

MANUAL

PLC LINUX

CHIPMUNK 3

© 2024 Ibercomp S.A.





INDICE

1 INTRODUCCIÓN.....	3
1.1 Características Técnicas.....	3
1.2 Aplicaciones.....	5
2 INSTALACIÓN.....	5
2.1 Conexionado.....	5
2.2 RS485.....	6
2.3 RS232.....	8
2.4 USB.....	9
3 CONFIGURACIÓN.....	10
3.1 Localización IP.....	11
3.2 Programa ChipMunk.....	13
3.3 Grabar SDCard desde cero.....	15
3.4 Arranque inicial.....	18
3.5 Primeras configuraciones.....	20
3.6 Directorios.....	22
3.7 Librerías WiringOP.....	24
3.8 Activar reloj en tiempo real DS3231.....	25
3.9 Java.....	28
3.10 Pantalla Virtual.....	28
3.11 Node-red.....	29
3.12 RustDesk.....	30
3.13 Dummy Video Driver.....	32
3.14 Servicio ChipMunk.....	35
3.15 Configurando red.....	36
4 COMANDOS DE INTERES.....	37
4.1 Control del Sistema.....	37
4.2 Información del Sistema.....	38
4.3 Gestión de Archivos y Directorios.....	38
4.4 Red y Conectividad.....	39
4.5 Manejo del Puerto Serie y Diagnóstico.....	39
5 CONEXIONADO.....	40
5 CONDICIONES DE GARANTIA.....	41



1 INTRODUCCIÓN

Dicen que en tiempos antiguos, una ardilla podía cruzar España de norte a sur sin tocar el suelo, saltando de árbol en árbol. Una de estas ardillas, inquieta y amante de la innovación, que pasó su vida entre los naranjos de la costa mediterránea, se encontró en la arena de una playa a un pingüino, llamado Tux, con una maleta llena de código abierto.

De esta amistad nació un equipo ágil como una ardilla y sólido como el Linux que ofrece una solución alternativa, económica y potente para automatización de cualquier tipo de proceso. Estamos convencido que este equipo le permitirá investigar, aprender, desarrollar y crear sistemas novedosos.

El ChipMunk 3 es un controlador de automatización basado en Linux, diseñado para ofrecer la flexibilidad de un ordenador con entradas y salidas (E/S) en un formato compacto y robusto. A diferencia de los PLCs tradicionales, el ChipMunk 3 proporciona un entorno abierto que permite la ejecución de aplicaciones avanzadas, integración en redes industriales y comunicación con dispositivos IoT.

Gracias a su arquitectura, abierta, basada en Orange Pi Zero 3, este equipo combina la potencia de un procesador Quad-Core ARM Cortex-A53 con conectividad industrial, facilitando la implementación de soluciones en automatización, control de procesos y supervisión remota.

Compatible con sistemas operativos Debian/Ubuntu, puede ser programado en Node-RED, Home Assistant, C/C++, Java, Codesys y otros entornos de automatización, permitiendo un sin fin de desarrollos personalizados.

1.1 Características Técnicas



1.1.1 Procesador y Memoria

- CPU: Allwinner H618 Quad-Core ARM Cortex-A53
- GPU: Mali G31 MP2, compatible con OpenGL ES 3.2
- RAM: 4GB LPDDR3

1.1.2 Almacenamiento

- Ranura para MicroSD (hasta 128GB)

1.1.3 Conectividad



- RS485 con control de flujo automático mediante PIC10
- Ethernet 1000M para comunicaciones rápidas y confiables
- Wi-Fi 802.11 b/g/n/ac (2.4GHz/5GHz)
- Bluetooth 5.0
- 2 puertos USB para periféricos adicionales
- Salida HDMI (Mini HDMI, soporta hasta 4K a 30Hz)

1.1.4 Alimentación y Consumo

- Entrada: 12V a 24V DC (compatible con sistemas industriales)

1.1.5 Software y Programación

- Sistema Operativo: Debian / Ubuntu
- Compatible con:
 - ✓ Node-RED: Creación de flujos de automatización IoT.
 - ✓ Home Assistant: Integración con dispositivos de domótica y automatización.
 - ✓ Webswing: Ejecución de aplicaciones Java accesibles vía navegador.
 - ✓ Bases de Datos: MySQL, SQLite y otras.

1.1.6 Diseño y Montaje

- Montaje: Caja DIN de 6 módulos
- Indicadores LED:
 - ◆ LED de alimentación
 - ◆ LED de estado de comunicaciones
 - ◆ 2 LEDs configurables por el usuario

1.1.7 Otras Características

- Reloj en Tiempo Real (RTC) con respaldo mediante condensador de 1F
- Temperatura de operación: 0°C a 70°C
- Soporte para decodificación de video H.265/H.264



1.2 Aplicaciones

El equipo es de propósito general y es ideal para un amplio abanico de aplicaciones.

- Control y automatización industrial.
- Monitoreo y gestión de sistemas HVAC.
- Gestión de energía y eficiencia energética.
- Supervisión remota en entornos industriales.
- Integración en redes IoT y Smart Home.

2 INSTALACIÓN

El equipo está montado en una caja de cuatro módulos, integrable dentro de un cuadro doméstico/industrial con carril DIN. Requiere alimentación de 12V a 24V corriente continua y tiene un consumo máximo de 100mA.

2.1 Conexionado

Todos los contactos están en regletas enchufables paso 5.08mm que aceptan cables de 1.5mm. Para las comunicaciones RS485, como indicaremos más adelante, debe utilizarse cable de par trenzado para las señales A+ y B- y otro par para unir la tierra (GND).



Debe prestar mucha atención a la polaridad de las conexiones RS485, porque en muchos dispositivos la señal negativa está en A y la positiva en B. Nosotros seguimos la nomenclatura utilizada en los datasheets de los transceptores.

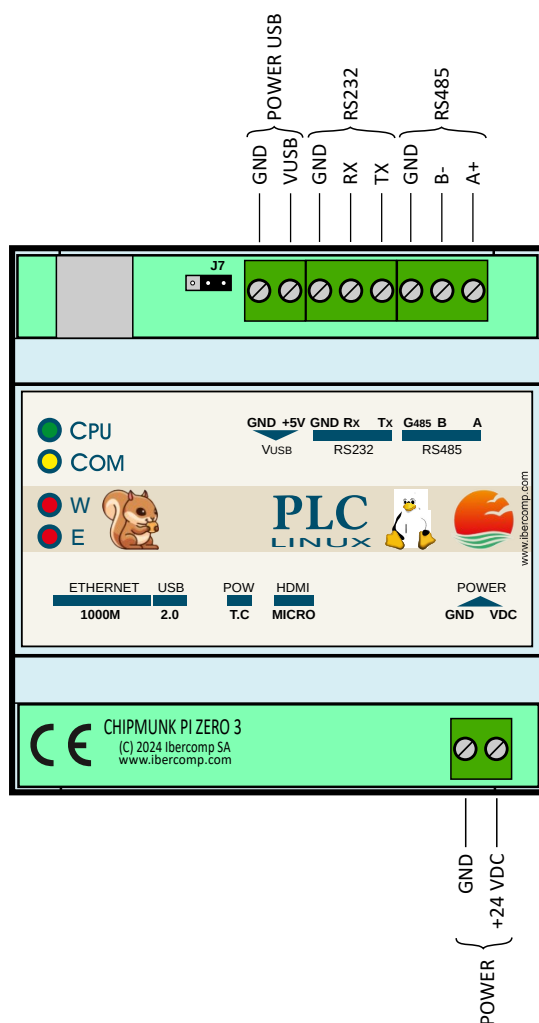
Las comunicaciones serie RS232c tienen niveles de tensión de puerto serie estándar, no se puede conectar directamente a un puerto serie TTL como los utilizados en otros dispositivos tipo Arduino. Es así para que sea más robusto y un pico de tensión no dañe inmediatamente el equipo. También permiten una longitud de cable algo mayor.

Para anclarlo sobre el carril DIN Omega dispone en la parte posterior de dos pestañas fijas que lo cogen por la parte superior y una pestaña móvil por la parte inferior. Primero hay que meter la parte superior y luego la parte inferior que debe hacer un click.

Para volver a extraer el módulo en la parte inferior hay que mover un poco la pestaña con ayuda de un destornillador plano.



El equipo dispone del siguiente conexionado:



2.2 RS485

Incorpora un puerto RS485 industrial basado en el transceptor Analog Devices ADM2587, un módulo aislado que permite una comunicación robusta en entornos exigentes.

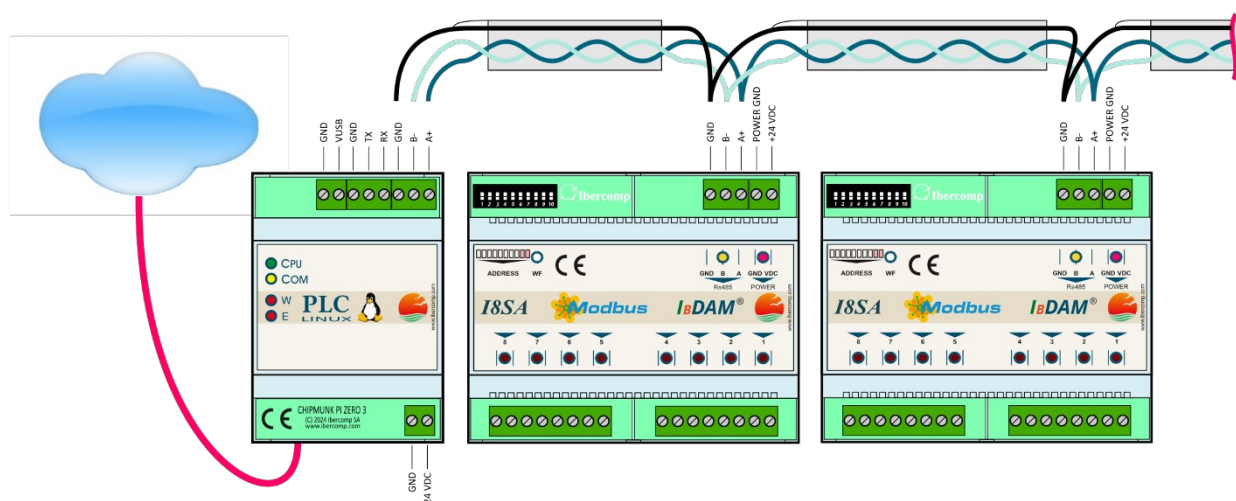
Este transceptor ofrece aislamiento galvánico de 2500V y soporta hasta 256 nodos, asegurando compatibilidad con buses de comunicación Modbus RTU y otros protocolos industriales.

