

# ESP32 draadloze spelletjes (ESP-Mesh)

Programma op github:

<https://github.com/thieu-b55/ESP32-Wireless-Touch-Screen-Games>

Speel:

TIC-TAC-TOE

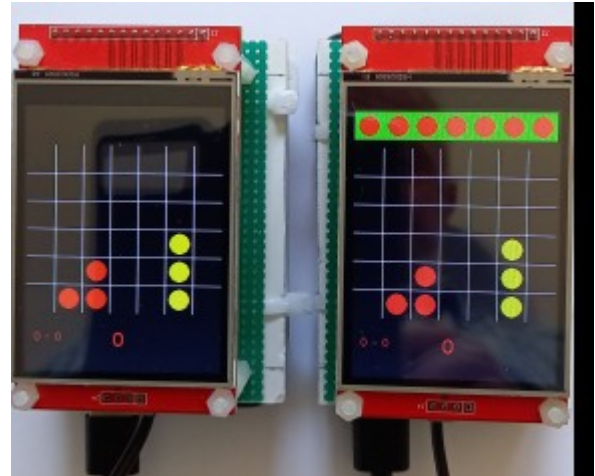
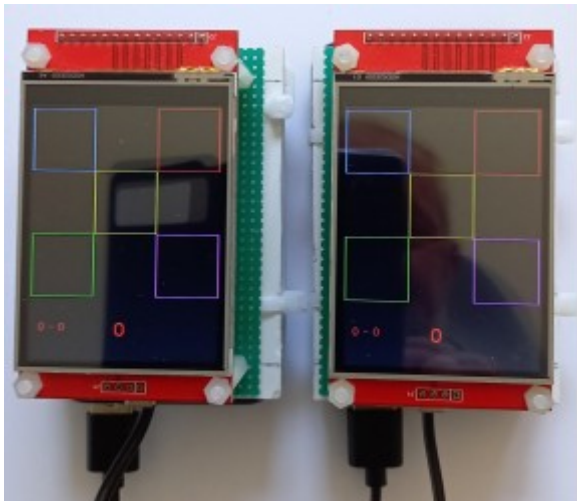
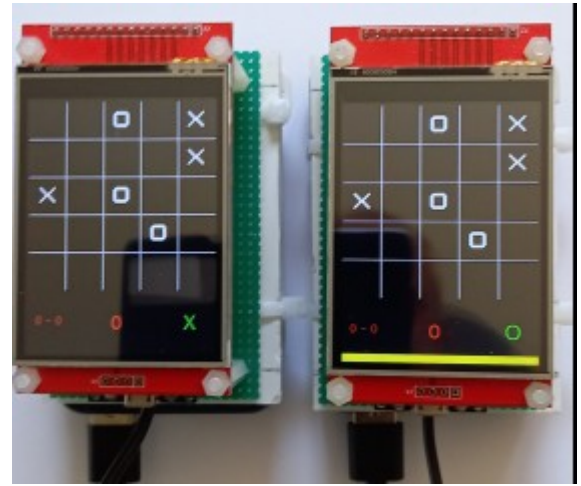
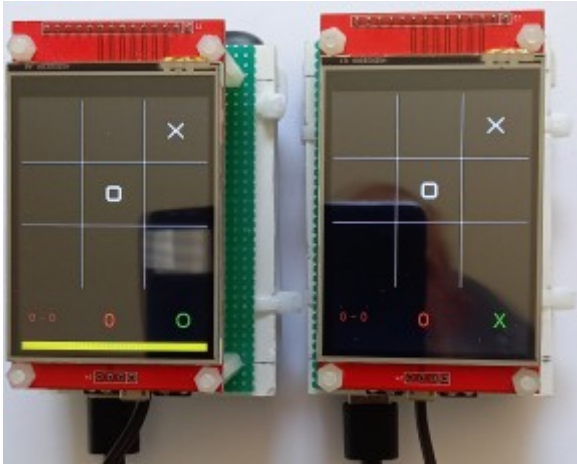
Vier Op Een Rij

