User guide for the manual controller

PacoMouseCYD by Paco



v0.14 08/2026

Paco Cañada https://usuaris.tinet.cat/fmco/

Content

Introduction 3
1. Safety instructions 3
2. Warranty 3
3. Support and assistance 3
4. Before assembly 4
5. Connections 6
6. Locomotives 7
   6.1 Driving 7
   6.2 Selecting a Locomotive 9
   6.3 Editing a Locomotive 10
7. Accessories 13
8. Programming a CV 17
9. Configuration 20
10. Utilities 24
11. Automation 33
Appendix I: Schematics 35
Appendix II: Programming with the Arduino IDE 37
Appendix III: SD Card contents 47
Appendix IV: Tips 52

The software uses libraries with different licences, please follow all terms includes in those. Loconet is a registered trademark of Digitrax, Inc. Xpressnet is a registered trademark of Lenz Elektronik GmbH. Z21 is a trademark of Modelleisenbahn GmbH. Arduino is a registered trademark of Arduino S.r.l., ECoS is a trademark of ESU electronic solutions Ulm GmbH & Co. KG, Märklin is a registered trademark of Gebr. Märklin & Cie GmbH

Introduction

PacoMouseCYD was created after discovering the *Cheap Yellow Display* (CYD) board. It offered a simple additional controller for locomotives. It has WiFi connectivity.

Commercial use of this design is absolutely **prohibited**.

The *Cheap Yellow Display* model **2432S028** is a compact board. It contains an ESP32 processor, an ILI9341 controller driving an **2,8"**colour screen (320x240 pixel), an XPT2406 controller for a **resistive** touch screen, an SD card reader (up to 32Gb), connector for a speaker, an RGB LED and an LDR. Just add a rotary encoder like the EC11 or KY040 with a button and power this with a battery to create a wireless controller.

1. Safety instructions

This product is not suitable for unsupervised use by children under the age of 14 years. Because of some small parts that might come loose, keep it out of reach of children under 3 years of age. Improper use may result in injury because of sharp edges and points. It is recommended to use a pencil-type soldering iron with a very fine tip to avoid damage to the components.

2. Warranty

All information for the construction and use of the **PacoMouseCYD** being a DIY device (*DIY - Do it Yourself*) is provided "as is," without warranty of any kind, expressed or implied, including and not limited to damages caused by misinterpretation, improper assembly, use, or handling.

3. Support and Assistance

Information for the construction and/or use of the **PacoMouseCYD** controller is available on the following websites, where you can ask for assistance without any guarantee or obligation:

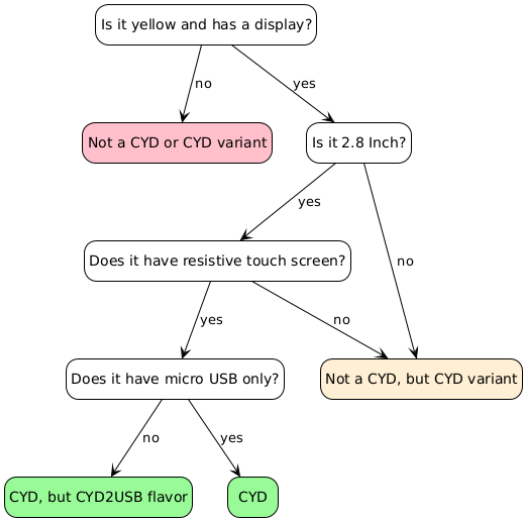
https://usuaris.tinet.cat/fmco
https://github.com/fmcopaco/PacoMouseCYD
https://www.iguadix.es/

Translating this manual involves a considerable amount of work, as does modifying the images. Consequently, the text within the images has generally not been altered; the original (Spanish) text remains in many of the illustrations. Most of them are self-explanatory or can be easily located on the PacoMouseCYD. [Guus Assmann]

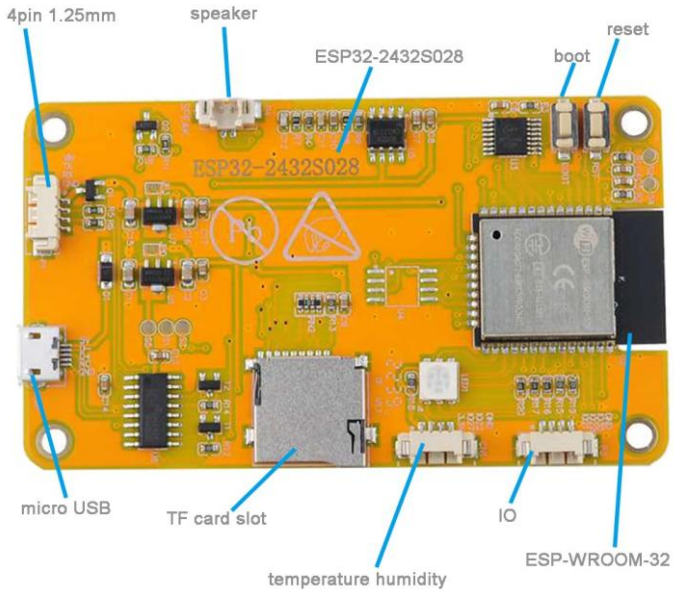
4. Before assembly

Make sure the *Cheap Yellow Display* is compatible with the **PacoMouseCYD,** firmware. If in doubt, please consult this webpage:

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







Cheap Yellow Display

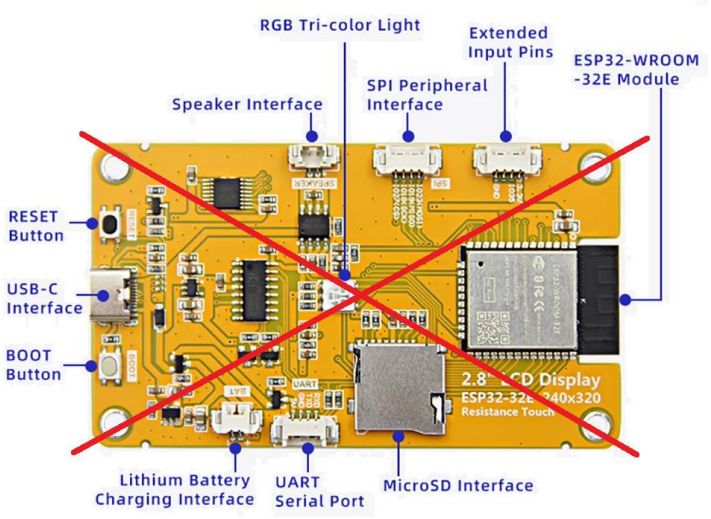
Model: **2432S028**

Processor **ESP32**

Screen **2,8”** (320x240 pixel), controller **ILI9341**

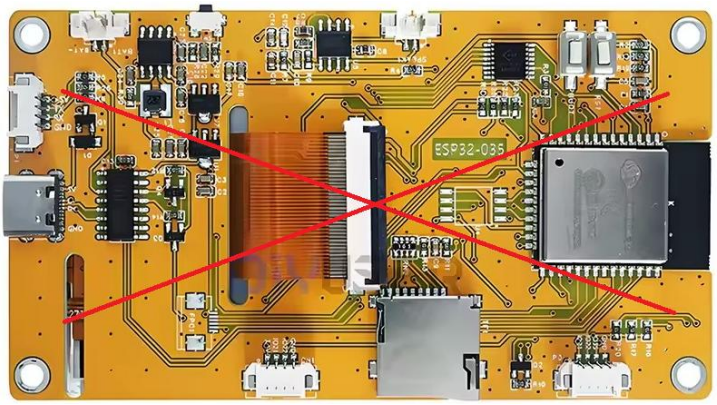
Touch panel **resistive,** controller **XPT2406**

SD card reader (up to 32Gb),



There are other similar screens that may be unsuitable because they have different connections and don't leave enough free pins to connect the encoder.

There are also CYD models with different TFT screens, such as 2.4" o 3.2", but these do not guarantee correct operation with the **PacoMouseCYD** firmware, due to different pin configurations.



The larger CYD models with TFT screens, such as the 3.5" ones may also be unsuitable because of different resolutions like320 x 480 and not 240x320, so they will not display correctly.

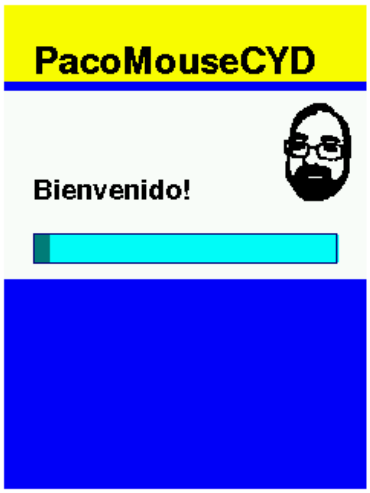
If you have one of these board, you can try using it for the **PacoMouseCYD** by adapting the configuration to your board (see Appendix II), but proper function is not guaranteed.

In the Appendices of this document, you'll find the schematics for assembling the **PacoMouseCYD** Controller.

The Arduino IDE is used for programming, allowing for upload of the program and necessary libraries, as well as the firmware for the **PacoMouseCYD**.

5. Connection

Once the **PacoMouseCYD** is powered, the welcome screen will be displayed. Initialization of the components will be performed and a connection to the control unit will be set up. The progress of this operation can be seen in the sliding bar.





The system will attempt to connect to the user's configured WiFi network. If a connection cannot be established, up to four networks that can be chosen from in the **PacoMouseCYD** will be displayed, allowing the user to choose a different network or modify the connection.



If the configuration was changed or a different network was chosen, the **PacoMouseCYD** must be turned off and on for initialization.

Pressing the button will open a window to configure the selected network's name, SSID, *password*, IP address, protocol.

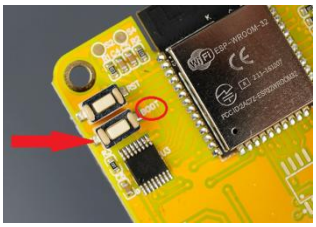
Tapping the SSID will invoke a search for currently present WiFi networks



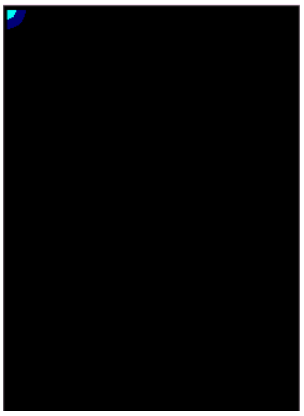


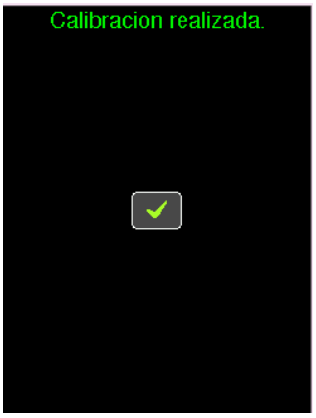


Pressing the **'BOOT'** button at any time (See section 9 – SCREEN) will access the screen calibration option. For more precise calibration, use the stylus to touch the corners that appear, as close as possible to the centre of the circle.







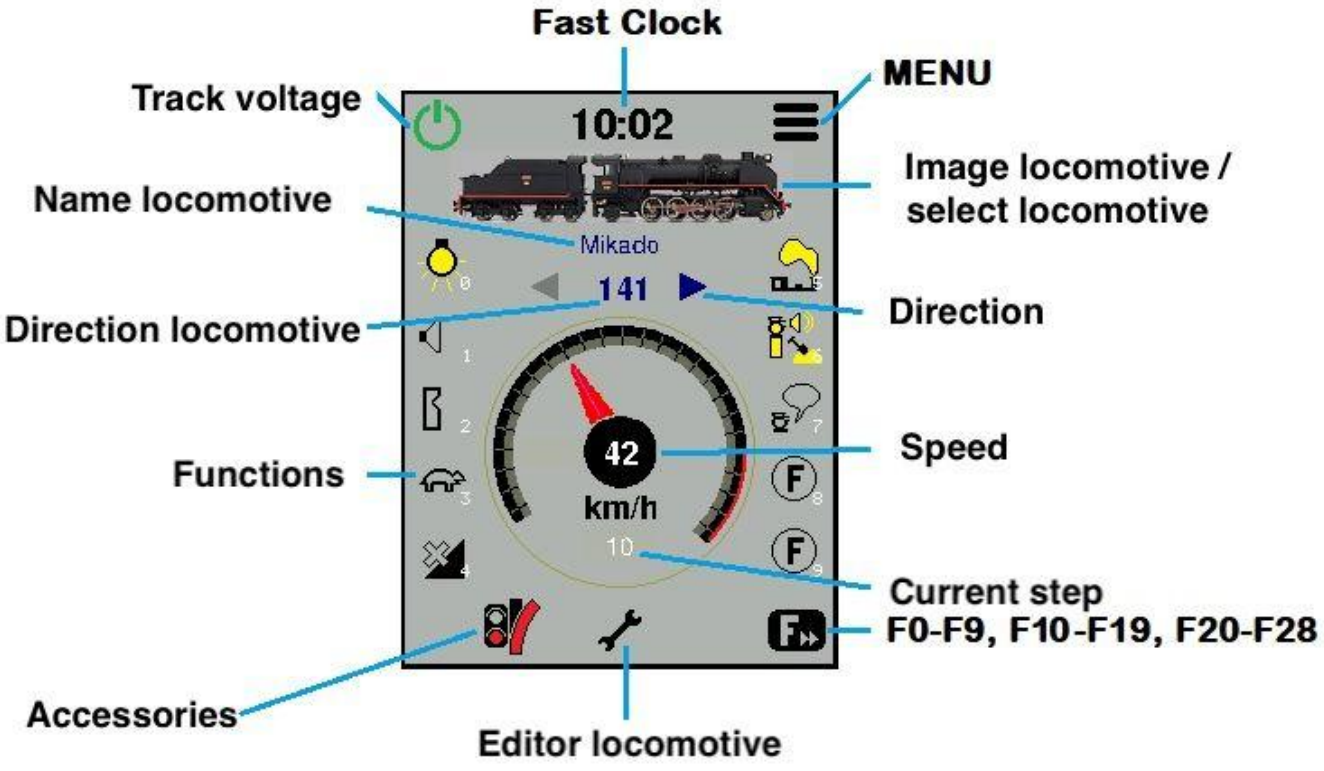


Once the connection to the control unit is established, the locomotive control screen will be shown.

6. Locomotives

6.1 Driving

The locomotive driving screen allows for control of speed for the selected locomotive and the touch screen gives access to other options:



The rotary encoder allows you to increase and decrease the locomotives speed. To stop a moving locomotive, push the button. With the locomotive stopped, pressing the encoder button again changes the direction of travel.

While manoeuvring mode is active (the speed value is displayed in yellow) change the speed by rotating the encoder one step at a time. To stop the locomotive, press the encoder button. This allows for more precise control of stops during manoeuvres.

The icon allows for selecting the display functions: F0-F9, F10-F19 or F20-F28.
Tapping the icon toggles the corresponding function.

Tapping on the locomotive image  will display the locomotive selection screen.

If the ECoS protocol is selected, icons are read from the control unit, **PacoMouseCYD** attempts to display the most appropriate icons and images loaded from the SD card according to the locomotive ID provided by the control unit.

