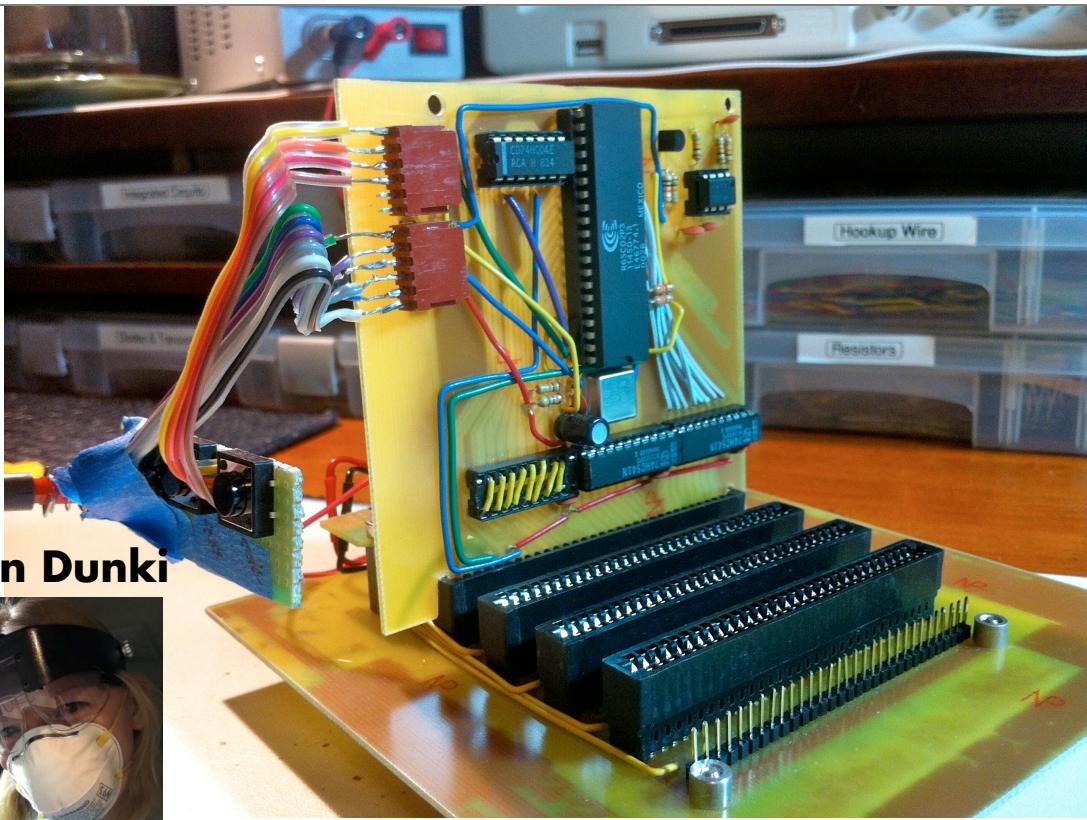


Computer Systems from the Ground Up

**Dawson Engler
Pat Hanrahan
Phil Levis
Isabel Bush**

Goal:

Understand Computer Systems



Quinn Dunki



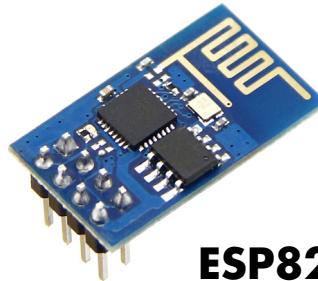
<http://quinndunki.com/blondihacks/>



Embedded Computing/Internet of Things

Communication (802.11 b/g/n)

Sensors



Actuators

ESP8266 <\$3

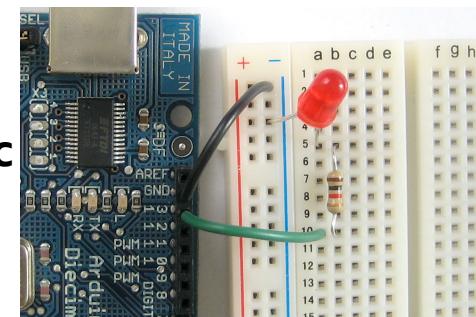
Computer : 32-bit Processor, 80 Mhz, 32 KB RAM

