

# Kurzanleitung für den Handregler

**PacoMouseCYD** by Paco



v0.9 02/2026

## EINFÜHRUNG

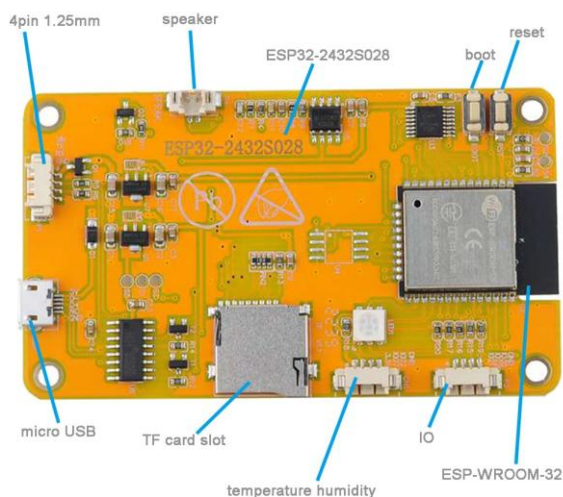
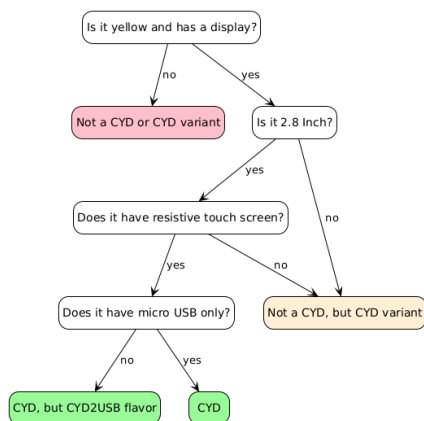
**PacoMouseCYD** entstand nach der Entdeckung des sogenannten „*Cheap Yellow Display*“ (CYD), um eine einfache zusätzliche Fernsteuerung für Lokomotiven über WiFi bereitzustellen. Eine kommerzielle Nutzung ist **NICHT** gestattet.

Das CYD-Modell **2432S028** enthält auf einer kompakten Platine einen ESP32-Prozessor, einen ILI9341-Display-Controller für den **2,8"**-Farbmonitor (320×240 Pixel), einen XPT2406-Controller für den **resistiven** Touchscreen, einen SD-Kartenslot (bis 32 GB), einen Lautsprecheranschluss, eine RGB-LED und eine Lichtsensor-Diode (LDR). Für den Betrieb benötigt man lediglich einen EC11- oder KY040-Drehencoder mit Taster sowie eine Batterie, um ein kabelloses Handgerät zu erhalten.

### 1. VOR DEM ZUSAMMENBAU

Stellen Sie sicher, dass Ihr *Cheap Yellow Display* mit der Firmware von **PacoMouseCYD** kompatibel ist. Bei Unsicherheit besuchen Sie bitte die folgende Seite:

<https://github.com/witnessmenow/ESP32-Cheap-Yellow-Display/tree/main>



#### **Cheap Yellow Display**

Modell: **2432S028**

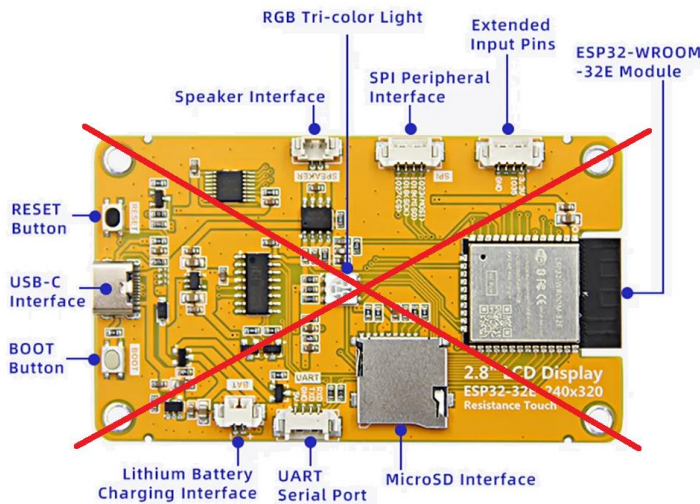
**ESP32**-Prozessor

**2,8-Zoll**-Bildschirm (320 x 240 Pixel), Controller

**ILI9341**

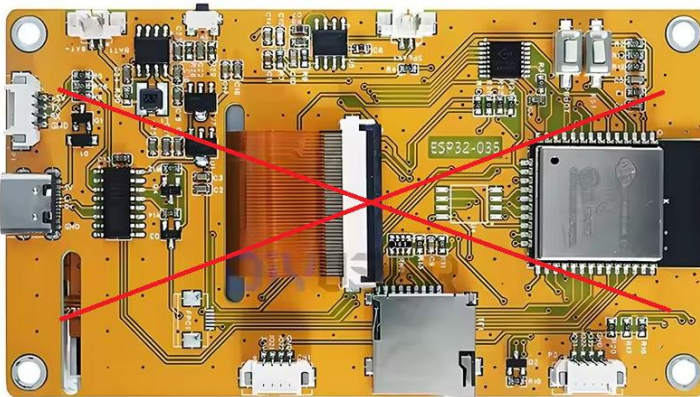
**Resistiver** Touchscreen, **XPT2406**-Controller

SD-Kartenleser (bis zu 32 GB)



Einige ähnliche Displays sind nicht geeignet, da ihre Pin-Belegung nicht genügend freie Pins für den Encoder zulässt.

Es gibt auch CYD-Modelle mit anderen Größen, z. B. 2,4" oder 3,2", jedoch ist die korrekte Funktion der **PacoMouseCYD**-Firmware bei diesen aufgrund der unterschiedlichen Pinbelegung nicht garantiert.