If *Fast Clock* is activated on the control unit and the current time has been transmitted, it will be displayed at the top.

Tapping this icon de-energizes the track output (*Emergency Off*). If the WiFi connection is lost this icon, will appear instead. When the connection is restored **PacoMouseCYD** will either automatically reset or needs to be restarted. This depends on the connection type.

Tapping this icon will display the accessories control panel.

Tapping this icon will allow for editing the locomotives characteristics. With the ECoS protocol, this icon will be displayed to show the features provided by the control unit.

Tapping this icon will display the**PacoMouseCYD**options menu.



6.2 Selecting a locomotive

Tapping the locomotive image in the driving screen will open the locomotive selection window. Now a name and number are displayed:

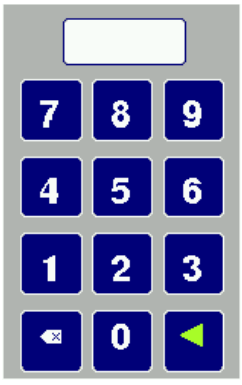


PacoMouseCYD saves the locomotive that was selected last, for quick access without having to enter the number.Using the encoder, the list can be navigated for selection of the desired locomotive.

Tapping this icon allows for changing the order in which the locomotives are displayed. The selections are listed in the table below:

Icon	Order
	Last selected
	Ascending enumeration
	Descending enumeration
	Ascending locomotive name
	Descending locomotive name

A manual address can be entered by tapping the icon , a numeric keypad appears (except for the ECoS protocol). If no number is entered, the current locomotive will remain active.



6.3 Editing the locomotive

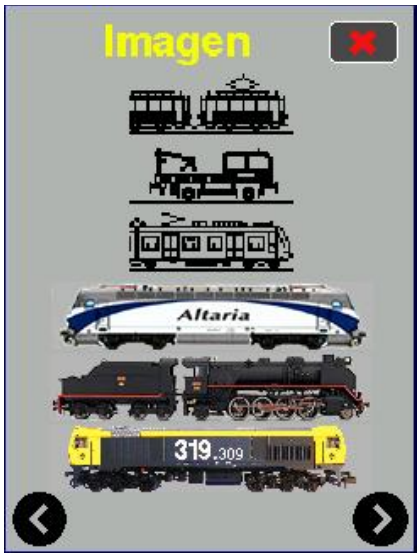
In the locomotive screen, by tapping the icon allows for editing your locomotive's data: Name, Image, Maximum speed and the icons for its functions



When the ECoS protocol is active, this information cannot be altered. The data from the ECoS are displayed. Activation of the CS2 protocol allows for choice of the data format (DCC, MM o MFX) by tapping the button next to the address.

Make a selection by tapping the locomotive image from the ones included in the **PacoMouseCYD** (generic black and white images) or from the images you have stored on the SD card in the **/image** directory. (see Appendix III for information about these images)





Tapping these icons or will scroll through the available images. Tap the one of your choice.

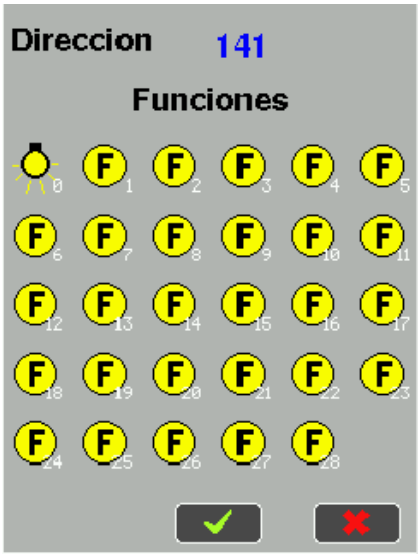
Tapping the locomotive name field will display a keyboard thus allow an entry of a 16 character name.

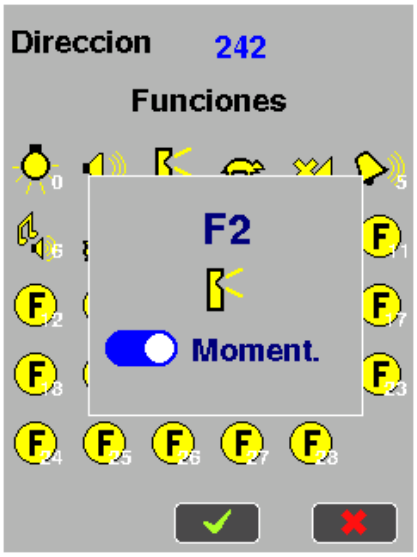


Tapping the speed field in the centre will bring up a keypad that allows entry of a maximum speed up to 999 km/h. This maximum speed will be displayed in the speedometer, ideally it corresponds to the maximum speed of the real locomotive and to that of the model locomotive if it's parameters have been properly set in the CV.

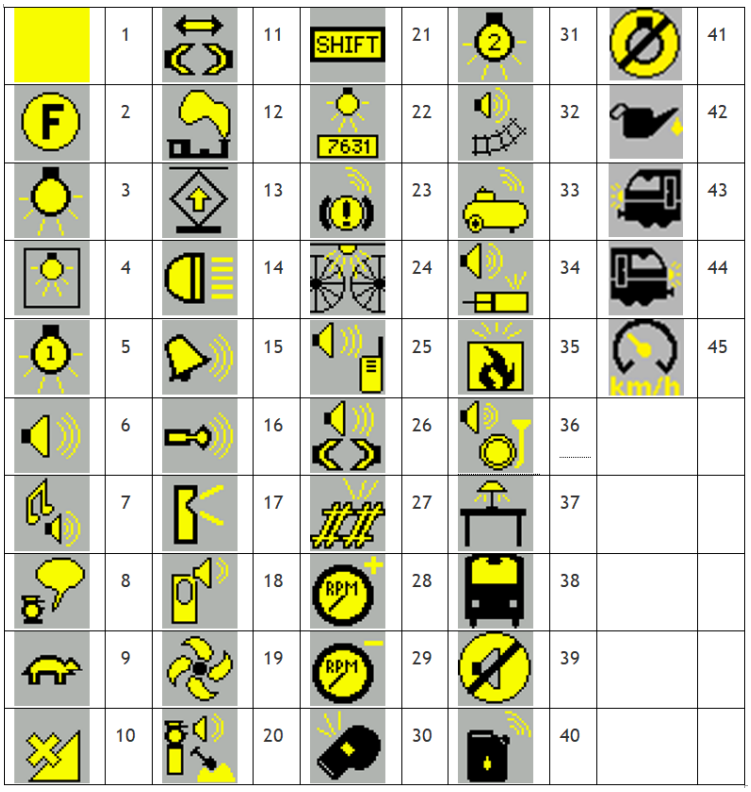


Tapping the functions field opens a window for selection of the icons to be displayed for each of function F0 to F28. Tapping on a function icon will open a selection window that can be scrolled using the rotary encoder. Selection of a momentary activation is also possible. Clicking the encoder or tapping a new icon will change this:





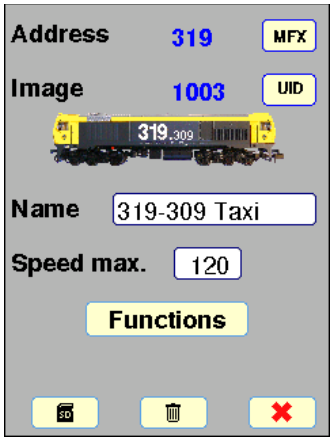
Available icons for the supported functions in the **PacoMouseCYD** are shown below:



After selecting the appropriate options, tapping SD will save the selections to the SD card (if available) or to the internal hard drive.

Locomotives can be deleted from the SD card and the **PacoMouseCYD** by tapping 🗑 icon. A query window will open for confirmation. Once accepted, it is deleted and the next locomotive in the *stack* will be selected.





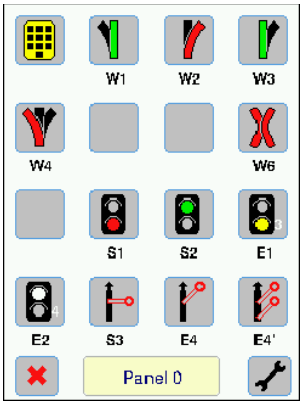
With the CS2 protocol, controlling a DCC or MM locomotive only requires it's address. However for the MFX protocol, it's UID must also be entered.

The locomotive address will correspond to the one indicated on the control unit for the MM protocol. Enter the locomotive name exactly as it appears on the control unit and tap the UID field. If it's found, the UID (a number greater than 16384) will appear in the image field. The image of the locomotive should be saved on the SD card with this number.

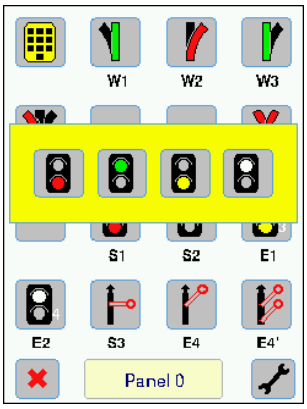
The locomotive address can still be used for manual selection and automation.

7. Accessories

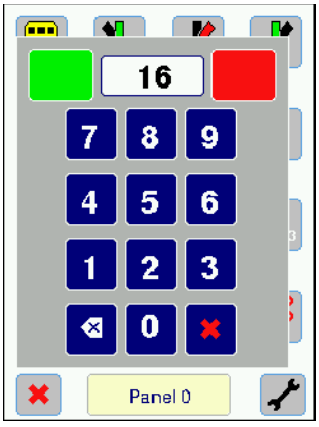
On the locomotive driving screen, tap the  icon to select "**Accessories**". This will display a panel for controlling accessories:



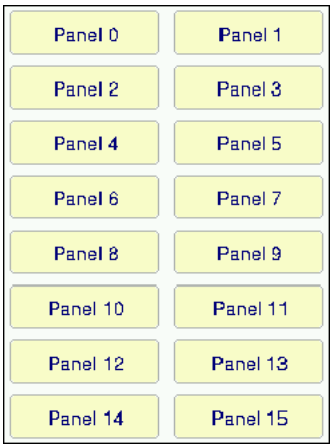
Tapping an accessory field will operate the corresponding accessory. If an accessory has multiple aspects, they will be displayed in order to choose a selection:



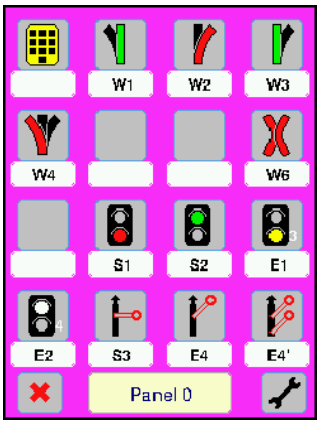
Tapping the  icon will open a window for manual choice of the accessory number to operate. Operation is performed by tapping the red and/or green fields for the appropriate action:



Tapping the panel name will open a window that allows for selection of another panel out of the 16 available:



To adapt a panel to your needs, tap the icon:



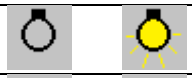
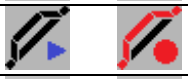
By tapping an icon in the panel of choice, a new icon can be selected using the rotary encoder. Select it by pushing the button or tapping the field:



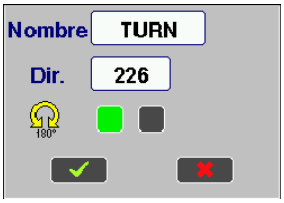
If an accessory type has three or four aspects, the number of aspects is indicated by a number in the lower right corner:

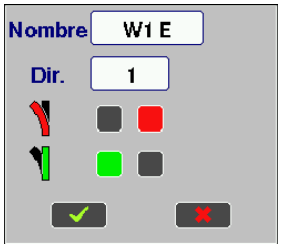


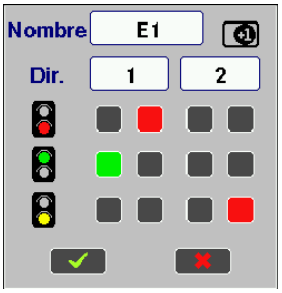
These are the accessories that can be chosen:

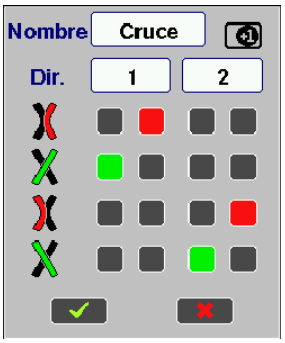
ID	*Aspect*	*Description*
0		Deleted accessory
1		Left turnout
2		Right turnout
3		Triple turnout
4		Crossing
5		Double crossing
6		Switchboard
7		Signal with 2 aspects
8		Signal with 3 aspects
9		Signal with 4 aspects
10		Arm signal with 2 aspects
11		Arm signal with 3 aspects
12		Level crossing
13		Turntable: Left track
14		Turntable: Right track
15		Turntable: 180º rotation
16		Turntable: Exit track
17		Light
18		Sound
19		Power supply
20		Manual accessory selection
21		Switch
22		Automation (Routes)
23		Lock

Selecting an accessory will open a screen to define its name (up to 6 characters), its address, and activation of the corresponding output:





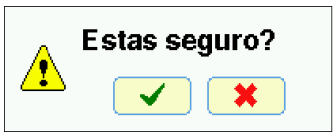




Entering the second address can be skipped, if it's consecutive, by tapping the  icon

If no accessory is selected, confirmation to delete the item will be asked for:



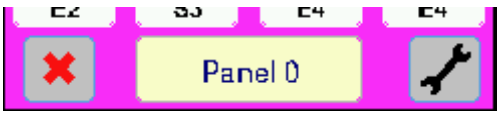






A name and number for automatic routes can be entered (up to 6 characters):

Tapping the panel name allows for entry of a name (up to 12 characters):





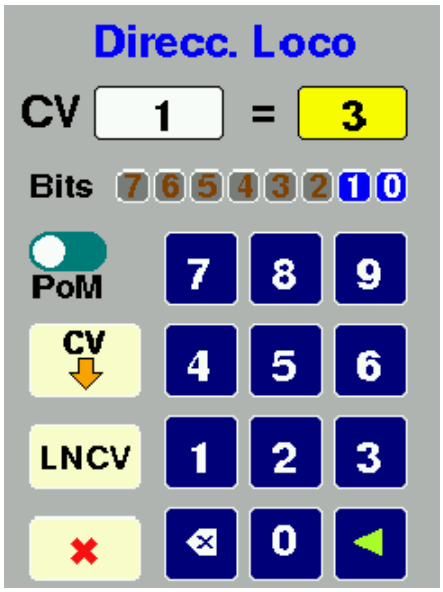
To finish modifying the panel, tap the  icon again.