Este puerto está conectado al puerto serie TTY55 o TTYA55, dependiendo si es Linux 6.1 o 5.7



Para mayor seguridad, añadimos protección de nivel 4 mediante un esquema de tres etapas:

1. TBU (Transient Blocking Unit): Protege contra sobrecorrientes transitorias, limitando la corriente a un nivel seguro.
2. TVS (Transient Voltage Suppression): Absorbe picos de voltaje rápidos, protegiendo contra descargas electrostáticas y picos transitorios.
3. GDT (Gas Discharge Tube): Dispositivo de descarga de gas que actúa como una protección secundaria frente a sobretensiones severas.



Este sistema de protección avanzada protege el bus RS485 frente a descargas electrostáticas, transitorios eléctricos rápidos y sobretensiones, cumpliendo con las normativas industriales más exigentes.

El puerto RS485 ha sido diseñado para cumplir con los estándares internacionales de inmunidad electromagnética, garantizando un funcionamiento estable en entornos industriales adversos.

Prueba	Nivel de Protección	Normativa Aplicable
ESD (Electrostatic Discharge)	± 15 kV	IEC 61000-4-2
EFT (Electrical Fast Transients)	± 2 kV	IEC 61000-4-4
Surge (Sobretensión transitoria)	± 6 kV (4/80 μ s)	IEC 61000-4-5

El puerto RS485 es polarizado por el conversor, con lo que se minimiza la susceptibilidad al ruido.



A la hora de realizar la instalación RS485 debe tener en cuenta unas exigencias mínimas:

- El cable que de comunicaciones es para pasar datos y no intensidad. No utilice cables gruesos usados habitualmente en instalaciones eléctricas para potencia. Utilice cables finos.
- Para mejor inmunidad eléctrica utilice cable trenzado y apantallado. Una buena opción es el cable Ethernet STP categoría 5. Es económico, fácil de adquirir y con características superiores a las necesarias.
- Conecte la pantalla del cable a tierra, pero solo en un lado. Por la pantalla del cable no debe circular corriente. Aunque es preferible que circule corriente a que no estén conectados los GNDs de los distintos equipos.
- Además de las señales RS485 A y B debe conectar también la señal GND. Sino lo hace corre el riesgo de ante tormentas o sobretensiones se dañen los transceptores. Una buena opción es utilizar un par de cables trenzados para la conexión GND. Otro par lo puede utilizar para transmitir alimentación +24V, pero tenga presente que son cables finos y que no pueden soportar mucha intensidad.
- Evite pasar los cables de datos en paralelo con cables de tensión. Aunque técnicamente puede que funcione, se incumple la normativa y puede ocasionar problemas de ruidos.
- Evite tener puntos de los que salen 3 mangueras cables. 485 debe ser una topología de bus, y cada uno de los nodos no debe tener un cable con una longitud superior a los 20 cm.

Los puntos anteriores son recomendaciones, no son dogmas de fé. Lo más seguro es que ignorando algunos puntos la red funcione, aunque lo correcto es seguir los puntos.

2.3 RS232

Dispone un puerto serie con niveles RS232, implementado mediante el integrado MAX202 sobre el puerto de depuración TTY0.

El puerto RS232 del ChipMunk 3 está diseñado para proporcionar acceso directo al sistema mediante una terminal, especialmente útil en situaciones donde las comunicaciones Ethernet o Wi-Fi puedan fallar.

Para establecer una conexión con un ordenador que carece de puerto serie, se puede utilizar un adaptador RS232-USB.



Configuración Predeterminada del Puerto Serie RS232:

- Velocidad de transmisión (Baud Rate): 115200 baudios
- Bits de datos: 8
- Bit de paridad: Ninguno
- Bits de parada: 1
- Control de flujo: Ninguno

Conexión mediante PuTTY

Para acceder al sistema a través del puerto serie utilizando PuTTY, siga los siguientes pasos:

1. Conexión Física:

- Conecta el ChipMunk 3 al ordenador utilizando un adaptador RS232-USB.
- Identifica el número de puerto COM asignado en el ordenador. Esto se puede verificar en el "Administrador de dispositivos" de Windows, bajo la sección "Puertos (COM y LPT)".

2. Configuración de PuTTY:

- Abre PuTTY.
- En la sección "Session":
 - Selecciona "Serial" como tipo de conexión.
 - En "Serial line", ingresa el puerto COM identificado (por ejemplo, COM3).
 - En "Speed", ingresa 115200.
- En la categoría "Connection" > "Serial":
 - Configure los parámetros mencionados anteriormente: 8 bits de datos, 1 bit de parada, sin paridad y sin control de flujo.
 - Haz clic en "Open" para iniciar la sesión.

3. Interacción con el Sistema:

- Una vez establecida la conexión, debería aparecer la consola del sistema.
- Si la pantalla está en blanco, presiona Enter para obtener el prompt de inicio de sesión.

2.4 USB



El equipo dispone de 3 conectores USB, uno que está en la propia placa Orange PI Zero 3. Los otros dos conectores USB están situados en la parte superior del equipo. Puede conectar dispositivos tales como teclado, ratón o disco USB de estado sólido.

Si desea conectar equipos que consumen más de 0.5A deberá proporcionar una tensión de alimentación adicional. Eso lo puede hacer con la entrada de +5V y moviendo el jumper a la posición cercana al conector de los USB.

Si lo hace debe estar muy seguro de que la corriente sean 5VDC, si mete por error una tensión mayor dañará con un 100% de probabilidad los dispositivos conectados al USB.

3 CONFIGURACIÓN

La unidad que entregamos tiene grabada en la sdcard interna el sistema operativo Debian Bullseye 6.1 con salida HDMI. Para poderlo conectar a un monitor HDMI necesitará un cable de micro HDMI a HDMI.

El usuario por defecto es:

orangeipi

y su password es:

orangeipi

También puede conectarse como root y su contraseña también es orangeipi.

El equipo dispone preinstalado además del sistema operativo de las aplicaciones:

Java JDK 11

Rust Desk

Node-red

chipMunk

Y tiene configurado el reloj en tiempo real y el puerto serie RS485.

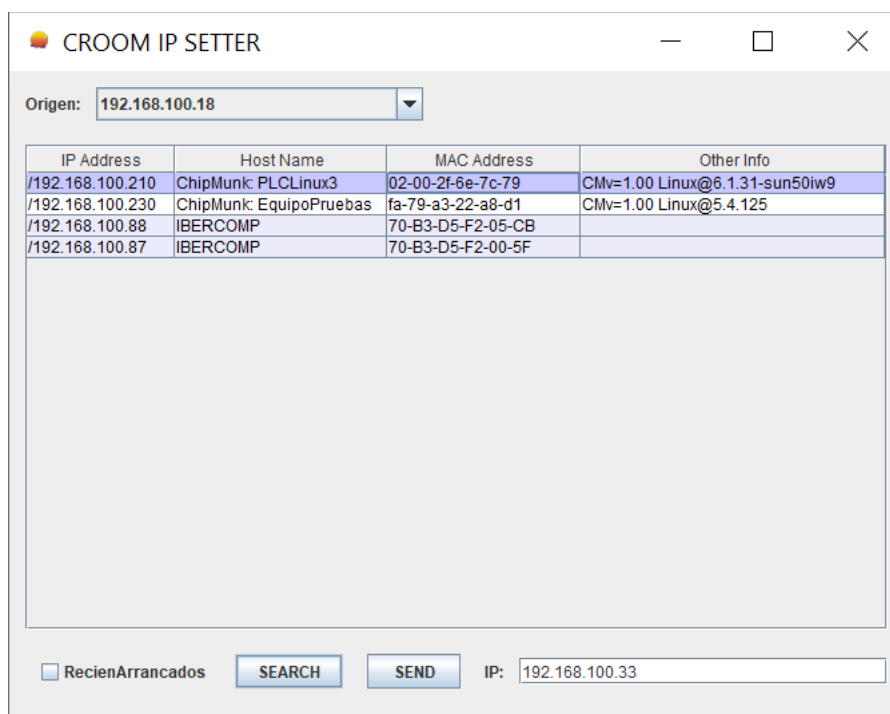
El programa Rust Desk permite realizar un acceso remoto al equipo, para ello debe conectar el equipo a una red que tenga servidor DHCP y un ordenador con Rust Desk en la misma red.



3.1 Localización IP

Lo primero que deberá hacer es averiguar la IP de su equipo.

Esto lo puede hacer con un terminal conectado al puerto RS232 o un terminal abierto en un monitor local y el comando `sudo ipconfig`, utilizando la aplicación IbercompDiscover o la aplicación Advanced IP Scanner.



