IOT – AN INTRODUCTION

RAMESH PERI
ARCHITECT OF IOT AND ANDROID TOOLS
SSG/DPD/TCAR
AUSTIN, TX
AGENDA – PART 1

• Introduction & Motivation
  • My Stories
  • Promise of IoT
  • Micro-Controllers & Sensors
• The Google & Nest Story
• Systems I built
  • Sprinkler Controller
  • Security System
  • IP Camera
• Philosophical Rumination
• Projects that one can attempt as part of this course
AGENDA – PART 2

• Boards & equipment we have for our labs
• Boards on markets today
• Circuit Design Tools and PCB Manufacturing
  • Demo of Fritzing
  • Demo of 123dcircuits
• Some Resources
  • Books, IoT development kits, Where to buy stuff, Magazines, techshop
• Detailed Examples
  • Blinking an LED with RPI
  • Blinking LED with Galileo
  • Digital Temperature sensor with RPI
  • Digital Temperature sensor with Arduino
PART 2
BOARDS AND EQUIPMENT FOR OUR LABS
ARDUINO MEGA

- Digital I/O + PWM
- ATmega 1280
- 4x Serial UARTS + I2C
- +5 Volts
- 32 More Digital I/O Pins
- Backward Compatible Arduino Headers
- 16 Analog Inputs
- USB
- USB or External Power
- FTDI USB - Serial
- Reset
- ICSP
- GND
INTEL GALILEO

- Ethernet Port
- USB Client
- USB Host
- Digital Pins
- 5V Power In
- μSD Card
- SD Activity Indicator
- Power Indicator
- Arduino Sketch Reset
- Pin 13 LED
- Reboot Button
- IOREF Select
- Analog Pins
- Power Pins

- Digital Ground
- Analog Reference Pin
- Digital I/O Pins (2-13)
- Serial Out (TX)
- Serial In (RX)
- ATmega328 Microcontroller
- External Power Supply
- USB Plug
- Reset Button
- In-Circuit Serial Programmer
- Reset Pin
- 3.3 Volt Power Pin
- 5 Volt Power Pin
- Voltage In
- Ground Pins
- Analog In Pins (0-5)
AVAILABLE KITS FOR LABS
SUNFOUNDER SENSOR KIT (TBP) – ABOUT $72
BOARDS ON MARKET TODAY
BOARDS ON THE MARKET TODAY

- pcduino
- Arduino Yun
- zedboard
- ifc6410
- BeagleBone Black
- Odroid-u3
- gumstix
- netduino
- raspberrypi
- Galileo
<table>
<thead>
<tr>
<th>Company</th>
<th>website</th>
<th>cores</th>
<th>arch</th>
<th>freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino Yun</td>
<td>Qualcomm <a href="http://www.pcduino.com">www.pcduino.com</a></td>
<td>MC+1</td>
<td>MIPS</td>
<td>400M</td>
</tr>
<tr>
<td>PCDuino</td>
<td>Allwinner <a href="http://www.inforceboard.org">www.inforceboard.org</a></td>
<td>2</td>
<td>ARM</td>
<td>1G</td>
</tr>
<tr>
<td>Odroid-u3</td>
<td>Samsung <a href="http://www.hardkernel.com">www.hardkernel.com</a></td>
<td>4</td>
<td>ARM</td>
<td>1.7G</td>
</tr>
<tr>
<td>zedboard</td>
<td>Xilinx <a href="http://www.zedboard.org">www.zedboard.org</a></td>
<td>n/a</td>
<td>FPGA</td>
<td></td>
</tr>
<tr>
<td>ifc6410</td>
<td>Qualcomm <a href="http://www.inforceboard.org">www.inforceboard.org</a></td>
<td>4</td>
<td>ARM</td>
<td>1.7G</td>
</tr>
<tr>
<td>Arduino Uno/Zero/Due</td>
<td>Atmel maker</td>
<td>MC+1</td>
<td>ARM</td>
<td>16M</td>
</tr>
<tr>
<td>netduino</td>
<td>STMicro netduino.com</td>
<td>1</td>
<td>ARM-M3</td>
<td>168M</td>
</tr>
<tr>
<td>gumstix</td>
<td>TI gumstix.com</td>
<td>1</td>
<td>ARM</td>
<td>600M</td>
</tr>
<tr>
<td>raspberrypi</td>
<td>Broadcom raspberrypi.org</td>
<td>1</td>
<td>ARM</td>
<td>400M</td>
</tr>
<tr>
<td>Galileo</td>
<td>Intel maker.intel.com</td>
<td>1</td>
<td>x86</td>
<td>400M</td>
</tr>
<tr>
<td>Beaglebone</td>
<td>TI Beagleboard.org</td>
<td>1</td>
<td>ARM</td>
<td>1G</td>
</tr>
</tbody>
</table>
CIRCUIT DESIGN TOOLS AND PCB MANUFACTURING
# Circuit Design and Board Fabrication

