ECE 471 – Embedded Systems
Lecture 14

Vince Weaver

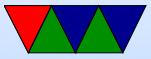
http://www.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

24 October 2013

Homework #2 Comments

- Code Commenting: be sure to comment well!
  Make sure comments match the code!
  More is (usually) better.
  Quality more important than quantity, though with assembly code it's best to over-comment. Think of you (or someone else) trying to follow the code 5 years from now.



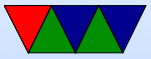
Do *not* just restate the code:

- **bad:** `add r1,r1,#1 // add 1 to r1`

Rather, describe what the code does at a high level:

- **better:** `add r1,r1,#1 // Increment string pointer`

- I was impressed with the different ways used to save r0.
  push/pop
  save/restore from 32-bit variable in memory

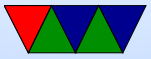


use a higher register (r6 was popular)

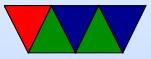
- Also the various Ways to divide by 16
  $1/16 * 2^{32}$
  shift to divide, mask with 0xf
  shift to divide, double shift to get remainder
  shift, shift, subtract

- Also impressed with variety of ways in C code

- Decimal Printing: why 10 chars?

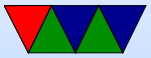


Because maximum 2^{32} is $\sim$4 billion, fits in 9 digits



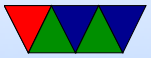
Homework #3 Concerns

- Have you started?
- If you see weird intermittent problems and your 5V line drops to a much lower value, you may need to have me re-flow the solder on the power connectors.
  As a workaround you can use 3.3V to power the display instead of 5V.



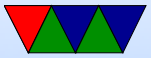
Gumstix 40-pin header

- The Gumstix board has a TI OMAP3503 CPU on-board, with lots of I/O. 400-500 pins. Only a subset can be routed to the user.
- In addition there is a TPS65950 Voltage Regulator chip which provides signals too, mostly the A/D converter and audio
- The full documentation for these chips can be found and easily accessed online.

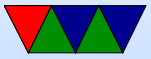


One-Wire Bus

- From Dallas Semiconductor
- Low speed data and power over one wire (you also need ground)
- Devices have capacitor to provide power when data line low
- One master
- 16.3kbit/s

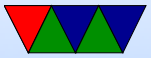


- Up to 300m twisted pair

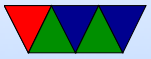


One-Wire Protocol

- Master sends reset pulse; holds low for 480us. Resets all slaves
- A slave shows it is there by holding low for another 60us
- A 1 is a short ($\sim$10us) pulse. A 0 is a longer (60 us) pulse.
- To receive data, master sends 10us low. If slave wants wants a 1, it does nothing. If it wants a 0, it pulls low for 60us.

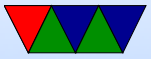


- Typically 8-bit command followed by 8-bit data chunks
- Each device has unique 64-bit ID
- Process to enumerate the bus by probing bits with binary search; faster than probing all 2^{64} possible.



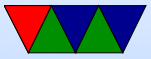
Linux Interface

- "w1" driver merged in 3.6 kernel (about a year ago)
  - Driver for various interfaces, including bit-banging over GPIO (w1-gpio)
  - `/sys/bus/w1/devices/22-0000001d84f2/w1_slave`
  - read value and get ASCII dump of transaction
- OWFS – another drive, not in main kernel. Lets you export one-wire devices as a filesystem

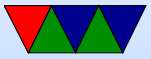


One-Wire on Gumstix

- Despite pin 19 being labeled as “1WIRE” the signal is not multiplexed by default.
- By default it is configured as “HDQ” (a protocol similar to 1-wire but not exactly the same)
- In theory you could write code to properly multiplex in the bootloader or kernel. The gumstix wiki says no one has done this yet.
- People have used 1wire (also on the similar beagleboard)

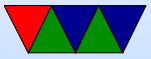


by just using a GPIO pin and w1-gpio driver.

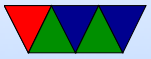


SPI bus

- Synchronous full-duplex serial bus named/formalized by Motorola. No real standard.
- Master/slave with multiple slave select lines
- 4-wire bus
- SCLK from master
- MISO – master in, slave out

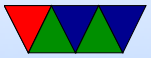


- MOSI – master out, slave in
- CS0, CS1, etc – slave chip selects



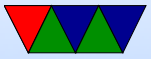
SPI protocol

- Master starts clock
- Pulls chip-select of desired slave low
- Send and receive (at same time over MISO/MOSI wires



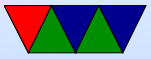
SPI advantages

- Full-duplex, fast, arbitrary message size, low power (no pullup resistors), no arbitration, no unique ids, unidirectional signals
- Disadvantages: more pins, one master, short distances



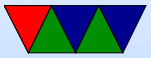
SPI uses

- Interfaces that need more bandwidth than other embedded busses
- Can interface to SD card this way (slower)

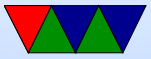


SPI bus on Linux

- SPIDEV
- /dev/spidevB.C
- sys/devices/.../spiB.C
- /sys/class/spidev/spidevB.C
- open/close /dev/spidevB.D
- ioctl SPI_IOC_RD_MODE, SPI_IOC_WR_MODE

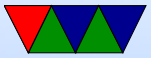


- ioctl SPI_IOC_RD_LSB_FIRST, SPI_IOC_WR_LSB_FIRST
- ioctl SPI_IOC_RD_BITS_PER_WORD, SPI_IOC_WR_BITS
- ioctl SPI_IOC_RD_MAX_SPEED_HZ, SPI_IOC_WR_MAX_
- read/write the device node

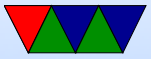


SPI bus on Gumstix

- SPI1 brought to expansion with cs0 and cs1 lines
- cs0 is the touch screen, cs1 is the display. Those have to be disabled if want to use SPI.
- Pin 3 - SPI1_CLK
- Pin 5 - SPI1_MOSI
- Pin 7 - SPI1_MISO

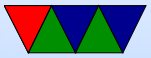


- Pin 6 - SPI1_CS0
- Pin 8 - SPI1_CS1
- Also nIRQ pin



Analog Digital Converters on Overo

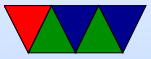
- 0-1 battery charging
- 2-7 general purpose, input range 0-2.5V, 10bits
- in8 is from USB, in12 is 3.3V rail
- AGND, ADCIN2/3/4/5/6/7
- Voltages in millivolts
- in12_input is 3.3V rail



```
cat /sys/class/hwmon/hwmon0/device/in12_input
3272
```

- madc Linux driver CONFIG_TWL4030_MADC TPS65950 talks over i2c but kernel handles that
- /sys/class/hwmon/hwmon0/device/

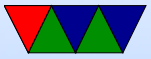
```
cat /sys/class/hwmon/hwmon0/device/temp1_input
56
```



Other pins on Overo Header

For details see the documentation for the chip, as these signals in general come right off the chip.

- PWM – pulse-width modulation. On board timers can generate exact length pulses. Useful for servos? Need custom kernel module
- TXD1/RXD1 – ttyS0 serial port
- TXD3/RXD3/CTS3 – ttyS2 serial port, IR (infrared) compatible



- TSnIRQ – interrupt?
- gpsPPS – typically sends a pulse once per second, for use in timing. Often a GPS signal provides this?
- POWERON – says when power is on?
- WAKEUP – ?
- VBACKUP – ?
- SYS_EN – ?