Die CYD-Modelle mit größeren TFT-Bildschirmen, wie z. B. die 3,5-Zoll-Modelle, sind ebenfalls ungeeignet, da ihre Auflösung 320x480 statt 240x320 beträgt und das Bild daher nicht korrekt angezeigt wird.

Wenn Sie ein solches Modell besitzen, können Sie versuchen, es für **PacoMouseCYD** zu verwenden, indem Sie die Konfiguration an Ihre Platine anpassen. Ein korrekter Betrieb kann jedoch nicht garantiert werden.

In den Anhängen dieses Dokuments finden Sie die Schaltpläne für den Aufbau des **PacoMouseCYD**-Controllers.

Die Programmierung erfolgt mit der Arduino IDE, über die Sie das Programm, die benötigten Bibliotheken und die **PacoMouseCYD**-Firmware hochladen können.

Diese Software und die zugehörigen Dateien sind ein DIY-Projekt und nicht für die kommerzielle Nutzung bestimmt.

DIE SOFTWARE WIRD OHNE JEGLICHE AUSDRÜCKLICHE ODER STILLSCHWEIGENDE GEWÄHRLEISTUNG BEREITGESTELLT.

Die Quelltexte dienen ausschließlich zum Kompilieren und Hochladen auf das Gerät.

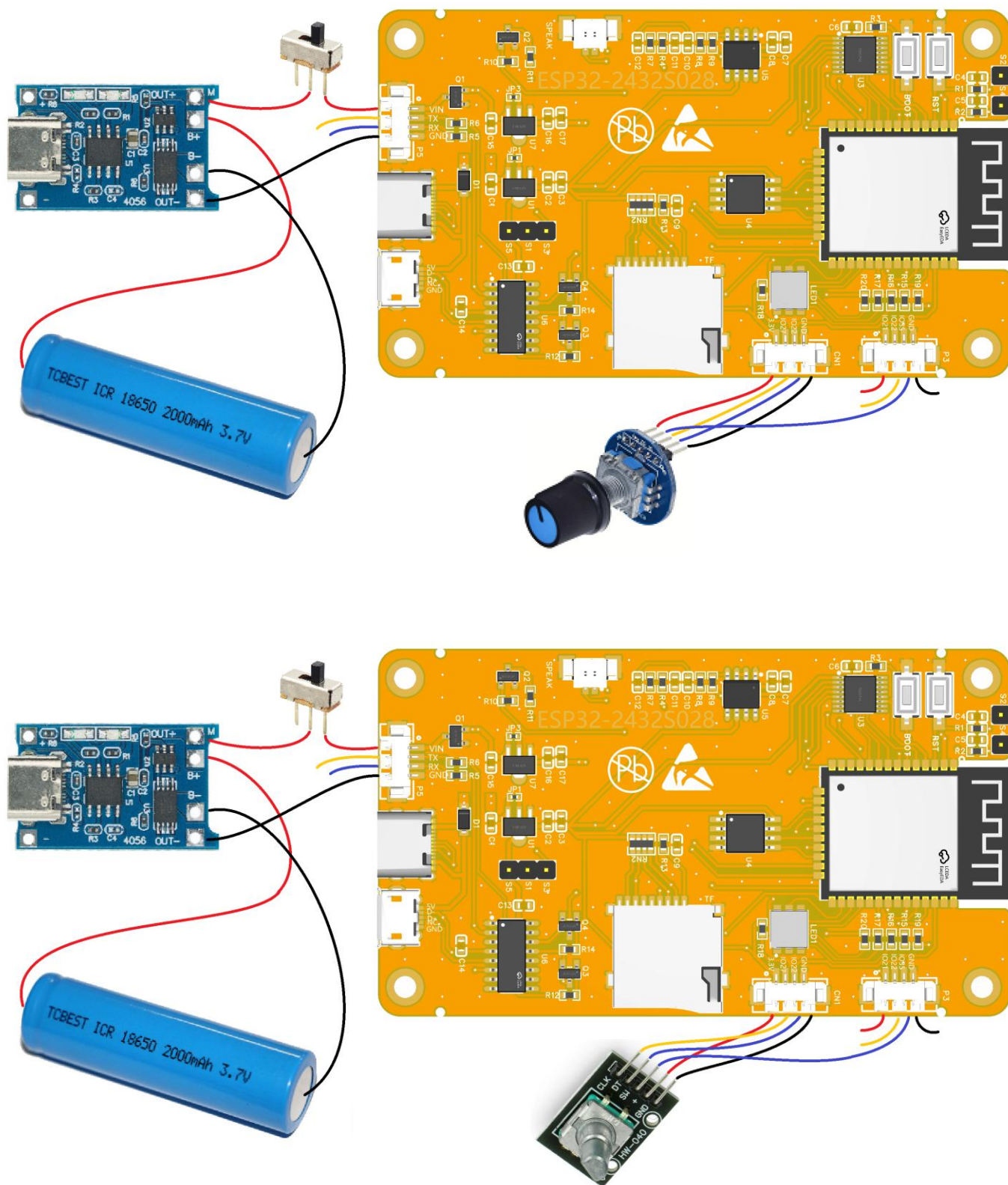
Es ist Ihnen nicht gestattet, den Quellcode zu verändern oder dieses Projekt zu forken/veröffentlichen.

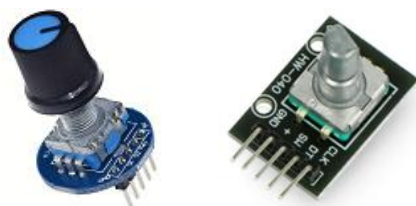
**KOMMERZIELLE NUTZUNG IST VERBOTEN.**



## Anhang I: Diagramme

### Anschlussdiagramm





ENCODER		CYD	Conector
GND	GND	GND	CN1
S1	CLK	IO22	CN1
S2	DT	IO27	CN1
Key	SW	IO35	P3
5V	+	+3V3	CN1

**ACHTUNG**

Es gibt viele verschiedene Arten von Encodern, und sie sind möglicherweise nicht gleich verdrahtet; wenn der Encoder rückwärts arbeitet, tauschen Sie die Drähte **S1/CLK** und **S2/DT**.