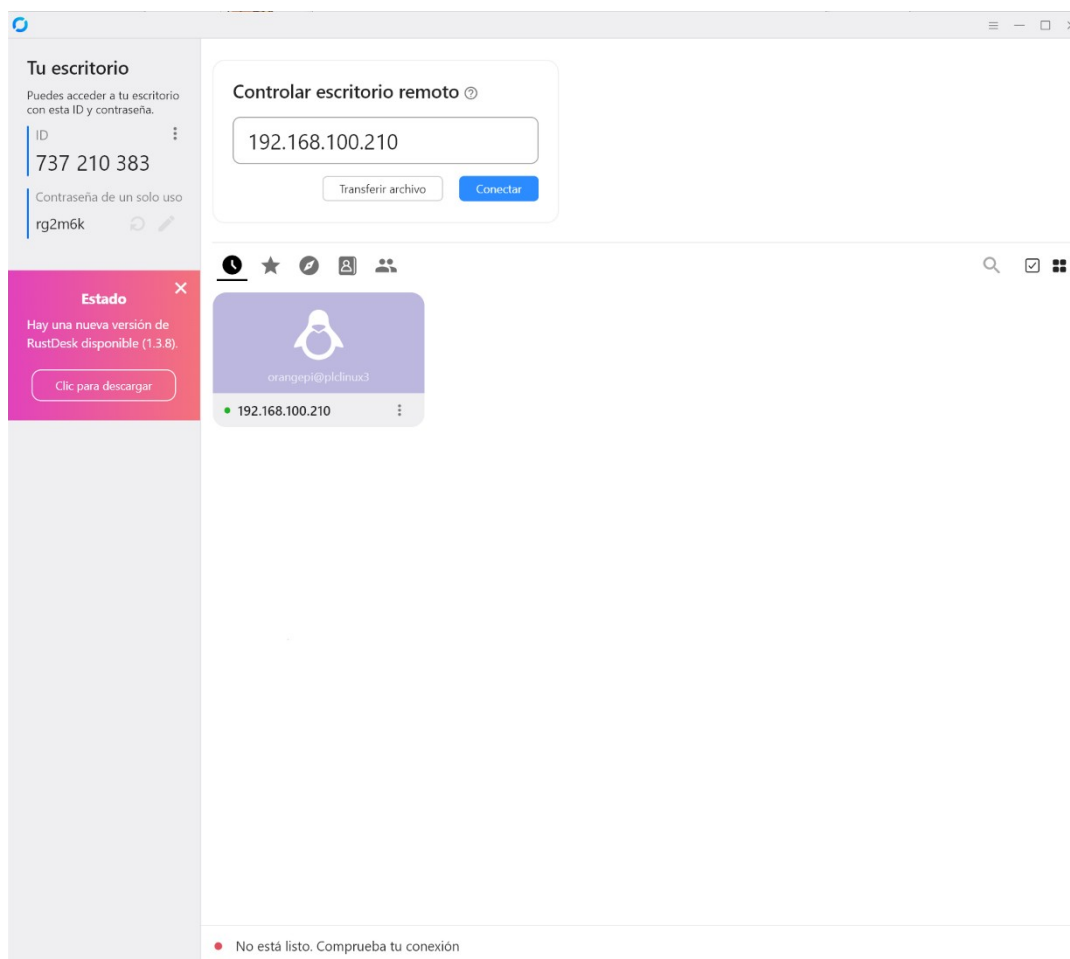
Para poder acceder al equipo con Rust Desk, poniendo su número IP, deberá tener conectado un monitor HDMI o instalar el driver dummy video.



Si instala este driver en principio ya no lo podrá volver a desinstalar, y deberá flashear una SD Card desde cero. Por tanto lo recomendable es sacar copia de seguridad de la SD Card con Win32DiskImager.



Una vez localizada la IP puede acceder sin temor mediante RustDesk al escritorio del PLC.



La contraseña del Rust Desk es: orangePI2025

Una vez haya entrado dispondrá de un escritorio normal en el que podrá ejecutar distintos programas.

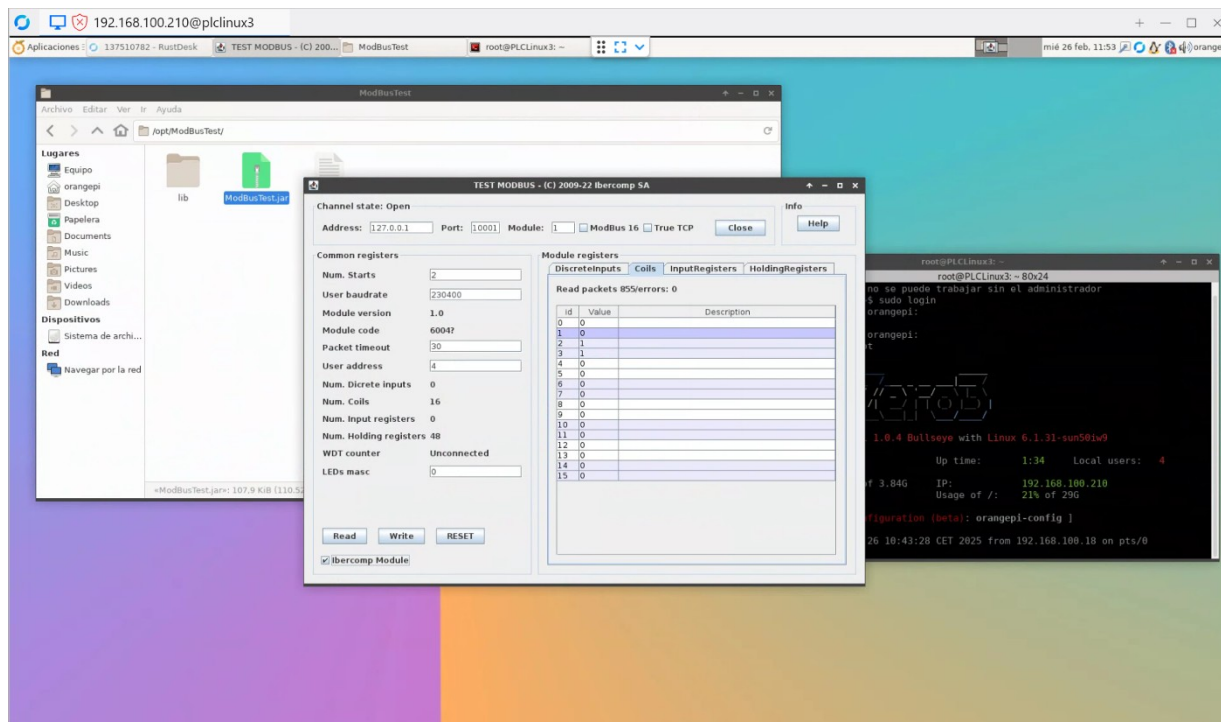


Debe tener cuidado en no tocar las configuraciones del Rust Desk, de las comunicaciones Ethernet o Wifi o contraseñas. Si lo hace se arriesga a que el equipo quede inaccesible. En este caso siempre podrá acceder por puerto RS232c.

Recomendamos que estudie con detenimiento la documentación y manuales de las distintas aplicaciones, para que pueda analizar todas las posibilidades que le brinda nuestro equipo.



A continuación de una pantalla de acceso al Chipmunk remotamente a través de RustDesk.



3.2 Programa ChipMunk

La aplicación ibChipMunk hace lo siguiente:

- Permite localizar la placa usando la aplicación ibDiscover. Se trata del puerto UDP 40404 que al recibir un carácter devuelve su identificación. Lo utilizan todas nuestras placas ethernet.
- Permite acceder a través del puerto TCP 10001 a un modulo modbus RTU, de un modo compatible al denuestro conversor estándar. Este puerto se puede manejar a través del puerto TCP 10001. Inicialmente está configurado como 9600 baudios, pero esto se puede cambiar mediante el fichero /etc /ibercomp /chipMunk /chipMunk.cfg. Si se cambia hay que ejecutar un reboot.
- Ofrece un módulo modbus virtual, en el puerto 10000, en el que se pueden leer en los registros algunos parámetros tales como temperatura, número de cores, frecuencias, cargas de trabajo, ...

Esta aplicación la puede descargar de nuestra Web, y está instalada en la carpeta /opt/ibercomp/chipMunk. Se ejecuta automáticamente al reiniciar el sistema como demonio.



El módulo virtual es accesible con protocolo modbus RTU over Ethernet, y de él se podrán leer las siguientes tablas:

Coils

nº	Tipo	Descripción
0		=1 Los LEDs son controlados por el sistema. Esto quiere decir que el LED CPU parpadeará, el COM por defecto indicará datos del puerto RS485 y que los LED W/E indicarán acceso a modulo virtual.
1		Estado del LED CPU
2		Estado del LED COM
3		Estado del LED W
4		Estado del LED E

Holding Registers

nº	Tipo	Descripción
0	EE	Número de arranques del programa chipMunk que debería coincidir con el número de arranques del equipo. Cada vez que el módulo arranca o es reiniciado se incrementa el valor de este registro en una unidad. Con ello se puede conocer si un módulo se está reiniciando de forma anómala.
1	RO	Versión del equipo. La versión viene codificada como versión*10 + subversión. La versión 1.0 se representa como 10.
2	RO	Código del módulo virtual, 50000
3	EE	Identificador de usuario. Es un registro de lectura y escritura persistente que puede ser utilizado para almacenar un número que identifica un módulo en la instalación.
4		R E S E R V A D O
5		R E S E R V A D O
6	RO	Número de registros tipo "discrete input"
7	RO	Número de registros tipo "coil"
8	RO	Número de registros tipo "input register"
9	RO	Número de "holding registers".
10		R E S E R V A D O
11		R E S E R V A D O
12		R E S E R V A D O
13		R E S E R V A D O
14		R E S E R V A D O
15		Hora RTC
16		Minutos RTC
17		Segundos RTC
18		Año RTC
19		MES RTC



nº	Tipo	Descripción
20		Día del mes RTC
21		Día de la semana RTC
22	RO	Temperatura del procesador en décimas de grado centígrado.
23	RO	Frecuencia de funcionamiento del procesador en MHz
24	RO	Número de usuarios conectados.
25		Carga media del último minuto.
26	RO	Carga media de los últimos 5 minutos.
27	RO	Carga media de los últimos 15 minutos.
28	RO	Número de núcleos que dispone el procesador.
29	RO	Carga de trabajo del procesador 1
30	RO	Carga de trabajo del procesador 2.
31	RO	Carga de trabajo del procesador 3.
32	RO	Carga de trabajo del procesador 4.
33		R E S E R V A D O
34		R E S E R V A D O
35		R E S E R V A D O
36		R E S E R V A D O

El programa no ofrece ni discrete inputs ni de input registers.

LEYENDA:

- EE Indica que son registros persistentes que se almacenan en la memoria E2PROM. Estos registros pueden ser rescritos pero debe tener presente que después de 1.000.000 de escrituras su contenido puede no ser correcto.
- ME Indica que el registro es almacenado en E2PROM y que puede ser reescrito un máximo de 1.000.000 de veces. Después la memoria puede no funcionar correctamente.
- RO Indica que se trata de un registro de solo lectura. Las escrituras sobre este registro serán ignoradas.

