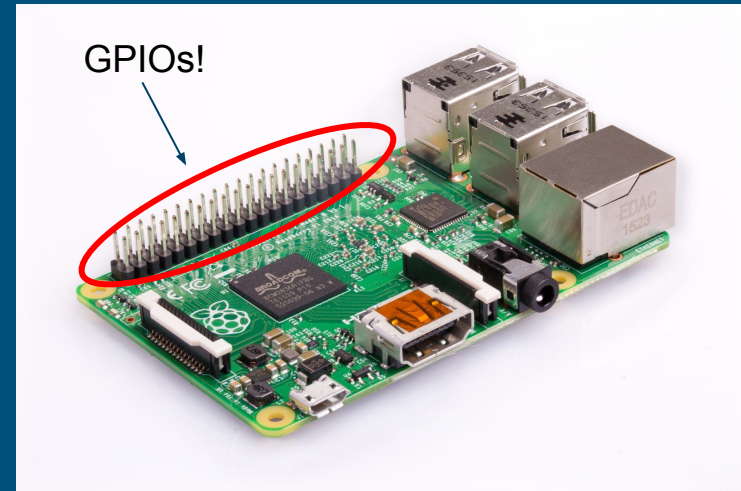# Raspberry PI GPIO Kernel Programming

William (BJ) Blair
CS 528 - Linux Kernel Development

Raspberry PI - Introduction

- What is it?
  - A small, cheap, low power usage computer
  - Originally designed to teach kids about computers and programming, also popular for hobbyist projects
  - Specs (Model B+, v1.2) [1]:
    - ARM 4x Cortex-A53 900 MHz CPU
    - 1 GB Ram (shared with the GPU)

GPIOs!

Raspberry PI - Introduction

- GPIOs (General Purpose Input-Output)
  - Programmable signal input and output (3.3V or 5V)
    - Controlled using a set of registers which you can directly access (DMA) (physically mapped addresses)
    - Numerous libraries available for interfacing (preferred!)
  - Can also be configured as an interrupt source

```c
// Register name:
// GPIO Function Select 0
#define GPFSEL0 0x7E200000

// set the first three bits
// as 001
*((int*)GPFSEL0) |= 0b001;
*((int*)GPFSEL0) &= ~0b110;
```

Example code to set GPIO 0 as an output pin [2]

Linux Kernel GPIOs

- The kernel since version 2.6.21 includes a generic GPIO API (Kernelspace only, NOT userspace) [3]
- Example uses (besides Raspberry PI) include [4]:
  - Monitoring MMC/SD card insertion/removal
  - Detecting MMC/SD card write protect status
  - Transceiver configuration
  - Switch sensing

```
#include <linux/gpio.h>

gpio_request();                          // reserve a GPIO
gpio_free();                             // make it available again
gpio_direction_<input|output>();         // set a GPIO pin as input or output
gpio_<get|set>_value();                  // read or set the value of a specific pin
gpio_to_irq();                           // gets a new interrupt number and maps it to
                                         //   a GPIO pin
```

Project

- Implemented a set of system calls to make the GPIO API available in userspace
  - More simple and direct interface than reading/writing to sysfs
    - E.g.
      - BJ_SetOutput(42);
        - Vs.
      - fd = open("/sys/class/gpio/gpio42", O_RDRW);
      - write(fd, "out")

- Implemented a device driver to display binary numbers on a set of LEDs
  - E.g. "echo 7 > /dev/bjrpi" will set the LEDs as 00111

Performance

How long each method took to write 0-31 to the GPIOs N times
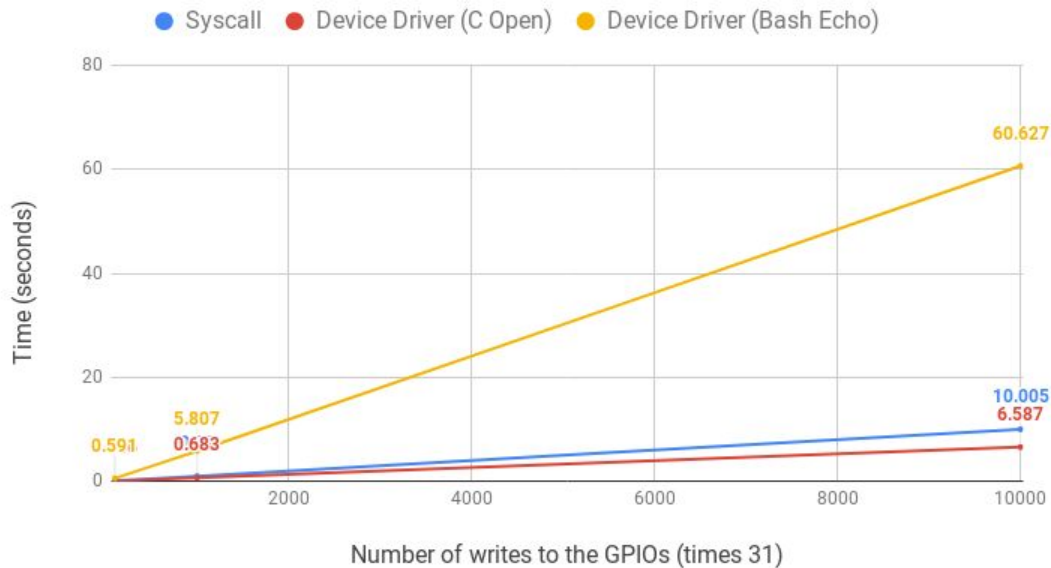
CODE!!!

References

1. https://en.wikipedia.org/wiki/Raspberry_Pi
2. Broadcom BCM2835 ARM Peripherals. https://www.raspberrypi.org/app/uploads/2012/02/BCM2835-ARM-Peripherals.pdf
3. Jonathan Corbet. "GPIO in the kernel: an introduction". Online: https://lwn.net/Articles/532714/
4. GPIO Interfaces. Android kernel documentation. Online: https://android.googlesource.com/kernel/omap/+/glass-omap-xrr02/Documentation/gpio.txt