Introduction to Informix and the Internet of Things

Mike Walker

mike@advancedatatools.com

Internet of Things (IoT) Webcasts

1. **Introduction to Informix and the Internet of Things**
2. Using Informix TimeSeries and the Internet of Things – June 2nd
3. Running an Informix Database Server on an ARM Computer – June 23rd
4. Installing and Configuring Informix on an ARM Computer – July 28th

Introduction to Informix and the Internet of Things

- Brief overview of what IoT is
- IBM & IoT
- Demonstration of sensor data and IoT
- Overview of an IoT solution – how the pieces fit together
- Microcontrollers – connect to the physical world
- Example of reading a temperature sensor using an Arduino
- Use an ARM-based computer as a low-power, low-cost gateway
- Brief overview of installing Informix on a Beaglebone Black
- Wireless communication to/from ARM gateway using XBee radios
- Receiving sensor data at the gateway

Internet of Things (IoT)

- Connecting together the Physical and Digital World
- Network of objects, exchanging information

Internet of Things - Examples

- Smart Meters
  - Real time power usage monitoring
- Virgin Atlantic
  - Use internet connected sensors on engines, flaps, landing gear, etc. Can generate 0.5 TB from a single flight *(Computerworld UK)*
- DisneyWorld
  - MagicBand : check into hotel room, buy lunch, go through the turnstiles at the amusement parks. Disney can collect data on visitor movement *(ZDNet)*

Internet of Things – Consumer Examples

- Lightbulbs
- Thermostat
- Garage Door Controller
- Door Lock
- Power Strip
- Wall Switch
- Propane Tank Gauge
- Piggybank

All connected…and all accessible with a smartphone

Internet of Things (IoT)

Nearly 26 billion devices on the Internet of Things by 2020

Gartner, Inc

More than 30 billion devices will be wirelessly connected to the Internet of Things (Internet of Everything) by 2020

ABI Research

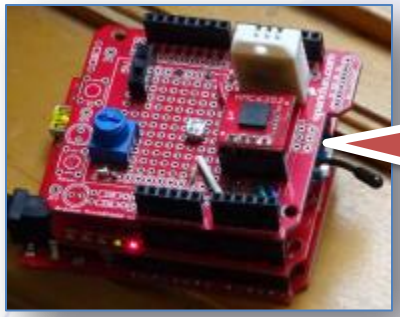
Why do we Care?

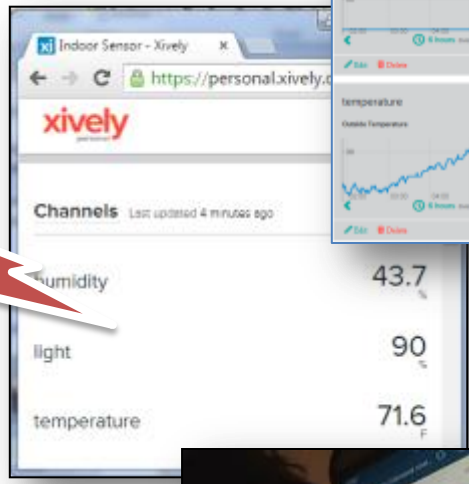
- Monitoring and control of all sorts of things
- Interaction through smartphones/tablets anywhere in the world
- Enormous amounts of data

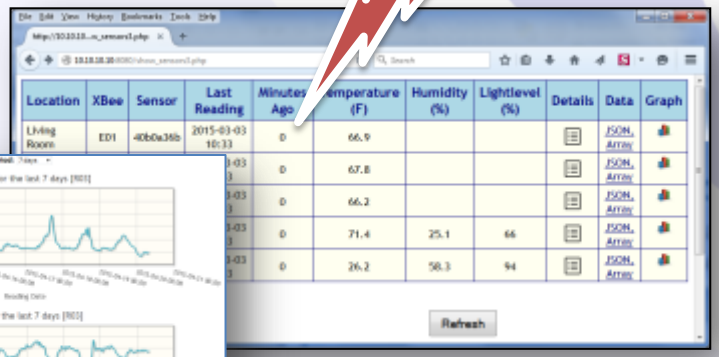
IBM and IoT

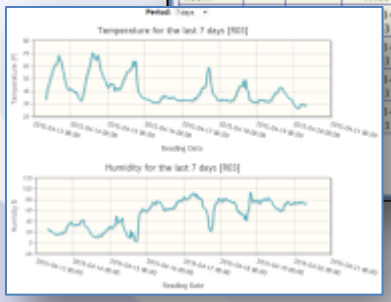
- Informix is IBM's database of choice for Internet of Things implementations
  - Informix TimeSeries
    - Very efficient storage of large volumes of sensor data
  - Informix on ARM and Intel Quark
    - Run Informix on a small footprint, cheap hardware
  - Partnerships
    - Example: Informix available as part of the Intel Gateway developer SDK
  - Informix Docker image available
    - Try out Informix quickly and easily
- Internet of Things Foundation
  - Cloud based repository for storing IoT data
- Node-RED
  - Configure flows and events through a browser