Arduino UNO (\$25)

```
// Blink an led
int led = 13;

void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  digitalWrite(led, HIGH);
  delay(1000); // 1000 msec
  digitalWrite(led, LOW);
  delay(1000);
}
```



Goal:

Learn Your Tools



<http://amhistory.si.edu/juliachild/>



<http://dans-woodshop.blogspot.com/>

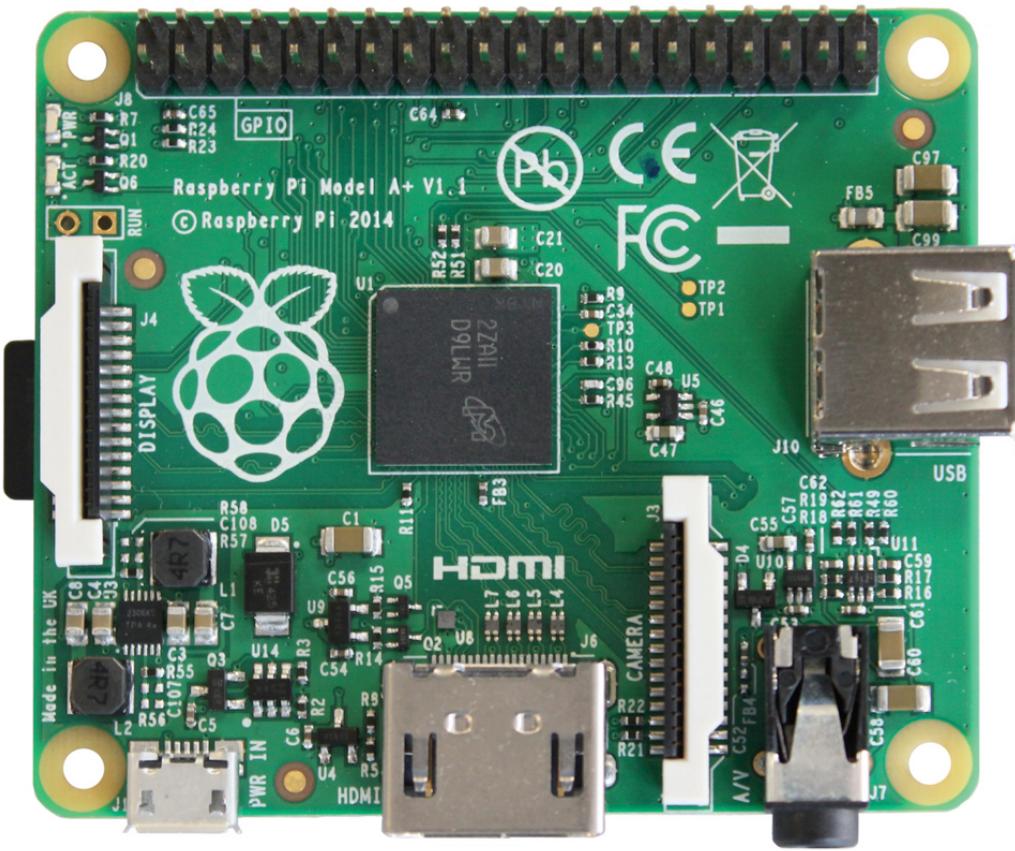


<https://paulsellars.com/tag/gouge/>

Bare Metal on the Raspberry Pi

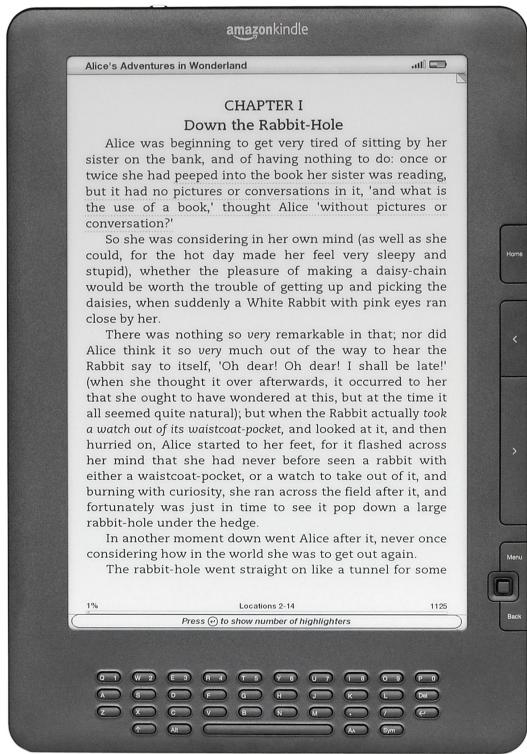
Definition: Bare metal programming involves no operating system and minimal use of libraries.