3.3 Grabar SDCard desde cero

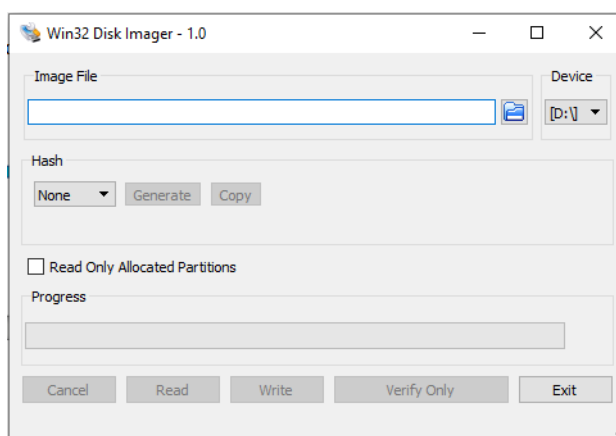
Si el sistema que le hemos suministrado no se adapta a sus necesidades, o bien se ha corrompido y no tiene copia de seguridad, no se preocupe, dando unos pasos podrá a partir de un sistema estándar configurar un nuevo sistema operativo en unos tres cuartos de hora (casi todos dedicados a descargar el sistema operativo y grabar físicamente la SDCard).



La sdcard debe grabarse con un sistema operativo derivado de Debian. La versión actual que hemos verificado es la bullseye 6.1 que se puede descargar de nuestra página (ya preconfigurada), de la página de armbian o de la página oficial de Orange PI Zero 3 (en estos dos últimos casos sin preconfigurar).

Existen otros sistemas operativos que puede explorar por su cuenta.

La grabación de la SDCARD puede hacerse con el programa Win32DiskImager.



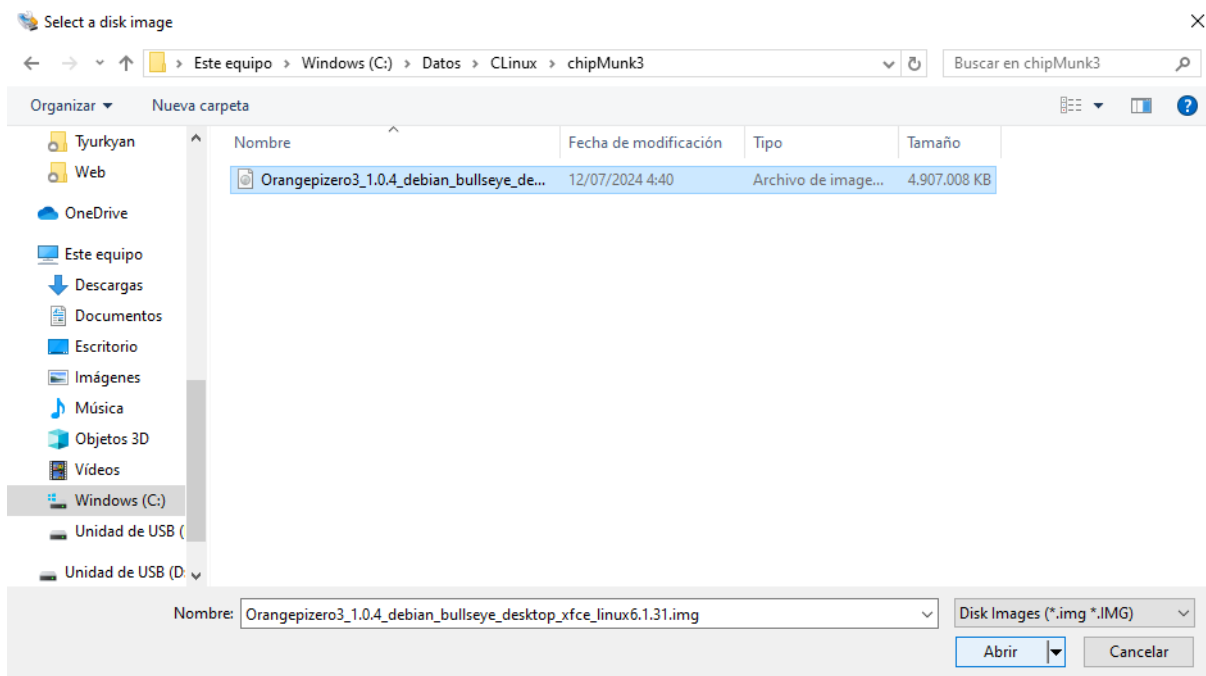
Lo primero es seleccionar la imagen descargada, para lo cual requerirá disponer al menos de unos 6GB de disco duro.



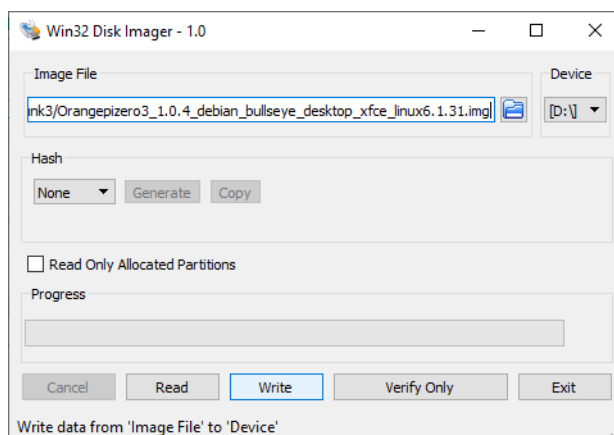
Debe tener presente que la imagen descargada está en un fichero comprimido 7z, este debe ser previamente descomprimido para obtener un fichero de imagen img. Este fichero de imagen es el que tiene que seleccionar desde la aplicación WIN32DiskImager.



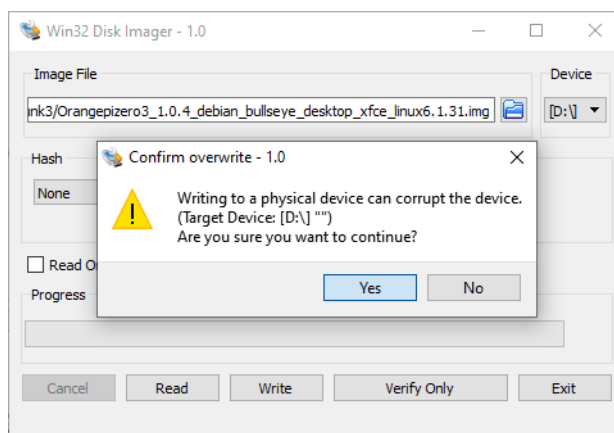
A continuación seleccione la imagen y presione Abrir.



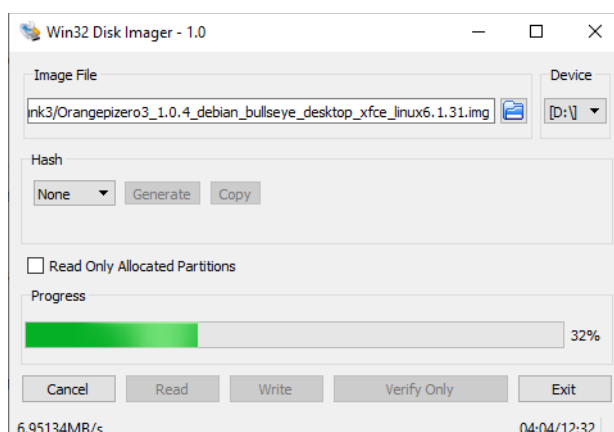
Y una vez la haya seleccionado debe darle a la opción write.



Le diré que puede corromperse el dispositivo. Hace referencia a que el contenido de la SD Card anterior se pierde. Si no tiene datos relevantes no se preocupe, indique yes. El programa lo que va a hacer es sobrescribir todos los sectores.



El proceso de grabado es muy lento, puede tardar en algunos equipos hasta 30 minutos. Para asegurarse que sea “rápido” seleccione un puerto y grabador USB3.



3.4 Arranque inicial

Es recomendable que en el primer arranque el equipo esté conectado por puerto serie, así podrá ver lo que está haciendo. Si eso no fuese posible puede conectarlo a una red con DHCP y averiguar que IP tiene con Advanced IP Scanner.



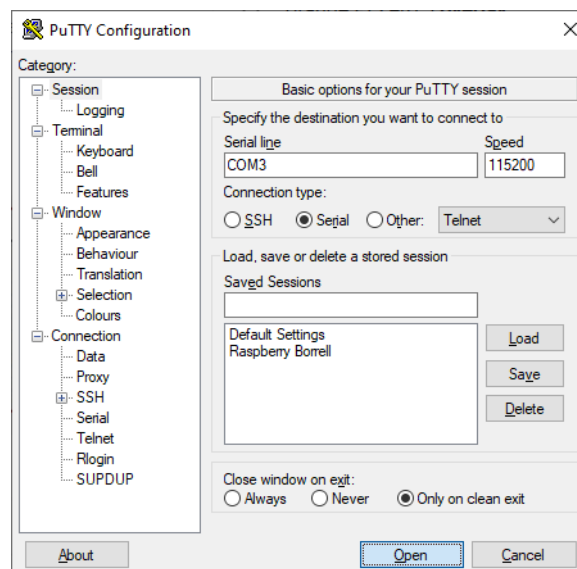
Si no dispone en la red de un servidor DHCP no se configurará el puerto Ethernet y no se podrá conectar al equipo. Normalmente si conecta un cable ethernet directamente desde el ordenador al PLC no tendrá servicio DHCP.

Insistimos, lo más sencillo es puerto serie a un terminal RS232 configurado a 115200 baudios. Como le comentamos anteriormente, esto lo puede hacer desde Windows con la



aplicación libre denominada putty. Esta puede descargarla de internet, y en el ordenador no es necesario instalarla para ejecutarla.

Si deberá buscar en el administrador de dispositivos de Windows el puerto serie que le ha asignado el sistema operativo. En nuestro caso utilizamos un conversor USB RS232c que está en el COM3.