AKKU	TP4056	CYD	Stecker
<b>+V</b>	B+		
<b>GND</b>	B-		
	OUT+	VIN	P5
	OUT-	GND	P5

**ACHTUNG**

Prüfen Sie vor dem Anschließen der einzelnen Elemente die Beschriftung der Pins auf Ihrer CYD-Platine.

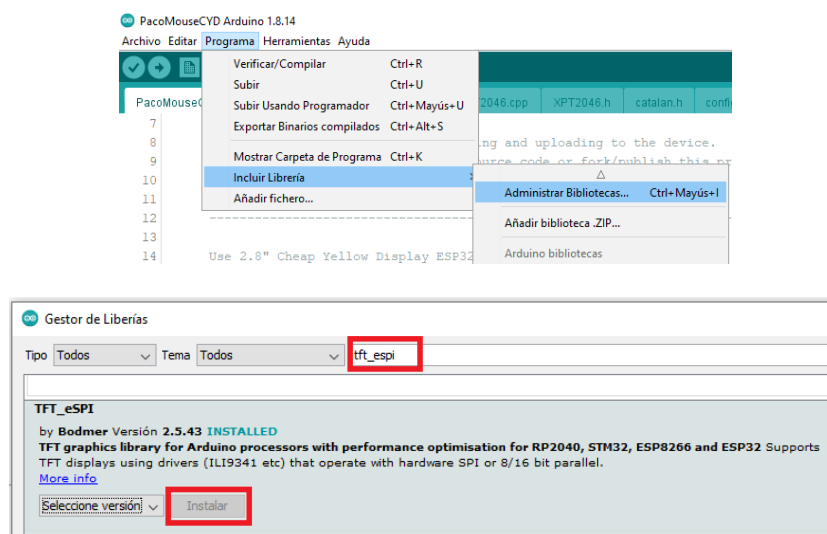
## Anhang II: Programmierung mit der Arduino IDE

### Programmierung

Das Programm **PacoMouseCYD** wird über die USB-Verbindung des *Cheap Yellow Display* Boards mithilfe der Arduino IDE (<https://www.arduino.cc/en/Main/Software>) auf dieses Board übertragen.

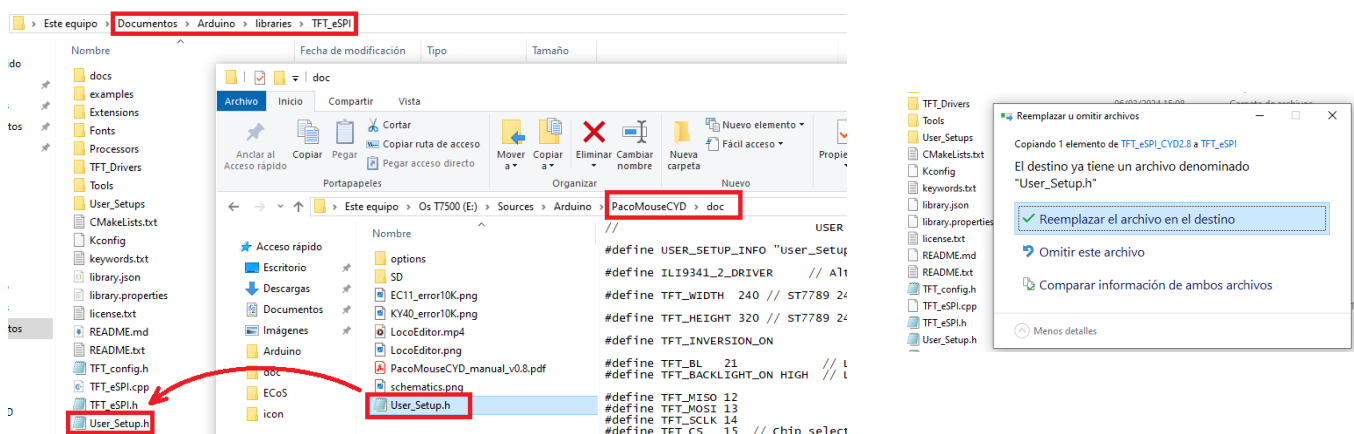
Das *Cheap Yellow Display* verwendet einen ESP32-Prozessor und verfügt über ein TFT-Display mit einem ILI9341-Controller und einem vom XPT2046 gesteuerten Touchpanel. Daher müssen die benötigten Bibliotheken und Boards in der Arduino IDE installiert werden, um das Programm zu kompilieren.

Die Bibliothek **TFT\_eSPI.h** (Version 2.5.43) muss über „Sketch“ → „Bibliothek einbinden“ → „Bibliotheken verwalten...“ installiert werden.



