATA\_PIN\_R1>(leds, NUM\_ROWS \* NUM\_COLS);

// Ajoutez les autres lignes de code pour les autres colonnes (G1, B1, R2, G2, B2).

}

void loop() {

// Votre code de contrôle de la matrice LED ici.

}

d) Utilisez les fonctions de la bibliothèque pour contrôler les LED de la matrice, telles que leds[x] = CRGB::Red pour allumer une LED en rouge à la position x.

e) Dans la fonction loop(), on peut mettre à jour les couleurs et les motifs des LED en modifiant les valeurs dans le tableau leds[] et en appelant FastLED.show() pour mettre à jour la matrice LED.

1. Téléversez le code sur votre ESP32

Une image contenant capture d’écran, point, carré

Description générée automatiquement

Une image contenant capture d’écran, noir, motif, noir et blanc

Description générée automatiquement