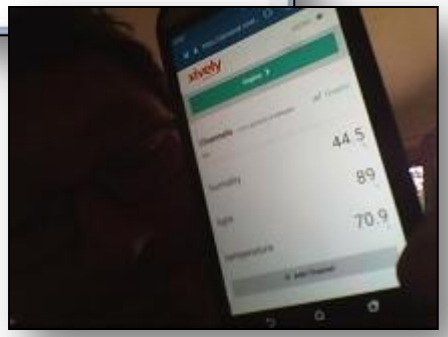
Demonstration











Advanced DataTools

That’s the WHAT…

Now the HOW…

Sensors

Sensors for everything

Electricity Usage

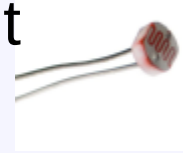
Accelerometer

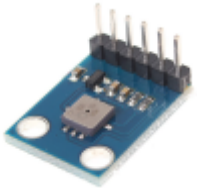
Moisture

RFID Reader

Temperature

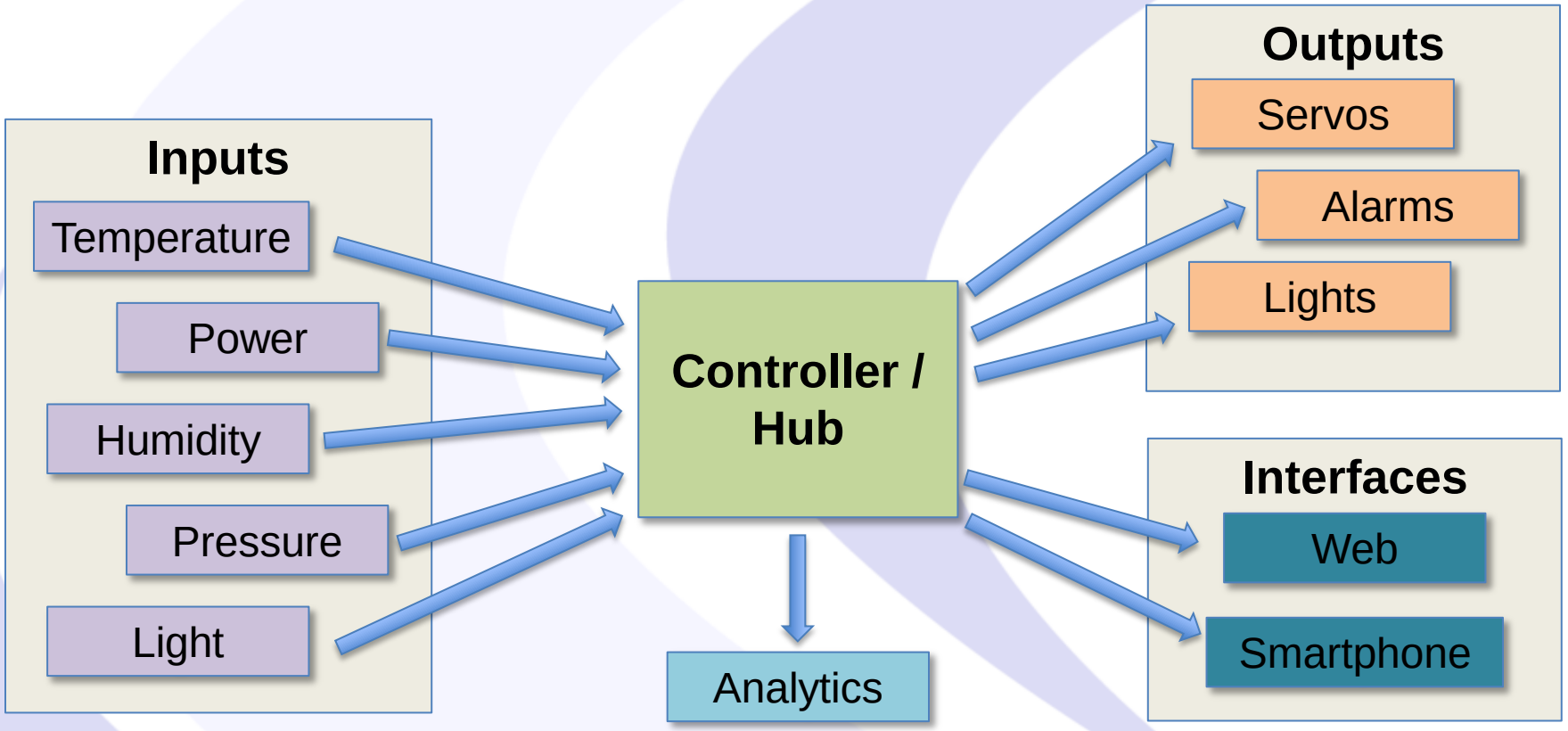
Light



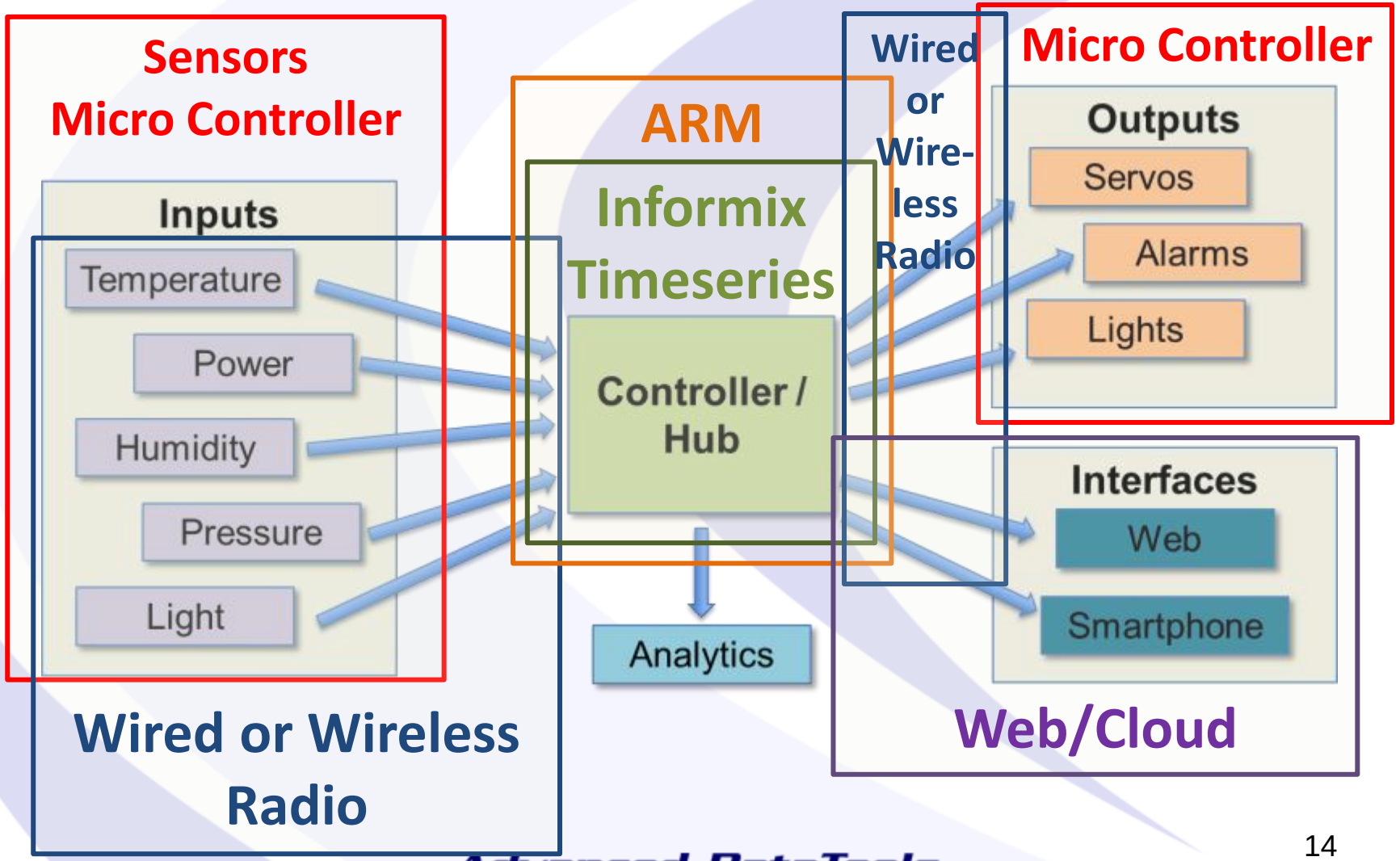


Barometric Pressure

Get Things Talking



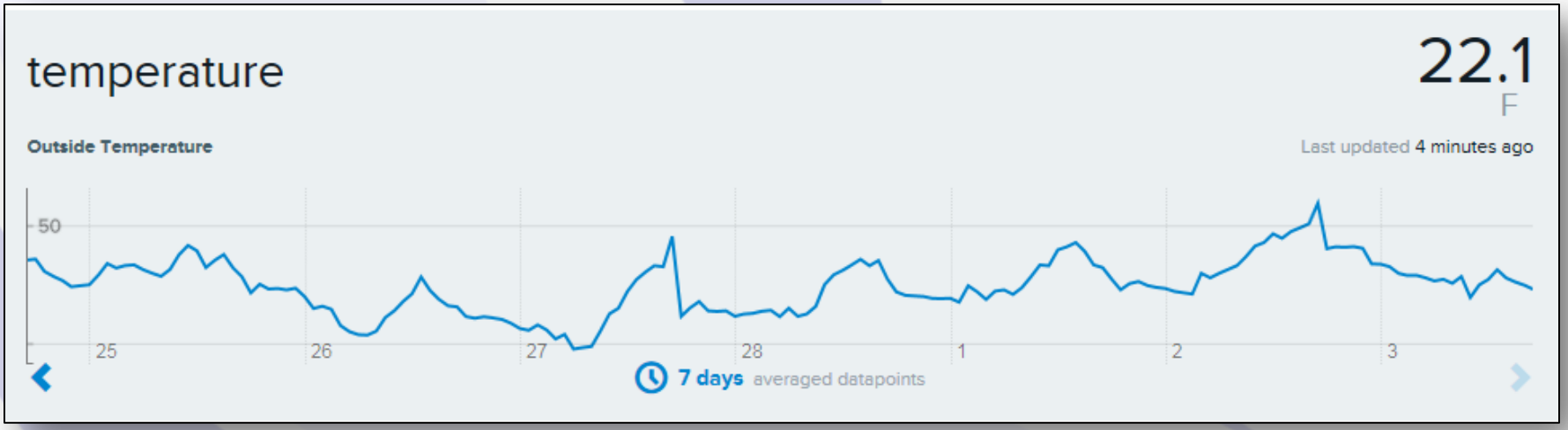
Technologies



Where to Start?

How do I go from this:

To this:



Advanced DataTools

Microcontroller Boards

- Electronics board with a microcontroller
- Can bridge the gap between the physical world around you and computers
- Input and Output pins
  - Read values from the environment
  - Turn things on and off
- Programmable

Microcontroller Boards

Intel Edison

TI LaunchPad

Beaglebone Black

Pickaxe

Arduino

Raspberry Pi

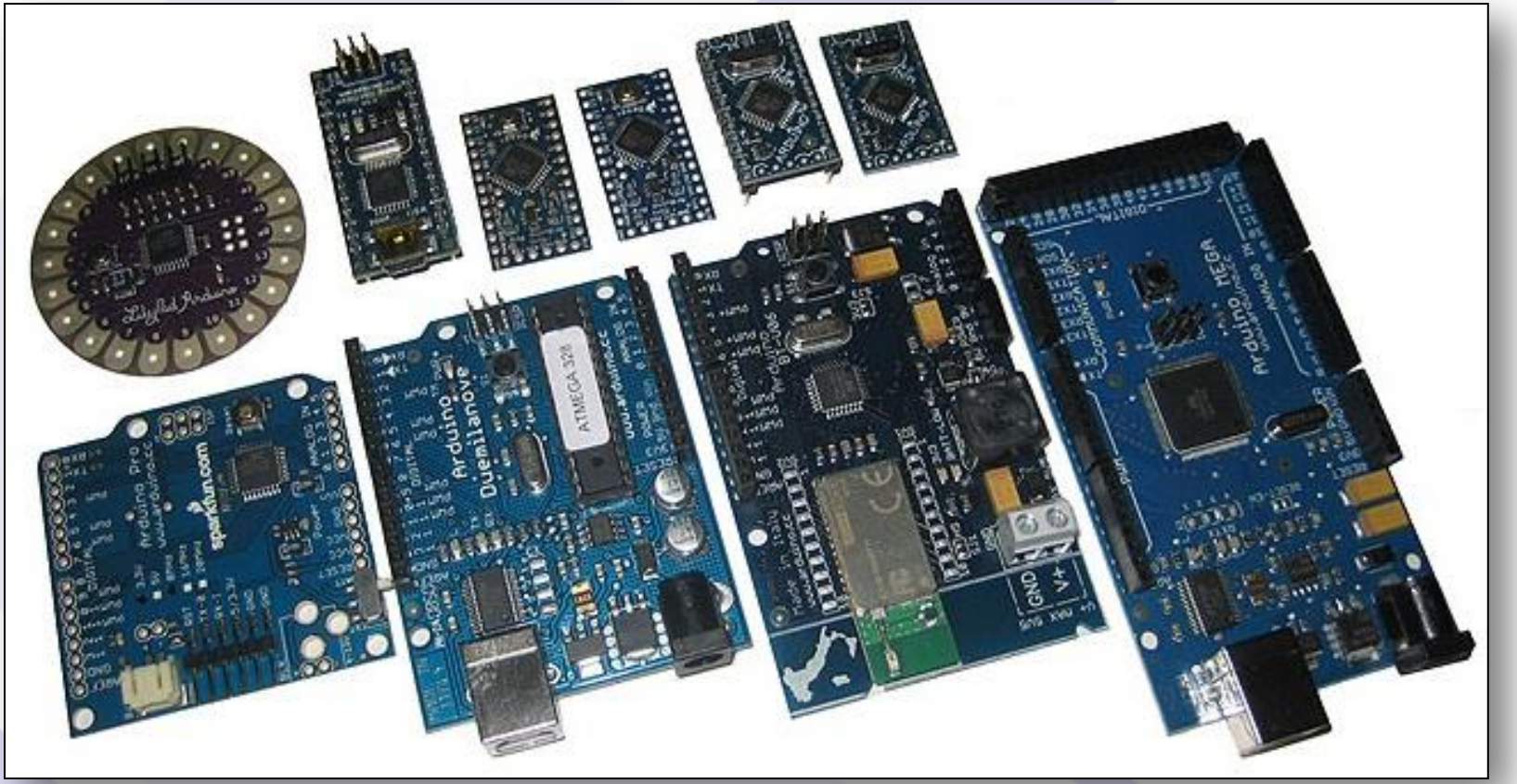
and many more…

Advanced DataTools

Arduino

- Electronics board & microcontroller
- Cheap ($10-45 depending on version)
- Programming language (based on Wiring)
- Arduino is Open Source/Open Hardware
- Many different varieties and clones

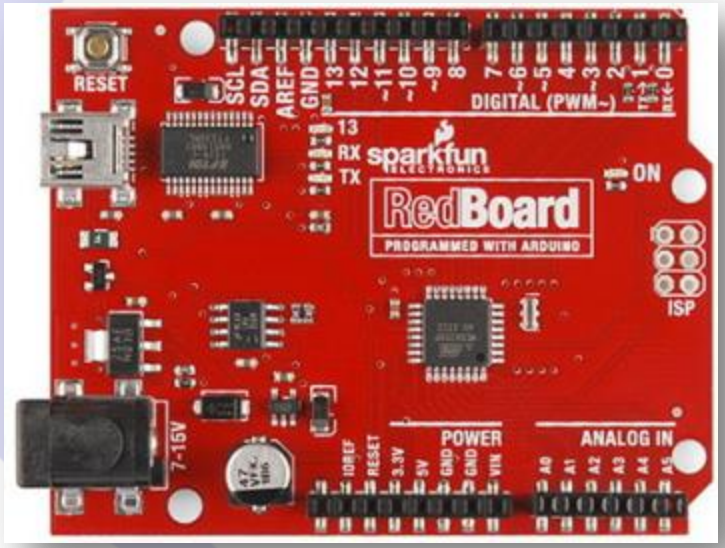
Arduino


LilyPad Arduino
sparkfun.com
Arduino Pro
Arduino
Duemilanove
ATMEGA328
www.arduino.cc
Made in Italy
GND
V+
Arduino
MEGA