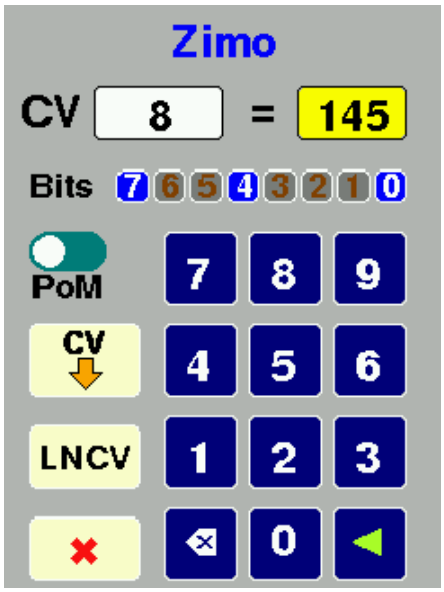
NOTE:

PacoMouse	Loconet	Lenz	Roco
			
			

8. CV Programming

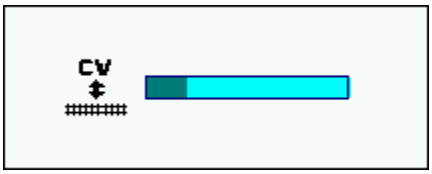
The decoder CV's can be programmed both on the programming track (read and write direct mode) as well as on the main track. (PoM, write only) In the *Loconet* version, LNCV's can also be programmed.





By tapping the CV field, indicated by the yellow background, the CV number may be entered with the numeric keypad.

Tapping this icon will read the CV from the programming track (PoM) and after a brief pause, the result will be displayed in the data field. The active bits corresponding to the read value will be shown.



Selecting the data field, indicated by the yellow background, the CV value can be edited using the keyboard. The active bits will change reflecting selection of the value. The active bits may also be changed by tapping them, this changes the corresponding data.

Tapping this icon will write the value to the CV on the programming track (PoM) or the main track (PoM)for the locomotive selected by the controller.

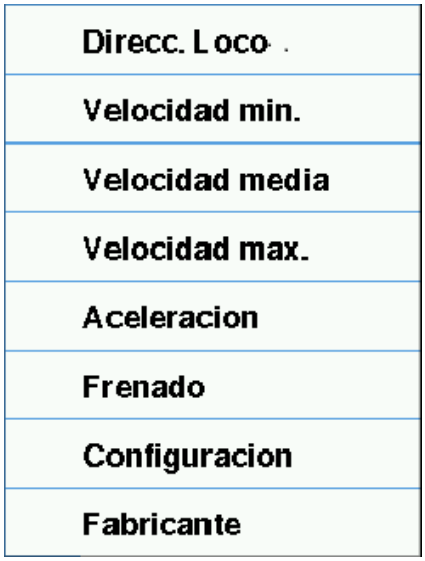
For the general CV's, the corresponding name will be displayed at the top. Entering an incorrect value in the CV number or data will result in a highlight with a pink background.

For the Xpressnet protocol, the maximum number of CV's is either 256 or 1024, depending on the controller type. Using the CS2 protocol will only allow CV programming if the selected locomotive is DCC or MM.

ATTENTION: If the loc. is MM, CV's can only be programmed, not read. The programming track uses address 80, so all locomotives will be programmed. Additionally, tapping the icon allows for modification of the main CV's, using the programming track..

Tapping this 

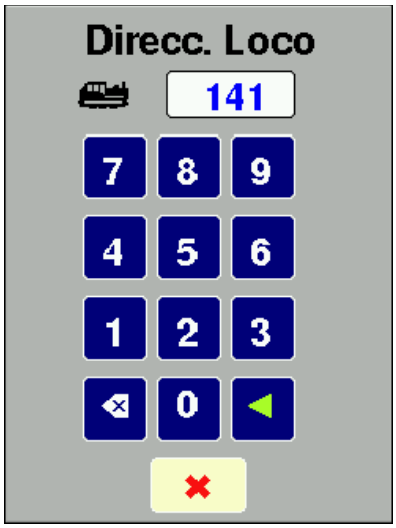
icon will read the CV from the programming track and display its value on the screen:



The main CV's are: Address (CV1, CV17, CV18), Minimum speed (CV2), Middle speed (CV6) Maximum speed (CV5), Acceleration (CV3), braking (CV4), Configuration (CV29) and manufacturer (CV8).

Reading CV8 will result in the manufactures name being displayed, if it's one of the more common ones..

The "locomotive address" option will read the current address from CV1 or CV17 and CV18, depending on CV29 designating a short or long address use.



The new address can be entered with the keypad and if  is tapped, CV1, or CV17 and CV18 as well as CV29 will be programmed. So if it's either a short or long address, it will be programmed correctly.

NOTE:

Using the ECoS protocol, CV programming is only available from firmware version 4.2.3 and onward.

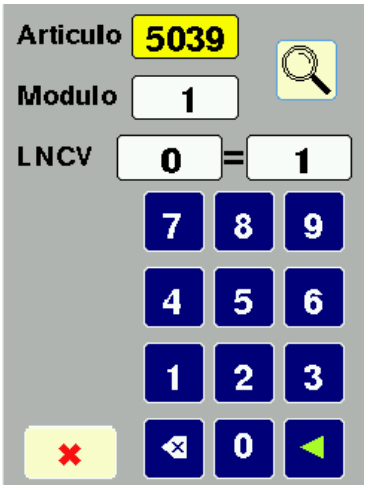
Make sure to consider this when assigning an address between 100 and 127:

- For loconet systems, the short address ranges from 1 to 127, while for Xpressnet systems the range is 1 to 99. The Z21 protocol allows for matching the settings to the control panel.

- Control units like the Digikeijs DR5000 or the Z21 allow for specifying the highest short address number.

LNCV (Option only for Loconet)

Using the Loconet protocol allows for programming of the LNCVin the modules.Tapping the LNCV icon will display the available options for the LNCV. Start by entering the item number (Refer to the manual) for the item that is to be changed and tap



Some modules have a feature to allow for display of their part number and module number. To use this feature simply connect to the module to be programmed to the bus and tap the icon. If your module has this function, its part number and module number will appear.

Now, the LNCV's can be programmed. First, enter the LNCV in question (Refer to the manual) tap the icon to read the corresponding LNCV. An LNCV that's not implemented, typically displays as 65535 when read. To modify a value, tap the icon to change it

ATTENTION: The LNCV contains the module number; if it's modified, reread it for verification and check that the module does exit programming mode while set correctly.

9. Configuration

The *Configuration* section displays a menu with different option the user can choose from to set the **PacoMouseCYD** preferences:



LANGUAGE:

Tapping the "**Language**" line will change the language in the menus to either *Spanish*, *Catalán*, *English, German, Czech, French or Dutch*. The appropriate flag is shown for clarity.

SCREEN:

The screen brightness level can be set here. Also rotating the screen by 180º degrees is possible. Calibration of the touch panel is another selection here.



Activating the RGB led on the **PacoMouseCYD** board will show the control unit status.

RED: Short circuit / Emergency stop (STOP)
GREEN: Normal mode (GO)
BLUE: Programming mode

A different function can be assigned to the **BOOT** on the board.

In addition to the functions F0 to F28, alternate functions can be assigned, such as: Calibrate, Power off, Stop and Emergency brake, Manoeuvres, Next and Previous. (The latter for function block, accessory panels, brightness etc.).

ATTENTION

To start initial screen calibration, press and holding the **BOOT** button on the initial logo screen. This will open the Calibration screen, even if another function has been selected for this button. When the unit is operational after the start, the assigned function will apply.

SPEED:

When operating a locomotive, pressing the encoder button will stop it if it's in motion. Choice of stopping in accordance with the brake CV setting, by sending a speed 0 or emergency stop is possible.

Maneuvering mode can be activated, allowing the speed to be reduced to step 1 using the encoder. To stop the locomotive, press the button of the encoder. This provides a precise control of stops during manoeuvres. Pressing again will change the driving direction.



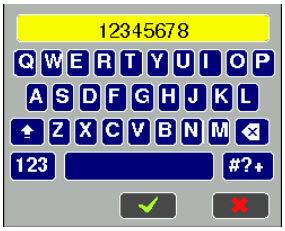
WIFI:

Selection of four networks and protocols is possible. Tap the icon for this. By tapping the field behind the WiFi sign, a name can be given. Tapping the field below SSID starts a search for the networks in range. Tap the field behind *Password* to enter this for the selected network.



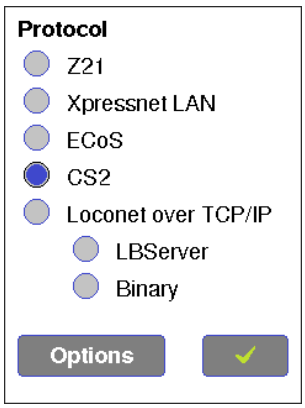




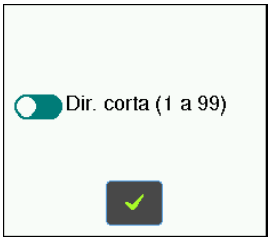


The IP address of the control unit and the protocol, along with the port to use can be entered using the numerical keypad.

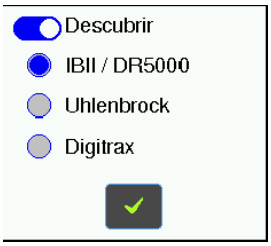
Tapping the protocol field, selects between available protocols and some inherent options, such as the maximum address and other options depending on the protocol of the control unit. E.g. the Z21 or*Loconet over TCP*:



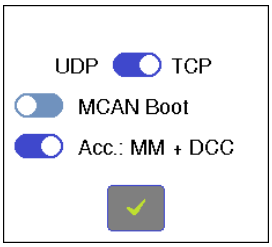
Options
Z21



Options
Loconet over TCP



Options
CS2



Select the server type for LocoNet over TCP/IP protocol between LBServer or Binary. The correct port must be selected to establish communication. The usual port numbers are shown in the table, but please also check in the server configuration.

Central	LBServer Port	Binary Port
DR5000	5550	5550
YD7001	1234	5560
Intellibox2neo	1234	
Daisy II WLAN	1234	
JMRI	1234	

For the CS2 protocol, choose between UPD or TCP, the latter being preferred in the latest version by Märklin. *MCAN Boot* allows for initiation of communication with the Märklin Gleisbox if there's no MS2 connected and use the CAN2WLAN. Select also if accessories will only be controlled by DCC (1 to 2048) or as MM (1 to 320) and DCC (321 to 2048).

For the changes to take effect, the **PacoMouseCYD** needs to reboot.



CLOCK:

The time and *Fast Clock* rate can be set. The *Fast Clock* will be displayed if it's activated on the control unit.

For the Xpressnet protocol, it's only supported when connected to a LZV200 v4. For the Z21 protocol, it needs firmware FW1.43 or higher. The LocoNet protocol indicates whether the control unit supports *Fast Clock* in *slot 123* (Intellibox II, DR5000, etc.). It's not supported in the ECoS protocol. In the CS2 protocol, when connected to a CS2/CS3 v2.5.1 or higher, the factor is the duration of 1 minute in the model time in seconds.



LOCKING:

Using the **PacoMouseCYD** in a club or lending it to a guest may warrant locking certain options to prevent unwanted situations.

The *Locomotive* selection locks accidental choice of a different locomotive.

The *Accessories* selection prevents turnout positions to be changed.

The *Programming* selection prevents modifications to CV, accessories panels and station run options. This is especially useful if the central station doesn't have a separate programming track, as all locomotives on the track could receive the same programming for the CV. We can leave the control to the guest.



ATENCIÓN

Activating any of these locks also **disables** the ⏻ button to prevent de-energizing the tracks on the entire club layout.

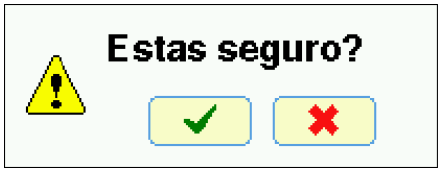
ABOUT:

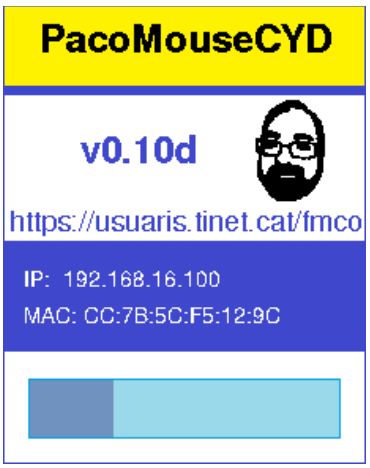
Tapping this will display the **PacoMouseCYD** info, like version, IP address and MAC address.



The **PacoMouseCYD** firmware can be updated from a CD card without a computer.

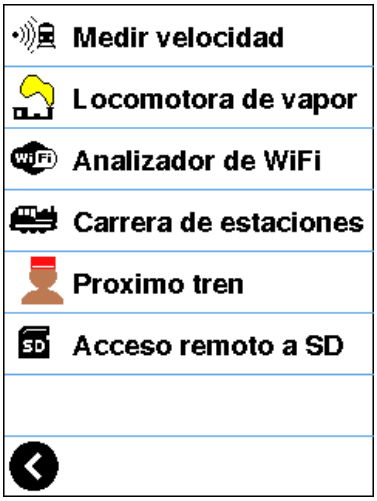
Tap the "*update firmware*" button. The root directory of the SD card will be searched for a file "PacoMouseCYD.ino.esp32.bin" (See Appendix II)





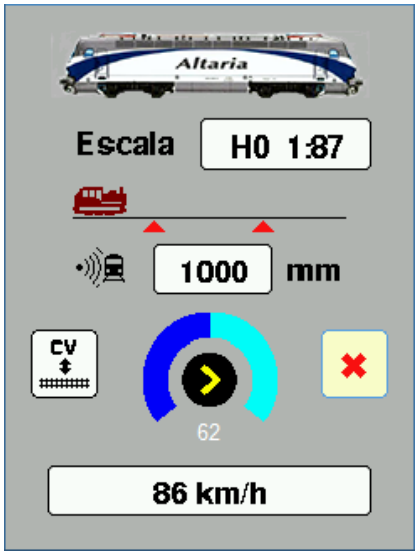
10. Utilities

The *Utilities* section displays a menu with different utilities offered by the **PacoMouseCYD:**

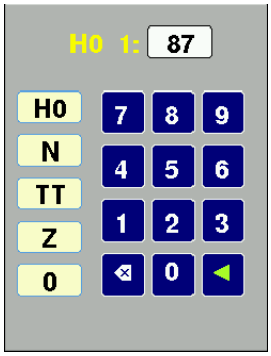


SPEED MEASURMENT:

This is the option to determine the scale speed of the locomotive so the speed can be properly calibrated in line with a real locomotive, so it corresponds to the configuration in the **PacoMouseCYD**.



1- Select the scale of the locomotive by tapping the field behind the scale, activate the keypad and enter the value. One of the common scales can be selected with the corresponding button.