Bare metal programs boot and startup on their own, and directly control peripherals.



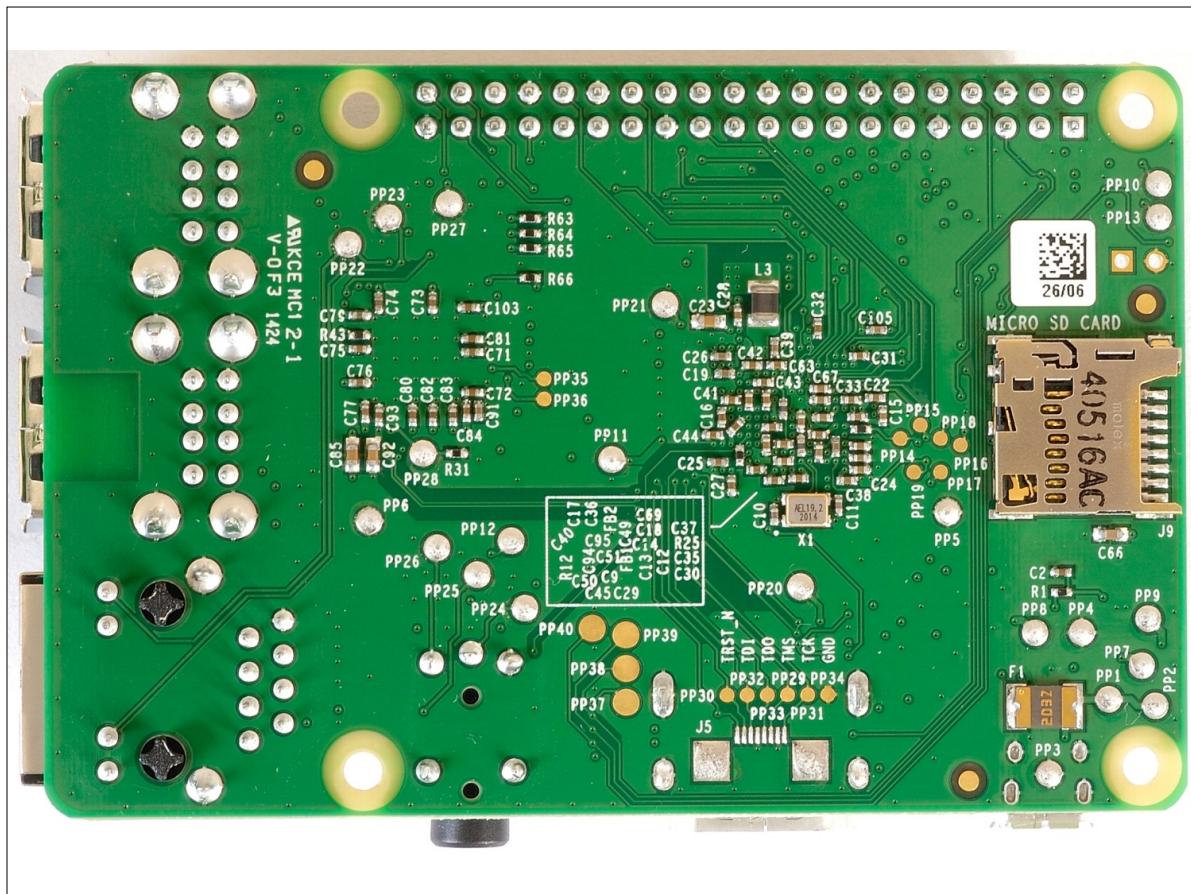
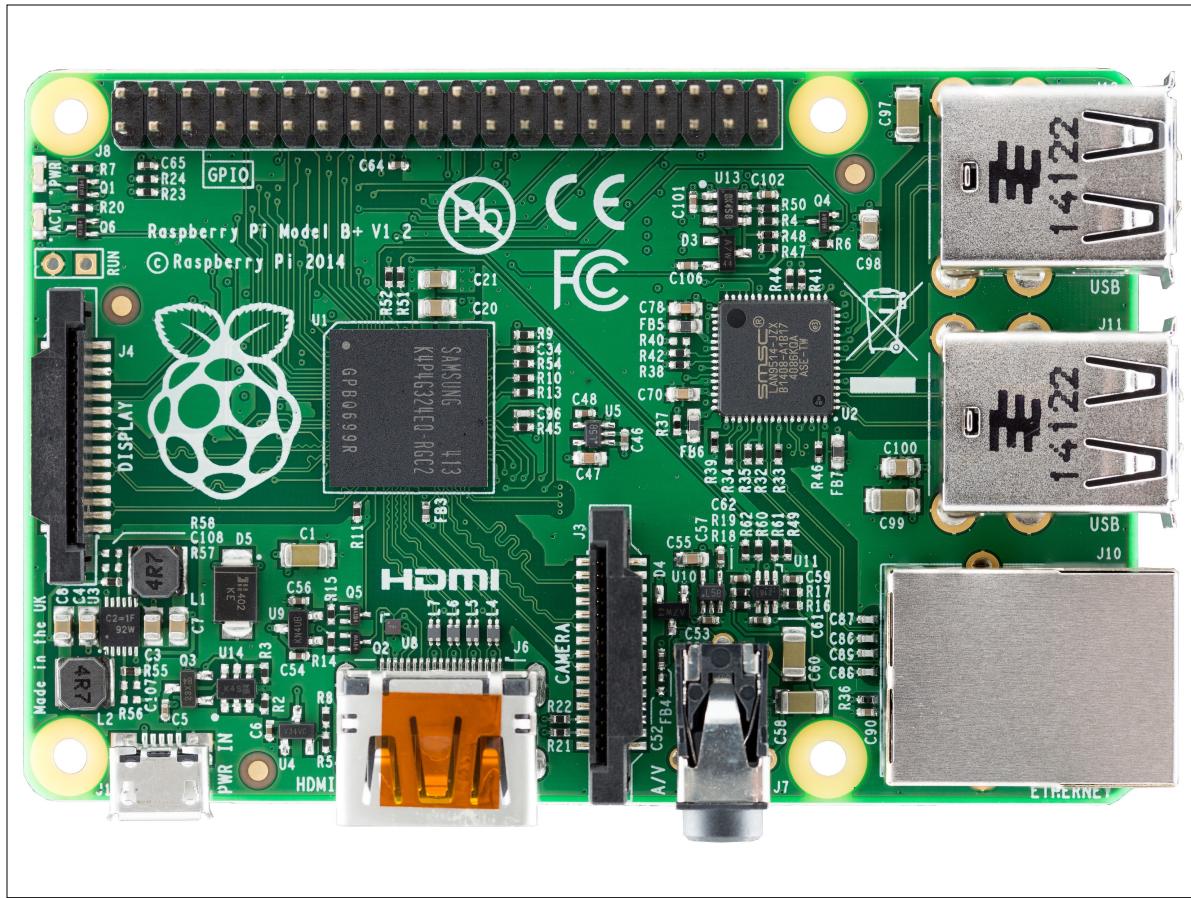


