Linux

- Opción alternativa - Instalación del parche PREEMPT RT usando Xanmod para Ubuntu 24.04 (x64)
- Instalación de ROS 2 y librerias
- Instalación de Ardupilot para ROS 2
- Compilar el parche de tiempo real para Ubuntu

Opción alternativa - Instalación del parche PREEMPT RT usando Xanmod para Ubuntu 24.04 (x64)

Instalación del parche PREEMPT RT usando Xanmod para Ubuntu 24.04 (x64)

El siguiente script de instalación se basa en el siguiente enlace:
https://linuxcapable.com/install-xanmod-kernel-on-ubuntu-linux/

Esta primera parte prepara el sistema con los paquetes necesarios, el repositorio y comprueba que versión es compatible con el kernel.

```
sudo apt update
sudo apt upgrade -y
sudo apt install dirmngr lsb-release ca-certificates software-properties-common apt-transport-https dkms curl -y
curl -fSsL https://dl.xanmod.org/gpg.key | sudo gpg --dearmor | sudo tee /usr/share/keyrings/xanmod.gpg > /dev/null
echo 'deb [signed-by=/usr/share/keyrings/xanmod.gpg] http://deb.xanmod.org releases main' | sudo tee /etc/apt/sources.list.d/xanmod-kernel.list
sudo apt update
wget https://dl.xanmod.org/check_x86-64_psabi.sh
chmod +x check_x86-64_psabi.sh
./check_x86-64_psabi.sh
```

En resultado de este ultimo comando será ***CPU supports x86-64-v2***, ***CPU supports x86-64-v3*** o **CPU supports x86-64-v4**. En función de la salida tendremos que adaptar la versión.

```
sudo apt install linux-xanmod-rt-x64v3
```

Para completar la instalación reiniciar el equipo.

OPCIONAL: ROS 2

En mi caso quiero una caracteristicas determinadas del parche de tiempo real, que son las siguientes:

```
# Enable CONFIG_PREEMPT_RT
 -> General Setup
  -> Preemption Model (Fully Preemptible Kernel (Real-Time))
   (X) Fully Preemptible Kernel (Real-Time)

# Enable CONFIG_HIGH_RES_TIMERS
 -> General setup
  -> Timers subsystem
   [*] High Resolution Timer Support

# Enable CONFIG_NO_HZ_FULL
 -> General setup
  -> Timers subsystem
   -> Timer tick handling (Full dynticks system (tickless))
    (X) Full dynticks system (tickless)

# Set CONFIG_HZ_1000 (note: this is no longer in the General Setup menu, go back twice)
 -> Processor type and features
  -> Timer frequency (1000 HZ)
   (X) 1000 HZ

# Set CPU_FREQ_DEFAULT_GOV_PERFORMANCE [=y]
 ->  Power management and ACPI options
  -> CPU Frequency scaling
   -> CPU Frequency scaling (CPU_FREQ [=y])
    -> Default CPUFreq governor (<choice> [=y])
     (X) performance
```

Lo que ocurre es lo siguiente al comprobar si tenemos estas caracteristicas

```
cat /boot/config-$(uname -r) | grep -E
"CONFIG_PREEMPT_RT|CONFIG_HIGH_RES_TIMERS|CONFIG_NO_HZ_FULL|CONFIG_HZ_|CONFIG_CPU_FREQ_DEFAU
LT_GOV_"
```

Podemos observar que no tenemos todas caracteristicas deseadas, 3 de las 5.

```
# CONFIG_HZ_PERIODIC is not set
# CONFIG_NO_HZ_FULL is not set
CONFIG_HIGH_RES_TIMERS=y
CONFIG_PREEMPT_RT=y
# CONFIG_HZ_100 is not set
CONFIG_HZ_250=y
# CONFIG_HZ_300 is not set
# CONFIG_HZ_500 is not set
# CONFIG_HZ_1000 is not set
CONFIG_CPU_FREQ_DEFAULT_GOV_PERFORMANCE=y
# CONFIG_CPU_FREQ_DEFAULT_GOV_POWERSAVE is not set
# CONFIG_CPU_FREQ_DEFAULT_GOV_USERSPACE is not set
# CONFIG_CPU_FREQ_DEFAULT_GOV_SCHEDUTIL is not set
```

Entonces, vamos a proceder a descargar de nuevo el kernel y esta vez compilarlo y instalarlo. Porque lo hago de esta forma, por asegurarme que el kernel era compatible y funcionaba.

