Python in the Real World: From Everyday Applications to Advanced Robotics

Jivitesh Singh Dhaliwal

PyCon India 2011

About Me

Electronics Enthusiast (To put it Mildly!)

- Final Year Electrical & Electronics Engineering Student- UIET, Panjab University, Chandigarh
- I love Python (Don’t we all?)
- I also like C, C++
- Open Source Advocate

Overview

I’m going to talk about:

- Microcontrollers: What they are, and what they can do for us
- Interfacing them with Python
- Using Python to design Intelligent Systems
- Robotics and Python
- Scope of Python in Robotics and Embedded Systems

The Era of Processing

From the 8 bit 8051 to 32 bit Cortex M3

What has changed:

– Increased Processing Power

– Richer Instruction Sets

– (Much) Faster Speeds

– Increased Program Memory

– Low Power Consumption

What This Means

- Ability to Integrate Embedded Systems with High Level Languages
- Effective Designing of Intelligent Machines
- More Efficient Interfacing between Hardware and Software
- Much more proficient Hardware

If you can Imagine something, you can Create it…

Interfacing With Python

The Two Techniques:

- Serially:
  - » PySerial
  - » p14p (Python-On-A-Chip)

- Over the Internet:
  - » Python RPC over http (using Python’s inbuilt Library)

The Power of Serial

```
>>> import serial
>>> serdev = '/dev/ttyACM0‘
>>> s = serial.Serial(serdev)
>>> s.write("hello")
>>> s.close()
```

As simple as that!

Python-On-A-Chip

- p14p:
  - A flyweight python Virtual Machine
    - Python in a Billion Places!
- The Virtual Machine
  - Code

```
ipm> import mbed
ipm> pwm21 = mbed.PwmOut(21)
ipm> pwm21.period_us(1000)
ipm> pwm21.pulsewidth_us(500)
```

Over The Internet

An Example:

```
>>> from mbedrpc2 import *
>>> mbed = HTTPRPC("192.168.0.4")
>>> x = DigitalOut(mbed,LED1)
>>> x.write(1)
```

import pylab as awesomeness

Assigning Meaning to Data Acquired

» Interactive plotting

» Ability to call mathematical functions

» Complex data management

The Potential of Scipy, Numpy and Matplotlib

- Plotting:
  - » 2D plotting of given signals
- Mathematical Manipulation of data :
  - » Linear Algebra Routines
  - » Matrix Operations
  - » Integration, Differentiation packages
  - » Fourier Transformation
  - » Optimization
  - » Signal Processing

A Few Examples

1. Making LEDs Blink
2. Inputting data from sensors
3. Plotting of Data
4. Manipulation of data using scipy, numpy and matplotlib
5. Optimization of existing embedded systems- some examples