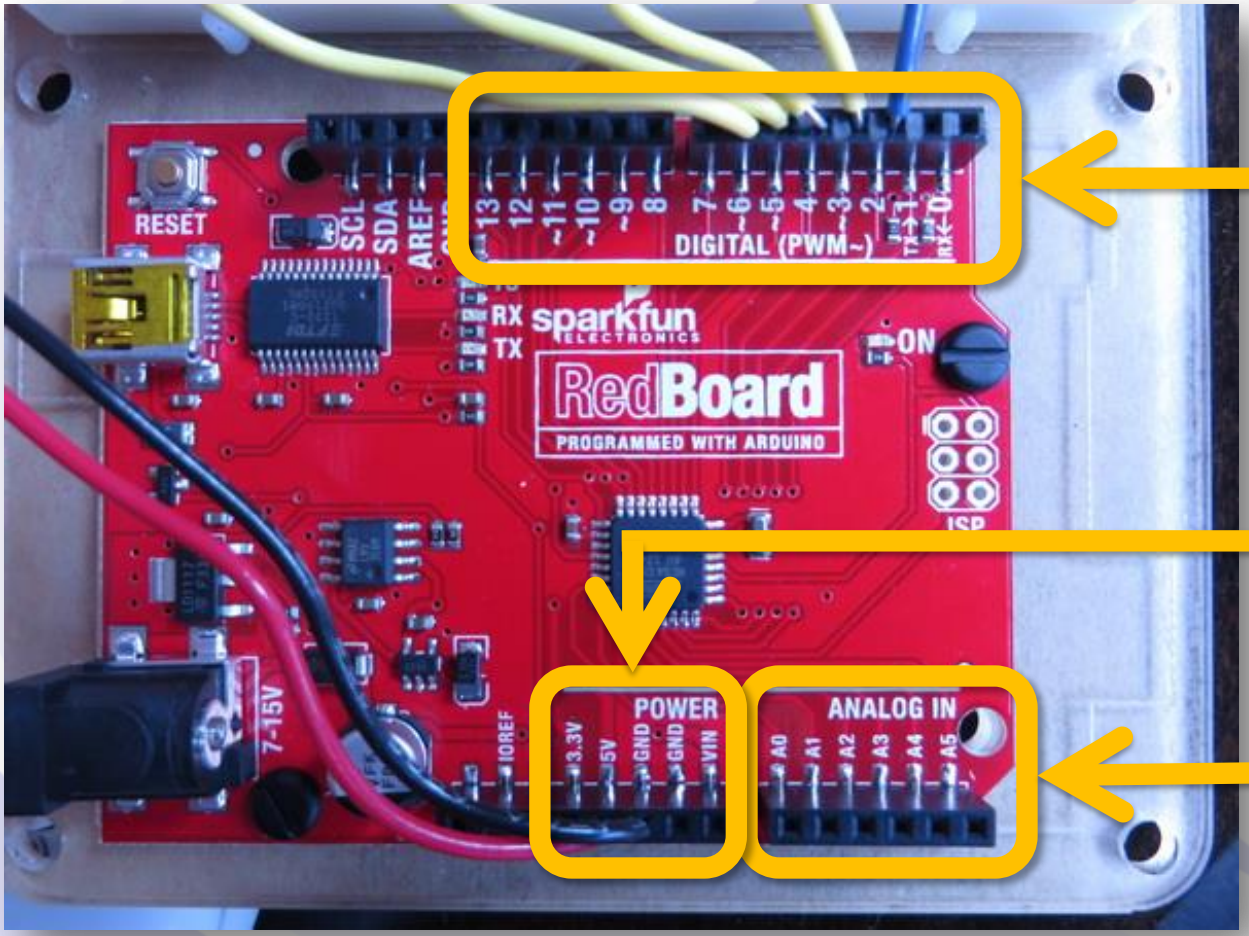
Arduino – Open Hardware

Arduino Uno R3

Sparkfun Redboard



Arduino – Connect to the Physical



Digital Pins

Power

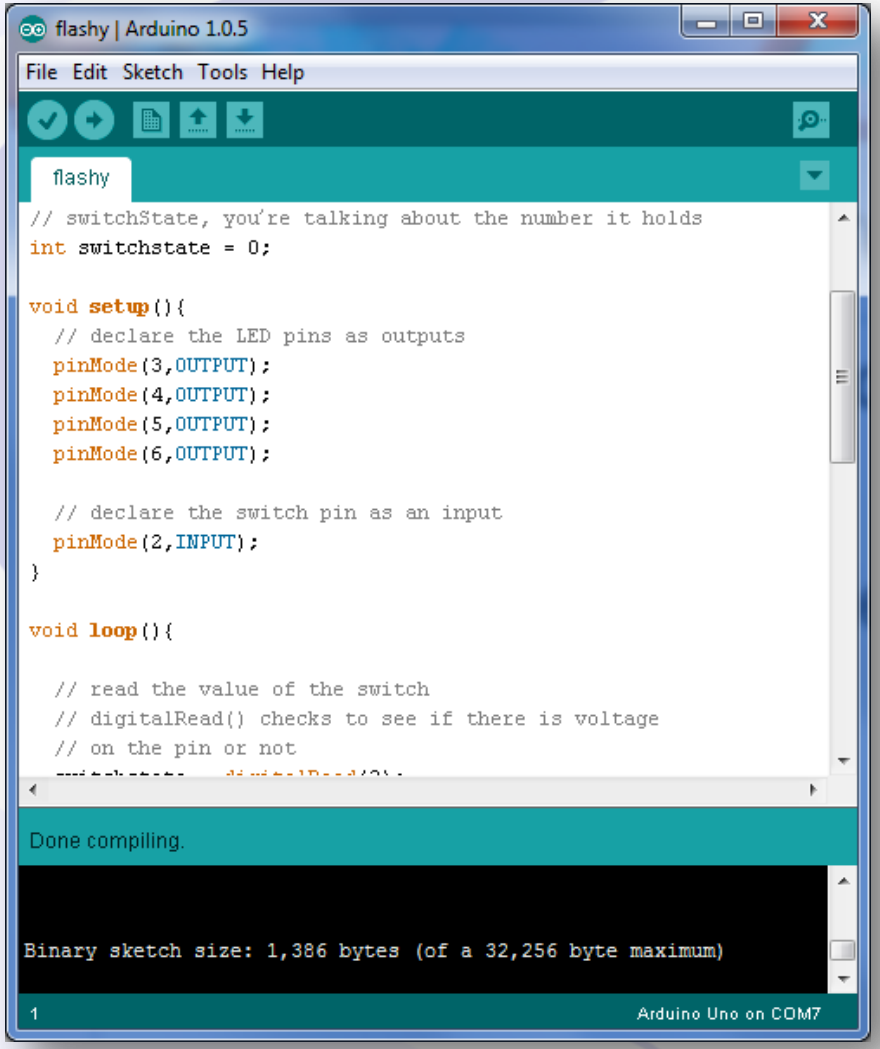
Analog Pins

Arduino Programming

- Program the Arduino from a PC using the Arduino IDE
- Free Download: http://www.arduino.cc/en/Main/Software
- Available for:
  - Windows
  - Mac
  - Linux