Una vez abierto el puerto serie, podremos ver los procesos que se van ejecutando durante el arranque del sistema. Si el sistema tiene pantalla HDMI, una vez arranque el sistema gráfico se dejarán de enviar. Estos aparecerán por la pantalla.

```
COM3 - PuTTY
U-Boot SPL 2024.01-orangepi (Jul 11 2024 - 16:36:36 +0800)
DRAM base address is defined as 0x40000000
DRAM has 16 b/raw, 10 b/col, 4 B/width, 2 #rank and 8 #bank
DRAM top address must be less than 0x100000000
DRAM: 4096 MiB
Trying to boot from MMC1
NOTICE: BL31: v2.11.0(debug):4b9be5ab
NOTICE: BL31: Built : 12:05:33, Jul 9 2024
NOTICE: BL31: Detected Allwinner H616 SoC (1823)
NOTICE: BL31: Found U-Boot DTB at 0x4a0b2878, model: OrangePi Zero3
INFO: ARM GICv2 driver initialized
INFO: Configuring SPC Controller
INFO: PMIC: Probing AXP305 on RSB
ERROR: RSB: set run-time address: 0x10003
INFO: Could not init RSB: -65539
INFO: BL31: Platform setup done
INFO: BL31: Initializing runtime services
INFO: BL31: cortex_a53: CPU workaround for erratum 855873 was applied
INFO: BL31: cortex_a53: CPU workaround for erratum 1530924 was applied
INFO: PSCI: Suspend is unavailable
INFO: BL31: Preparing for EL3 exit to normal world
INFO: Entry point address = 0x4a000000
INFO: SPSR = 0x3c9
INFO: Changed devicetree.
```



En cualquier caso, una vez haya arrancado el sistema le aparecerá en el terminal algo similar

```
COM3 - PuTTY
Loading Device Tree to 0000000048edc000, end 0000000048f4efff ... OK
Working FDT set to 48edc000

Starting kernel ...

orangezero3 login: orangepi (automatic login)

Welcome to Orange Pi 1.0.4 Bullseye with Linux 6.1.31-sun50iw9

System load:  102%      Up time:    0 min
Memory usage: 5% of 3.84G  IP:
CPU temp:    45°C      Usage of /:  13% of 29G

[ General system configuration (beta): orangepi-config ]

Last login: Fri Jul 12 02:35:01 UTC 2024 on tty1
orangepi@orangezero3:~$
```

Y ahí podrá ejecutar los comandos Linux que desee. Uno interesante es `top`, que muestra el estado del equipo.

3.5 Primeras configuraciones

En el primer arranque, el sistema amplía automáticamente la partición de la SDCard para utilizar todo el espacio disponible. A continuación, solicita configurar la contraseña del usuario `root`, seleccionar el intérprete de comandos (por defecto `Bash`), definir la zona horaria (Europa/Madrid) y elegir el juego de caracteres (UTF-8/ES). Solo es necesario seguir las instrucciones en pantalla para completar este proceso.

En la consola puede ejecutar:

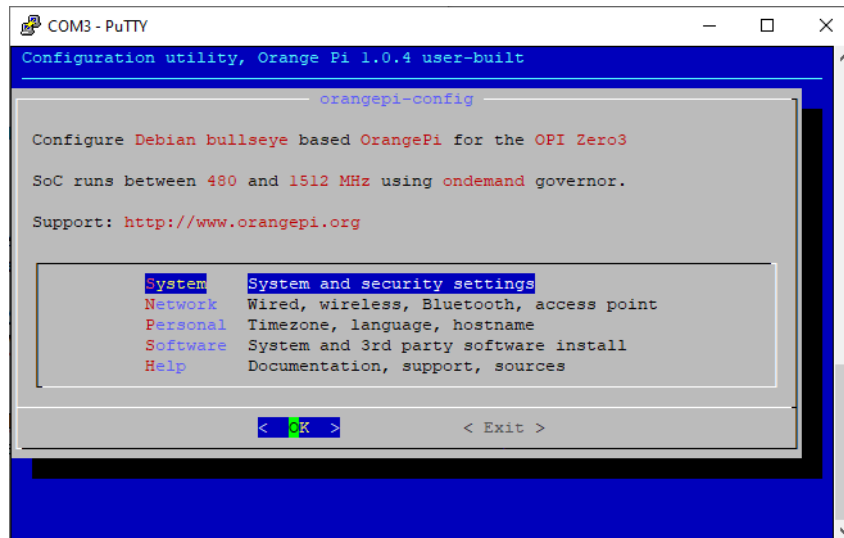
```
sudo orangepi-config
```

y nos pedirá la clave, inicialmente la clave es `orangepi`. Si el equipo no está conectado por Ethernet a un router o red con DHCP nos dirá que la configuración no funcionará correctamente. En cualquier caso hay que presionar una tecla.

Esa indicación hace referencia que algunas configuraciones que requieren descargar ficheros de internet no podrán ejecutarse, pero lo principal si que podrá ser.



A continuación un pantallazo de la aplicación orange-pi-config



Ahí podemos configurar IP fija o la red Wifi. Es bastante intuitivo.

Más adelante volveremos a ejecutar esta configuración para activar el bus I2C, puerto serie,

...

Como tendremos acceso, ahora toca comprobarlo.

```
Ping 8.8.8.8
```

```
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.  
64 bytes from 8.8.8.8: icmp_seq=1 ttl=116 time=27.6 ms  
64 bytes from 8.8.8.8: icmp_seq=2 ttl=116 time=28.1 ms  
64 bytes from 8.8.8.8: icmp_seq=3 ttl=116 time=27.1 ms
```

Si ha ido bien nos indicará el tiempo que se tarda en hacer una petición al servidor de dominios de google. Lo normal es que tarde por debajo de 25ms, pero dependerá de estado de la red. Si hay respuesta es que ha ido bien.

Ahora podemos actualizar los repositorios y actualizar todo el software instalado a su última versión.

```
apt-get update  
apt-get upgrade
```

Si en algún momento indica que para instalar requiere consumir espacio de almacenamiento le decimos que SI. El upgrade la primera vez suele tardar un tiempo.



Ahora

```
apt-get install git  
apt-get install build-essential
```

Esto es necesario para poder instalar módulos y drivers. En la versión bulleeye 6.1 ya los tiene instalados y no hace ningún efecto.

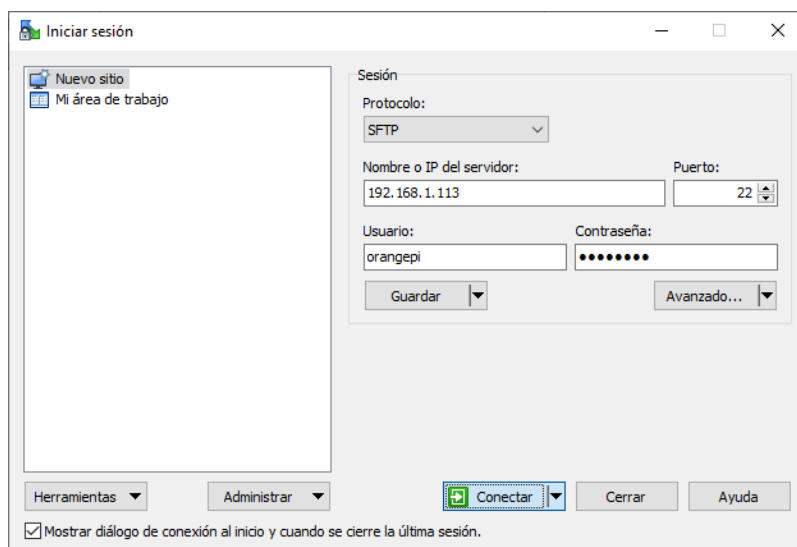
3.6 Directorios

Para la configuración que proponemos hay que crear dos directorios. Uno para almacenar la información de configuración y otro para guardar las aplicaciones. Para ello podemos usar el programa de windows WinSCP. Este nos permitirá navegar por dentro del directorio linux, crear carpetas, editar ficheros,

Al ejecutar la aplicación WinSCP nos pedirá que indiquemos la IP, el usuario y la contraseña. Como usuario podemos poner orangepi o root. La contraseña en ambos casos es root.



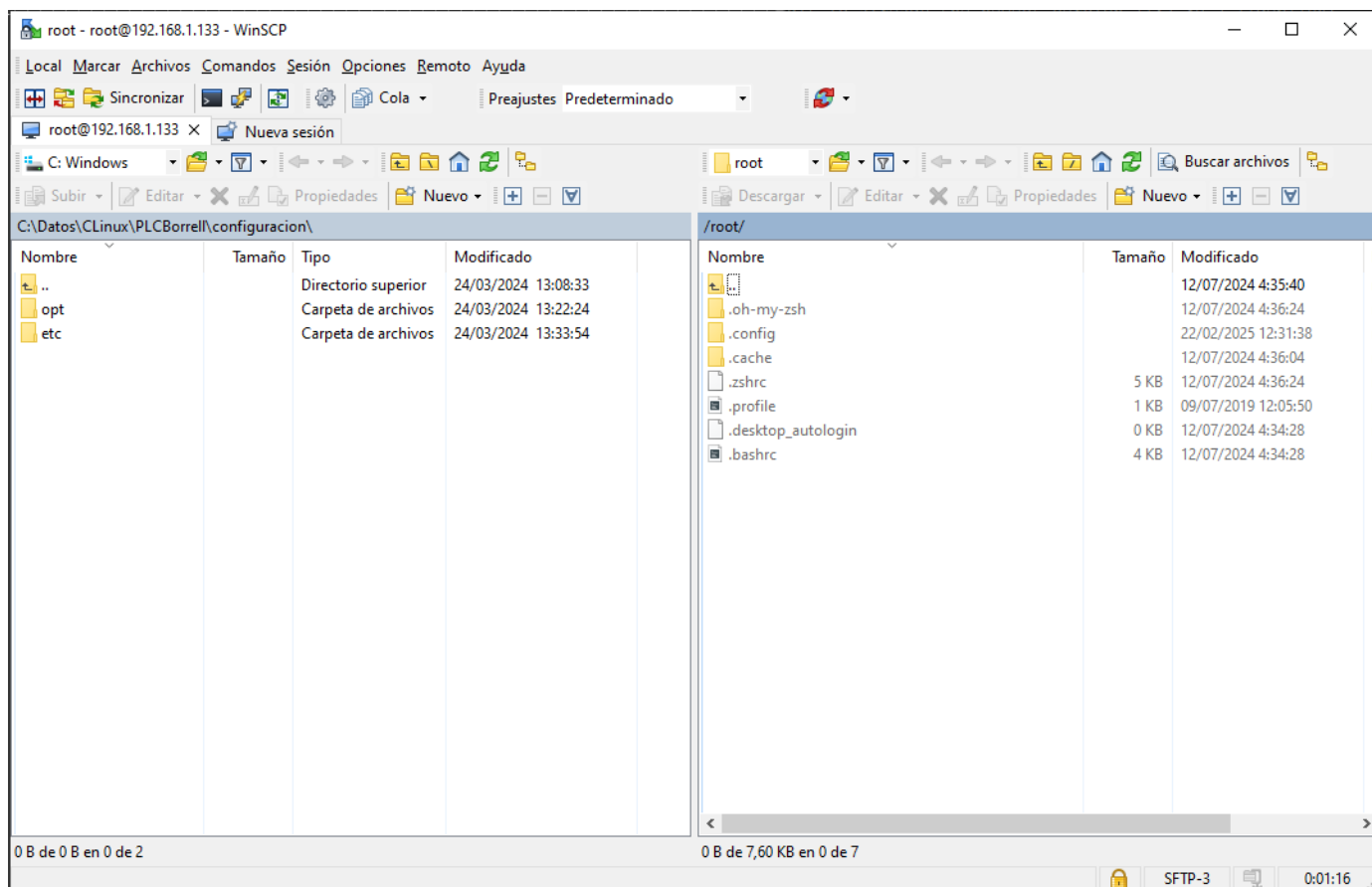
Como vamos a instalar ficheros en las carpetas del sistema, este acceso conviene hacerlo como root, aunque para ficheros de usuario mejor entrar como usuario. Por defecto está creado solo el usuario orangepi, aunque se pueden crear otros.