Geheugenspel

Vier Op Een Rij Vallende Munten

Met 2, met elkaar verbonden via een Mesh netwerk, ESP32 modules





## Onderdelen

WROOM Devkit module

2x 2.8 TFT SPI 240x320 TFT **TOUCH** SCREEN

## Verbindingen

ESP32	5V	>>	5V voeding
ESP32	GND(3x)	>>	GND voeding
ESP32	EN(Reset)	>>	TFT RESET
ESP32	2	>>	TFT DC
ESP32	4	>>	T_CS
ESP32	12	>>	TFT SDO(MISO)
			T_DO
ESP32	13	>>	TFT SDI(MOSI)
			T_DIN
ESP32	14	>>	TFT_SCK
			T_CLK
ESP32	15	>>	TFT CS
ESP32	25	>>	T_IRQ
ESP32	3.3V	>>	TFT LED

## Settings User\_Setup.h TFT\_eSPI

```
// ##### EDIT THE PIN NUMBERS IN THE LINES FOLLOWING TO SUIT YOUR ESP32 SETUP #####

// For ESP32 Dev board (only tested with ILI9341 display)
// The hardware SPI can be mapped to any pins

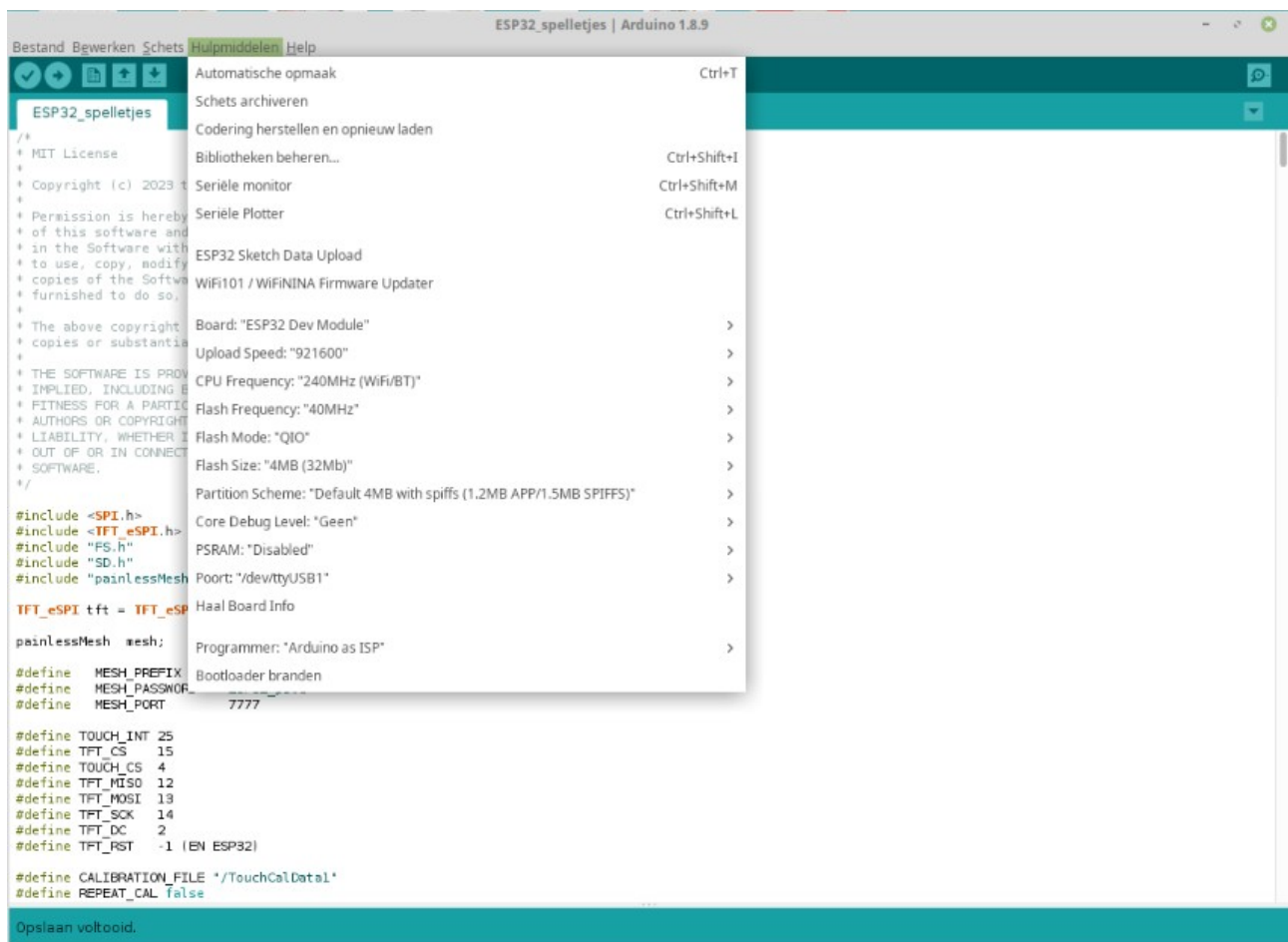
#define TFT_MISO 12
#define TFT_MOSI 13
#define TFT_SCLK 14
#define TFT_CS   15  // Chip select control pin
#define TFT_DC    2  // Data Command control pin
#define TFT_RST   -1 // Reset pin (could connect to RST pin)
// #define TFT_RST  -1 // Set TFT_RST to -1 if display RESET is connected to ESP32 board RST

// For ESP32 Dev board (only tested with GC9A01 display)
// The hardware SPI can be mapped to any pins

// #define TFT_MOSI 15 // In some display driver board, it might be written as "SDA" and so on.
// #define TFT_SCLK 14
// #define TFT_CS   4  // Chip select control pin
// #define TFT_DC    2  // Data Command control pin
// #define TFT_RST   -1 // Reset pin (could connect to Arduino RESET pin)
// #define TFT_BL    22 // LED back-light

#define TOUCH_CS 4  // Chip select pin (T_CS) of touch screen
```