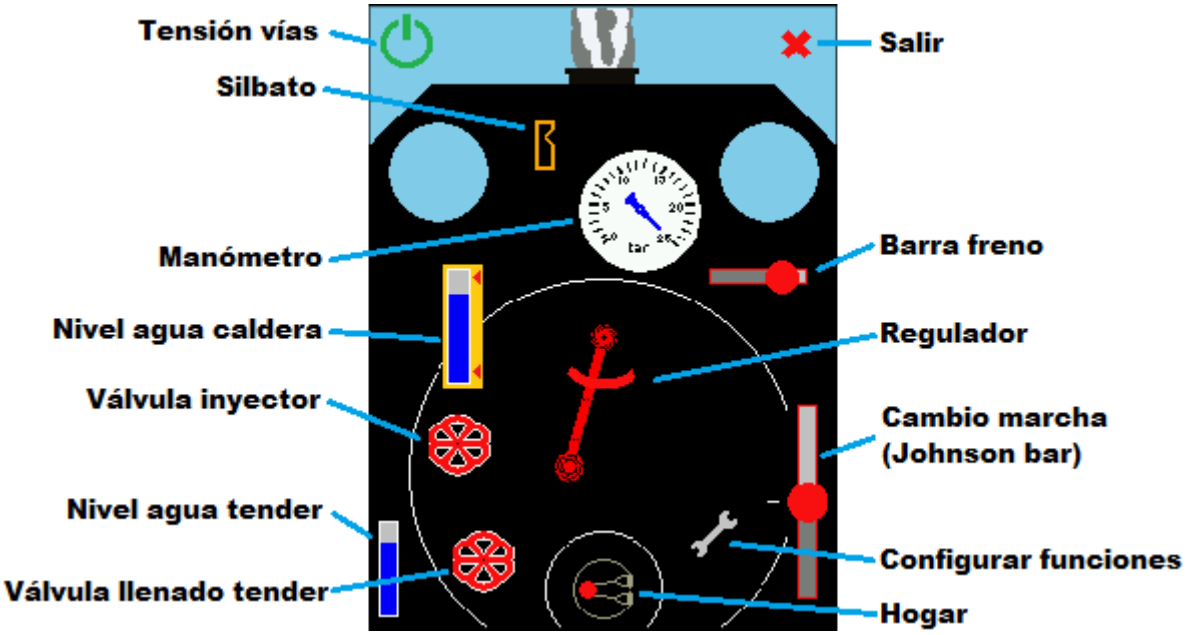
2- Choose the longest possible straight section of the layout so the locomotive can reach its maximum speed and measure the distance (in mm) between two easily identifiable reference marks (tree, figures, etc) Enter the measurement in the "*Length"* field.

3- Drive the locomotive at maximum speed and when it passes the first mark, press the encoder button. The texts "**Measuring**" will display in the lower field. At the second mark, press the encoder button again. The current scale speed of the locomotive will appear in the lower field. If the locomotive is stopped by pressing the encoder button, pressing it again will reverse the direction.

4- Ideally, the maximum speed reached by the locomotive should correspond to the speed of the actual locomotive. The speed can be adjusted with the corresponding CV. To do so, tap the  icon to directly access programming CV5 in PoM.

STEAM ENGINE (LOCOMOTIVE):

Using the rules and controls of a steam engine cab, it can be driven more realistically:



The regulator is operated by turning the encoder, this regulates the engines speed, provided that the brake  is released and the gear selector (Johnson bar) is far from the neutral position.

The gear selector regulates the amount of steam reaching the pistons, as well as the drive direction of the steam-engine. In neutral position (the middle of the bar) no steam will reach the pistons, so the engine will gradually loose speed. The further from neutral, the more steam thus resulting in more power (acceleration). This will increase the consumption of steam and coal. In turn water usage will increase and the pressure drops. Once the desired speed is reached, gradually move the gear towards its neutral position to reduce steam and coal consumption.

Keep the boiler water level between the two level indicator marks. If the boiler runs dry, the engine will stop to prevent damage. The boiler can be fed with water from the tender by opening the injector valve  This will introduce water into the boiler, against its pressure.

Refill the tender with water to increase the travel range. If the tender runs out of water, the engine will stop. Once a watering point is reached, open the tender water valve to get it filled Because the tender has a large capacity, filling it will take some time. Keep in mind that this operation cannot be performed with a moving engine!

Maintain boiler pressure. If the pressure drops, this will limit the maximum speed. At too low pressure, the engine will stop.

To increase boiler pressure, the fireman has to shovel coal. Open the fire box door by tapping this icon and shovel coal into the firebox .

To stop the steam engine, close the regulator by turning the encoder. Shifting gears to maximum will help reduce speed. Use the brake level for faster braking. In case of emergency, press the encoder button and the steam engine will stop as quickly as possible..

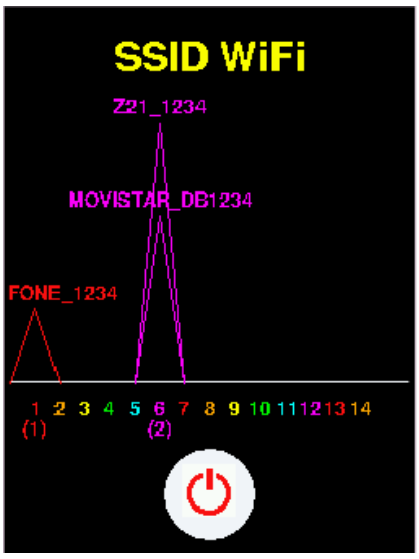
Remember to use the whistle while is required .

Tapping the icon will enable to configure the functions (sounds) associated with the whistle, injector, shovelling and filling the tender. The model should of course have these functions. **(TBD)**

WIFI ANALYZER:

There may be many active WiFi networks available. It may make using the remote difficult as sometimes there are multiple users for the same channels.

This tool provides checking the different networks and sees the relative signal strength. This way, verification of the conditions for good reception is possible.



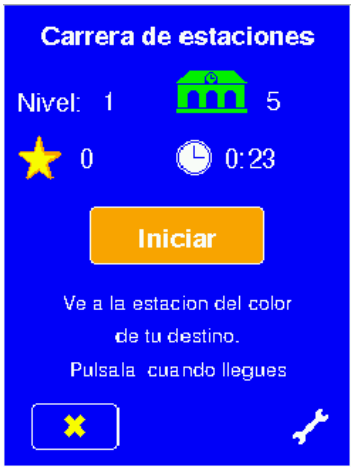
Tap and hold to close and initialize the **PacoMouseCYD**

STATION RUN:

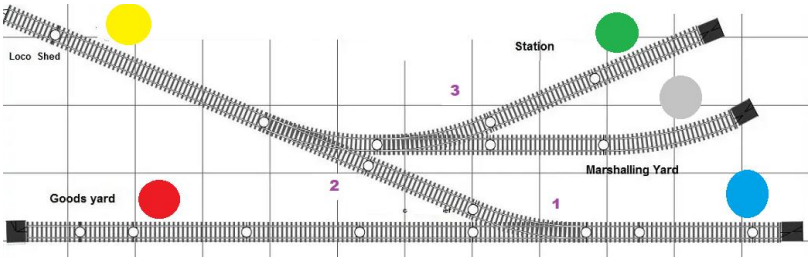
At the 2022 edition of *Expotren* fair, Alfred presented *Intelino*, an interesting smart train game system compatible with wooden train tracks. The intention is to encourage a love for trains among young children. It can be operated by using colour tiles on the track or form a tablet. It can even be programmed in *Scratch*. That day I was told a little anecdote:

"A kid who was playing said to him:"
- *I have reached level 4!*
- *What? - Alfred said in surprise. There aren't any levels in the game, show me.*
  The boy proudly showed him the tablet screen, indicating that he had already beaten level 4. Alfred saw that it was actuually one of the variations of the same game, but with a different track layout.
- *But for that, you have to change the tracks and set them up as shown in the diagram, he explained.*
  Then the boy got up and left.

The interesting thing about this anecdote is that you don't need to change the track layout of your model railway to overcome challenges, but you can also create engaging games for younger players on one of your layouts. I've adapted the *Station Run* game so that little ones can play on our layout



The game needs arranging between 3 and 5 stations (platforms, loading docks, etc) of different colours each (Red, Green, Yellow, white, blue) and a locomotive controlled by our little train driver. (It's the one selected by the **PacoMouseCYD,** where the maximum speed can be limited by programming it's CV to a value we choose).

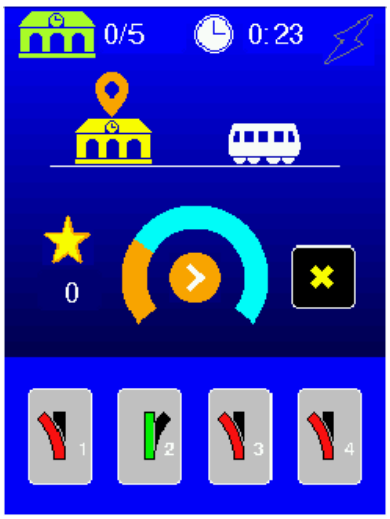


Within a limited time, all stations presented in the game must be reached.

The train can be driven freely and turnouts set until the desired station is reached. By tapping on the train icon, it can be stopped, if it hasn't already.

Once a mission is completed, a reward is earned with stars and levelling up, gaining an extra session and more time.

If time runs out before the mission is completed, the turn is forfeit and another player can play.











The game can be configured by tapping the icon. The number of stations can be modified between 3 and 5. The number or turnouts between 1 and 4.

The turnouts can be configured to correspond with our circuit or part of our model railway, displayed in the correct orientation, left/right as well as the control in case the position is inverted

Depending on the layout of the tracks, non digitized turnouts may be moved manually as well.

Take care to adapt the time provided to the actual time that is needed on the circuit. This ensures that levels can be finished and the game is fun, not a source of frustration.

Tapping the time or direction field will display a keypad for entry of the value.

The initial time is selectable between 10 sec and 255 sec (a little over 4 minutes) After each new level the time will be increased by a few seconds.

In case of a short circuit, for example by passing an incorrectly set turnout (frog polarisation) the icon will appear, but time will not be stopped. Remove the short circuit and tap the icon to resume the race.



ATTENTION

Adults can play too!

NEXT TRAIN:

Next train is a simplified adaption for the **PacoMouseCYD** of model train operations game prototype, based on cards and waybills inspired by the rules of the game "*Voll abgefahren: Der nächste Zug*."

Collect passengers and freight transport orders at the operating point of your model railway and transport them with the model railway trains to another operating point specified by the order.

Prepare any number of trains with up to 4 cars at any point of the operation (Platforms, loading track, depot) The recommendation is to have at least one passenger train and one freight train, or two mixed passenger and luggage trains for 1 or 2 players. For 3 or 4 players, take the dimensions of the layout and possible routes into account, to prevent a deadlock situation.

A train journey begins and ends at a point of operation, but must stop at its scheduled stops (Passenger trains on platforms, freight trains on loading tracks, mixed trains on both platforms and loading tracks). Only locomotives may stop at a depot. It will be able to continue its journey on the next run.

Plan the routes, considering the operating points capacity. Only one train journey per stop is allowed per round trip.

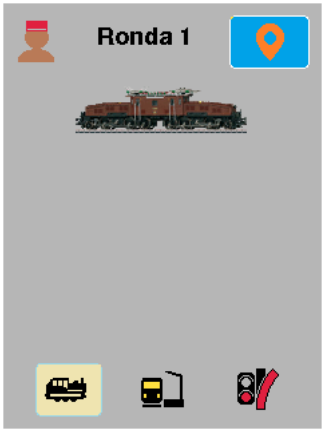


To play *Next train* an SD card has to be inserted in the **PacoMouseCYD**. The directory */game* contains files 0 to 9, each of them containing the cards for a different game.

Use **GameEditor** with the web browser to adapt the game to your layout specifics. (See Appendix III).

Choose the game number (from 0 to 9) and the number of players (from 1 to 4) and tap *Start*

ONE ROUND SEQUENCE:

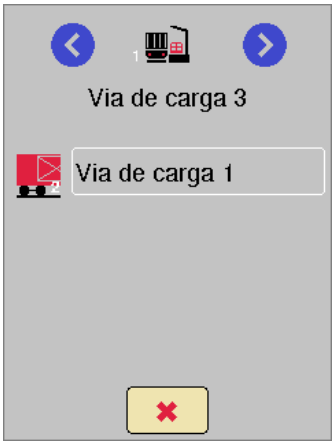


A different colour cap () identifies each player. The game is played in rounds until all cards are used.

The player and current round are indicated at the top.

Tap the image to choose the next train.

- Tapping the  icon show the other available cards at each point of the operation: Platform: , Loading track: , Depot: .



Select the starting point where the train is located.

Next to the operation point icon, you'll see the number of wagons (from 1 to 4) New transport bills exceeding the loading point capacity will be discarded automatically.
The points earned by a delivery are displayed next to the train car icon.

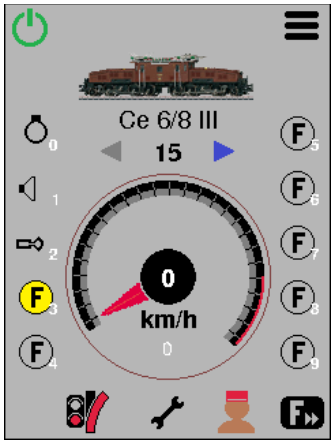
Select the appropriate loading orders that match the type of wagon being carried by the train.

Passengers:, Freight:, Luggage:, Mail:



Tap the icon or to perform one of the following actions:

a) **DRIVE**: Select a destination:
- Proceed in the current direction, from the operation point to the next scheduled stop accessible from this track.
- Change the signals and turnouts in the route to the next stop.
- Drive to the next scheduled stop.



b) **DIRECTION CHANGE**: Change the turnouts and signals needed for the locomotive to go from the current location to the other end of the train and couple it. On a push-pull train, simply change the drivers cab.

c) **MANEUVERS**: Perform the necessary manoeuvres (manoeuvre a car, etc) to have the train ready to depart from the current position to the next destination.



Tap the icon to continue the game and tap to indicate that this train round's moves has finished.



Indicate the operating point your train has reached (platform, loading bay, or depot).

A waybill is only considered complete once it's delivered to the destination.

Tap this icon to stop the game at any time.

Tap this icon  to have a new game card appear:





Order: Provided there's enough capacity, a new order for the car that's specified will be placed at the operation point.





Rewards: Use these to play when a related event appears. Any unused reward will increase the final score.



Event: Perform the described action.



This will be shown if a round is lost.



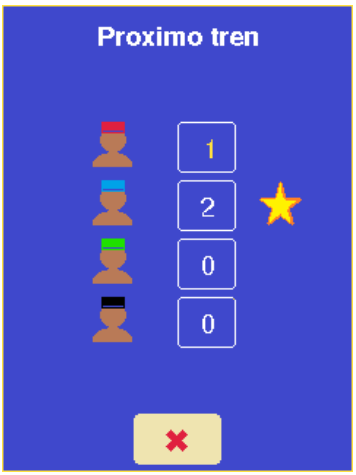
Losing a round can be avoided by using appropriate rewards.

Another played is up  provided there's no penalties for a lost turn.

END OF THE GAME

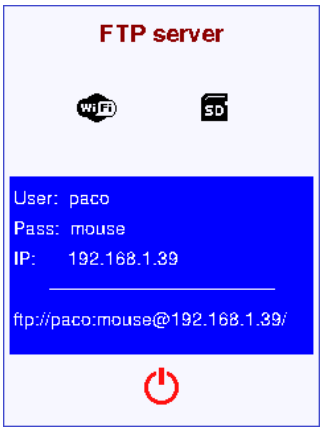
The game ends when all cards are used.

The players final score is displayed.
It's based on the amount finished orders and unused reward cards.



REMOTE ACCESS TO THE SD CARD:

The SD card in the **PacoMouseCYD** can be remotely accessed so it needs not be moved to the computer to access it to modify images and data files. (.csv)

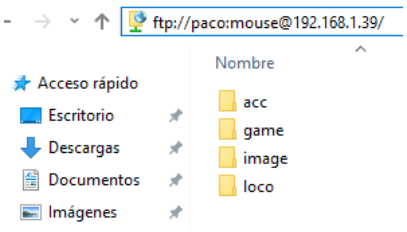


Access to the SD card is provided via a simple FTP server installed on the **PacoMouseCYD.** This supports file transfers between the computer and the card.