Una vez el programa haya conseguido conectar con el PLC, aparecerá una ventana en la que se puede ver los directorios del equipo remoto y equipo local. Esta ventana de forma intuitiva permite copiar ficheros de un equipo al otro.



Haciendo doble click sobre un fichero de texto se abrirá sobre el notepad o notepad++ donde lo podrá editar. También con el botón derecho accediendo a propiedades podrá indicar a quien pertenece el fichero así como si es ejecutable, solo lectura, ... Para mayor conocimiento consulte la documentación del programa.



Para empezar creamos los directorios:

```
/etc/ibercomp  
/opt/ibercomp
```

Dentro de opt copiamos

```
WiringOP-Zero
```

dentro de */opt/ibercomp* copiamos:

```
chipMunk
```

y dentro de */etc/ibercomp* copiamos los archivos de configuracion contenidos en la carpeta:



chipMunk

En principio las carpetas a pasar son las que podrá descargar de nuestra tienda, bajo ficherosAdicionales.zip. Debe descargarlas previamente.

3.7 Librerías WiringOP

Para que los programas en C, en especial ChipMunk, puedan acceder a los pines del micro es necesario instalar la librería WiringOP. Esta es una biblioteca de código abierto basada en WiringPi, diseñada para facilitar el acceso y control de los pines GPIO en las placas Orange Pi y otras placas compatibles con Allwinner.

Fue creada como una adaptación de WiringPi, la popular biblioteca para Raspberry Pi desarrollada por Gordon Henderson, pero ajustada para funcionar con los procesadores Allwinner H2+, H3, H5 y similares.

Al igual que su predecesora, WiringOP proporciona una interfaz sencilla en C y soporta el control de pines GPIO, SPI, I2C, UART y PWM, permitiendo que los desarrolladores interactúen con el hardware sin necesidad de trabajar directamente con registros de bajo nivel.

El uso de WiringOP permite escribir programas para control de hardware en C, C++, y otros lenguajes compatibles con librerías compartidas. Su compatibilidad con herramientas como Python mediante bindings y su facilidad de instalación la convierten en una opción ideal para desarrolladores que buscan un control optimizado del hardware en sus proyectos embebidos.

Para instalarla la forma más sencilla es:

```
cd /opt/wiringOP
chmod 777 ./build
./build
```

Una forma alternativa es situarse en /opt y ejecutar:

```
apt-get update
apt-get install -y git
git clone https://github.com/orangepi-xunlong/wiringOP.git

cd wiringOP
./build clean
```



```
./build
```

Compilará e instalará la librería. Podemos comprobarlo tecleando:

```
gpio readall
```

```
root@orangepi3:/opt/wiringOP# gpio readall
```

H616											
GPIO	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	GPIO	
		3.3V			1	2		5V			
229	0	SDA.3	OFF	0	3	4		5V			
228	1	SCL.3	OFF	0	5	6		GND			
73	2	PC9	OFF	0	7	8	0	OFF	TXD.5	3	226
		GND			9	10	0	OFF	RXD.5	4	227
70	5	PC6	ALT5	0	11	12	0	OFF	PC11	6	75
69	7	PC5	ALT5	0	13	14		GND			
72	8	PC8	OFF	0	15	16	0	OFF	PC15	9	79
		3.3V			17	18	0	OFF	PC14	10	78
231	11	MOSI.1	OFF	0	19	20		GND			
232	12	MISO.1	OFF	0	21	22	0	OFF	PC7	13	71
230	14	SCLK.1	OFF	0	23	24	0	OFF	CE.1	15	233
		GND			25	26	0	OFF	PC10	16	74
65	17	PC1	OFF	0	27	28					
272	18	PI16	ALT2	0	29	30					
262	19	PI6	OFF	0	31	32					
234	20	PH10	ALT3	0	33	34					
GPIO	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	GPIO	
H616											

3.8 Activar reloj en tiempo real DS3231

El DS3231 es un Reloj en Tiempo Real (RTC, por sus siglas en inglés) de alta precisión que utiliza una interfaz de comunicación I²C para proporcionar información precisa sobre la fecha y la hora. Diseñado para aplicaciones, sin conexión a internet, donde la estabilidad y exactitud del tiempo son esenciales, este módulo incorpora un oscilador de cuarzo integrado y compensado por temperatura (TCXO), lo que minimiza la variación en la frecuencia de oscilación debida a los cambios de temperatura, garantizando una precisión de ± 2 ppm en un amplio rango de temperaturas.

El DS3231 incorpora un condensador tampón, lo que le permite mantener su funcionamiento incluso en ausencia de alimentación externa durante aproximadamente tres meses. Esto es particularmente útil en sistemas que pueden experimentar cortes de energía, asegurando que el reloj siga funcionando y proporcionando la hora correcta cuando el sistema vuelva a encenderse.



Puede obtener más documentación acerca de este dispositivo en el siguiente enlace:

<https://orangepiweb.es/clock-ds3231.php>

El primer paso es con el orange-pi-config activar el bus I2C3. Aproveche también para activar para activar el ttyS5.

La librería i2c debería estar instalada:

```
sudo apt-get install i2c-tools
```

despues comprobamos que esté realmente activo con

```
sudo i2cdetect -l
```

```
root@orangepizero3:~# sudo i2cdetect -l
i2c-3  i2c          mv64xxx_i2c adapter          I2C adapter
i2c-4  i2c          mv64xxx_i2c adapter          I2C adapter
i2c-5  i2c          DesignWare HDMI          I2C adapter
```

Miramos a ver que hay conectado en el bus I2C

```
sudo i2cdetect -y 3
```

```
root@orangepizero3:~# sudo i2cdetect -y 3
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  68  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Si todo ha ido bien debería aparecer una tabla con un 68, indicando que esa es la dirección del integrado localizado en el bus i2c.

```
echo ds3231 0x68 > /sys/class/i2c-adapter/i2c-3/new_device
sudo i2cdetect -y 3
```



Ahora le decimos que hay conectado un chip ds3231 en la dirección 0x68. Volvemos a listar la tabla y en lugar de un 68 nos aparecerá UU. Eso indica que está “conectado”.

```
root@orangezero3:~# sudo i2cdetect -y 3
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- UU -- -- -- --
70: -- -- -- -- -- -- --
```

Ahora listamos los relojes en tiempo real que tiene el sistema:

```
dmesg | grep rtc
```

deberían aparecer dos, el de software y el de hardware. (rtc0 y rtc1).

```
root@orangezero3:~# dmesg | grep rtc
[  0.029027] platform 3001000.clock: Fixed dependency cycle(s) with
/soc/rtc@7000000
[  0.034699] platform 7010000.clock: Fixed dependency cycle(s) with
/soc/rtc@7000000
[  1.537549] sun6i-rtc 7000000.rtc: registered as rtc0
[  1.537573] sun6i-rtc 7000000.rtc: setting system clock to 1970-01-
02T00:00:07 UTC (86407)
[  1.537672] sun6i-rtc 7000000.rtc: RTC enabled
[ 207.194622] rtc-ds1307 3-0068: SET TIME!
[ 207.197293] rtc-ds1307 3-0068: registered as rtc1
```

Comprobamos con date que la hora sea correcta, y también la del reloj DS3231

```
sudo hwclock -r -f /dev/rtc1
```

si no coinciden podemos

```
sudo hwclock -w -f /dev/rtc1
```

En el fichero *etc/rc.local* hay que insertar

```
echo ds3231 0x68 > /sys/class/i2c-adapter/i2c-0/new_device
hwclock -s -f /dev/rtc1
```



```
exit 0
```

Esto se puede hacer con el WinSCP, haciendo doble click sobre el fichero. Este se abrirá en el notepad, aunque nosotros recomendamos que instale y utilice el Notepad++

Si hemos instalado un Linux 5 en lugar de un Linux 6 y deseamos que la fecha/hora se sincronice con un servidor NTP, podemos:

```
sudo apt install ntp
systemctl enable ntp
```

3.9 Java

Para instalar JAVA lo más sencillo es instalar la versión que propone debian, que a fecha de hoy es el Open Java 11. Todas las aplicaciones nuestras funcionan correctamente sobre este.