## Settings Arduino IDE



Laad het programma ESP32\_spelletjes.ino in beide ESP32 modules.

Pas eventueel de MESH\_PORT aan indien men meer dan 1 set gelijktijdig wil gebruiken.

```
painlessMesh mesh;  
  
#define MESH_PREFIX "ESP32"  
#define MESH_PASSWORD "ESP32_pswd"  
#define MESH_PORT 7777
```

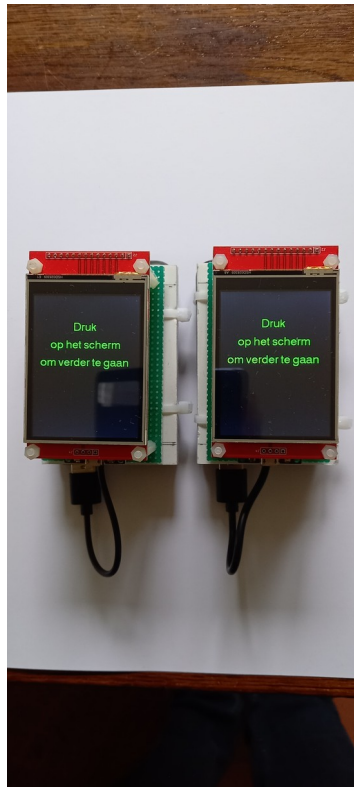
2 bij elkaar horende modules moeten dezelfde MESH\_.... data hebben.

### Hoe werkt het

Na de opstart verschijnt er dit scherm



Beide ESP32 modules kiezen nu een random nummer en wachten op verbinding met de tegenspeler. Na verbinding verschijnt het volgende scherm



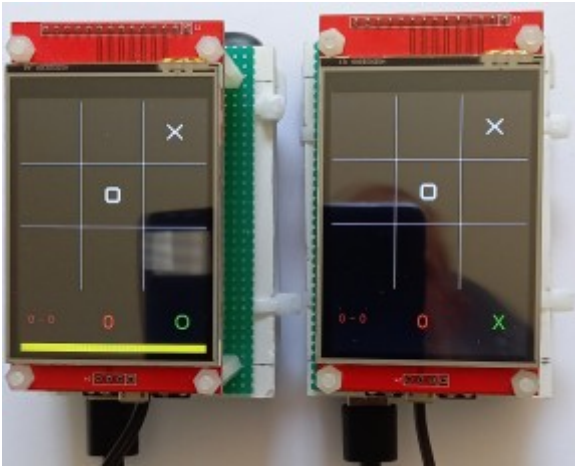
1 van de beide spelers moet nu het scherm aanraken (met de bijgeleverde touch pen geeft het beste resultaat). Wie heeft geen invloed op het verder verloop, dit wordt bepaald door het hoogste random gekozen getal van de ESP32 modules

Na aanraken scherm verschijnen de volgende schermen



De module met het hoogste random getal mag kiezen welk spel er gespeeld wordt en mag ook als eerste beginnen. Na elk spel wordt er gewisseld wie mag beginnen.