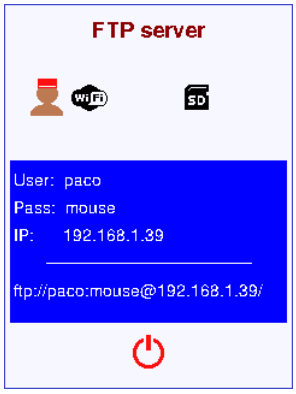
Connect the computer to the same router that the **PacoMouseCYD** is connected to. If this is not possible, connect one of the four networks to the router. (See the configuration section).

The window displays the credentials (username, password and IP address) for connecting to the FTP server via port 21 using an FTP client (FileZilla, WinSCP, etc)

To connect using Windows File Explorer, type the **ftp://** address shown on the **PacoMouseCYD** server screen into the address bar. (Always use lowercase letters)



Just wait a little while for the root directory of the SD card to appear.

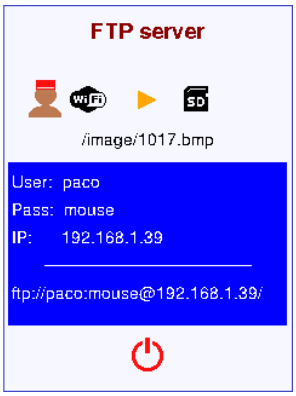


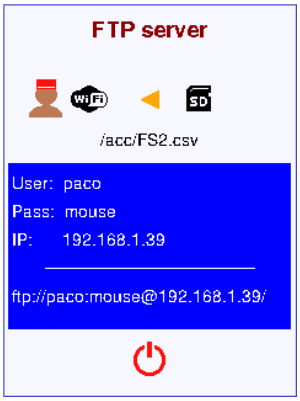
An icon will show that there's a user connected to the FTP server.

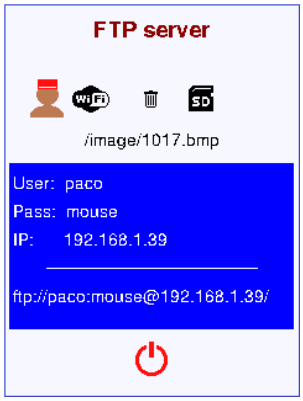
Now, the SD card files and folders can be accessed: saved, read, delete and renamed.

As this is a simple server, it is slow and only one simultaneous connection is possible So only establish a connection if large amounts of data are to be transferred using the FTP client (FileZilla, WinSCP,...).

Tapping the ⏻ icon will restart the **PacoMouseCYD** for proper access to the newly transferred files on the SD card.







11. Automation

PacoMouseCYD allows for an easy way of defining automations of the model railway, such as setting routes, controlling locomotives, etc.:



Definition of up to 50 different automations is provided. Next to each automation number, an icon displays its status:

 Automation not defined

 Automation defined

 Automation defined and active

For identification, a name of up to 16 characters can be given to an automation. Tapping the  icon or the name will start it. Tapping the  icon, will stop it.

To define a new automation or edit an existing one, tap the  icon.



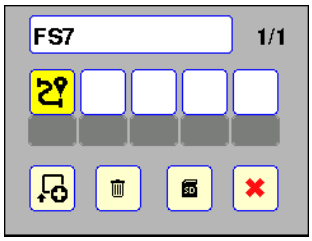
In edit mode, names and numbers are displayed in red to indicate edit mode. The following actions are available:

Editing: Tap the name of the automation

Delete: Tap the  icon

New: Tap the  icon

A newly created automation  is assigned a default name and contains only the end-of-sequence instruction.



Rotate the encoder to select different instructions and push the encoder to edit the highlighted instruction.

 Add a new instruction

 Delete an instruction

 Save the automation

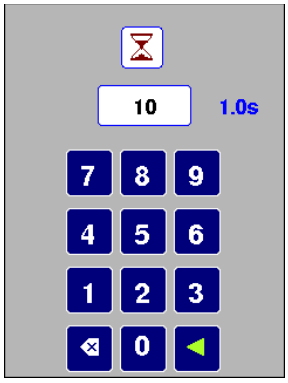
Available instructions are:

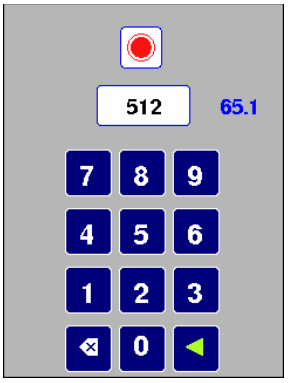
Instruction	Parameter	Description
	1..2048	Move accessory to red
	1..2048	Move accessory to green
	0..4095 *	Wait for contact to be inactive
	0..4095 *	Wait for contact to be active
	1..9999 **	Select a new locomotive
	0..100	Change the locomotives actual speed (0..100%)
	0..28	Deactivate locomotive function
	0..28	Activate locomotive function
		Locomotive reverse direction
		Locomotive forward direction
		Change locomotive direction
		Locomotive emergency stop
	1..60000	Wait a short time (in 100ms increases)
	1..50	Start a new automation sequence
	1..50	Start a new automation sequence and wait for completion
		Restart the current sequence
	0..2359	Wait for the clock time *(Fast Clock)*
	0..59	Wait for the clock time minutes *(Fast Clock)*
		End the automation

* The actual contact number and range depend on the protocol in use.

** The locomotive number is used in all protocols, except ECoS, where its ID (image number) must be entered. For MFX, the locomotive number is used and its correct UID (image number) must be defined.

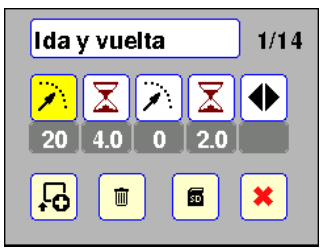
Tapping the icon will add a new instruction. For editing an instruction, press the encoder button to bring up a window for selection of the instruction and its parameter:





While entering a timeout and contact parameter, the changed value appears next to the field for easy interpretation.

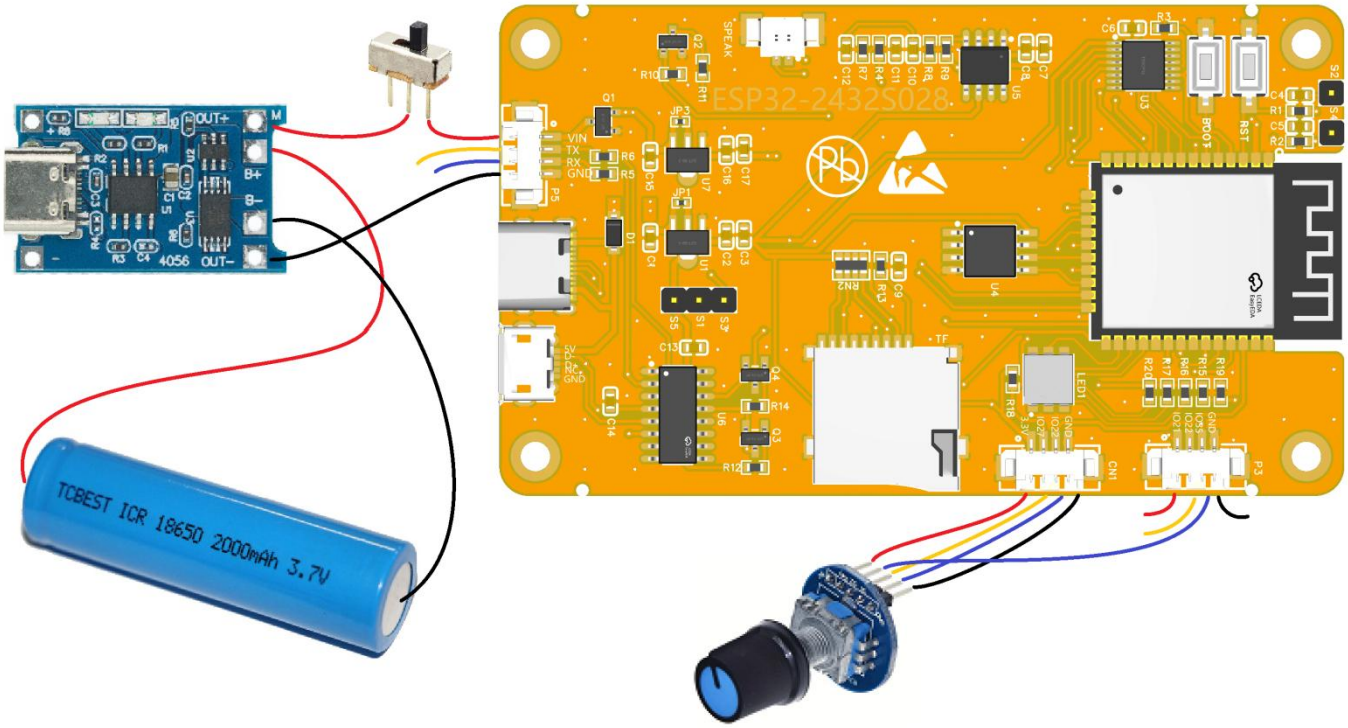
Tap the icon to save the instruction.

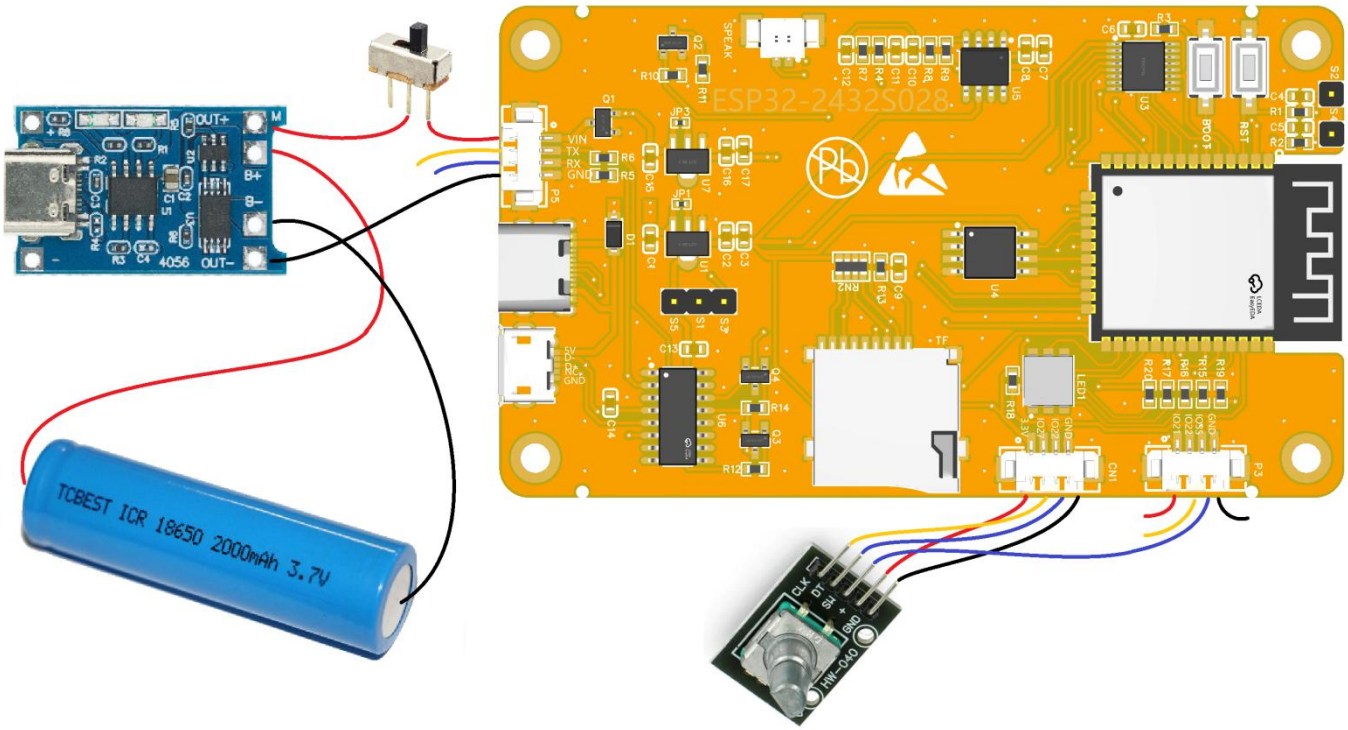


Once all instructions are entered, tap the icon to save them for later execution.

Appendix I: Diagrams

Connection diagram









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

ATTENTION



There are many different types of *encoders* and they may not be wired the same. If an *encoder* is working in reverse, swap the wires **S1/CLK** and **S2/DT.**





BATTERY	TP4056	CYD	CONNECTOR
+V	B+		
GND	B-		
	OUT+	VIN	P5
	OUT-	GND	P5

ATTENTION



Check the silkscreen of your CYD boards for the name of each pin before connecting a component

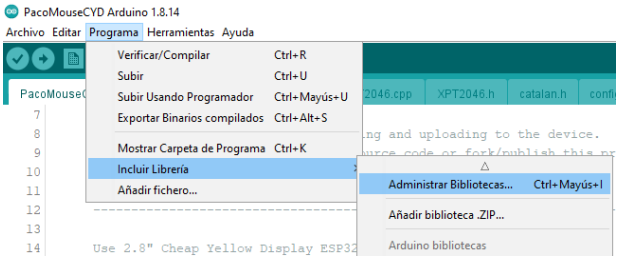
Appendix II: Programming with the Arduino IDE

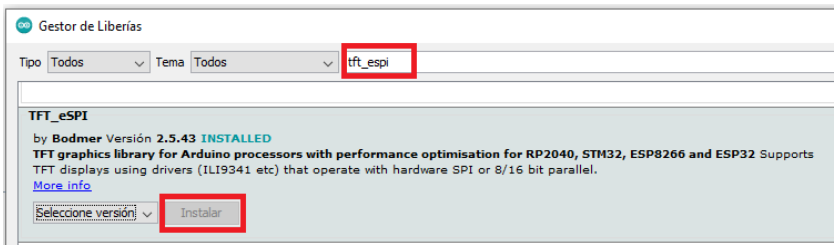
Program

Use the Arduino IDE (https://www.arduino.cc/en/Main/Software) to upload the **PacoMouseCYD** sketch for the *Cheap Yellow Display*, using the boards USB connection.

The *Cheap Yellow Display* uses an ESP32 processor and has a TFT screen with an ILI9341 controller and a touch panel controlled by an XPT2046. To compile and upload the program, the necessary libraries and boards have to be available to the IDE.