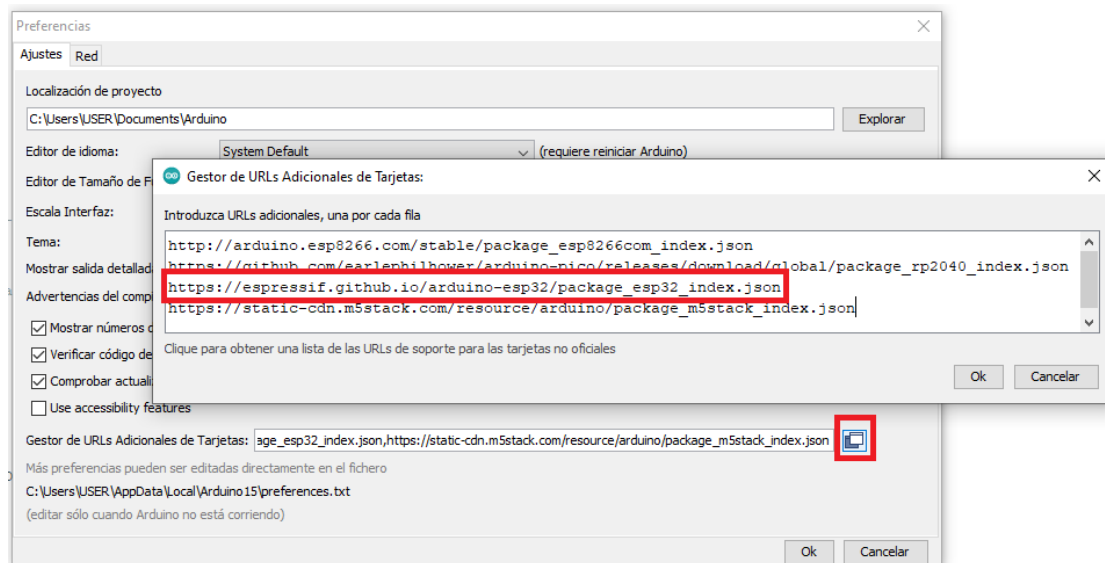
Die **TFT\_eSPI.h**-Bibliothek unterstützt TFT-Displays mit verschiedenen Controllern. Die standardmäßige Konfiguration ist jedoch nicht für das *Cheap Yellow Display* geeignet. Passen Sie die Datei *User\_Setup.h* im Verzeichnis „Eigene Dokumente/Arduino/libraries/TFT\_eSPI“ an und tragen Sie die Werte ein, die für den Displaytyp und die Pins des *Cheap Yellow Displays* relevant sind.

Die passende Datei *User\_Setup.h* für das *Cheap Yellow Display* finden Sie im Ordner „PacoMouseCYD\doc\“. Kopieren Sie diese in den Bibliotheksordner unter „Eigene Dokumente/Arduino/libraries/TFT\_eSPI“.

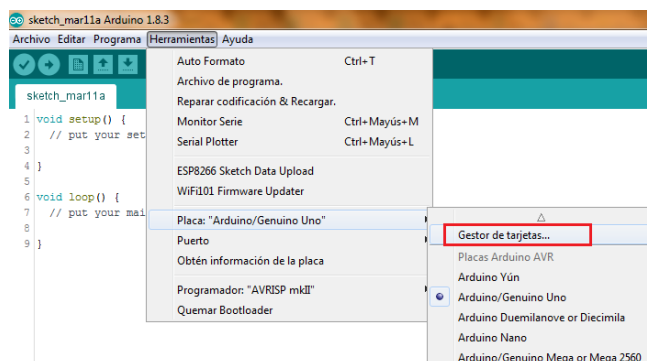


Sie müssen außerdem den *Arduino Core* für den ESP32-Chip installieren. Gehen Sie zu *Datei > Einstellungen* und kopieren Sie unter „Zusätzliche Board-URL-Manager“ den folgenden Link:

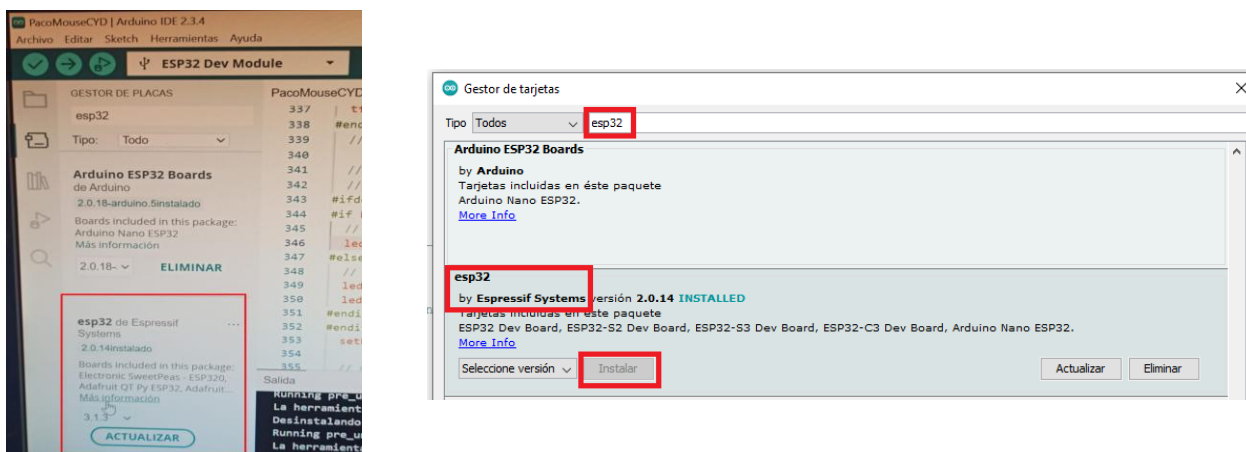
**[https://espressif.github.io/arduino-esp32/package\\_esp32\\_index.json](https://espressif.github.io/arduino-esp32/package_esp32_index.json)**



Gehen Sie zu *Tools > Board: „... > Board Manager...“*

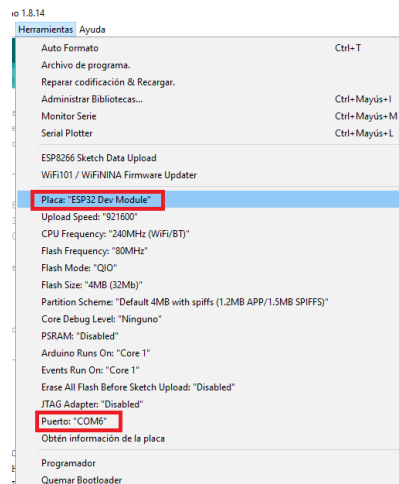


Suchen Sie im Board-Manager nach „esp32“ und installieren Sie **ESP32 von Espressif Systems (v2.0.14 oder v3.3.0)**.