iPhone 3



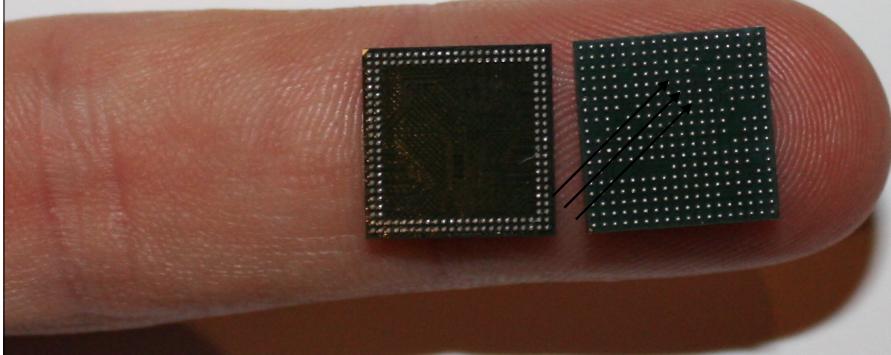
Kindle 2

Inspect Raspberry Pi

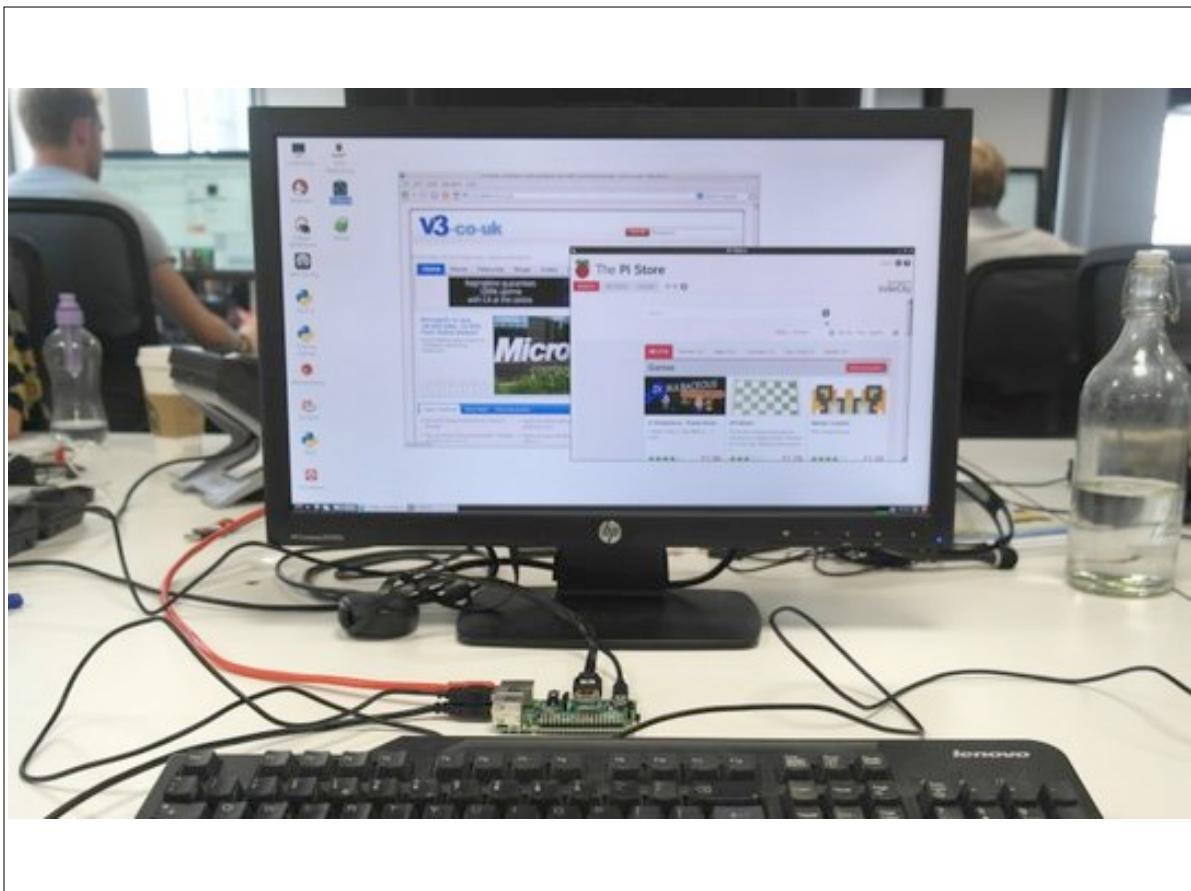