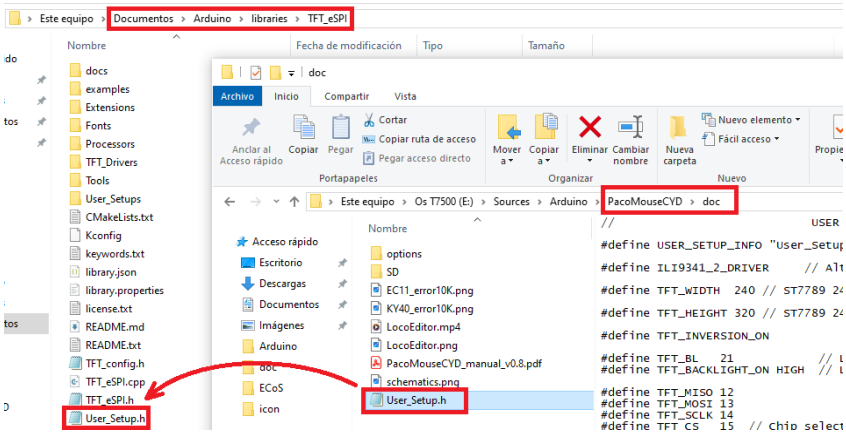
Open the **PacoMouseCYD.ino** file in the *PacoMouseCYD f*older. The **TFT_eSPI.h** (v2.5.43) library needs to be installed, using the option *Program -> Include Library ->Manage Libraries..*

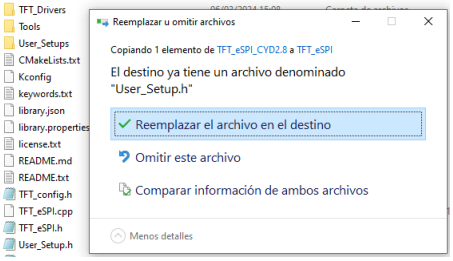




The **TFT_eSPI.h** supports more TFT displays with different controllers. The default configuration included with the library is not suitable for the *Cheap Yellow Display.* The *User_Setup.h* file located in *My Documents/Arduino/libraries/TFT_eSPI* needs adaption to the appropriate display type and pins used by the *Cheap Yellow Display.*

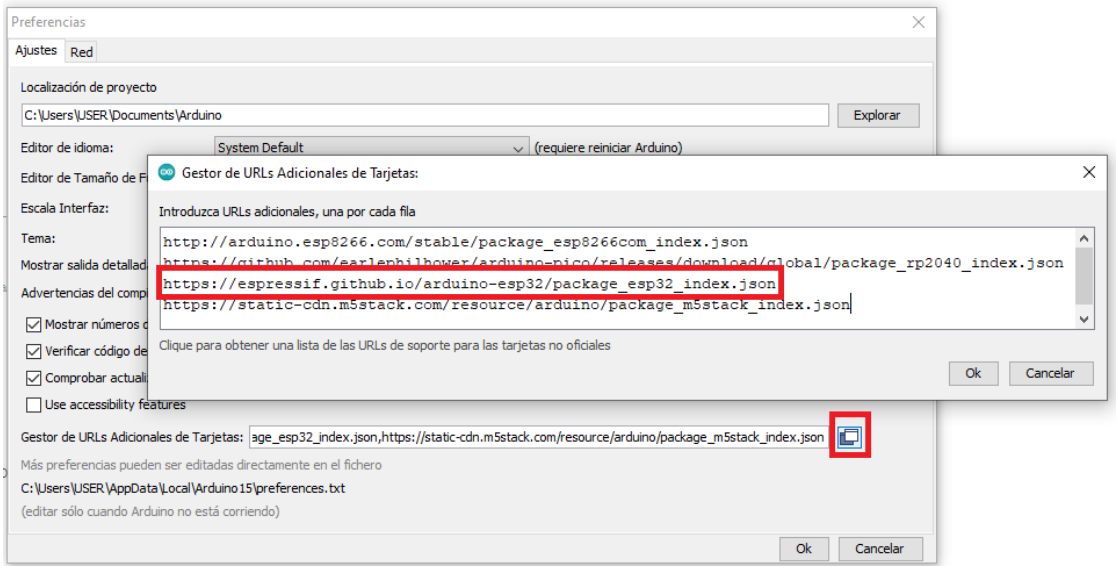
The suitable *User_Setup.h* file for the *Cheap Yellow Display* is located in the *PacoMouseCYD/doc/* . Copy it to the library in *My Documents/Arduino/libraries/TFT_eSPI*



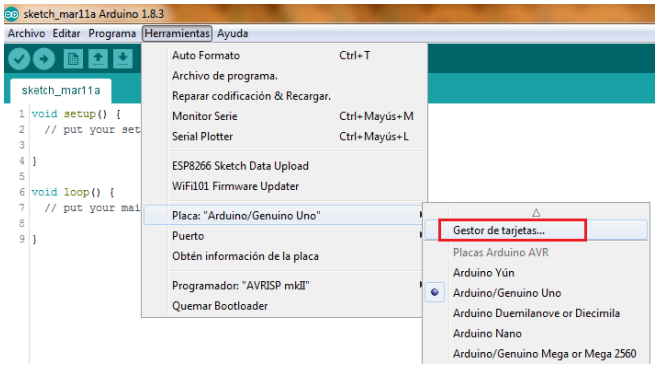


The *Arduino Core* for the ESP32 chip needs to be installed. In *File> Preferences* where it says additional boards manager, copy the following link:

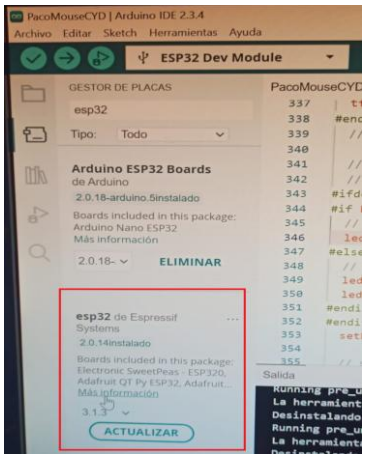
https://espressif.github.io/arduino-esp32/package_esp32_index.json

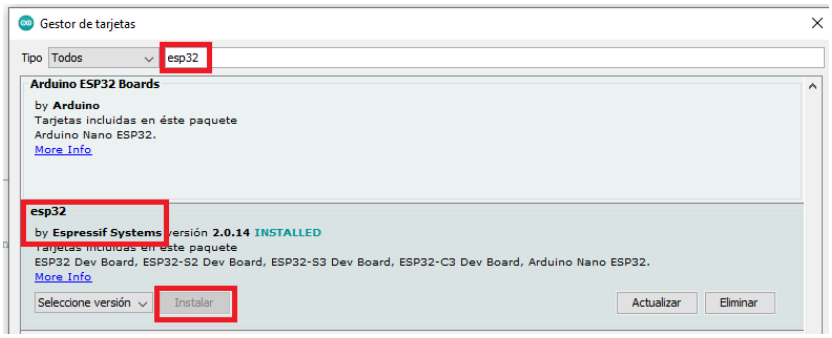


Go to *Tools > Boards: "…> Boards Manager…*



In the Boards Manager, search for "*esp32*" and install the ***ESP32 by Espressif Systems (v2.0.14 or v3.3.0)***





Configuration

Este software y los archivos asociados son un proyecto de bricolaje que no está diseñado para uso comercial.

EL SOFTWARE SE PROPORCIONA "TAL CUAL", SIN GARANTÍA DE NINGÚN TIPO, EXPRESA O IMPLÍCITA.

Las fuentes solo se proporcionan para compilar y cargar en el dispositivo.
No se permite modificar el código fuente ni derivar/publicar este proyecto.

EL USO COMERCIAL ESTÁ PROHIBIDO.

This software and associated files are a DIY project that is not intended for commercial use.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESSED OR IMPLIED.

Sources are only provided for building and uploading to the device.
You are not allowed to modify the source code or forward/publish this project.

COMMERCIAL USE IS FORBIDDEN.

Several configurations are available when programming the **PacoMouseCYD** from the Arduino IDE, depending on your setup.

Modify the ***config.h*** file, to adapt the code for your specific **PacoMouseCYD** setup.

ATTENTION: Non of the code for the**PacoMouseCYD** needs changing if the *Cheap Yellow Display* model **2432S028R** 2.8” is used, with a resistive touch panel. This is described in this manual.

Read this section only if you’re familiar with the subjects.

ATTENTION: Modify only this file (***config.h***) and only the section of code that starts with:

```
//////////////////////////////////////////////////////////
// ***** USER OPTIONS *****
//////////////////////////////////////////////////////////
```

And ends with:

```
//////////////////////////////////////////////////////////
// ***** END OF USER OPTIONS *****
//////////////////////////////////////////////////////////
```

Modifiable values for each option are mentioned below, shown in **red** and possible values are in **purple**:

NOTE: To **uncomment** a line, remove the // characters at the beginning of the line.

After setting the options for your version of the **PacoMouseCYD**, move on to the next chapter and program your board.

Cheap Yellow Display options

Select the hardware version of the CYD (*Cheap Yellow Display*). Use the CYD model 2432S028 a 2.8" with a resistive touch panel. It is also available in 2.4" and 3.2" sizes, although their functionality is not guaranteed. Other models may be tried if the configuration is known.

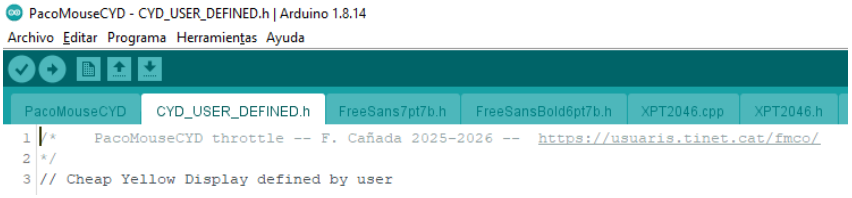
Use the ***User_Setup.h*** for the TFT_eSPI library that's compatible with your board.

```
#define CYD_TFT_28        0 // Cheap Yellow Display 2.8" (CYD 2432S028R). The only really supported.
#define CYD_TFT_24        1 // Cheap Yellow Display 2.4" (CYD 2432S024R)
#define CYD_TFT_32        2 // Cheap Yellow Display 3.2" (CYD 2432S032R)
#define CYD_USER_DEFINED  3 // Cheap Yellow Display other type. User defined: Define the pins in CYD_USER_DEFINED.h
#define CYD_ESP32_035     4 // Cheap Yellow Display 3.5" (ST7796 320x480)
#define CYD_ESP32_32E     5 // Cheap Yellow Display 2.8" with ESP-32E, no free pins for encoder, use LED RGB pins instead.

// Seleccione la version hardware del CYD (Cheap Yellow Display) de la lista    - Select the hardware version of CYD
(Cheap Yellow Display) from the list
// Use el archivo User_Setup.h correcto para la libreria TFT_eSPI - Use the correct User_Setup.h file for library
TFT_eSPI

#define CYD_HW_VERSION            CYD_TFT_28
```

For another type of screen than the CYD model 2432S028 de 2.8"or one different from the predefined ones, the pin configuration is needed (no guarantee for functionality) try selecting the **CYD_USER_DEFINED** and modify the data in the CYD_USER_DEFINED.h file accordingly.



To control the XPT2046 touch controller chip, indicate the most suitable access mode:

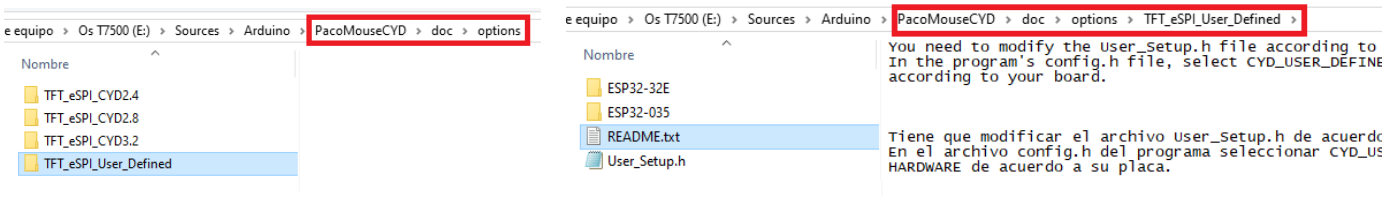
```
// Seleccione el modo de acceso al chip XPT2046 - Select XPT2046 chip access mode: USE_XPT2046_SPI /
USE_XPT2046_BITBANG
#define XPT_MODE          USE_XPT2046_BITBANG
```

If the XPT2046 touch panel response is rotated relative to the screen, change its orientation:

```
// Seleccione rotacion de la pantalla tactil - Select Touchscreen rotation: 0 / 1 / 2 / 3
#define XPT_ROTATION  0
```

The ***User_Setup.h*** file also needs to be modified to the board of your choice.

In the *PacoMouseCYD/doc/options* folder, find the *User_Setup.h* files and instructions for the *config.h* suitable for the different boards that were found. They may not be supported though, but included in case they may be useful.



RGB Led Options

The (*Cheap Yellow Display*) has an RGB LED at the back side. Although impractically placed, it may be used to display the status of the control unit.

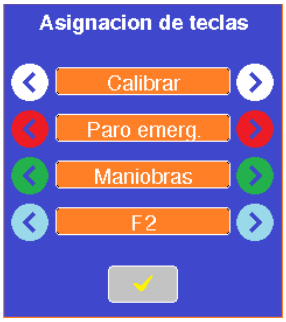
```
#define PRESENT           0      // LED RGB present. Normal use
#define UNUSED            1      // LED RGB removed. Using LED RGB pins for encoder
#define FUNC_BUTTONS      2      // LED RGB removed. Using LED RGB pins for pushbuttons (install pull-up resistors )


// Seleccione el uso del LED RGB - Select use of the RGB LED:PRESENT / UNUSED / FUNC_BUTTONS
#define USE_RGB_LED               PRESENT
```

If the board has insufficient pins to connect the encoder, an option (`UNUSED`) can be set, to use these pins for the encoder. Remove the code line and replace it with the pin numbers the encoder is connected to.

```
// Seleccione si usa el LED RGB - Select if use the RGB LED:
#define USE_RGB_LED   UNUSED
```

Another option (`FUNC_BUTTONS`) is to use the RGB LED pins to create three additional action buttons beside the BOOT button (See section 9) , in this case, remove the RGB LED and connect three buttons between the inputs and GND. Don't forget to install pull-up resistors, 10k is a good value, or reuse the ones from the RGB LED.




```
// Seleccione si usa el LED RGB - Select if use the RGB LED:
#define USE_RGB_LED   FUNC_BUTTONS
```

#define USE_RGB_LED	LED RGB pins	Buttons	Remarks
PRESENT	LED RGB	BOOT	CYD original 2.8"
UNUSED	Encoder	BOOT	If CYD has insufficient pins for the encoder
FUNC_BUTTONS	Buttons	BOOT, R, G, B	4 action buttons

Battery level reading options

An external battery level indicator can be used, so the **PacoMouseCYD** will not be required for this.

```
#define READ_UNUSED0     // Don't read Battery level. Normal use
#define READ_LDR          1     // Using LDR pin for battery level. Remove LDR. (max. 3.3V)
#define READ_XPT          2     // Using XPT2046 (pin 7). (max. 5V)


//Seleccione el modo de lectura del nivel de bateria-Select mode to read battery level: READ_UNUSED / READ_LDR / READ_XPT
#define BATT_MODE                    READ_UNUSED
```



Read the document about battery monitoring options: **PacoMouseCYD_Battery.pdf**

If any battery monitoring option is used, this  icon will appear in the upper left corner. Once the battery level drops below the configured minimum, recharge the battery as soon as possible.