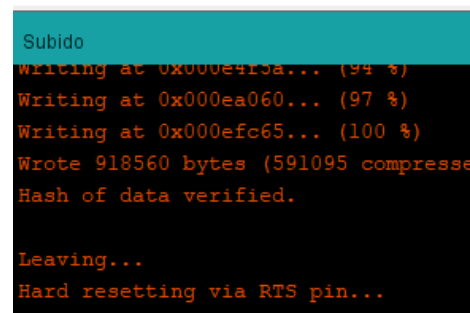
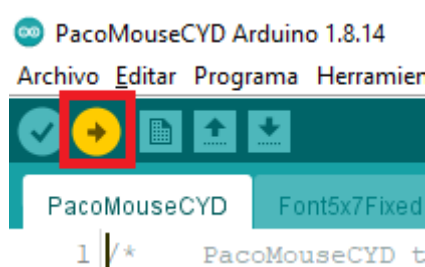


## Firmware-Programmierung über USB

Nachdem wir den Sketch geöffnet, überprüfen wir in der Arduino IDE, ob wir das **ESP32 Dev Module** Board und den zugehörigen Port ausgewählt haben. Möglicherweise müssen wir zuvor den CH340 USB-Chip-Treiber ([https://www.wch-ic.com/downloads/CH341SER\\_ZIP.html](https://www.wch-ic.com/downloads/CH341SER_ZIP.html)) installieren, damit der Port in der Liste erscheint.

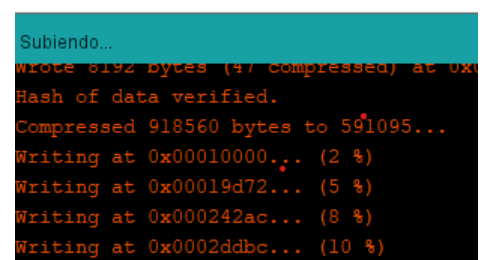
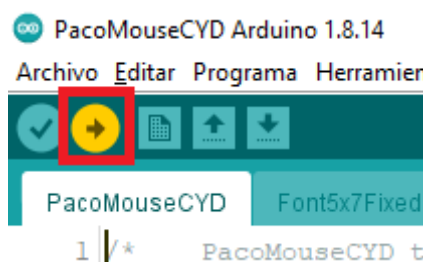
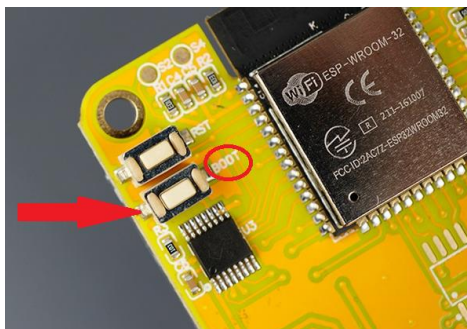


Um das Programm hochzuladen, klicken Sie auf die Schaltfläche „Hochladen“. Nach erfolgreicher Kompilierung des Programms erhalten Sie eine Benachrichtigung über den erfolgreichen Upload.



Der Vorgang sollte automatisch ablaufen, und der Begrüßungsbildschirm von **PacoMouseCYD** wird angezeigt.

Sollten Probleme beim Hochladen des Programms auftreten, überprüfen Sie, ob Sie den Port, an den das Gerät angeschlossen ist, korrekt angegeben haben. Falls eine Fehlermeldung erscheint, dass sich der Chip nicht im *Download-Modus* befindet, drücken Sie die BOOT-Taste, bis das Programm hochgeladen wird.



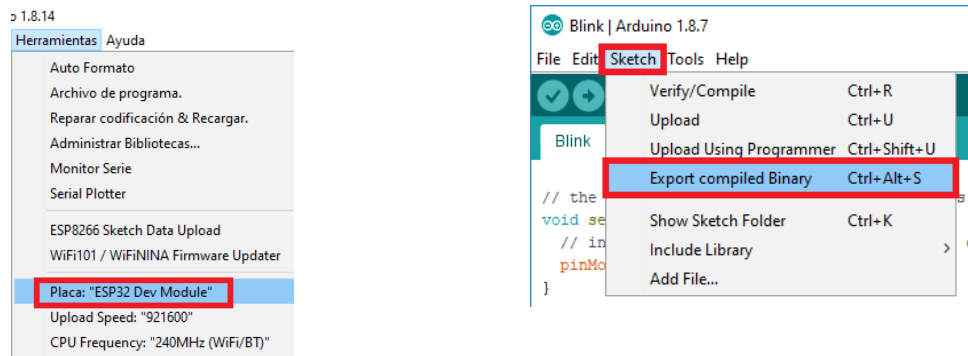


## Firmware-Update von der SD-Karte

Sobald die **PacoMouseCYD** für Updates programmiert wurde, kann sie wie oben beschrieben neu programmiert werden.

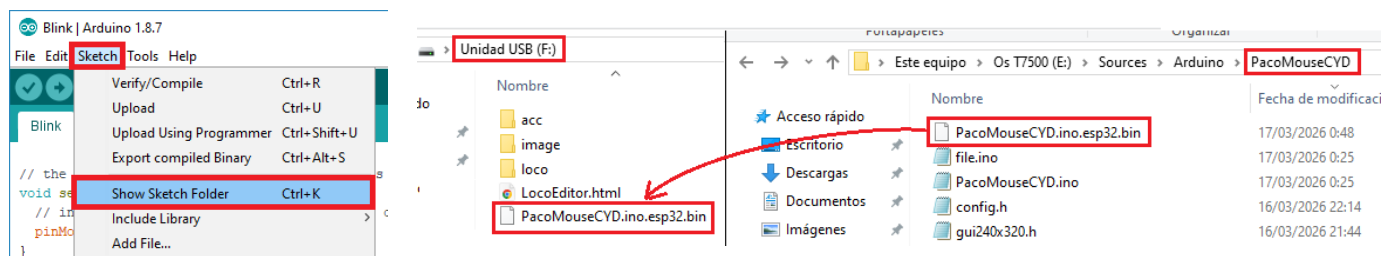
Wenn der USB-Anschluss des *Cheap Yellow Display* nicht zugänglich ist oder mehrere Handregler aktualisiert werden müssen, kann das Update über eine .bin-Datei im Stammverzeichnis der SD-Karte erfolgen.