Package on Package

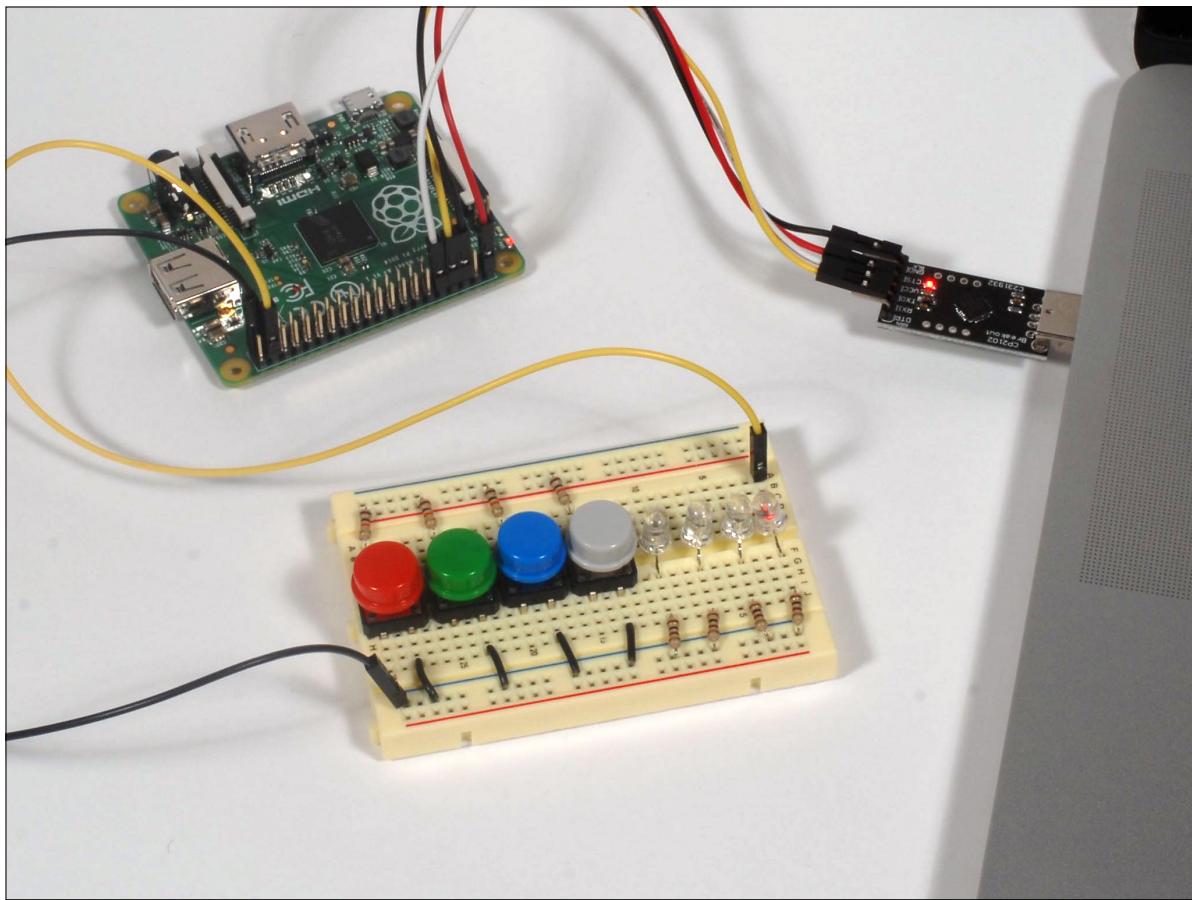
**Broadcom 2865 ARM Processor
ARM1176JZF-S 700 MHz
ARMv6 Architecture**