<table>
<thead>
<tr>
<th>Name</th>
<th>Website</th>
<th>Company</th>
</tr>
</thead>
<tbody>
<tr>
<td>123dcircuits</td>
<td><a href="http://www.123dapp.com">www.123dapp.com</a></td>
<td>Autodesk</td>
</tr>
<tr>
<td>Fritzing</td>
<td><a href="http://www.fritzing.org">www.fritzing.org</a></td>
<td>opensource</td>
</tr>
<tr>
<td>geppetto</td>
<td>geppetto.gumstix.com</td>
<td>gumstix</td>
</tr>
<tr>
<td>expresspcb</td>
<td><a href="http://www.expresspcb.com">www.expresspcb.com</a></td>
<td>expresspcb</td>
</tr>
</tbody>
</table>
DEMO OF FRITZING
DEMO OF 123DCIRCUITS
SOME RESOURCES
## IOT DEVELOPMENT KITS & PROGRAMMING MODELS

<table>
<thead>
<tr>
<th>Name</th>
<th>website</th>
<th>company</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino</td>
<td><a href="http://www.arduino.cc">www.arduino.cc</a></td>
<td>Opensource</td>
</tr>
<tr>
<td>ThingSquare</td>
<td><a href="http://www.thingsquare.com">www.thingsquare.com</a></td>
<td>ThingSquare</td>
</tr>
<tr>
<td>mbed</td>
<td><a href="http://www.mbed.com">www.mbed.com</a></td>
<td>ARM</td>
</tr>
<tr>
<td>Kinoma</td>
<td><a href="http://www.kinoma.com">www.kinoma.com</a></td>
<td>Marvell</td>
</tr>
<tr>
<td>Minibloq arduino</td>
<td>Blog.minibloq.org</td>
<td>opensource</td>
</tr>
<tr>
<td>Xively</td>
<td><a href="http://www.xively.com">www.xively.com</a></td>
<td>logmein</td>
</tr>
<tr>
<td>Onebox IoT</td>
<td><a href="http://www.freescale.com">www.freescale.com</a></td>
<td>Freescale &amp; Oracle</td>
</tr>
<tr>
<td>IOE</td>
<td><a href="http://www.qualcomm.com/solutions/ioe">http://www.qualcomm.com/solutions/ioe</a></td>
<td>qualcomm</td>
</tr>
<tr>
<td>IRIS</td>
<td><a href="http://www.lowes.com/cd_Iris_239939199">http://www.lowes.com/cd_Iris_239939199</a>_</td>
<td>lowes</td>
</tr>
<tr>
<td>Labview</td>
<td><a href="http://www.ni.com">www.ni.com</a></td>
<td>NI</td>
</tr>
<tr>
<td>raspiconnect</td>
<td>Milocreek.com</td>
<td></td>
</tr>
<tr>
<td>Wyliodrin</td>
<td>Wyliodrin.com</td>
<td></td>
</tr>
<tr>
<td>python</td>
<td>raspberrypi</td>
<td>opensource</td>
</tr>
</tbody>
</table>
## SOME BOOKS