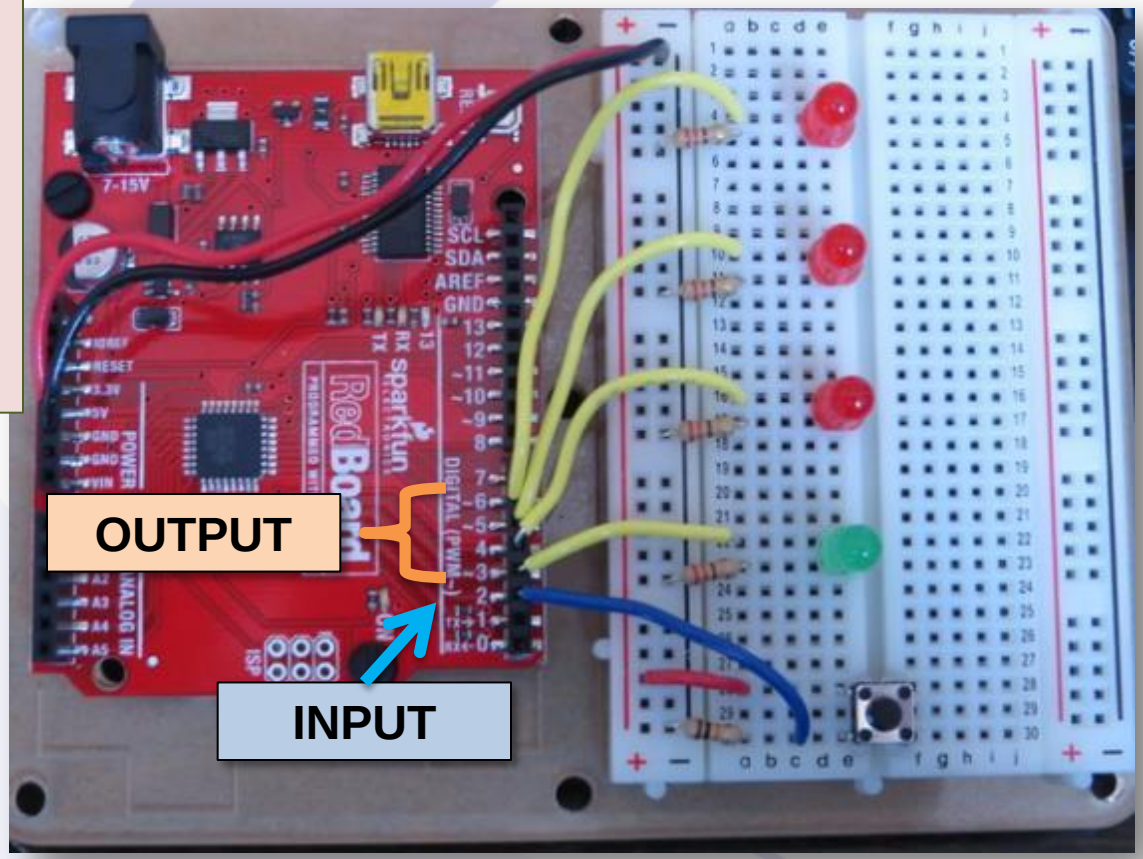
Arduino Programming

Upload “sketch” (Arduino program) via a USB cable

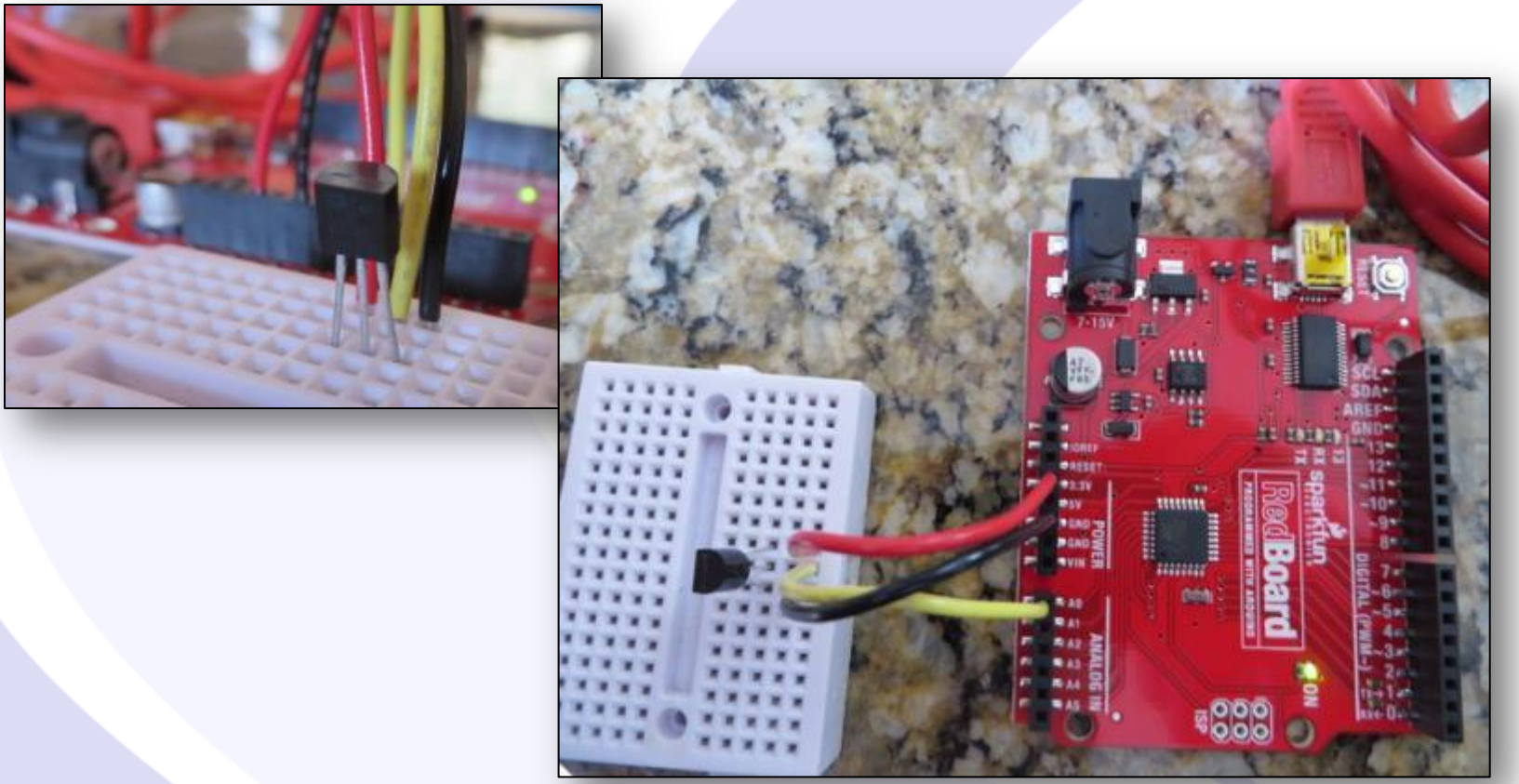


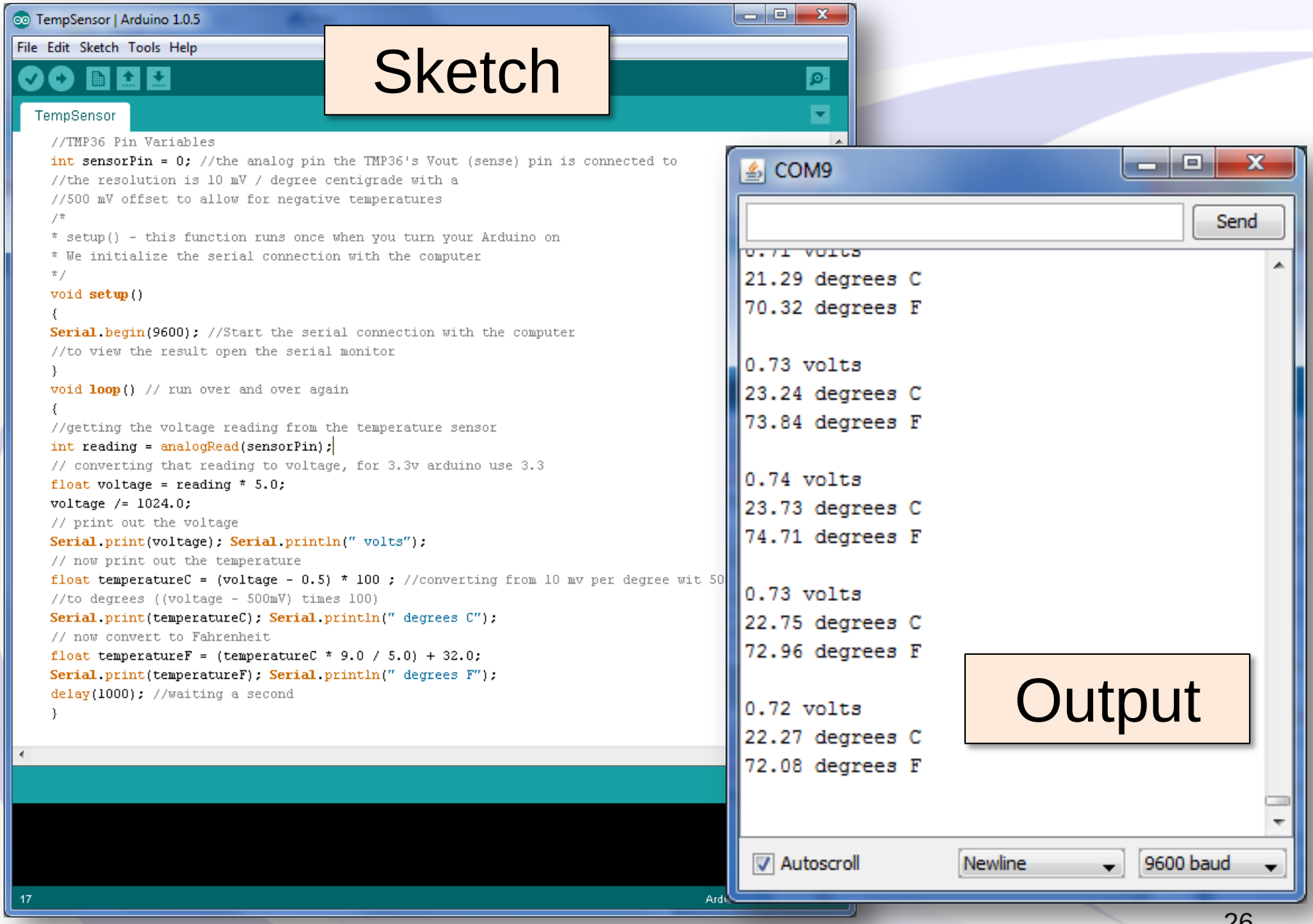
Arduino INPUT/OUTPUT

```
void setup(){
  // declare the LED pins
as outputs
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);
  pinMode(6,OUTPUT);

  // declare the switch
pin as an input
  pinMode(2,INPUT);
}
```



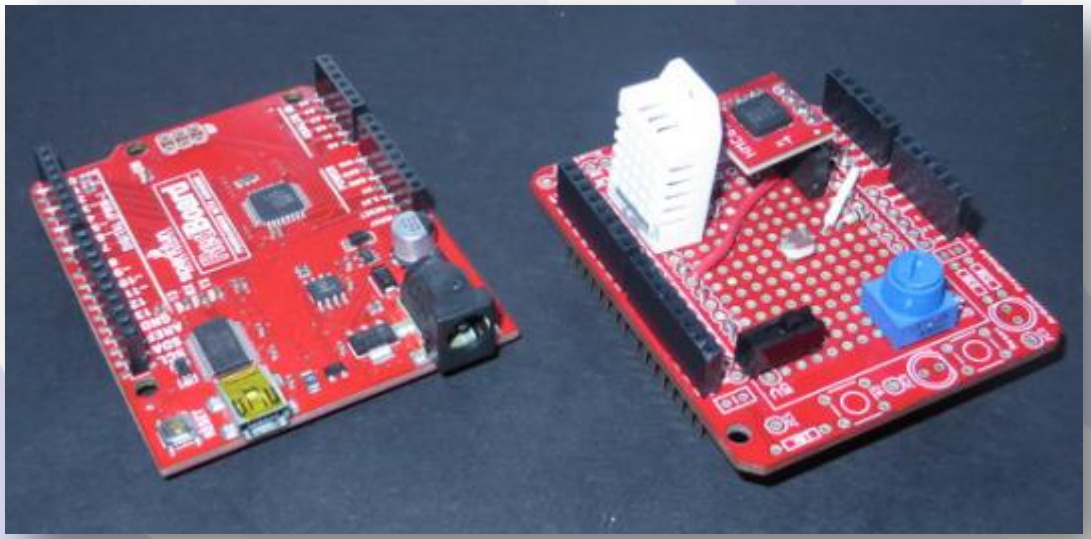
Arduino Temperature Sensor

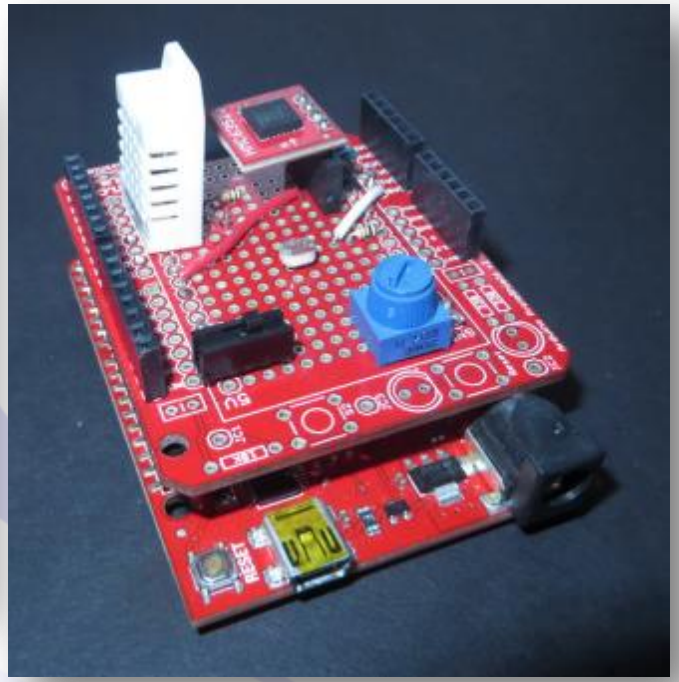



TempSensor | Arduino 1.0.5
File Edit Sketch Tools Help
TempSensor
//TMP36 Pin Variables
int sensorPin = 0; //the analog pin the TMP36's Vout (sense) pin is connected to
//the resolution is 10 mV / degree centigrade with a
//500 mV offset to allow for negative temperatures
/*
* setup() - this function runs once when you turn your Arduino on
* We initialize the serial connection with the computer
*/
void setup()
{
Serial.begin(9600); //Start the serial connection with the computer
//to view the result open the serial monitor
}
void loop() // run over and over again
{
//getting the voltage reading from the temperature sensor
int reading = analogRead(sensorPin);
// converting that reading to voltage, for 3.3v arduino use 3.3
float voltage = reading * 5.0;
voltage /= 1024.0;
// print out the voltage
Serial.print(voltage); Serial.println(" volts");
// now print out the temperature
float temperatureC = (voltage - 0.5) * 100 ; //converting from 10 mv per degree wit 50
//to degrees ((voltage - 500mV) times 100)
Serial.print(temperatureC); Serial.println(" degrees C");
// now convert to Fahrenheit
float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;
Serial.print(temperatureF); Serial.println(" degrees F");
delay(1000); //waiting a second
}
17
Sketch
COM9
Send
21.29 degrees C
70.32 degrees F
0.73 volts
23.24 degrees C
73.84 degrees F
0.74 volts
23.73 degrees C
74.71 degrees F
0.73 volts
22.75 degrees C
72.96 degrees F
0.72 volts
22.27 degrees C
72.08 degrees F
Autoscroll
Newline
9600 baud
Output

Step it up!

- Can expand the Arduino through “shields”
- Add more capabilities – WiFi, Bluetooth, etc …or more sensors





Next Step

- Now we've got a way read sensor information in a way we can understand it
- We can also drive devices in the same way and interact with the environment – actuators, lights, etc.
- To make this really useful, we want to put all this together so we can make better decisions, keep a history, publish the data and ***do stuff!***
- Need a controller…a hub/gateway

Smart Gateway

- Somewhere to collect data, store data, make decisions, trigger other actions and publish data
- Hardware depends on application

Controller

For home automation, a small Raspberry Pi, or Beaglebone Black is an ideal solution

- Run linux
- Small Footprint
- Low power
- Cheap ($35–55)
- Can run a database

Beaglebone Black

Rev C (latest version) - $55

- ARM Cortex-A8 (ARMv7)
- 512MB RAM
- 4GB internal flash storage
- Expand storage with microSD card
- Ethernet
- HDMI
- No built in WiFi, no Bluetooth
- Requires 5V power (can be powered via USB)
- Now ships with Debian Linux installed
- ***Capable of running a full version of Informix!***

Beaglebone – Setup

- Ready out of the box
  - Connect via USB and ssh into it
- Additional steps:
  - Flash Beaglebone with the latest version of Debian from a microSD card
  - Configure microSD card for additional storage
  - Set up networking (WiFi requires dongle)
  - Update packages
  - Set up user accounts

Why Install a Gateway Database?

- Maintain a history
- Advanced local reporting
- Make more informed decisions locally
- Don’t need to wait to get information from the cloud
- Don’t need to rely on an Internet connection

Why Informix?

- Over a 20 year history
- Full support from IBM
- Enterprise class database software
- Reliable & Robust
- Scalable
- Performance
- Ease of administration
- Features
  - High Availability/DR
  - Full NoSQL support – built in JSON & BSON data types, MongoDB APIs
  - Many, many more…

Why Informix for IoT?