Nachdem der Sketch geöffnet und gegebenenfalls die Optionen in der **config.h**-Datei angepasst wurden, wird in der Arduino IDE überprüft, ob das **ESP32 Dev Module** Board ausgewählt ist. Um die .bin-Datei zu generieren, wird anstelle des Klicks auf „Hochladen“ der Menüpunkt **Sketch > Kompilierte Binärdateien exportieren** ausgewählt.

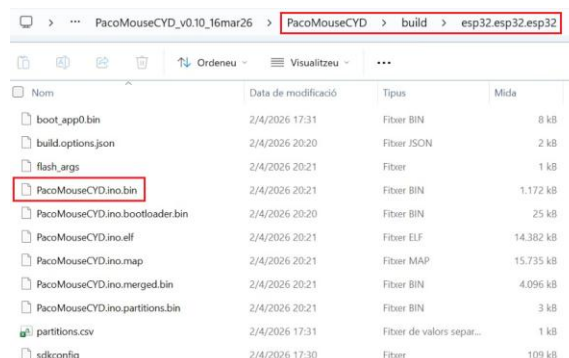


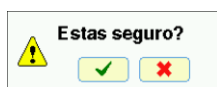
Die Datei **PacoMouseCYD.ino.esp32.bin** wird im Sketch-Verzeichnis von **PacoMouseCYD** erstellt.

Über **Sketch > Sketch-Ordner anzeigen** kann die generierte Datei **PacoMouseCYD.ino.esp32.bin** gefunden und in das Stammverzeichnis der SD-Karte kopiert werden.



Bei der Arduino IDE v2 befindet sich die .bin-Datei im **Build**-Ordner unter dem Namen **PacoMouseCYD.ino.bin**. Benennen Sie sie in **PacoMouseCYD.ino.esp32.bin** um und kopieren Sie sie auf die SD-Karte.

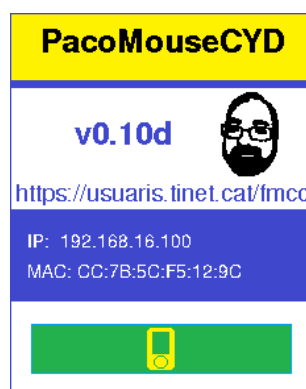
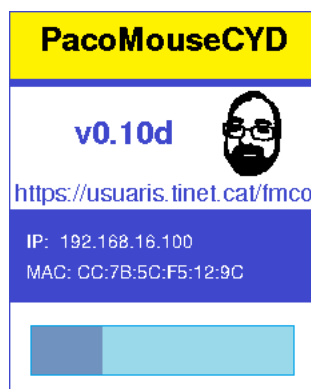




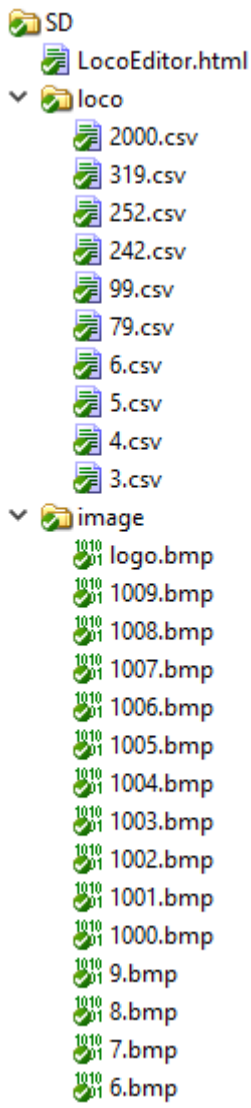
Legen Sie die SD-Karte in die **PacoMouseCYD** ein, schalten Sie sie ein und gehen Sie zu *Menü > Einstellungen > Über...* und tippen Sie auf die Schaltfläche *Firmware aktualisieren*.

Nach der Bestätigung wird die Firmware in wenigen Sekunden aktualisiert, und die **PacoMouseCYD** startet mit der neuen Version neu.

Wenn alles erfolgreich verlaufen ist, wird der Fortschrittsbalken grün angezeigt; im Fehlerfall rot.



## Anhang III: Inhalt der SD



Die SD-Karte speichert die Lokomotivbilder und -daten, die PacoMouseCYD auf dem Bildschirm anzeigt. PacoMouseCYD kann auch ohne SD-Karte verwendet werden, allerdings stehen Ihnen dann nur die standardmäßigen Schwarzweißbilder zur Verfügung.