```
apt-get install default-jdk
```

Una vez finalizado se comprueba con:

```
java -version
```

Y generará algo como:

```
root@EquipoPruebas:~# java -version
openjdk version "11.0.22" 2024-01-16
OpenJDK Runtime Environment (build 11.0.22+7-post-Debian-1deb11u1)
OpenJDK 64-Bit Server VM (build 11.0.22+7-post-Debian-1deb11u1, mixed
mode)
```

3.10 Pantalla Virtual

Solo en los sistemas operativos que no tengan escritorio, si quiere ejecutar aplicaciones de escritorio desde WebSwing deberá instalar una pantalla virtual, X11, para ello habrá que instalar el terminal virtual xvfb, para ello:

```
apt-get install libxext6
apt-get install libxi6
apt-get install libxtst6
```



```
apt-get install libxext6  
apt-get install libxext6  
apt-get install libxrender1  
apt-get install xvfb
```

Lo más probable es que todas las librerías estén previamente instaladas en su última versión. Solo habrá que instalar esta aplicación.

La forma de crear una pantalla virtual será:

```
Xvfb :1 -screen 0 1024x768x24 </dev/null &  
export DISPLAY=":1"
```

Solo habrá que hacerlo manualmente una vez para la prueba.

3.11 Node-red

Node-RED es una herramienta de programación visual basada en flujos, ideal para proyectos IoT y aplicaciones prácticas.

Pasos de Instalación

Instalar Dependencias Básicas

Se requiere contar con algunas herramientas para compilar y gestionar el código:

```
sudo apt install build-essential git curl
```

Este comando instala:

- build-essential: Herramientas de compilación esenciales.
- git: Para clonar repositorios.
- curl: Para transferir datos y ejecutar scripts remotos.

Ejecutar el Script Oficial de Instalación

Con el siguiente comando, se descarga y ejecuta el script oficial que se encarga de actualizar Node.js y de instalar Node-RED:



```
bash <(curl -sL  
https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)
```

Este script realiza lo siguiente:

- Verifica la versión actual de Node.js y, de ser necesario, la actualiza.
- Instala o actualiza Node-RED y sus dependencias.
- Configura Node-RED como un servicio para facilitar su ejecución.

Verificación y Ejecución

- Iniciar Node-RED

Una vez finalizada la instalación, puedes iniciar Node-RED con:

```
node-red-start
```

- Alternativamente, si el script configuró el servicio, puedes usar:

```
sudo systemctl start nodered.service
```

- Habilitar el Inicio Automático (Opcional)

Para que Node-RED se inicie automáticamente al arrancar el sistema, ejecuta:

```
sudo systemctl enable nodered.service
```

- Acceder a la Interfaz Web

Por defecto, Node-RED se ejecuta en el puerto 1880. Abre un navegador y accede a:

```
http://<dirección_IP_del_OrangePi>:1880
```

3.12 RustDesk

Es una herramienta para la administración remota de ordenadores. Es una aplicación libre y bastante intuitiva.



Para poder instalar el RustDesk es “recomendable” disponer de un monitor conectado a HDMI y un teclado/ratón. Si no quiere usar teclado y ratón local, deberá instalar alguna aplicación tipo VNC, algo que no le recomendamos.

Lo primero es descargar el fichero de instalación DEB. Nosotros hemos realizado las pruebas con la versión 1.3.7 para ARM64 sobre Ubuntu, la cual ha funcionado correctamente en Debian.

La URL de descarga se obtiene de la [página de releases de RustDesk](#).

Se puede descargar también desde la consola con comando para descargar el paquete:

```
wget  
https://github.com/rustdesk/rustdesk/releases/download/1.3.7/rustdesk-1.3.7-aarch64.deb
```

Para instalarlo debe teclear en la consola abierta en el propio escritorio:

```
sudo apt-get install -f  
sudo dpkg -i rustdesk-1.3.7-aarch64.deb
```

También puede hacer esta última parte abriendo el fichero .deb desde el administrador de archivos.

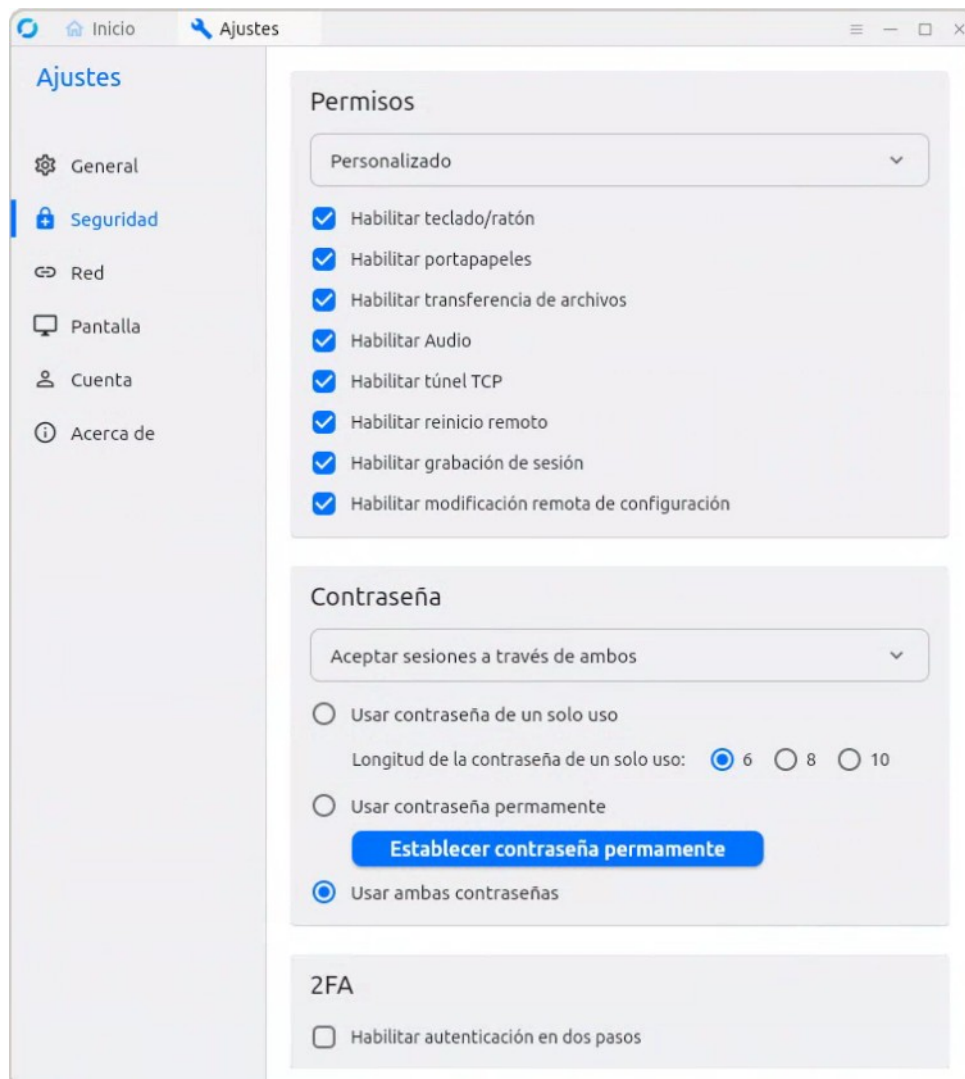
Con la instalación completada, puedes iniciar RustDesk desde la terminal o a través del menú de aplicaciones de tu entorno gráfico.



Lo primero que debe hacer cuando arranque el Rust Desk es ponerle una clave de arranque fija, activar el acceso directo por IP y permitir configurar el RustDesk remotamente. Si no lo hace puede encontrarse con la necesidad de empezar todo el proceso de instalación de nuevo.



En la siguiente pantalla le mostramos que opciones debe seleccionar para que funcione.



3.13 Dummy Video Driver

En muchas instalaciones no interesa disponer de un monitor, pero si poder acceder para realizar mantenimiento, cambios de aplicaciones, ...

El RustDesk es una buena opción, pues permite acceder por un número identificador desde cualquier parte del mundo si el equipo está conectado a internet. Si no dispone de monitor, podrá acceder para transferir ficheros desde su ordenador al PLC y viceversa, también podrá acceder por SSH o a cualquier servidor que tenga como MySQL o Nodered.



Pero si quiere acceder al escritorio, puede instalar un dummy video driver, que desconectará la salida HDMI y la redirigirá a un driver que no existe. Como alternativa puede mediante un adaptador conectar un simulador de monitor hardware.

El dummy video driver permite a Xorg generar una salida gráfica virtual, configurada mediante un archivo de configuración. Así, se habilita un entorno gráfico sin contar con un monitor real.

Para instalar el paquete del Dummy Video Driver deberá desde una consola ssh o serie:

```
sudo apt update
```

Y ahora instale el paquete xserver-xorg-video-dummy:

```
sudo apt install xserver-xorg-video-dummy
```

Este paquete contiene el controlador que permite a Xorg generar una salida gráfica sin monitor físico.

Configuración de Xorg para Usar el Dummy Driver

Para que Xorg utilice el dummy driver, debes crear un archivo de configuración. Se recomienda crear el archivo en el directorio `/etc/X11/xorg.conf.d/`.

Si el directorio no existiera debe crearlo. En el Debian Bulleye está definido:

```
sudo mkdir -p /etc/X11/xorg.conf.d
```

Ahora debe crear el fichero de configuración, puede hacerlo con WinSCP o con nano.

```
sudo nano /etc/X11/xorg.conf.d/10-dummy.conf
```

Dentro del archivo debe poner:

```
Section "Device"
    Identifier "DummyDevice"
    Driver "dummy"
    VideoRam 256000
EndSection
```



```
Section "Monitor"
    Identifier "DummyMonitor"
    HorizSync 28.0-80.0
    VertRefresh 48.0-75.0
EndSection

Section "Screen"
    Identifier "DummyScreen"
    Device "DummyDevice"
    Monitor "DummyMonitor"
    DefaultDepth 24
    SubSection "Display"
        Depth 24
        Modes "1024x768" "1280x1024" "1920x1080"
    EndSubSection
EndSection

Section "ServerLayout"
    Identifier "DummyLayout"
    Screen "DummyScreen"
EndSection
```

Explicación:

- Section "Device": Se define el dispositivo gráfico utilizando el driver dummy.
- Section "Monitor": Se configura un monitor virtual con rangos de sincronización.
- Section "Screen": Se asocia el dispositivo y el monitor. En la sub-sección "Display" se indican varias resoluciones (la primera será la predeterminada).
- Section "ServerLayout": Se establece la disposición general del servidor X.

Para aplicar los cambios debe guardar y cerrar el archivo. Si uso nano debe presionar Ctrl+O para guardar y Ctrl+X para salir. A continuación deberá reiniciar el servidor X o el sistema. La forma más sencilla de aplicar los cambios es reiniciar el equipo:

```
sudo reboot
```

Esto hará que Xorg lea la nueva configuración al iniciar.



Se se definió una resolución excesivamente alta puede ser necesario aumentar el valor de VideoRam en la sección "Device". Como la intención de disponer un dummy video es acceder por RustDesk remotamente para mantenimiento lo recomendable es utilizar la resolución más baja posible.

Si el entorno gráfico no se inicia correctamente, revise /var/log/Xorg.0.log para identificar los posibles errores en la configuración.

3.14 Servicio ChipMunk

La aplicación ChipMunk puede ejecutarse como servicio, de forma que siempre tendrá acceso al puerto RS485 por Ethernet, así como localizar el equipo en la red con IbercompDiscovery.

Para ejecutarlo como servicio el primer paso será copiar los ficheros:

```
chipMunk.service
```

en la carpeta:

```
/etc/systemd/system/
```

Estos ficheros deben de tener propiedades de ejecución (chmod 777). Una vez realizado, debemos activarlos, para ello:

```
systemctl enable chipMunk
```

y lo ejecutamos con:

```
systemctl start chopMunk
```

y lo comprobamos:

```
systemctl status chipMunk
```

El servicio chipMunk arrancará directamente la aplicación:

```
/opt/ibercomp/chipMunk/chipMunk
```

que deberá tener los permisos de ejecución activados.



El puerto ttyS5 deberá haber sido habilitados con el orange-config. Pueden definirse puertos adicionales, que puede sacar directamente de los pines del Orange PI Zero. La definición se hace en el fichero:

/etc/ibercomp/chipMunk/chipMunk.cfg

El fichero tiene un aspecto:

```
//
//
// Archivo de configuración para el demonio ETH <-> puertos RS485
//
// Se pueden asociar a puertos ethernet hasta 16 puertos serie, para ello hay definir
// parametros
// La primera linea forzosamente debe ser:
// serial=/dev/ttySn   En chipMunk/orangePIZero solo están conectados a RS485 n=1 y n=2
// baud=19200          Velocidad en baudios del puerto serie,
//                     Solo admite 9600, 19200, 38400, 57600, 115200, ...
//                     no admite 28800 baudios.
// ip=0.0.0.0          IP del ordenador que se puede conectar al puerto,
//                     0.0.0.0 = desde cualquier puerto, 127.0.0.0 = solo desde el
//                     propio equipo
// port=10001          Puerto del ordenador al que se ha de asociar el puerto serie
//                     indicado anteriormente
// LED=1              LED al que se ha de asociar el puerto. Puede ser 1, 2, 3, 4 o NONE

//
//
// Configuramos el primer puerto
serial=/dev/ttyS5
baud=9600
ip=0.0.0.0
port=10001
LED=1

//
//
// Aquí pueden continuar otros puertos serie
// aunque esta versión de equipo tan solo tiene un puerto serie
```

3.15 Configurando red

Además del DNS indicado con el programa de configuración armbian-config podemos añadir otros DNS en el fichero

/etc/resolv.conf

Dentro del fichero



```
# Generated by NetworkManager
nameserver 212.166.211.4
nameserver 212.166.132.96
nameserver 8.8.8.8
```

Dentro del fichero

```
/etc/network/interfaces
```

podemos definir las número MAC que deseamos que tenga el equipo. Esta debe ser única para cada unidad, salvo que estemos 100% seguros que no coexistirá otro equipo con la misma MAC.

```
auto eth0
iface eth0 inet static
hwaddress ether 00:50:c2:c1:90:55
address 192.168.100.101
netmask 255.255.255.0
network 192.168.100.0
broadcast 192.168.100.255
gateway 192.168.100.128
```

Esta parte puede variar de una versión Linux a otra y conviene consultar la documentación de la versión en concreto.

4 COMANDOS DE INTERES

Estos comandos básicos permiten administrar y diagnosticar el sistema para aquellos que no están familiarizados con Linux.

4.1 Control del Sistema

- reboot

Reinicia el equipo, deteniendo todos los procesos y volviendo a arrancar.

- shutdown now



Apaga el equipo inmediatamente. Para encenderlo de nuevo, es necesario desconectarlo y volver a alimentarlo.

- `shutdown -h now`

Apaga el equipo de forma segura y lo deja en estado de halt (detenido).

- `shutdown -r now`

Apaga y reinicia automáticamente el sistema.

4.2 Información del Sistema

- `uname -a`
Muestra información sobre el núcleo (kernel) de Linux, arquitectura y versión del sistema operativo.
- `cat /etc/os-release`
Muestra detalles sobre la distribución Linux instalada (por ejemplo, Debian o Ubuntu).
- `top`
Muestra en tiempo real el uso de CPU, memoria y procesos activos.
- `df -h`
Muestra el espacio en disco disponible en cada partición.
- `free -h`
Indica la memoria RAM libre y en uso.

4.3 Gestión de Archivos y Directorios

- `ls -l`

Lista archivos en el directorio actual con detalles como permisos y tamaño.

- `cd /ruta/deseado`

Cambia al directorio especificado.

- `cp archivo1 archivo2`

Copia archivo1 a archivo2.



- `mv archivo1 archivo2`

Mueve o renombra archivo1 como archivo2.

- `rm archivo`

Borra un archivo (⚠ ¡Irrecuperable!).

4.4 Red y Conectividad

- `ip a`

Muestra las direcciones IP asignadas a las interfaces de red.

- `Ping 8.8.8.8`

Comprueba si hay conexión a Internet enviando paquetes a Google DNS.

- `netstat -tulnp`

Muestra los puertos abiertos y qué servicios están usándolos.

- `curl ifconfig.me`

Muestra la IP pública del equipo.

4.5 Manejo del Puerto Serie y Diagnóstico

- `ls /dev/ttyS*`

Lista los puertos serie disponibles en el sistema.

- `dmesg | grep tty`

Muestra mensajes del sistema relacionados con puertos serie.

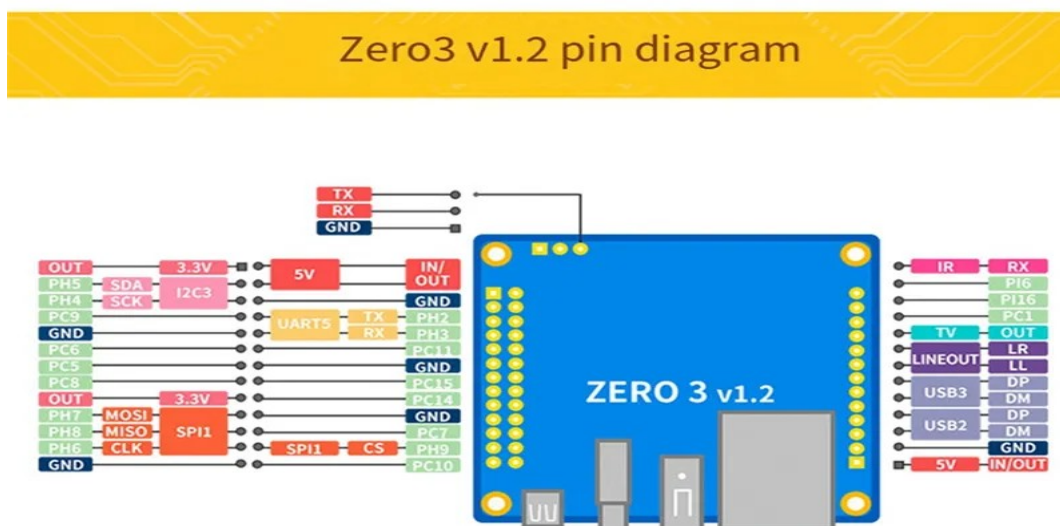
- `screen /dev/ttyS0 115200`

Abre una sesión de terminal en el puerto serie (reemplazar ttyS0 según corresponda).



5 CONEXIONADO

Si desea explorar otros sistemas operativos o bien realizar modificaciones a la electrónica necesitará conocer donde está conectada internamente cada cosa. Genérico el conexionado de una Orange PI Zero 3 vista desde arriba es:



PIN	FUNCIÓN
PC1	LED COM
PI16	LED W
PI6	LED E
IR-RX	SEL RS485 (selecciona modo RS485)
PH5	I2C SDA
PH4	I2C SCL
PH2	TX RS485 (TTY5)
PH3	RX RS485 (TTY5)
PC14	DE RS485 (TTY5)
PC10	LED CPU

Los puertos USB2 y USB3 están disponibles en los conectores de la parte superior.



Si El PIN SEL RS485 está a uno, el DE RS485 controlará la transmisión del transceptor RS485, y será la aplicación o sistema Linux el encargado de gestionar la dirección de los datos RS485. Si la señal se pone a cero, el control de flujo se hace automáticamente mediante un procesador externo tipo PIC10.

El resto de las señales que no son utilizadas puede utilizarlas para cualquier propósito.



Pero no debe olvidar que son señales al procesador, que cualquier tensión por encima de 3V3 o por debajo de 0V dañará irreversiblemente el equipo. Si eso sucediera le sugerimos que adquiera un nuevo Orange PI Zero 3 en cualquier tienda online.

5 CONDICIONES DE GARANTIA

Nosotros hemos puesto el máximo interés y esmero en el diseño de este PLC y está concebidas para funcionar todos los días 24 horas sin que se manifiesten problemas.

Ibercomp SA se compromete a dar garantía contra todo defecto de fabricación de 3 años desde la fecha de factura en este equipo.

La tarjeta procesadora Orange PI Zero 3, memoria SD CARD, sistema operativo y aplicaciones incluidas no son elementos fabricados por nosotros, con lo que dispondrán únicamente la garantía indicada por ley.

Debe entenderse que es un equipo muy abierto, que permite experimentar e innovar en desarrollos, esa es su finalidad, pero ha de tener presente que I+D el riesgo de rotura es alto y que no asumimos responsabilidad alguna de equipo manipulado o modificado.

Queda exenta de garantía cualquier rotura que se haya realizado físicamente, o sea, resultado directo de una mala instalación, alimentación a una tensión incorrecta o sobretensiones inducidas a través de la red eléctrica o líneas de datos.

La garantía incluye todos los elementos que hayan de sustituirse y la mano de obra necesaria para realizar dicha reparación. Si el equipo no tuviese arreglo bajo garantía se sustituirá por otro nuevo idéntico o modelo superior sustitutivo.

La garantía no incluye los gastos de transportista hacia nuestros talleres.

Ibercomp SA, como empresa fabricante del equipo, no se hace responsable de:

- 1) Cualquier daño o perjuicio que pueda ocasionar el equipo como consecuencia de un fallo de su funcionamiento o pérdida de datos.



- 2) Los fallos o defectos que pudieran hallarse en el software o ejemplos asociados al mismo. Estos están en continuo desarrollo y podrán actualizarse de forma gratuita siempre que haya una nueva versión disponible.

Ibercomp S.A. garantiza la aceptación de la devolución del equipo hasta 30 días después de la fecha de adquisición con la condición de que el equipo y accesorios así como su embalaje se encuentren en su estado original.

Si la adquisición fue realizada a través de tienda o distribuidor, la devolución del paquete deberá realizarse a través de la misma.