For all the same reasons, and…

- Informix TimeSeries
  - Store large volumes of sensor data very efficiently
- Easily Embeddable
  - No administration – DBA free
  - Remote management through SQL
- Runs on low-cost hardware
  - Full version – no loss of features
  - Small footprint – disk and memory
  - OEM licensing options

Beaglebone – Informix Install

Download Informix for ARM from IBM website

	Name	Version	Type	Release date	Size	Operating system
1.	IBM Informix Dynamic Server Developer Edition (Informix Developer Edition for Linux Quark 32 (DK 50)) Exceptional, low maintenance online transaction processing (OLTP) data server for enterprise and workgroup computing.	12.10UC4DE	Released product	30 Jan 2015	1kb	Linux
2.	IBM Informix Dynamic Server Developer Edition (Informix Developer Edition for Linux ARM v6 32 (Raspberry PI)) Exceptional, low maintenance online transaction processing (OLTP) data server for enterprise and workgroup computing.	12.10UC4DE	Released product	05 Jan 2015	1kb	Linux
3.	IBM Informix Dynamic Server Developer Edition (Informix Developer Edition for Linux ARM v7 32) Exceptional, low maintenance online transaction processing (OLTP) data server for enterprise and workgroup computing.	12.10UC4DE	Released product	07 Jul 2014	1kb	Linux

Quark, e.g. Intel Galileo

ARM v6, e.g. Raspberry Pi

ARM v7, e.g. Beaglebone, Raspberry Pi 2

Beaglebone – Informix Install

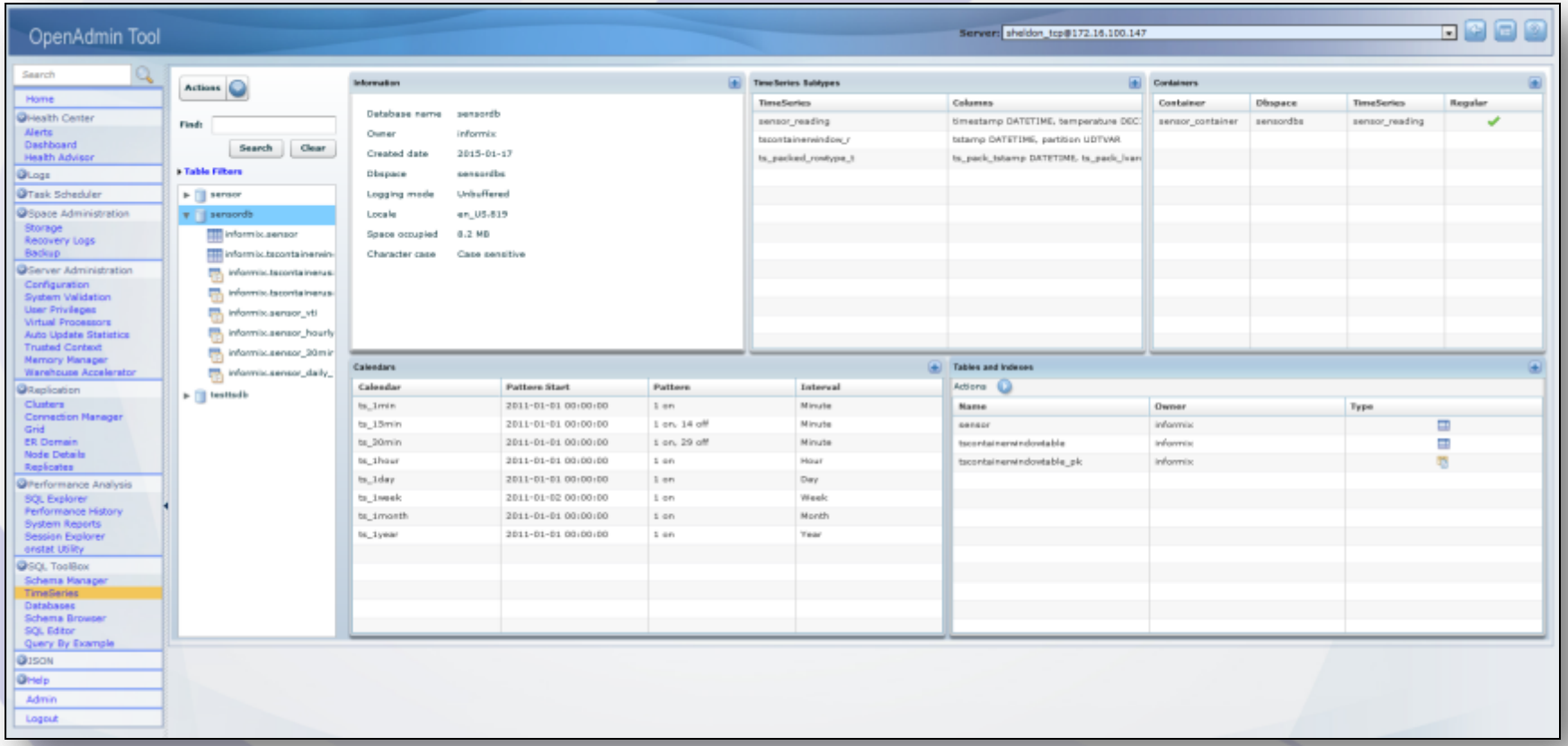
- Install the ksh package
- Upgrade java
- Copy the Informix tar file to the Beaglebone
- Unpack the tar file
- Run ./ids_install
- Perform the normal (simple!) Informix install

```
informix@beaglebone:~$ onstat -

IBM Informix Dynamic Server Version 12.10.UC4DE -- On-Line -- Up 1 days 09:34:46 -- 58544 Kbytes

informix@beaglebone:~$
```

Beaglebone – OAT

Run OAT against Beaglebone instance

Configure/monitor Informix from a desktop



Advanced DataTools

Informix Install - Docker Container

- Informix is available as a Docker Container for simple setup

  https://registry.hub.docker.com/u/ibmcom/informix-innovator-c/

- Innovator-C Edition
  - Free to use
  - Limited to 1-core, 2 GB memory
- Informix database products:

  http://www-03.ibm.com/software/products/en/informix-family

Now we have…

- Arduino reading sensor data
- Beaglebone, ARM-based, Linux computer, running an Informix database

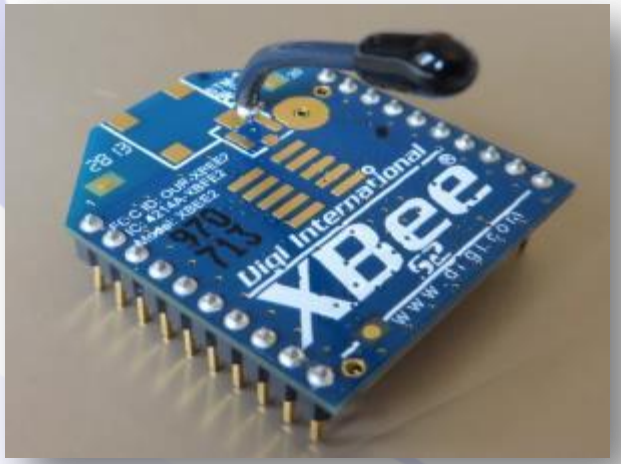
- Want to ***wirelessly*** connect the Arduino to the hub

Send/Receive Information Wirelessly

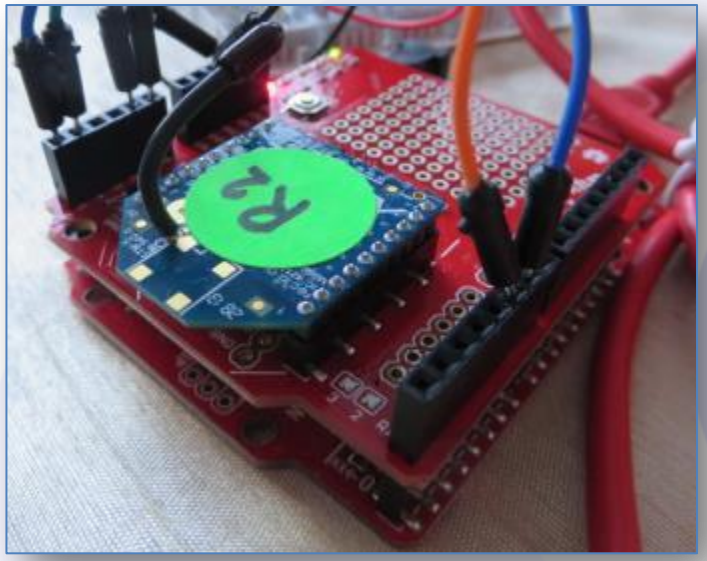
- Many options:
  - WiFi/Low Power WiFi
  - Cellular
  - Bluetooth Low Energy (Bluetooth Smart)
  - Other wireless protocols
    - ZigBee
    - Z-Wave

Send/Receive Information Wirelessly

- ZigBee
  - IEEE 802.15.4 Standard
  - Personal Area Network
  - Low power
  - Mesh Capable
- Zigbee implementation with XBee radios



XBee Radios

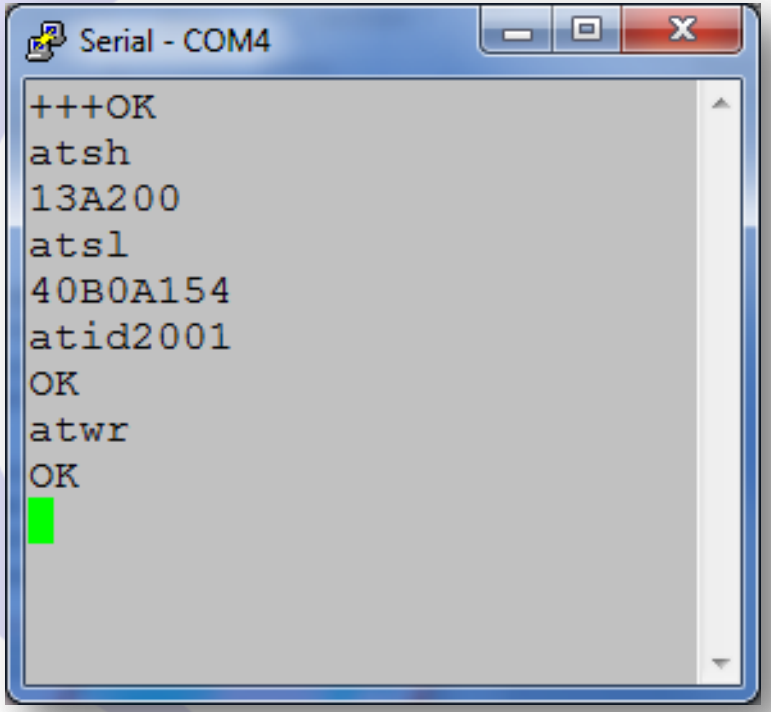


Need an XBee module on each sensor

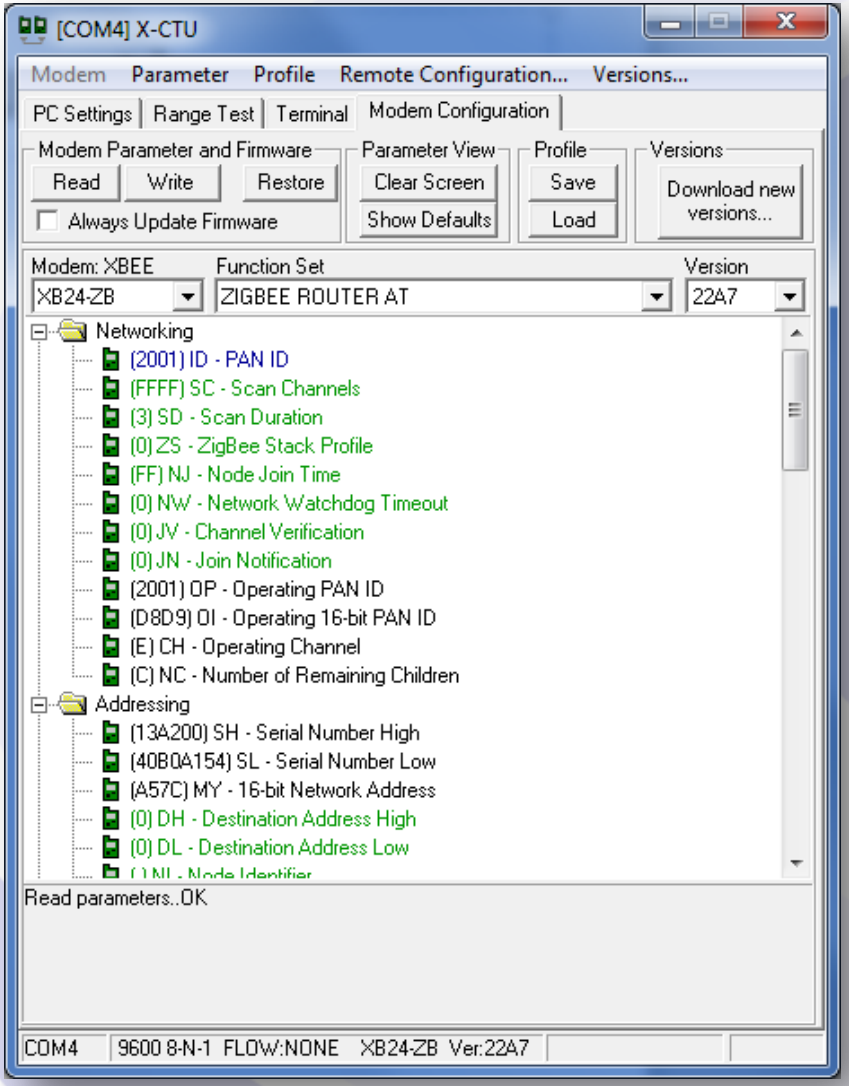
Need an XBee module on the hub

XBee Radios

- Configure the modules to talk to each other
  - Can use AT commands in a terminal window
  - Set a PAN ID (like a network name)
  - Simple to set up each XBee to echo output to the other (transparent/command mode)
  - Use API mode for more advanced applications