The CYD (*Cheap Yellow Display*) has an LDR on top of the PCB. As its operation is affected by the screen backlight, the operation is inadequate, also because of unsuitable resistor values. For this reason, the **PacoMouseCYD** doesn't use it.

The input is analogue (effective range of 125mV to 3100mV) Once the LDR is removed, the input is free for alternate usage, such as measuring battery voltage (`READ_LDR`) modify the circuit for this. (Consult the battery document).

```
// Seleccione el modo de lectura del nivel de bateria - Select mode to read battery level: NOT_READ / READ_LDR / READ_XPT
#define BATT_MODE                    READ_LDR
```

ATTENTION



DO NOT connect any pin of the ESP32 to a voltage higher than **3.3V**, This will cause irreparable damage. Use an appropiate method to protect the input from voltages higher than 3.3V (resistor divider, zener, clamping diode, etc.)

The CYD (*Cheap Yellow Display*) contains an XPT2046 chip to read the touch panel. Pin 7 is assigned to connect to the battery voltage. The ESP32 can read its analogue value in a register on this chip.

The chip size makes it a challenge to solder a wire to pin 7 (pitch 0.65mm) Connect it to the VIN pin of connector P5 or pad S3. (Consult the battery document).

```
// Seleccione el modo de lectura del nivel de bateria - Select mode to read battery level: NOT_READ / READ_LDR / READ_XPT
#define BATT_MODE                    READ_XPT
```

ATTENTION



Prior to connecting something to pin 7 of the XPT2046 chip, check that it's not connected to any point of the CYD board.

General Options

LOCOMOTIVES AND THE STACK

PacoMouseCYD saves the locomotive that was selected last, for quick access. So there's no need to enter the number on the keyboard. Here the number of stack positions can be selected.

```
// Max. locomotoras guardadas en stack (hasta 254) - Max. locomotives saved in stack (up to 254):
#define LOCOS_IN_STACK            100
```

Data Options

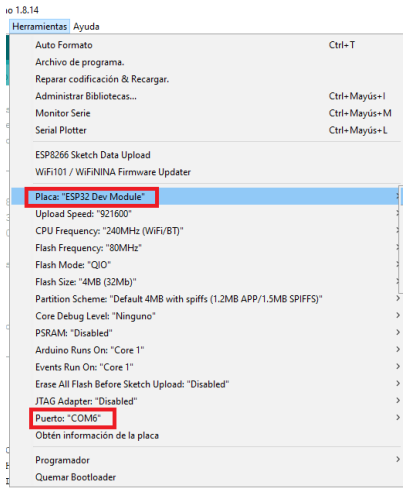
CSV file

PacoMouseCYD uses the SD card to store locomotive data and images. The data files are **.csv** so they can be opened in Excel for easy editing. The delimiter that's used, depends on the PC region settings and can be changed here.

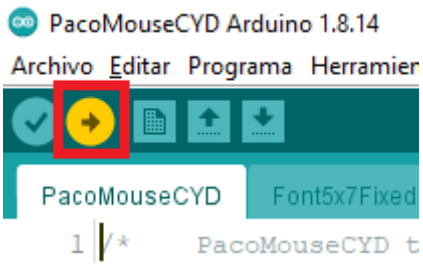
```
// Delimitador en fichero CSV - CSV file delimiter: ';' / ','
#define CSV_FILE_DELIMITER            ';'
```

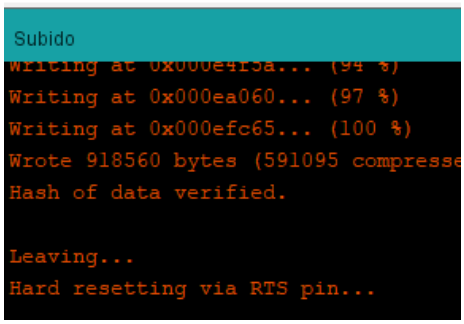
Programming the firmware using USB

Open the *sketch* and set the options in the ***config.h*** file. Make sure the ***ESP32 Dev Module*** is selected and connected to the output port in the Arduino IDE. The USB CH340 needs to be installed (https://www.wch-ic.com/downloads/CH341SER_ZIP.html) to make sure the connected port appears in the list



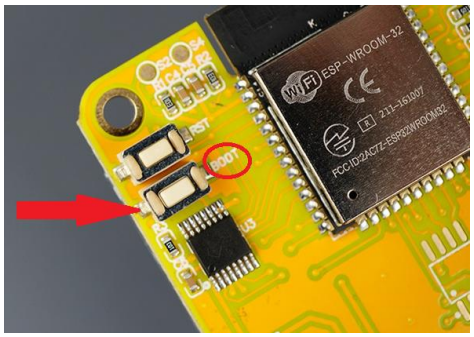
Click the "*Upload*" button. If all goes well, the upload will be confirmed to be 100% done.

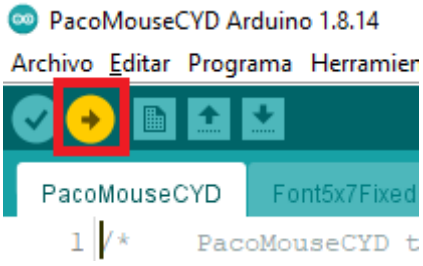


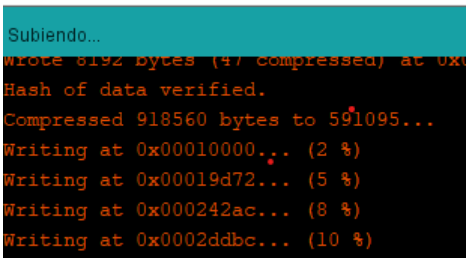


The processor will be reset and this starts the **PacoMouseCYD**.

In case of problems, start by checking if the port is correctly specified. The error message that the chip is not in *Download mode*, can be remedied by pressing and holding the Boot button, then press and release reset. Finally release BOOT and the correct mode is set:





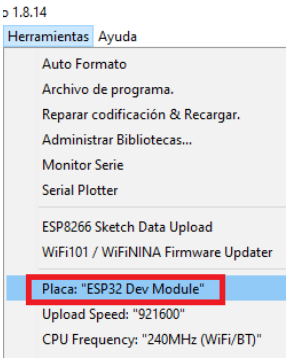


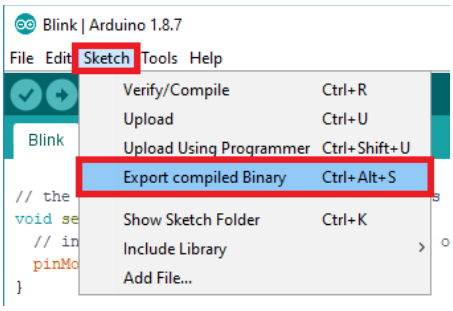
Firmware update from the SD Card

After programming the **PacoMouseCYD** for the first time, a **new** update version can be done using an SD card.

If the USB connector of the *Cheap Yellow Display* cannot be accessed easily or if multiple controllers need an update, use a .bin file in the root directory of an SD card.

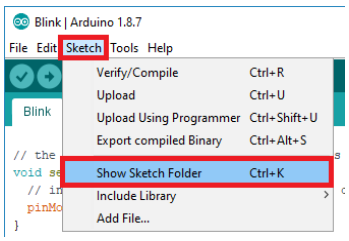
Open the *sketch* and update the options in the ***config.h*** file if needed. Check in the Arduino IDE that the ***ESP32 Dev Module*** is selected. To generate a .bin file, instead of clicking the update button, go to *Program > Export Compiled Binaries*

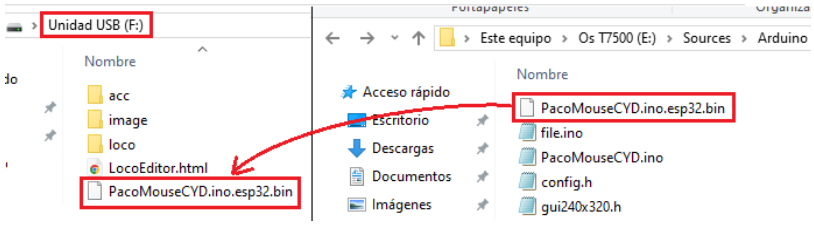




The file **PacoMouseCYD.ino.esp32.bin** will be created in the **PacoMouseCYD** *sketch* directory.

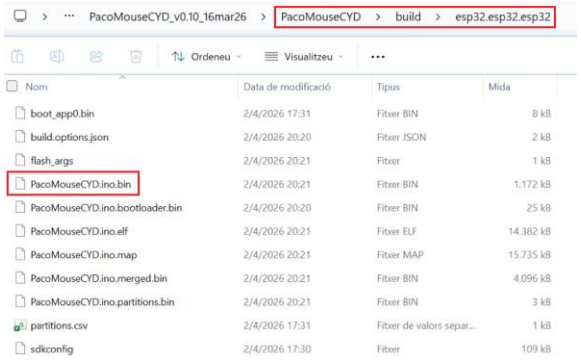
Go to *Program>Show program folder* to find the generated file *PacoMouseCYD.ino.esp32.bin* and copy this to the root directory of the SD card.





Arduino IDE v1.8.x

With the Arduino IDE v2 the .bin file is located inside the *build* folder with the name *PacoMouseCYD.ino.bin,* change it to **PacoMouseCYD.ino.esp32.bin** and copy it to the SD card.



Arduino IDE v2.x

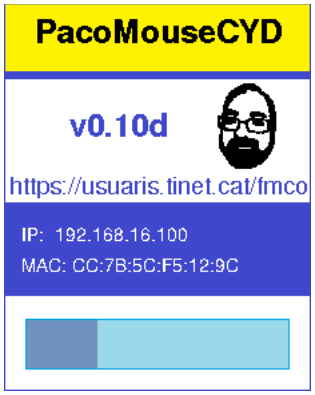


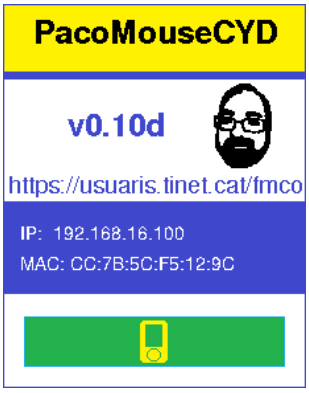


Insert the SD card in the **PacoMouseCYD**, activate it and go to *Menu> Configurtion>Settings...* and tap the *Update firmware* button.

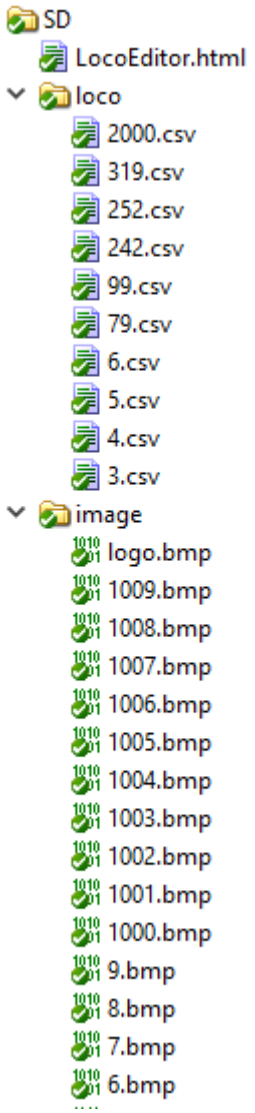
Tap to accept and the **PacoMouseCYD** will update and start with the new version.

A successful update will be indicated by a green bar. Failure will be shown with a red bar.





Appendix III: Content of the SD card



Images and data of the locomotives available to the **PacoMouseCYD**are stored on the SD card. The **PacoMouseCYD** can be used without an SD card. Then there will only be default black and white images available for the **PacoMouseCYD**.

LOCOMOTIVE IMAGES

Locomotive images are stored on the SD card in the **/image** directory. They are **.bmp** images with a size of 190x40 pixels (24bit), The file name consists of a number larger than 1000, corresponding to the locomotive's ID. Images with number 0 to 9 are defaults for the **PacoMouseCYD** and cannot be modified; they are needed for the *LocoEditor* to function correctly.

Some compatible images can be found on these websites. (Please comply with the terms and conditions for usage):

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

https://www.stayathome.ch/Lokbilder_ECoS.htm

https://fucik.name/PacoMouseCYD.img/

Use this tutorial to create your own images:

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

The background colour for the images is defined by 0xB6B6B6 (R:182, G:182, B:182)

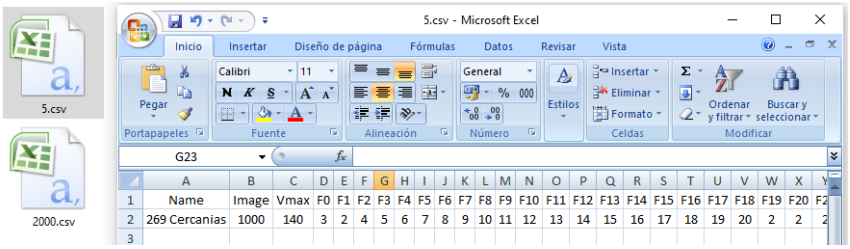
WELCOME SCREEN IMAGE

The welcome screen image is stored on the SD cards **/image** directory. The file name is **logo.bmp** with a size of 240x140 pixels (24bit)

LOCOMOTIVE DATA

The locomotives data (address, name, maximum speed, function icons) are stored as **.csv** files on the SD card in the **/loco** directory; the filename is the locomotive's address.