### LOKOMOTIVENBILDER

Die Lokomotivbilder werden auf der SD-Karte im Verzeichnis /image gespeichert. Es handelt sich um .bmp-Dateien mit einer Größe von 190x40 Pixel (24 Bit). Der Dateiname besteht aus einer Zahl größer oder gleich 1000 und entspricht der Lokomotiv-ID. Bilder mit Dateinamen von 0 bis 9 sind die Standardbilder von PacoMouseCYD und können nicht verändert werden; sie sind für die korrekte Funktion des LocoEditors erforderlich. Sie finden einige passende Bilder auf den folgenden Webseiten (bitte beachten Sie deren Nutzungsbedingungen):

<https://www.esu.eu/en/downloads/loco-pictures/>

[https://www.stayathome.ch/Lokbilder\\_ECoS.htm](https://www.stayathome.ch/Lokbilder_ECoS.htm)

Wenn Sie eigene Bilder erstellen möchten, können Sie dieser Anleitung folgen:

<https://www.youtube.com/watch?v=vMFVdSWTAbg>

Die Hintergrundfarbe des Bildes ist 0xB6B6B6 (R:182, G:182, B:182)

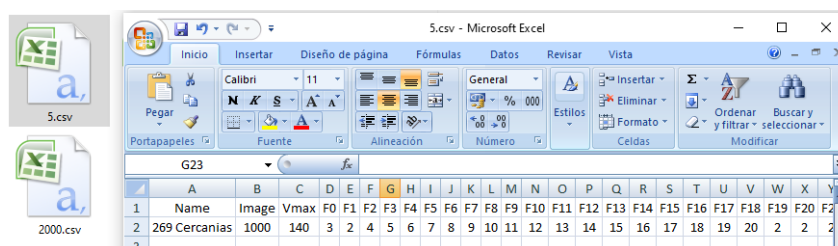
### WILLKOMMENSBLD

Das Willkommensbild ist auf der SD-Karte im Verzeichnis /image gespeichert; es handelt sich um eine Datei namens **logo.bmp** mit einer Größe von 240x140 Pixel (24 Bit).

### LOKOMOTIVENDATEN

Die Lokomotivdaten (Adresse, Name, Höchstgeschwindigkeit, Funktionssymbole) werden in einer .csv-Datei im Verzeichnis /loco der SD-Karte gespeichert. Der Dateiname entspricht der Adresse der Lokomotive.

Diese Datei kann in Excel geöffnet werden.

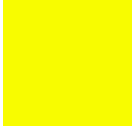







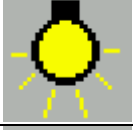







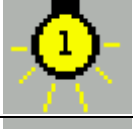



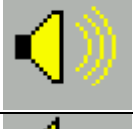
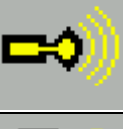


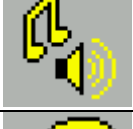
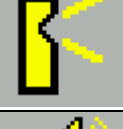





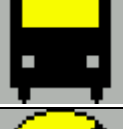










Spalte A: Lokomotivname (maximal 16 Zeichen)

Spalte B: Lokomotiv-ID. Diese entspricht dem Namen des Lokomotivbildes im Verzeichnis /image der SD-Karte, sofern sie größer oder gleich 1000 ist (entspricht auch der ECoS-ID). Werte zwischen 0 und 9 kennzeichnen die Standardbilder von PacoMouseCYD.

Spalte C: Höchstgeschwindigkeit (in km/h), angezeigt auf dem Tachometer. Idealerweise entspricht diese der Höchstgeschwindigkeit der realen Lokomotive und, bei korrekter Leistungsanpassung, der des Modells.

Spalte D und folgende Spalten: ID des Funktionssymbols für die Funktionen F0 bis F28.

	1		11		21		31
	2		12		22		32
	3		13		23		33
	4		14		24		34
	5		15		25		35
	6		16		26		36
	7		17		27		37
	8		18		28		38
	9		19		29		39
	10		20		30		40

## PacoMouseCYD LocoEditor

PacoMouseCYD Loco Editor

Castellano

Locomotora:

Nombre:

Vel. max.:

km/h

Archivo CSV: loco/  79.csv  
Cargar ...

Imagen:

Archivo BMP: image/  1002.bmp  
Cargar ...

Funciones:

F0	F1	F2	F3	F4	F5	F6	F7	F8	F9
F10	F11	F12	F13	F14	F15	F16	F17	F18	F19
F20	F21	F22	F23	F24	F25	F26	F27	F28	

Icono para F6

Guardar ...

Design  
by  
**Paco**

Um die Bearbeitung von Lokomotivdaten anschaulicher als in Excel zu gestalten, enthält die SD-Karte den **PacoMouseCYD LocoEditor**. Mit diesem können Sie die Daten bequem über einen Webbrowser bearbeiten.

Öffnen Sie die Datei **LocoEditor.html** von der SD-Karte.

Geben Sie die Lokomotivnummer, ihren Namen (maximal 16 Zeichen) und die Höchstgeschwindigkeit der Lokomotive ein.

Alternativ können Sie eine vorhandene .csv-Datei aus dem Lokomotivordner auf der SD-Karte auswählen. Klicken Sie auf „Laden...“, um die gespeicherten Daten zu laden.

Geben Sie die ID-Nummer des Lokomotivbildes ein oder wählen Sie es aus dem **image**-ordner auf der SD-Karte aus. Klicken Sie auf „Laden...“, um das Bild anzuzeigen. Die Bilder 0 bis 9 können nicht bearbeitet werden, da sie von **PacoMouseCYD** standardmäßig verwendet werden. Bilder ab 1000 sind benutzerdefiniert.

Die Funktionssymbole lassen sich durch Anklicken ändern. Alle verfügbaren Symbole werden angezeigt, sodass Sie das für die Funktion Ihrer Lokomotive am besten geeignete auswählen können.

Sobald Sie alle Daten Ihrer Lokomotive erfasst haben, klicken Sie auf die Schaltfläche „Speichern...“, um die .csv-Datei auf Ihren Computer herunterzuladen. Kopieren Sie diese in den **loco**-ordner auf Ihrer SD-Karte, damit sie für **PacoMouseCYD** verfügbar ist.