XBee Radios - Configuration



X-CTU

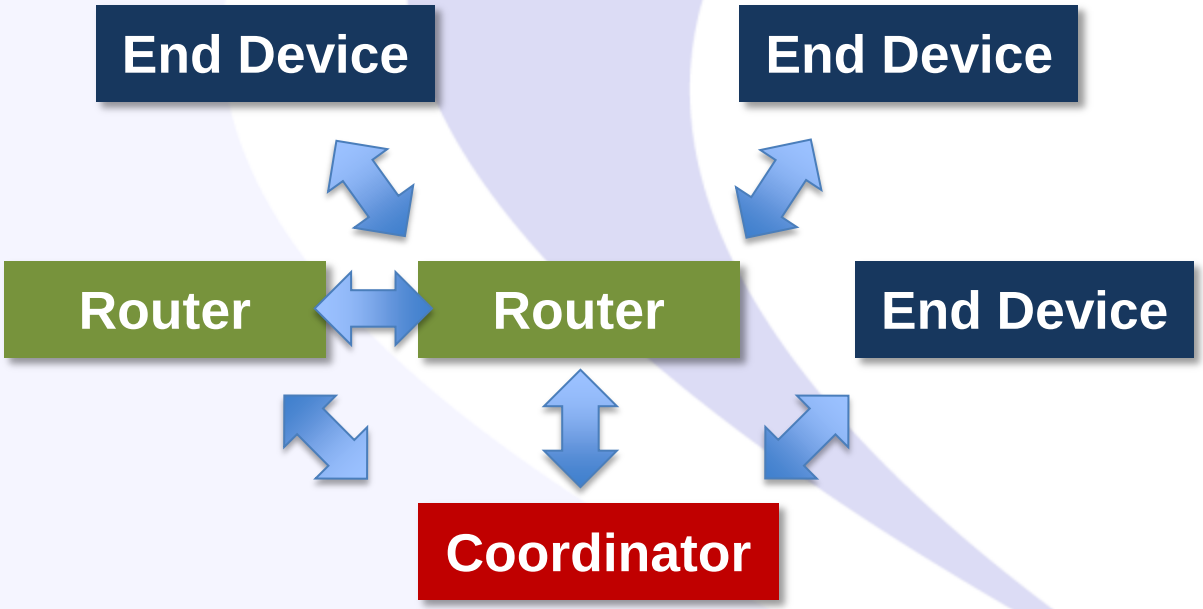
Interface to configure all the XBee parameters and basic troubleshooting

XBee Network

- Define an XBee as one of Coordinator, Router or End Point
  - **Coordinator:**
    - Destination/Root node [connected to the gateway]
  - **Router:**
    - Usually always on
    - Can connect to any other node
  - **End Device:**
    - May sleep – to save power
    - Can connect to any router or coordinator

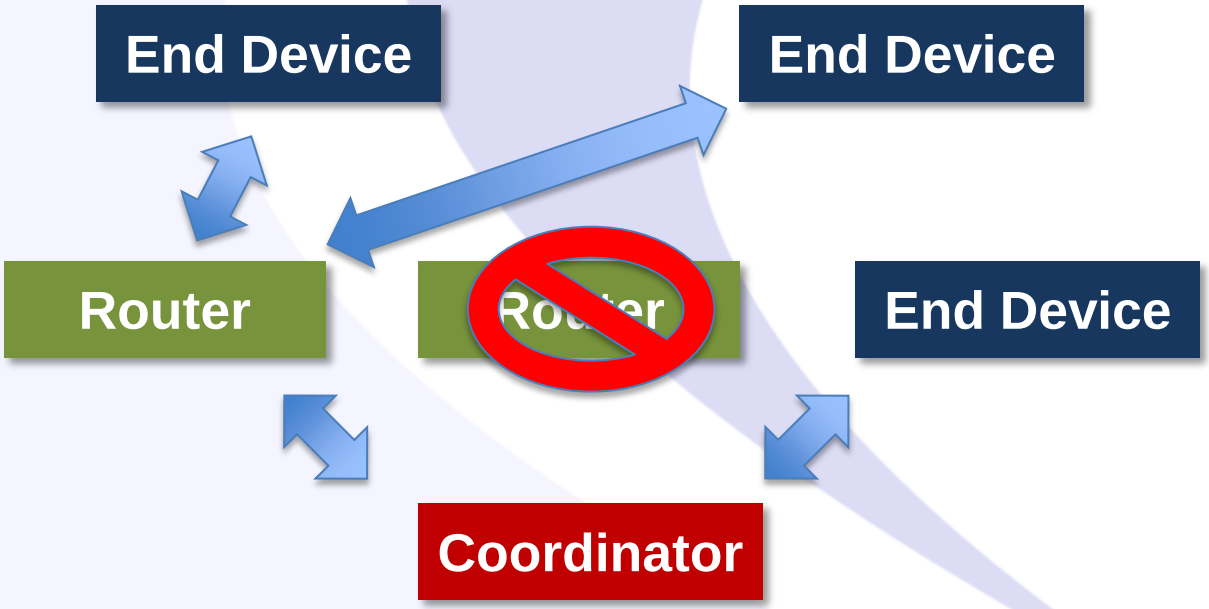
XBee Network

- Routers will pass on traffic from End Device, allowing End Points to be situated further away
- Routers will store messages for End Points and deliver when they wake up

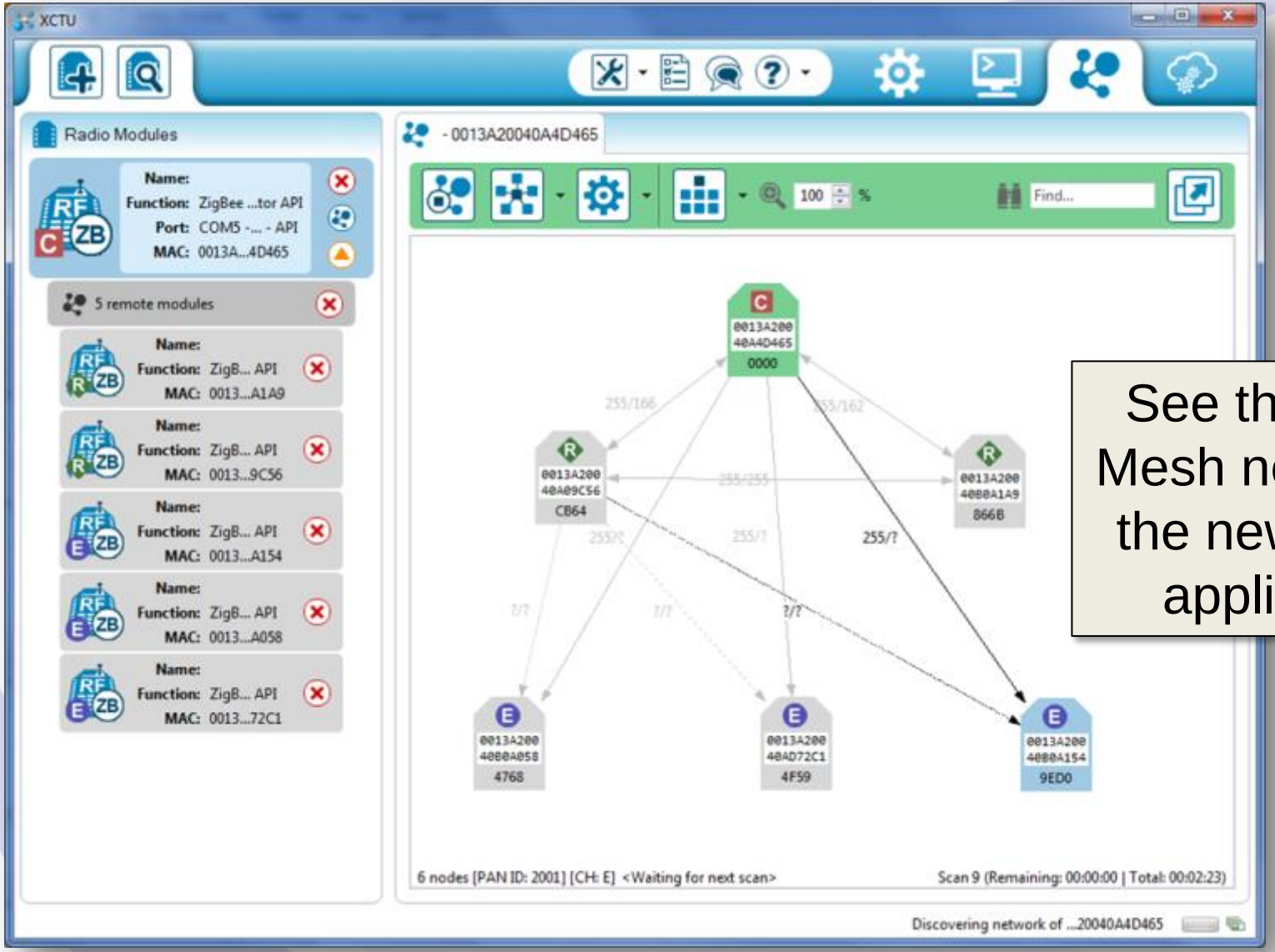


XBee Mesh Network

- XBees will automatically re-route traffic if nodes are unavailable
- All done without any additional programming