The files can be opened in Excel:

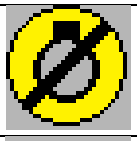
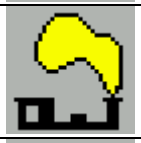
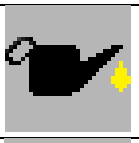
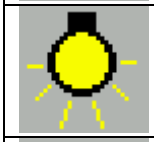
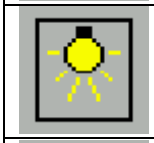
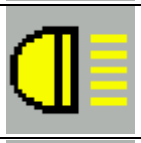
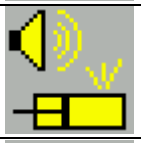
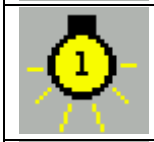
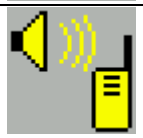
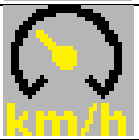
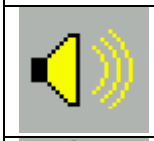
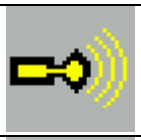
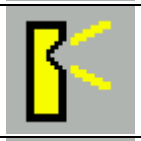
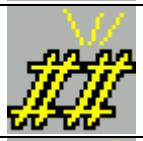
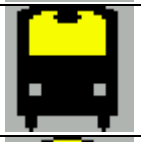


Column A: Locomotive name (maximum 16 characters)

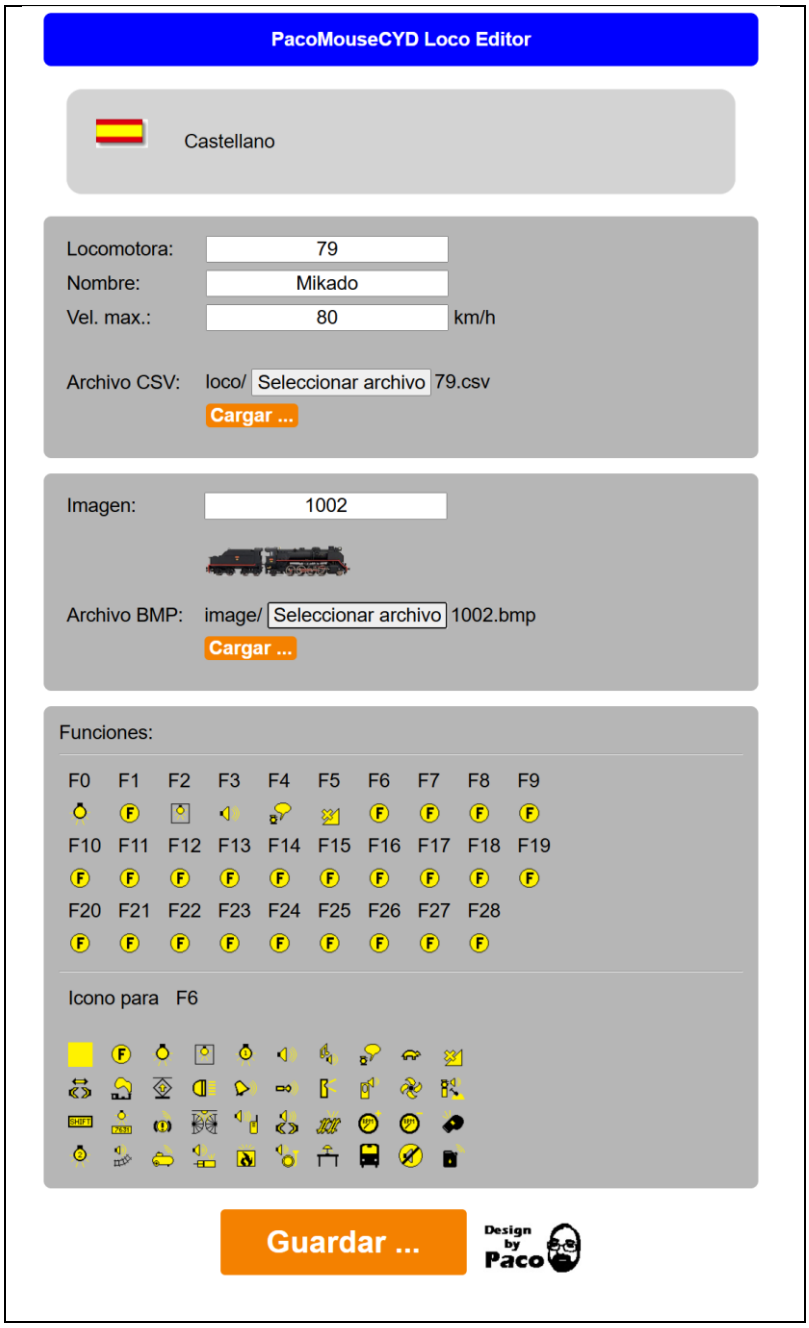
Column B: Locomotive ID. This corresponds to the name of the locomotive image on the SD card's **/image** directory if it's 1000 or higher (it also corresponds to the ECoS ID). Values 0 to 9 are default images for the **PacoMouseCYD.**

Column C: Maximum speed shown on the speedometer (in km/h). Ideally, this corresponds to the maximum speed of the real locomotive and to that of the scale model, provided the CV settings are correct.

Column D and further: Function icon ID for functions F0 to F28.

	1		11		21		31		41
	2		12		22		32		42
	3		13		23		33		43
	4		14		24		34		44
	5		15		25		35		45
	6		16		26		36		
	7		17		27		37		
	8		18		28		38		
	9		19		29		39		
	10		20		30		40		

PacoMouseCYD LocoEditor



To facilitate modification of locomotive data in a more visual way than Excel offers, the **PacoMouseCYD LocoEditor** has been included on the SD card to offer convenient modification of data using the web browser.

Open the **LocoEditor.html** on the SD card.

Enter the locomotive number, its name (maximum 16 characters) and the actual locomotives speed.

Select a **.csv** file from the SD card's loco folder and select **Load...**to read the stored data.

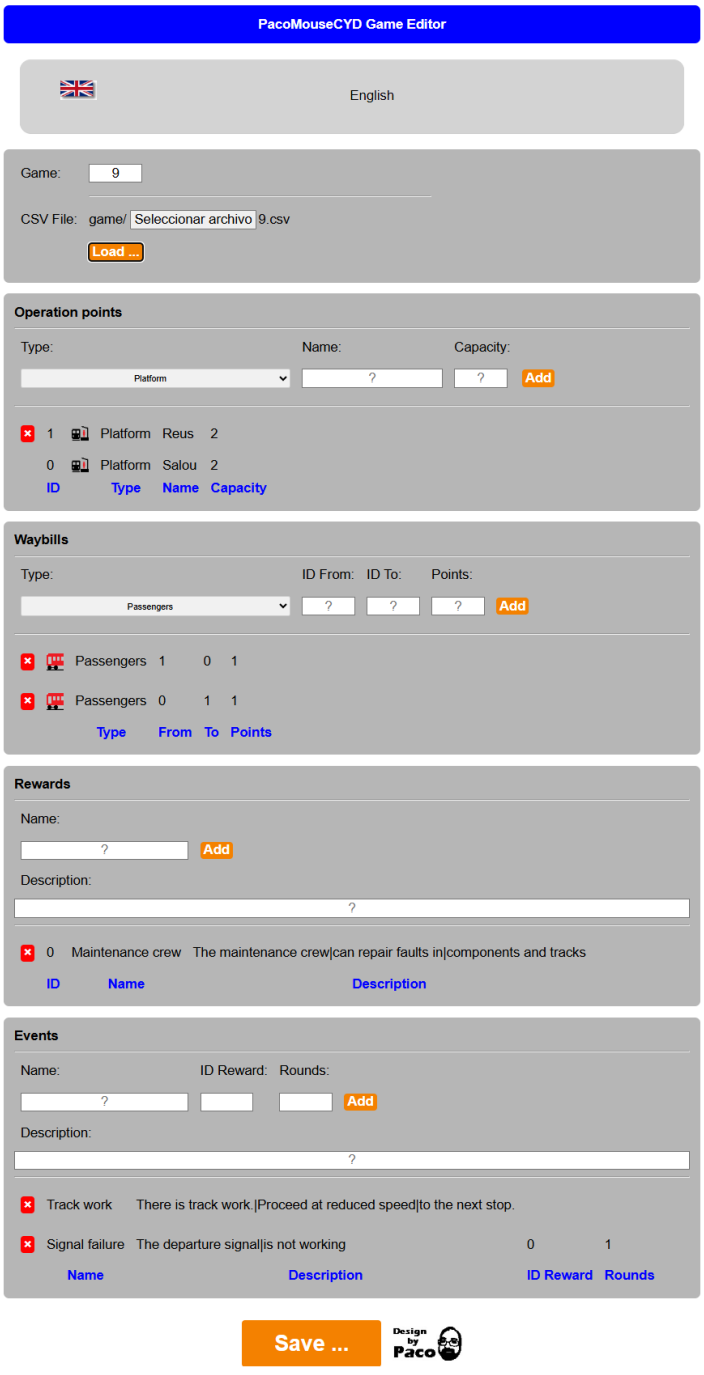
Enter the locomotive image ID or select it from the SD card's **image** folder. Pressing the **LOAD...** button will display the image. Images 0 to 9 cannot be modified as they are defaults used by **PacoMouseCYD**. Images 1000 and up are user-defined.

The function icons can be changed by clicking on them. All available icons will appear so the one that best suits the locomotives function can be selected.

Once all locomotive data is complete, click **Save...** to download the .csv file to the computer Copy it to the /**loco** folder on the SD card to make it available to the **PacoMouseCYD**.

With the ECoS protocol, only the images in the images/ folder are used where the filename corresponds to the locomotive ID provided by the central station. So the **PacoMouseCYD LocoEditor** is only useful for the other protocols.

PacoMouseCYD GameEditor



The cards for the game *next train* are defined in a **.csv** file in the **/game** directory of the SD card. Edit with Excel.

Card Type	*Card Subtype*	*Card description*
0 (Point of operation)	0	Platform
	1	Loading track
	2	Depot
1 (Transport order)	3	Passengers
	4	Goods
	5	Luggage
	6	Mail
2 (Event)	7	Event
	8	Reward

Name: Name (Operation points, rewards, and events)

Description: Description of rewards and events

NOTE: Don't use **;** in these fields. Use **|** to indicate a line break

Depending on the card type, the following fields will have different values:

Card Type	*Column E*	*Column F*	*Column G*
0	Capacity		
1	Origin	Destination	Points
2	Rewards	Turns	

Use the **PacoMouseCYD GameEditor** supplied on the CD card for more convenient editing of the game data is possible with Excel. This utility works with the web browser.

Open the **GameEditor.html** file from the SD card.

Enter the game number (0 to 9) to be edited.

An existing **.csv** file in the **/game** folder of the SD card can also be selected; press **Load...** to read the stored data.

Define the cards for the loading point capacity (1 to 4), the transport orders indicating the type of wagons (passengers, goods, luggage, mail) with the ID of origin and destination of the transport and the points that will be obtained by completion of the order.

With a reward card, losing a round from a relevant card can be avoided. Event cards add variety to the game by introducing unusual situations (both positive and negative) that can result in a player losing turns.

Once finished assembling all data for the card game, press **Save...** to save the **.csv** file to the computer. Copy the **/game** folder to the SD card to make it available to the **PacoMouseCYD**.

ACCESORY PANELS

Accessory panel data is stored in the **/acc** directory of the SD card. Panel names are stored in the **panel.csv** file and each panel's data (type, address, name, outputs) are saved in a **.csv** file with the panel number as file name.

Panel.csv

Column A: Name of the panel (max 12 characters)

Accessory Panel:

Column A: Accessory type number (See table in Chapter 7)

Column B: Address

Column C: Second address if needed

Column D: Name (max 6 characters)

Column E: Output configuration 0..65535 (one bit per active output)

Address2								Address							
Aspect 4				Aspect 3				Aspect 2				Aspect 1			
G	R	G	R	G	R	G	R	G	R	G	R	G	R	G	R

AUTOMATIONS

The data for each automation (name, instructions) is saved on the SD card's **/acc** directory in a **.csv** file, where the filename is FS followed by the automation number.

ColumnA: Name (max 16 characters)

ColumnB: Instructions (alphanumeric characters without spaces, ending with the character X)

Instruction	***Codeo***	***Parameter***	***Range***	***Description***
	t	Accessory Number	1..2048	Move accessory to red position
	T	Accessory Number	1..2048	Move accessory to green position
	f	Contact Number	0..4095 *	Wait for the contact to be inactive
	F	Contact Number	0..4095 *	Wait for the contact to be active
	L	Locomotive Number	1..9999 **	Selects a new locomotive
	S	Speed	0..100	Changethe speed of the current locomotive (0..100%)
	f	Function Number	0..28	Deactivates the locomotive function
	F	Function Number	0..28	Activates the locomotive function
	I	0		Reverse direction for locomotive
	I	1		Forward direction for locomotive
	I	2		Change locomotive direction
	I	3		Locomotive emergency stop
	D	Time (100ms multiples)	1..60000	Wait time (in 100ms multiples)
	R	Automation number	1..50	Starts a new automation
	C	Automation number	1..50	Starts a new automation and waits for it to finish
	r			Restart the current sequence
	H	Hours/Minutes	0..2359	Waits for the clock time hours*(Fast Clock)*
	M	Minutes	0..59	Waits for the clock time minutes*(Fast Clock)*
	X			End of sequence

* The actual contact number and range depend on the protocol in use.

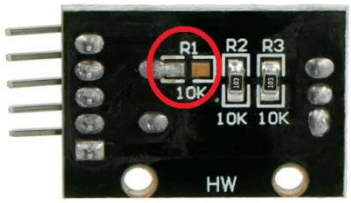
** The locomotive number is used in all protocols, except ECoS, where its ID (image number) must be entered. For MFX, the locomotive number is used and its correct UID (image number) must be defined.

Appendix IV: Tips

ENCODER

ATTENTION: Don't use and encoder without pull-ups or leave the encoders push button disconnected; hence the encoder movement will be unreliable.





Some KY-40 type encoders don't have 10K resistors installed on the push button pin.

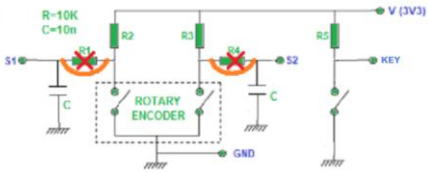
In that case, install a 10K resistor for R1.

The EC-11 type encoders may behave erratically because they have a series resistor of 10K between the encoder pin and the output.





In that case, bridge the 10K resistors and make a direct connection between the encoder pin and the corresponding output. The capacitors may also be removed.



BATTERY

Being a wireless controller, the **PacoMouseCYD** uses a rechargeable LiPo batter. Preferred 18650 type or 103450 with 2000mAh capacity is used for long battery operation.

18650 (Ø18 x 65mm)

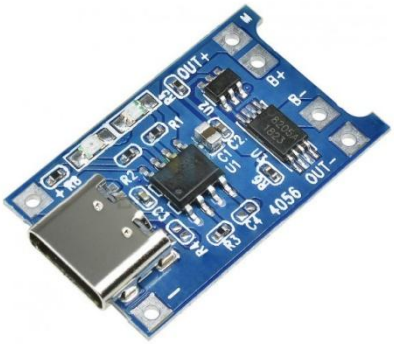


103450 (34mm x 50mm x 10mm)



The battery charging module uses a TP4056 type chip. This provides protection against over-discharge (preventing the battery to discharge below 2,4V) and also over charge (safeguard against exceeding 4,2V).

It has indicator LED's, one for charging (red) and one (blue) if charging is complete.



FAST CLOCK

The *Fast Clock* is displayed if it's enabled on the control panel. In the Xpressnet protocol, it's only supported when connected to an LZV200 v4. In the Z21 the firmware must be 1.43 or above. In the Loconet protocol, the control panel must support Fast Clock in *slot* 123 (Intellibox II, DR5000, etc.). In the CS2 version, it's displayed when connected to a CS2/CS3 v2.5.1 or newer. It's not supported in the ECoS version.

SD CARD

The **PacoMouseCYD** can be used without an SD card; however, the colour locomotive images will not be available then. The maximum supported SD card size is 32GB (FAT32 format). Insert the SD card before turning on the **PacoMouseCYD** otherwise the internal hard drive of the *Cheap Yellow Display* will be used for all operations / data.

The SD card stores colour images of the locomotives as well as their data. (With the ECoS protocol, only images are used, since the data is provided by the central unit). Data of accessory panels and games are also stored on the SD card.

INVERTED COLORS

If colours appear inverted after programming the **PacoMouseCYD**, this can be remedied by changing a setting in the *User_Setup.h* file in the library of the TFT_eSPI.h for the screen. (See Appendix II: Programming with the Arduino IDE).



In the *Documents\Arduino\libraries\TFT_eSPI* folder, open the file *User_Setup.h* and find the following lines:

```
// 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.
```

Change the following definitions to obtain the correct colours:

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

Or

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