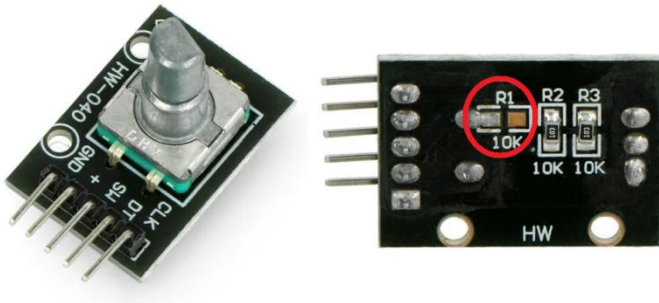
Beim ECoS-Protokoll werden nur die Bilder im Ordner **images/** verwendet, wobei der Dateiname der von der Steuereinheit bereitgestellten Lokomotiv-ID entspricht. Daher ist der **PacoMouseCYD LocoEditor** nur bei Verwendung anderer Protokolle sinnvoll.



## Anhang IV: Tipps

### ENCODER

**ACHTUNG:** Verwenden Sie keinen Encoder ohne Pull-up-Widerstände und lassen Sie den Encoder-Druckknopf nicht angeschlossen; die Encoderbewegung ist sonst unvorhersehbar.

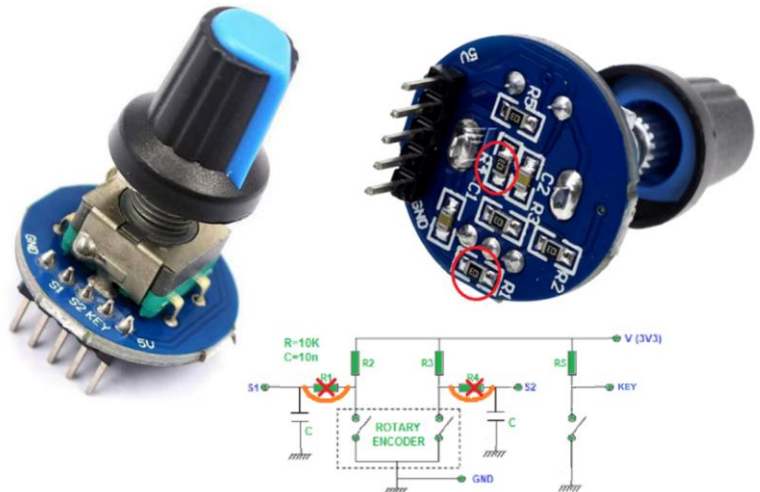


Bei einigen KY-40-Encodern fehlt der 10-k $\Omega$ -Widerstand am Drucktasteranschluss.

In diesem Fall muss ein 10-k $\Omega$ -Widerstand an R1 installiert werden, um den korrekten Betrieb zu gewährleisten.

EC-11-Encoder können unregelmäßig funktionieren, da zwischen dem Encoder-Pin und dem Ausgang ein 10-k $\Omega$ -Widerstand liegt.

In diesem Fall müssen die 10-k $\Omega$ -Widerstände R1 und R4 überbrückt werden, um den Encoder-Pin direkt mit dem entsprechenden Ausgang zu verbinden.



### AKKU

PacoMouseCYD mit einem wiederaufladbaren LiPo-Akku betrieben. Für eine längere Akkulaufzeit kommt ein 2000-mAh-Akku vom Typ 18650 oder 103450 zum Einsatz.

18650 (Ø18 x 65mm)

103450 (34mm x 50mm x 10mm)



Als Ladegerät dient ein Modul mit dem TP4056. Dieses Modul verfügt über einen Schutz gegen Tiefentladung (die Batteriespannung wird nicht unter 2,4 V entladen) und Überladung (die Batteriespannung wird nicht über 4,2 V hinausgehen).

Es besitzt Status-LEDs, die aufleuchten: eine während des Ladevorgangs und eine weitere, wenn die Batterie ausreichend geladen ist.



## **FAST CLOCK**

Die *Fast Clock* wird angezeigt, wenn sie am Bedienfeld aktiviert ist. In der Xpressnet-Version wird sie nur unterstützt, wenn sie mit einem LZV200 v4 oder einem Z21 mit Firmware 1.43 verbunden ist. In der Z21-Version wird sie nur unterstützt, wenn sie mit einem Z21 mit Firmware 1.43 verbunden ist. In der Loconet-Version wird sie angezeigt, wenn das Bedienfeld die *Fast Clock* in Slot 123 unterstützt (Intellibox II, DR5000 usw.).

## **SD-KARTE**

**PacoMouseCYD** kann auch ohne SD-Karte verwendet werden; in diesem Fall stehen die Farbbilder der Lokomotiven jedoch nicht zur Verfügung. Verwenden Sie eine SD-Karte mit bis zu 32 GB (FAT32-Format). Legen Sie die SD-Karte ein, bevor Sie **PacoMouseCYD** einschalten; andernfalls wird die interne Festplatte des *Cheap Yellow Displays* für die verschiedenen Daten verwendet.

Die Karte speichert die Farbbilder der Lokomotiven sowie deren Daten (beim ECoS-Protokoll werden nur die Bilder verwendet, da die Daten von der Steuereinheit bereitgestellt werden) und die Daten für die Zubehörtafeln.

## **INVERTIERTE FARBEN**

Wenn die Farben nach der Programmierung von **PacoMouseCYD** invertiert erscheinen, bedeutet dies, dass Sie nicht die korrekte Datei *User\_Setup.h* aus der Bibliothek „TFT\_eSPI.h“ für Ihren Bildschirm verwendet haben (siehe Anhang II: Programmierung mit der Arduino IDE).

Öffnen Sie im Ordner *Dokumente\Arduino\Bibliotheken\TFT\_eSPI* die Datei *User\_Setup.h* und suchen Sie nach den folgenden Zeilen:

```
// If colours are inverted (white shows as black) then uncomment one of the next
// 2 lines try both options, one of the options should correct the inversion.
```

Ändern Sie die folgenden Definitionen, um die korrekten Farben zu erhalten:

```
#define TFT_INVERSION_ON
// #define TFT_INVERSION_OFF
```

Oder

```
//#define TFT_INVERSION_ON
#define TFT_INVERSION_OFF
```