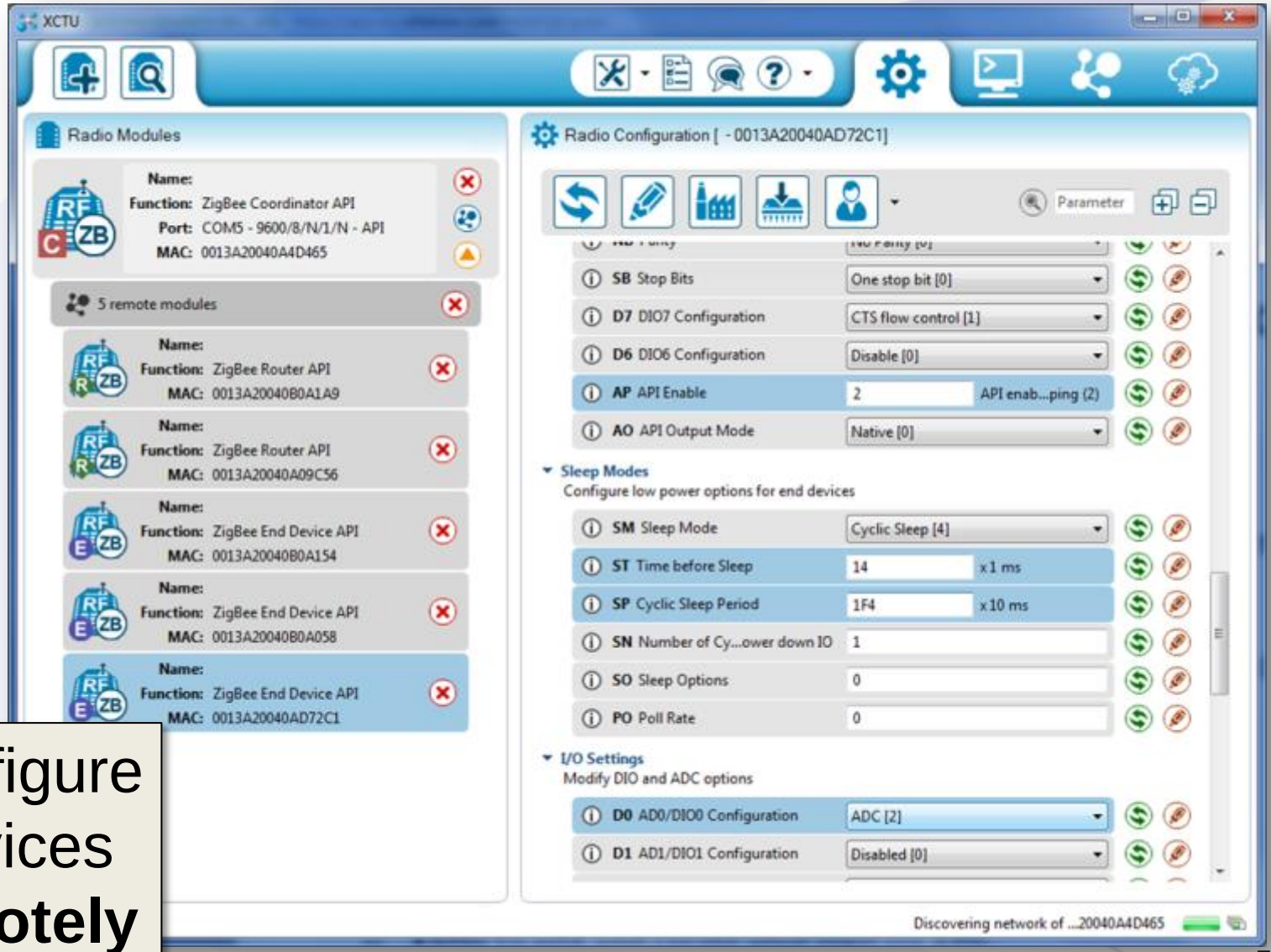


XBee Network, with the new XCTU



See the XBee Mesh network in the new XCTU application

XBee Network, with the new XCTU



Configure devices **remotely**

XBee Communication

Libraries for Arduino, python, C, etc to give programmatic control of the XBee

Arduino – Sensor (Send)

```
#include <XBee.h>
XBee xbee = XBee();
XBeeAddress64 addr64 = XBeeAddress64(0x00000000, 0x00000000);

Populate a “payload”:
ZBTxRequest zbTx = ZBTxRequest(addr64, payload, sizeof(payload));
xbee.send(zbTx);
```

XBee Communication

Python – Gateway (Receive)

```
from xbee import ZigBee
PORT = '/dev/ttyUSB0'
BAUD_RATE = 9600
ser = serial.Serial(PORT, BAUD_RATE)
xbee = ZigBee(ser,escaped=True)
…
response = xbee.wait_read_frame()
```

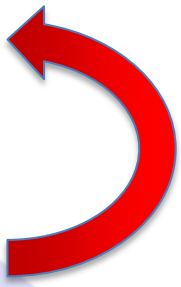
Parse response, e.g.

```
        sa = hex(response['source_addr_long'][4:])
```

XBees & Sending Simple Data

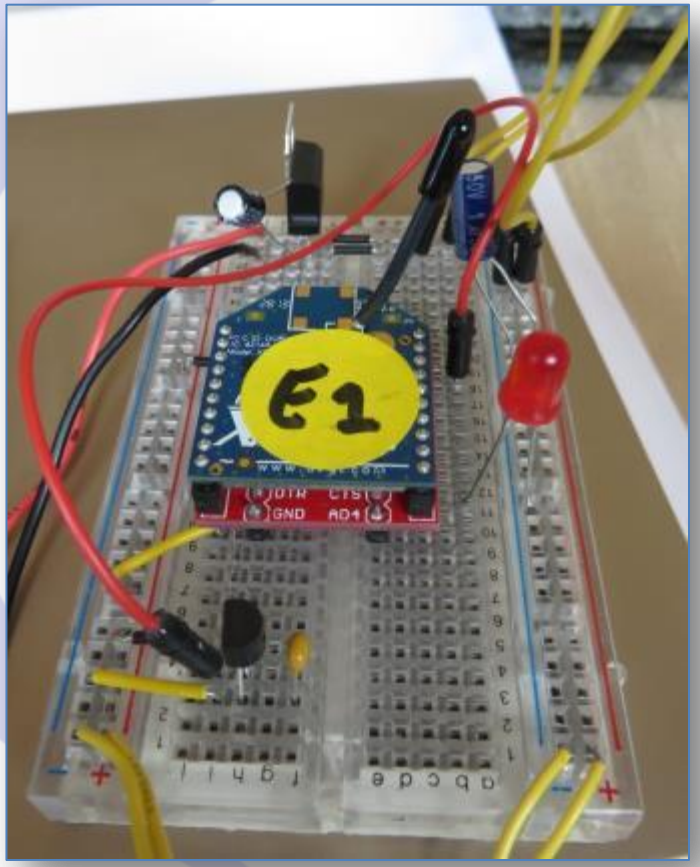
- XBee can be configured to send the analog value on a particular pin
  - Allows us just to connect it to a simple sensor and send the voltage reading back to base
- Can also configure it to sleep for a set period

Sleep
Wake Up
Send value on pin

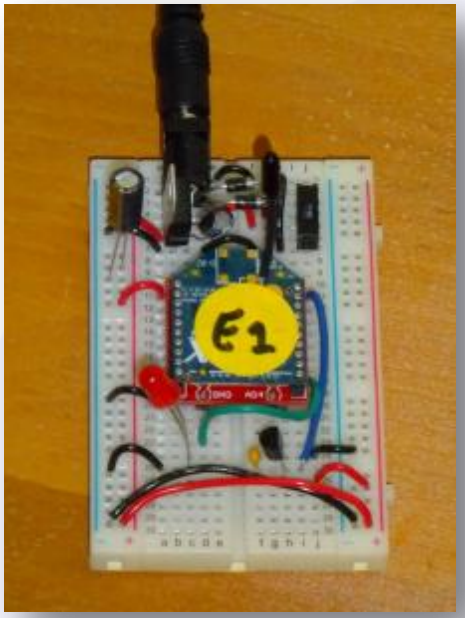


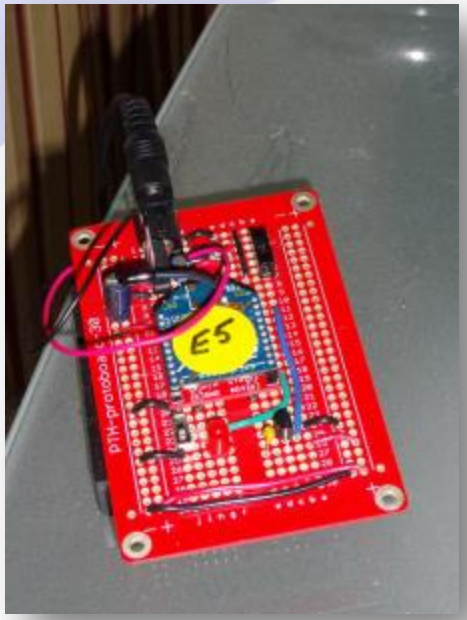
XBees & Sending Simple Data

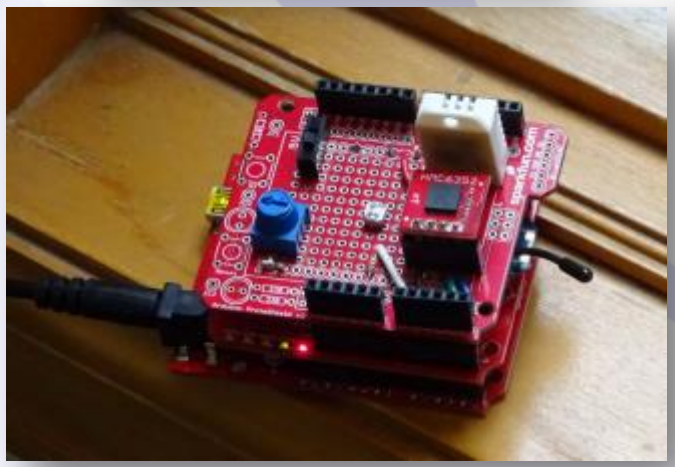
- The sleep allows us to save power
- Can run on batteries
- NOT always on – best for an End Device
- Simple setup – do not need an Arduino