## TIC-TAC-TOE



Weergave beneden op het scherm

Links

puntentelling met eerst de eigen punten en dan de tegenspeler

Midden

nogmaals de eigen punten

Rechts in het groen

met welke letter men speelt

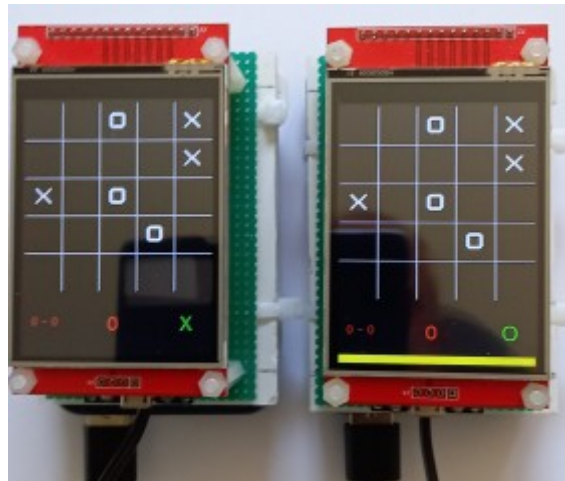
Gele balk duidt aan wie er aan zet is.





Een juiste combinatie wordt in het blauw weergegeven.  
Een nieuw spel begint na 5 seconden.

## Vier Op Een Rij



Weergave beneden op het scherm

Links

puntentelling met eerst de eigen punten en dan de tegenspeler

Midden

nogmaals de eigen punten

Rechts in het groen

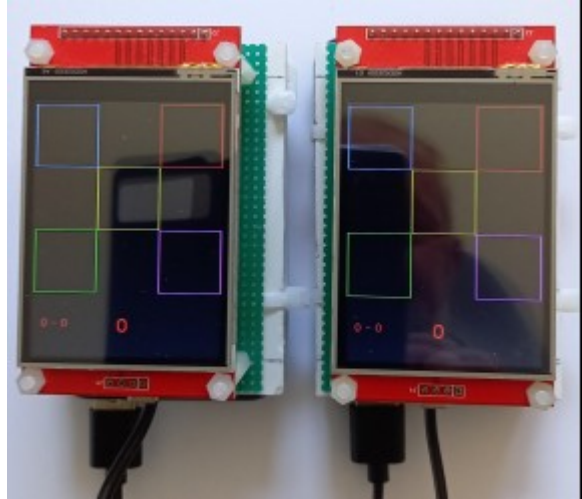
met welke letter men speelt

Gele balk duidt aan wie er aan zet is.

Een juiste combinatie wordt ook hier in het blauw weergegeven.

Nieuw spel na 5 seconden

## Geheugenspel



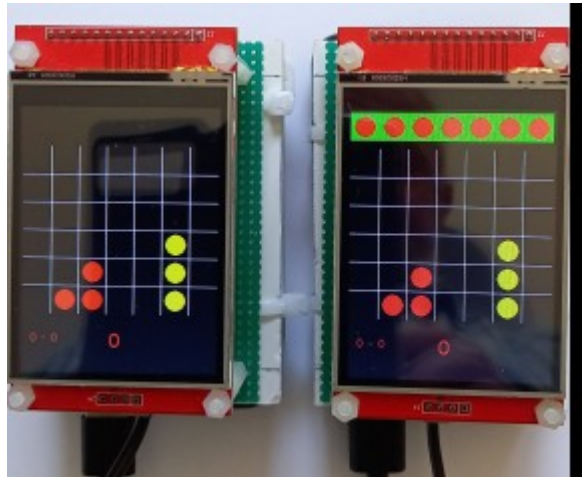
herspeel in de juiste volgorde de door de ESP32 getoonde kleuren.

Spel begint met 3 kleuren.

Beide spelers krijgen dezelfde kleuren.

Als **beide** spelers 5 punten of een meervoud van 5 punten bereiken komt er 1 kleur bij, maximum is 15 kleuren.

## Vier Op Een Rij, vallende munten



Laat een munt vallen door boven in de balk een rode of gele munt aan te raken.

Juiste combinatie wordt in blauw weergegeven.  
Een nieuw spel begint na 5 seconden.

Veel speelplezier,  
groeten,  
thieu-b55