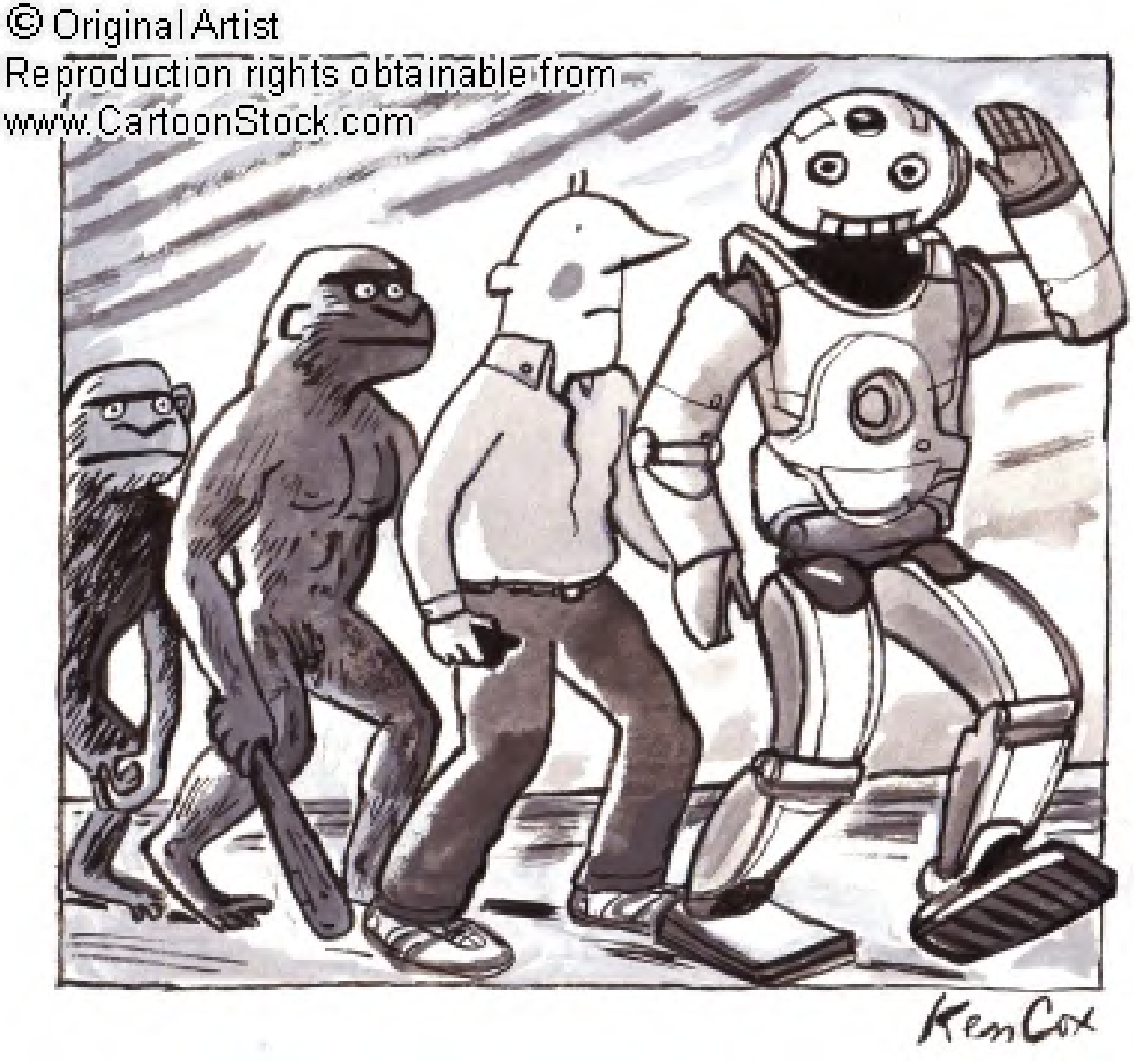
Summary So Far

We know now:

- What Microcontrollers are
- How to Interface them with Python
- How to use scipy, numpy and matplotlib to interpret data and take decisions

Robotics

The ultimate goal:



What are Robots?

Defining Robots:

"The technology developed to combine software, mechanical manipulators, sensors, controllers and computers to provide programmable automation."

- Essentially, cluster of multiple smaller embedded applications

Python and Robotics

- Python as the brain of a robot:
  - Complete in terms of Scientific tools available
  - Extremely Intuitive and Simple Syntax
- Microcontrollers as the External Interface to Python
  - Ability to Interface with Python
  - Ability to ‘Perceive’ External Data
  - Take Decisions based on Environment through Python

Making a Simple Robot using Python

(Demonstration)

Born With The Brains

How to Train Your Robot!

- Machine learning
  - The PyML Module
- Neural Networks
  - The PyBrain Module
- Support Vector Machines
  - PySVM Module

Implications

- Advanced Intelligent Robotics Systems
  - Support Vector Machine/ Neural Networks/ Other Pattern Recognition Modules available for python
  - Use of Prolog, Lisp within Python for logic reasoning
- Highly Optimized Results
  - Scipy, Numpy and Matplotlib Support
- Simpler Interface for the Programmer
  - The way you think is the way you write code!
  - The Pylab interface to know what’s going on visually

Where Python Lags

1. Slower Execution Cycles
2. Requires more Program Memory
3. Modules such as scipy, etc. not ported onto the chip (memory constrictions)

How Python Makes Up

- Cython!
  - Same code, convertable to C!
  - Faster Execution Cycle
- Use in More advanced Microcontrollers:
  - Have greater memory space

Scope of Python In Embedded Applications

- Demand and not supply= Scope :)
- Untapped Market
- Need for more aggressive development

Thank You

Find me if you want to:

- Talk about Embedded Systems and/ or Robotics
- Talk about Python
- Or a Combination of the Two!

Jivitesh Singh Dhaliwal

jiviteshdhaliwal@gmail.com