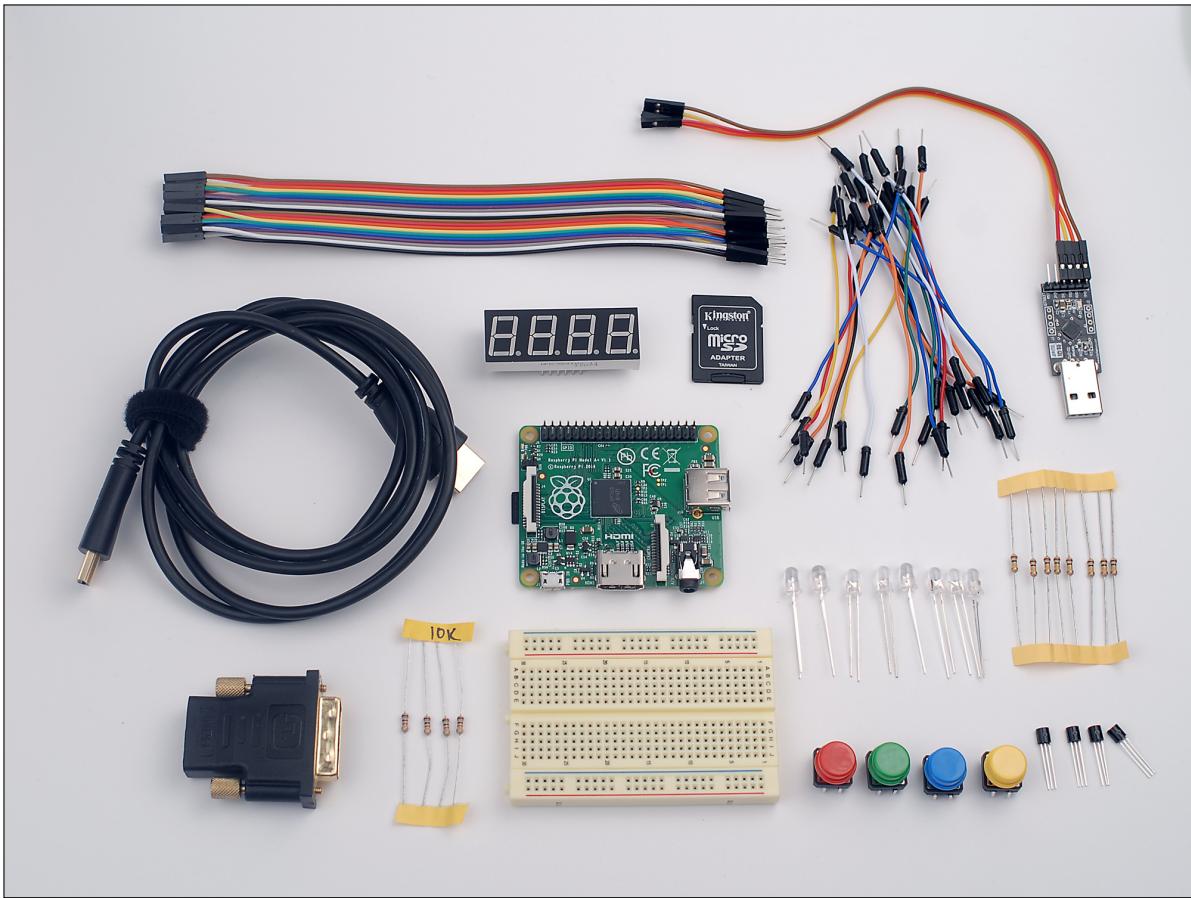


Samsung 1Gb SDRAM



Pi Kit





Manifest

Raspberry Pi A+

4GB SD card

USB Serial UART

HDMI cable and HDMI-DVI adapter

Breadboard

Jumpers: female-male, male-male

LEDs, pushbuttons, resistors, and transistors

Policy

Kit

- Complete the course - it's yours**
- Withdraw - return it**

You break it or lose it, you replace it

Course Organization

github.com/phanrahan/cs107e

§1 Baremetal Programming

- 1. ARM architecture**
- 2. ARM Assembly language**
- 3. C**
- 4. Functions**
- 5. Strings and serial communication**
- 6. Linking and loading**

Quinn Dunki Guest Lecture: Veronica

§2 Personal Computer

- 1. Graphics and the framebuffer**
- 2. Keyboard input**
- 3. Interrupts**
- 4. Audio**
- 5. MIDI input**

Computer History Museum Tour

§3 Systems Programming

- 1. Networking**
- 2. Memory allocation**
- 3. Caches**
- 4. Linux and beyond**

Administration

Modules

Weekly cadence

- 2 lectures on Fri and Mon**
- Mandatory lab on Tue evening 7-9 pm.**
- Assignment due following Mon at 12 midnight**

Laboratories

Attendance is mandatory

A set of exercises

Hand-in completed check-list

Leave ready to do assignment

Philosophy: lots-of-help, hands-on, collaborative

Lab: access to tools and supplies



Assignments

8 Assignments

Two parts

■ Basic

■ Extensions

Final project

NO EXAMS

Grading

A

- Complete all labs and basic assignments correctly
- Complete 1/2 of assignments with extensions
- Outstanding final project

B

- Complete all labs and basic assignments that work
- Good final project

C or below

- Partial credit on a significant amount of the work

Extra Credit

This is an 1st-time offering of the course

**Extra credit will be given for materials
that improve the next offering**

- Better version of an assignment or lab; alternative versions
- Written guides or explanations of hardware and software

Review

Basic Electricity

Voltage and current

Ohms Law : $V = I R$

Power : $P = I V$

Driving an LED

Transistor switches

Breadboarding

Guide: electricity.md

Number Representations

Binary representation

Hexadecimal

Bit operators

Guide: number.md

Unix Command Line

Moving around the file system

Creating, moving, and deleting files

Compiling and running programs

Guide: unix.md