Wireless Sensor Network

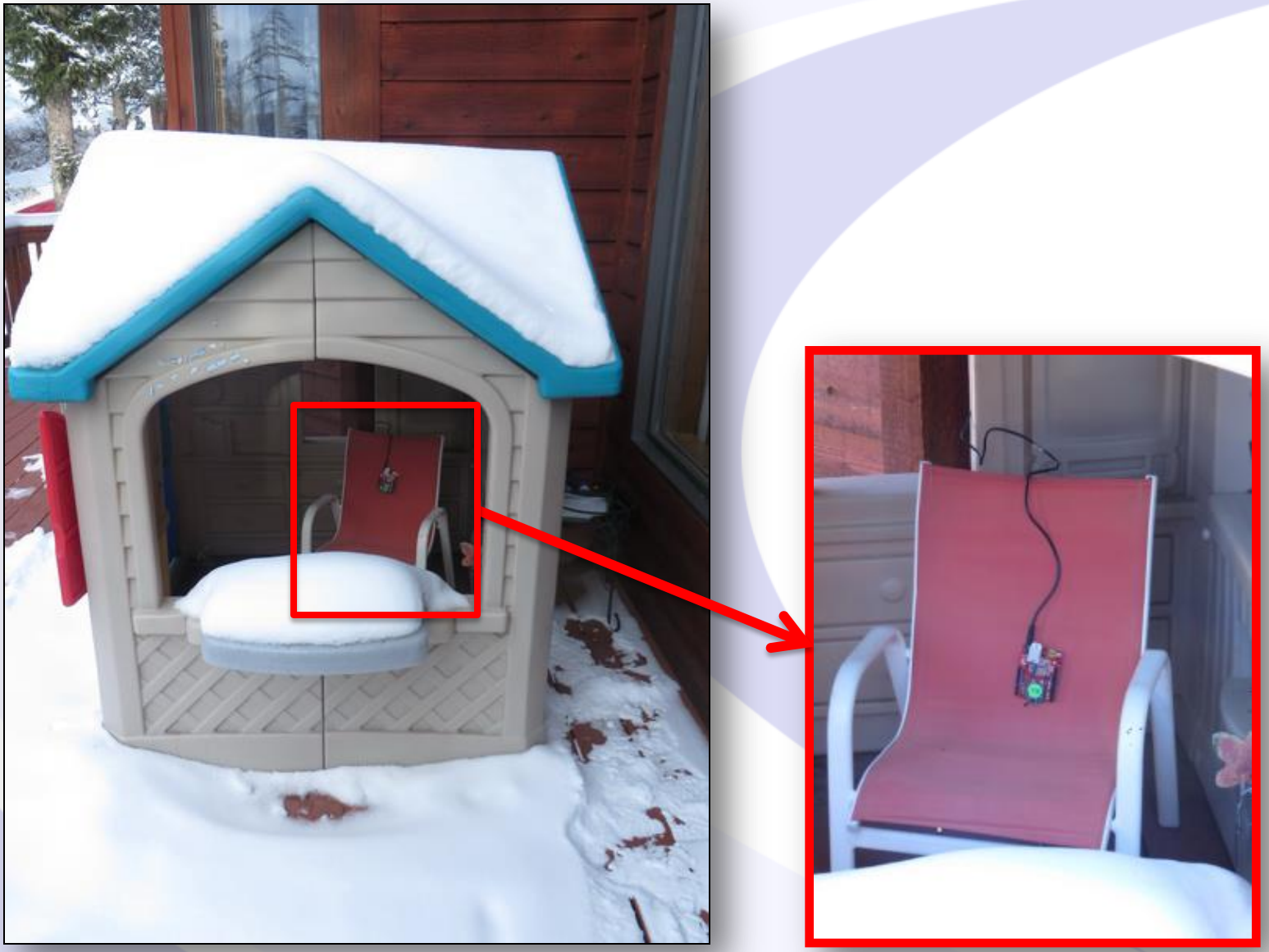






Advanced DataTools

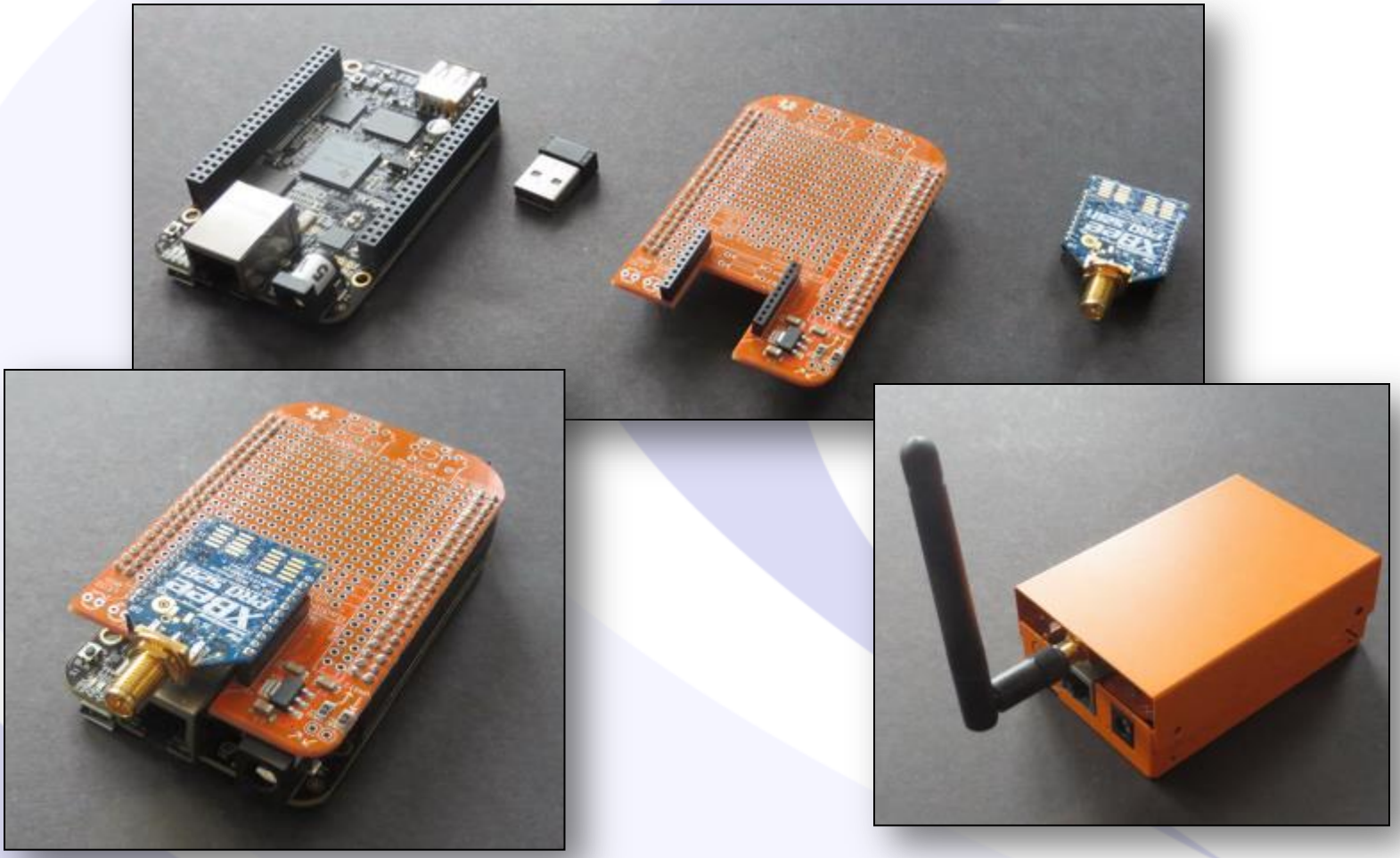
Wireless Sensor Network



Putting it together

Add a Coordinator XBee to the Beaglebone using a “cape”

(could also use USB or wire directly to the headers)



Receive Data from Remote Sensors

On the gateway (Beaglebone), read the data packets from the XBees

```
40b0a36b   2015-03-01 11:44:29  T: 66.2
40ad72c1   2015-03-01 11:44:30  T: 66.7
40a09c56   2015-03-01 11:44:31  T: 72.5  H: 24.6  L:  83  S: True   D:  61  Hdg:  72
40b0a058   2015-03-01 11:44:33  T: 66.2
40ad72c1   2015-03-01 11:44:35  T: 66.7
40a09c56   2015-03-01 11:44:36  T: 72.5  H: 24.5  L:  83  S: True   D:  61  Hdg:  72
40b79f8e   2015-03-01 11:44:36  T: 39.7  H: 39.8  L:  95  S: False  D:   0  Hdg:   0
40ad72c1   2015-03-01 11:44:40  T: 66.7
40a09c56   2015-03-01 11:44:40  T: 72.5  H: 24.3  L:  83  S: True   D:  61  Hdg:  72
40b0a058   2015-03-01 11:44:43  T: 66.0
40a09c56   2015-03-01 11:44:45  T: 72.5  H: 24.2  L:  83  S: True   D:  61  Hdg:  72
40ad72c1   2015-03-01 11:44:45  T: 66.7
40b79f8e   2015-03-01 11:44:46  T: 39.7  H: 39.9  L:  95  S: False  D:   0  Hdg:   0
```

Doing Something

- We now have data from multiple sensors streaming to our base station
  - Can trigger alerts – emails, texts, etc
  - Can trigger other actions…

Turning a Switch On/Off

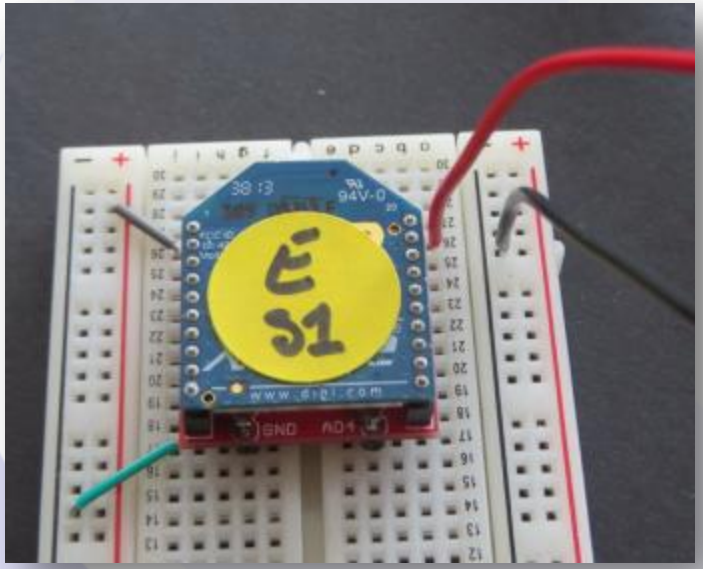
- Remotely set a pin on the XBee to Low/High

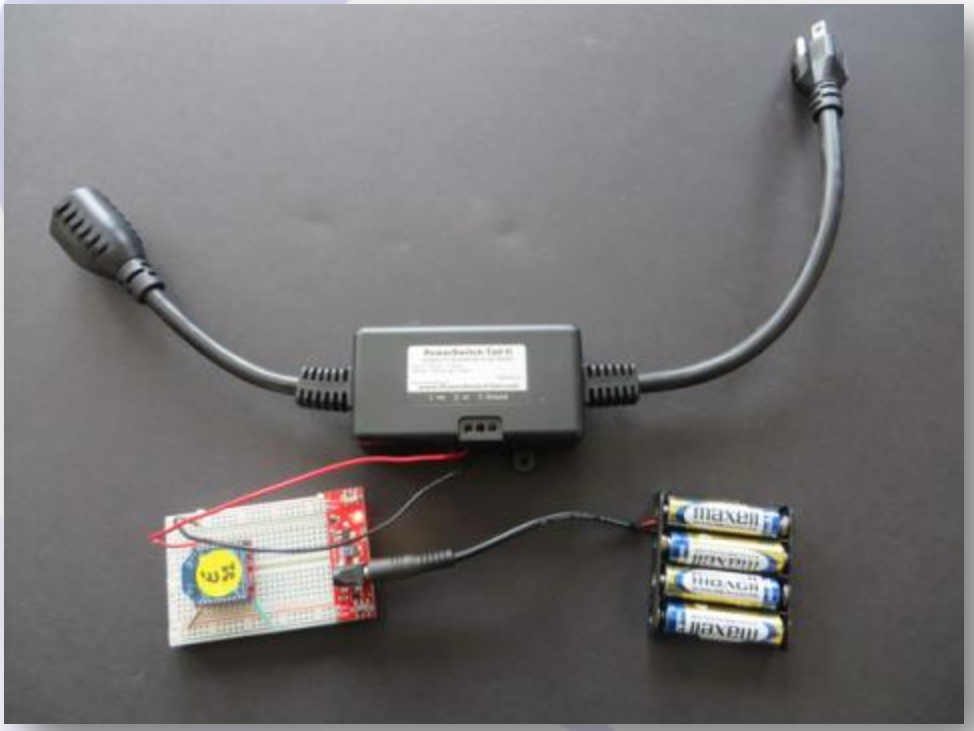
```
# D0=Digital Pin 0
# parameter: 4=Low, 5=High

xbee.remote_at(
dest_addr_long='\x00\x13\xa2\x00\x40\xb3\x42\xbb',
command='D0',
parameter='\x05')
```

- Use to control a relay to power a device

Turning a Switch On/Off

Pin high/low can control a relay to power a device





Now we have…

- Data from sensors (Arduino and “simple”)
- ARM based, Linux gateway, running Informix
- Sensor data sent/received wirelessly

- What are we going to do with this data?
  - Store it in the database…using TimeSeries
    - Keep a history
    - Values accessible from multiple applications
  - Push to the cloud
    - Access from anywhere

Next Webcast

Using Informix TimeSeries and the Internet of Things – June 2nd

- TimeSeries – what is it?
- How to set up a simple, regular TimeSeries
- Creating a TimeSeries for sensor data
- Virtual Table Interface and expression-based VTI
- Example of displaying sensor data on a web page
- Publishing sensor data to the cloud
- View data in the cloud
- Using APIs to pull data from the cloud
- Example of using APIs to display sensor data and graphs on a website

WAIUG

Next Meeting

Free Informix Technical Workshop New Features and Multi-tenancy – June 25 2015

Advanced DataTools

Questions?

Mike Walker

mike@advancedatatools.com