<table>
<thead>
<tr>
<th>Author</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Margolis</td>
<td>Arduino Cook Book</td>
</tr>
<tr>
<td>Simon Monk</td>
<td>Raspberry Pi Cook Book</td>
</tr>
<tr>
<td>Steven Goodwin</td>
<td>SmartHome Automation with Linux and Raspberry Pi</td>
</tr>
<tr>
<td>Andrew K Dennis</td>
<td>Raspberrypi Home Automation with Arduino</td>
</tr>
<tr>
<td>Gareth Halfacree &amp; Eben Upton</td>
<td>Raspberry Pi User Guide</td>
</tr>
<tr>
<td>Donald Norris</td>
<td>Raspberry Pi projects for evil genius</td>
</tr>
<tr>
<td>Matt Richardson</td>
<td>Getting Started with Intel Galileo</td>
</tr>
</tbody>
</table>

Subscribe to safaribooksonline where all these books are available online for free
<table>
<thead>
<tr>
<th>Magazine</th>
<th>Cost</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>MagPI</td>
<td>Free</td>
<td><a href="http://www.themagpi.com">www.themagpi.com</a></td>
</tr>
<tr>
<td>maker</td>
<td>Subscription</td>
<td><a href="http://www.makedigital.com/make">www.makedigital.com/make</a></td>
</tr>
<tr>
<td>Odroid Magazine</td>
<td>Free</td>
<td><a href="http://www.hardkernel.com">www.hardkernel.com</a></td>
</tr>
<tr>
<td>Raspi newsletter</td>
<td>Free</td>
<td><a href="http://www.raspberrypi.org">www.raspberrypi.org</a></td>
</tr>
</tbody>
</table>
## WHERE TO BUY STUFF

<table>
<thead>
<tr>
<th>URL</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>amazon.com</td>
<td>Can get lot of stuff here shipped very quickly</td>
</tr>
<tr>
<td>adafruit.com</td>
<td>Lot of cool info about arduino, rpi, educational material, kits, components</td>
</tr>
<tr>
<td>makershed.com</td>
<td>Maker community, kits, educational material</td>
</tr>
<tr>
<td>raspberrypi.org</td>
<td>All info about raspberrypi</td>
</tr>
<tr>
<td>maker.intel.com</td>
<td>Info about galileo board</td>
</tr>
<tr>
<td>dexterindustries.com</td>
<td>Robotics for raspberrypi and lego</td>
</tr>
<tr>
<td>Sparkfun.com</td>
<td>Lot of sensors and kits for arduino, rpi</td>
</tr>
<tr>
<td>Coocox.org</td>
<td>More peripherals and boards</td>
</tr>
<tr>
<td>Freetronics.com</td>
<td>Lot of arduino compatible stuff</td>
</tr>
<tr>
<td>Sainsmart.com</td>
<td>Lot of peripherals and sensors for arduino</td>
</tr>
<tr>
<td>Codeduino</td>
<td>Lot of info about arduino and projects</td>
</tr>
<tr>
<td>Avnet.com, digikey.com, mouser.com, frys, RS</td>
<td>Lot of electronic peripherals</td>
</tr>
</tbody>
</table>

Check Shieldlist.org for a complete list
<table>
<thead>
<tr>
<th><strong>Notes</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Dealextreme.com</td>
</tr>
<tr>
<td>Can get lot of stuff here shipped very “slowly” from china but very inexpensive</td>
</tr>
<tr>
<td><a href="http://www.pimoroni.com">www.pimoroni.com</a></td>
</tr>
<tr>
<td>UK based company sells lot of cool rpi stuff</td>
</tr>
<tr>
<td><a href="http://www.project-laika.com">www.project-laika.com</a></td>
</tr>
<tr>
<td>Rpi and robotics – an expansion board for rpi</td>
</tr>
<tr>
<td>Modmypi.com</td>
</tr>
<tr>
<td>A shop for rpi based stuff</td>
</tr>
<tr>
<td><a href="http://www.kitronik.co.uk">www.kitronik.co.uk</a></td>
</tr>
<tr>
<td>Lot of cool stuff</td>
</tr>
</tbody>
</table>
TECHSHOP