RECOMENDADO UNA RUTA COMO

~/Documentos/kernel o ~/kernel

Comprobar que sea la misma versión que el kernel actual, en este caso, he empleado la versión actual a dia de hoy. No borrar la instalación.

```
wget https://gitlab.com/xanmod/linux/-/archive/6.12.31-rt-xanmod1/linux-6.12.31-rt-xanmod1.tar.gz
tar -xvzf linux-6.12.31-rt-xanmod1.tar.gz

cp /boot/config-$(uname -r) .config
yes "" | make oldconfig

scripts/config --enable CONFIG_NO_HZ_FULL
scripts/config --enable CONFIG_HZ_1000
scripts/config --set-val CONFIG_HZ 1000

make olddefconfig

make -j$(nproc)
sudo make headers_install
sudo make modules_install
sudo make install
```

```
sudo update-grub
sudo reboot
```

Con esta serie de comandos realizamos la compilación y la instalación.

```
cat /boot/config-$(uname -r) | grep -E
"CONFIG_PREEMPT_RT|CONFIG_HIGH_RES_TIMERS|CONFIG_NO_HZ_FULL|CONFIG_HZ_|CONFIG_CPU_FREQ_DEFAU
LT_GOV_"
```

Luego queda comprobar que efectivamente las caracterisiticas son las deseadas.

```
# CONFIG_HZ_PERIODIC is not set
CONFIG_NO_HZ_FULL=y
CONFIG_HIGH_RES_TIMERS=y
CONFIG_PREEMPT_RT=y
# CONFIG_HZ_100 is not set
# CONFIG_HZ_250 is not set
# CONFIG_HZ_300 is not set
# CONFIG_HZ_500 is not set
CONFIG_HZ_1000=y
CONFIG_CPU_FREQ_DEFAULT_GOV_PERFORMANCE=y
# CONFIG_CPU_FREQ_DEFAULT_GOV_POWERSAVE is not set
# CONFIG_CPU_FREQ_DEFAULT_GOV_USERSPACE is not set
# CONFIG_CPU_FREQ_DEFAULT_GOV_SCHEDUTIL is not set
```

OPCIONAL: Drivers AMD

Radeon™ Software for Linux® version 25.10.1 for Ubuntu 24.04.2 HWE Version 25.10.1

```
wget https://repo.radeon.com/amdgpu-install/6.4.1/ubuntu/noble/amdgpu-install_6.4.60401-1_all.deb
sudo dpkg -i amdgpu-install_6.4.60401-1_all.deb
```

OPCIONAL: Drivers NVIDIA

Linux x64 (AMD64/EM64T) Display Driver 570.153.02 | Linux 64-bit

MANUAL

```
wget https://es.download.nvidia.com/XFree86/Linux-x86_64/570.153.02/NVIDIA-Linux-x86_64-570.153.02.run
chmod +x NVIDIA-Linux-x86_64-570.153.02.run
sudo IGNORE_PREEMPT_RT_PRESENCE=1 bash ./NVIDIA-Linux-x86_64-570.153.02.run
sudo reboot
```

OPCIONAL: Problemas con las pantallas

Eliminar archivos de configuración de Xorg: Si existe un archivo `/etc/X11/xorg.conf`, puede estar causando conflictos. Puedes eliminarlo o moverlo temporalmente:

```
sudo mv /etc/X11/xorg.conf /etc/X11/xorg.conf.backup
sudo apt update
sudo apt upgrade
sudo systemctl restart display-manager
sudo prime-select on-demand
```

Luego, reinicia el sistema con la pantalla externa conectada. Esto permitirá que el sistema detecte y configure automáticamente las pantallas durante el arranque.

Enlaces de referencia:

- https://linuxcapable.com/install-xanmod-kernel-on-ubuntu-linux/
- https://docs.ros.org/en/jazzy/Tutorials/Miscellaneous/Building-Realtime-rt_preempt-kernel-for-ROS-2.html

Instalación de ROS 2 y librerias

Instalación principal de ROS 2 Jazzy para Ubuntu 24.04

https://docs.ros.org/en/jazzy/index.html

Instalación de ROS 2 Control

https://control.ros.org/jazzy/doc/getting_started/getting_started.html

https://control.ros.org/jazzy/doc/ros2_control/controller_manager/doc/userdoc.html

Recomendable testear las demos

```
sudo apt-get install ros-$ROS_DISTRO-xacro
sudo apt-get install ros-$ROS_DISTRO-rqt*
sudo apt-get install ros-$ROS_DISTRO-joint*

mkdir -p ~/ws_dev/src
cd ~/ws_dev/src
git clone https://github.com/ros-controls/ros2_control_cmake.git
git clone https://github.com/ros-controls/ros2_control_demos.git
cd ..
colcon build
```

Importante leer:

https://control.ros.org/jazzy/doc/ros2_control/hardware_interface/doc/hardware_components_userdoc.html

https://design.ros2.org/articles/node_lifecycle.html

Instalación de Ardupilot para ROS 2

https://ardupilot.org/dev/docs/ros2.html

Compilar el parche de tiempo real para Ubuntu

https://github.com/2b-t/linux-realtime