

# ECEN 427: Lab 2

**BYU** Electrical & Computer  
Engineering  
IRA A. FULTON COLLEGE OF ENGINEERING

## What's a driver?

In Lab2, you must create drivers for:

1. Buttons
2. Switches (same as buttons)
3. Interrupt Controller

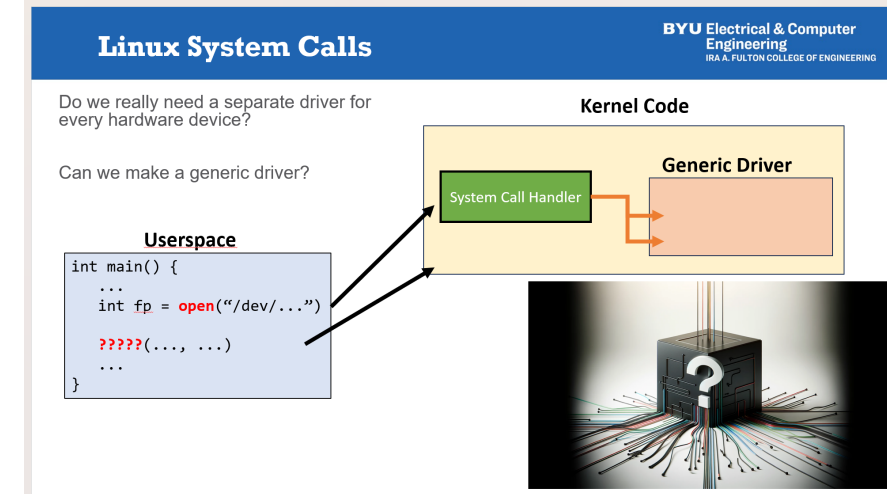
[https://github.com/byu-cpe/ecen427\\_student/tree/main/userspace/drivers](https://github.com/byu-cpe/ecen427_student/tree/main/userspace/drivers)

## What is the User space I/O (UIO) driver in Linux?

- Provides generic user space access to a hardware device
- Allows drivers to be written
- Knows nothing about the hardware device, except its physical base address and interrupt line

## What API (functions) does it provide to user space?

- Read/Write device addresses (via mmap)
- Enable interrupt line on CPU (via write)
- Wait/Check for interrupt activation (via read/poll)



- <https://byu-cpe.github.io/ecen427/documentation/hardware/>

1. How do you read/write to the GPIO controller registers?
2. How do you use the GPIO controller registers to read the current value of the buttons?

1. How do you enable the GPIO controller to generate an interrupt when a button is pressed?
2. For how long will the GPIO controller generate an interrupt?
3. Is an interrupt generated on press? On release? Or both?

# Interrupt Controller

1. What is an interrupt controller?
2. How many interrupt lines does the interrupt controller watch?
3. Suppose you want the interrupt controller to generate an interrupt when an interrupt is detected on input 1. What do you need to do to set this up?

# Interrupt Controller

1. For how long will the interrupt controller generate an interrupt?
2. What do you need to do to stop the interrupt controller from sending an interrupt?
3. If a button is pressed in our system (and assuming interrupts are enabled), what do you need to do to handle it?