Abrasives
Arts & crafts
Automotive
Electronics
Fabrication
Fabrics & sewing
Hand tools
Layout
Machining
Measurement
Plastics
Prototyping
Sheet metal
Surface finishing
Welding
Woodworking
DETAILED EXAMPLES
BLINKING LED ON RASPBERRY PI

Watch the video

- Section 9.01
BLINKING LED ON GALILEO (ARDUINO SKETCH)

Decent tutorial on getting started with Galileo at sparkfun

```cpp
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 * This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup()
{
    // initialize the digital pin as an output.
    pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop()
{
    digitalWrite(led, HIGH);  // turn the LED on (HIGH is the voltage level)
    delay(1000);  // wait for a second
    digitalWrite(led, LOW);   // turn the LED off by making the voltage LOW
    delay(1000);  // wait for a second
}
```
**BLINKING LED ON GALILEO (SHELL)**

```
Get a shell command by telneting into the galileo board
GPIO pin to Arduino pin mapping complexity

echo -n "28" > /sys/class/gpio/export
echo -n "out" > /sys/class/gpio/gpio28/direction
echo -n "strong" > /sys/class/gpio/gpio28/drive
for i in {1..5} do
    echo -n "1" > /sys/class/gpio/gpio28/value
    sleep 2
    echo -n "0" > /sys/class/gpio/gpio28/value
    sleep 2
done
echo -n "28" > /sys/class/gpio/unexport
```
TEMPERATURE SENSOR DS18B20

One Wire Digital Temperature Sensor - DS18B20
SEN-00245 ROHS

Description: This is the latest DS18B20 1-Wire digital temperature sensor from Maxim IC. Reports degrees C with 9 to 12-bit precision, -55°C to 125°C (±0.5°C). Each sensor has a unique 64-bit Serial number etched into it - allows for a huge number of sensors to be used on one data bus. This is a wonderful part that is the cornerstone of many data logging and temperature control projects.

Features:

- Unique 1-Wire® interface requires only one port pin for communication
- Each device has a unique 64-bit serial code stored in an onboard ROM
- Multidrop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Measures temperatures from -55°C to +125°C (-67°F to +257°F)
- ±0.5°C accuracy from -10°C to +85°C
- Thermometer resolution is user-selectable from 9 to 12 bits
- Converts temperature to 12-bit digital word in 750ms (max.)
- User-definable nonvolatile (NV) alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive environments
TEMPERATURE SENSOR ON RPI

- Watch the video
  - Section 12.09
TEMPERATURE SENSOR ON ARDUINO

Install the onewire library into Arduino

```cpp
#include <OneWire.h>
#include <DallasTemperature.h>
// Data wire is plugged into port 2 on the Arduino
#define ONE_WIRE_BUS 2
// Setup a oneWire instance to communicate with any OneWire devices
OneWire oneWire(ONE_WIRE_BUS);
// Pass our oneWire instance to Dallas Temperature.
DallasTemperature sensors(oneWire);
void setup(void)
{
  // start serial port
  Serial.begin(9600);
  Serial.println("Dallas Temperature IC Control Library Demo");
  // Start up the library
  sensors.begin();
}
void loop(void)
{
  // call sensors.requestTemperatures() to issue a global temperature
  // request to all devices on the bus
  Serial.print("Requesting temperatures...");
  sensors.requestTemperatures(); // Send the command to get temperatures
  Serial.println("DONE");
  Serial.print("Temperature for the device 1 (index 0) is: ");
  Serial.println(sensors.getTempCByIndex(0));
}
```
OTHER STUFF

• Check the video at http://vimeo.com/18539129
  • Good documentary on Arduino and make movement
• If you have techshop in your city check it out
• Become a member of Intel maker group at your site
  • If there none look at starting one
  • Start a galileo enthusiast group at your site
• Participate in maker/raspberrypi meetups in your city
  • Start a galileo meetup in your city
• Check out makerfaire in your city
  • Intel is a sponsor and you can volunteer there
• Become a member of Intel planet blue maker site