MicroPython for the IoT
16th Fribourg Linux Seminar - 2 may 2019

Jacques Supcik • @supcik
jacques.supcik@hefr.ch
Computers vs Embedded Systems
Typical 19” Rack Server
“Big” embedded systems
Traditional “small” embedded systems
Small embedded system with “mini” OS

C / C++

Application Software

Operating System

Hardware

User
Small embedded system with MicroPython
Why Python?
C was good in 1972

Computers changed a lot in 47 years
PiDP-11

Turn a Raspberry Pi into a blinktastic classic 1970s computer? PJ Evans puts on his Paisley shirt and heats his soldering iron

SPECS

DIMENSIONS: 170x245cm

MODELS: PDP-11/70

ARCHITECTURE: ICL 1963

DISK: DISK-100 Plus

BLINKENLIGHTS: 16

Remarkable replica

For this new kit, a painstaking process has resulted in an injection-moulded replica of the original PiDP-11’s case. If not for the one-third scale, you would struggle to tell it apart from the real thing. A perfect facsimile, this custom-built switchgear completes the package. You even get a key and lock, just like the real thing.

Once built, the PiDP-11 PCB comprises 64 LEDs, two rotary encoders, and an array of switches that connect to the Pi’s GPIO. Running a special version of the Simul emulator, the Pi accurately handles input and output from the panel. You can hook up a screen if you wish, use SSH, or go old-school and implement RS-232. The back panel is provided with different cut-outs to suit your cabling.

Digital-it-yourself

The PiDP-11 is supplied in kit form and there’s a lot to do. You’ll need to have some experience in soldering to put this together, the focus being on accurately fitting the switches and LEDs. This is tricky, but Oscar has provided tips that make the

An essential purchase for anyone with an interest in computing history

alignment of all these components much easier than with the PiDP-8. The instructions are in an alpha stage, but they are clear and the switch section is especially detailed. It took us about five hours to complete.

Full instructions are provided on how to prepare the Pi for its new career in 1970s computing. At the time of publication, a one-stop SD card image should be available. Otherwise, there are a few hoops to jump through, but nothing too arcane and the steps are well explained.

Once you log in, you’re straight into the PiDP-11’s operating system, an early form of UNIX. A number of alternative OSes are available, with more promised soon. You can switch back to Raspbian any time you like. In fact, as Simul doesn’t put a lot of strain on the Pi, it is unlikely to struggle with other server tasks. As a result, many users have their PiDPs doubling as file or media servers. These kits are a labour of love for Oscar and the attention to detail shines through, from the quality of the casing to the extensive labelling on the PCB. You may find the price high, but the quality is there to match. An essential purchase for anyone with an interest in computing history.
Guido van Rossum

*Benevolent Dictator For Life (BDFL)*

“Computer Programming for Everybody”

- An easy and intuitive language just as powerful as major competitors
- Open source, so anyone can contribute to its development
- Code that is as understandable as plain English
- Suitability for everyday tasks, allowing for short development times
Improved Programmer’s Productivity
Extensive Support Libraries
Easy to learn

High Level

Object-oriented
Extensible (in C/C++)

Community
Portable
Python core library
Python is almost 30 years old

Implementation started
1 Dec, 1989

Python 0.9
20 Feb, 1991

Python 1.0
26 Jan, 1994

Python 2.0
16 Oct, 2000

Python 3.0
3 Dec, 2008

MicroPython
3 May, 2014

Today
2 May, 2019

31 Dec, 2019

Python 2.7 EOL
From Python 2 to Python 3

Python 2.7

```python
>>> print "Hello"
Hello

>>> 5/2
2

>>> map(lambda x: 2**x, range(6))
[1, 2, 4, 8, 16, 32]

>>> [round(i+0.5) for i in range(6)]
[1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
```

Python 3.7

```python
>>> print "Hello"
SyntaxError: Missing parentheses in call to 'print'. Did you mean print("Hello")?

>>> 5/2
2.5

>>> map(lambda x: 2**x, range(6))
<map object at 0x1026d34a8>

>>> [round(i+0.5) for i in range(6)]
[0, 2, 2, 4, 4, 6]
```
Python 2 EOL in 2020 and Google “Grumpy”

Python 2.7 to Go source transpiler and runtime

https://github.com/google/grumpy
Python is one of the most loved language in 2018

<table>
<thead>
<tr>
<th>Loved</th>
<th>Dreaded</th>
<th>Wanted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rust</td>
<td>78.9%</td>
<td></td>
</tr>
<tr>
<td>Kotlin</td>
<td>75.1%</td>
<td></td>
</tr>
<tr>
<td>Python</td>
<td>68.0%</td>
<td></td>
</tr>
<tr>
<td>TypeScript</td>
<td>67.0%</td>
<td></td>
</tr>
<tr>
<td>Go</td>
<td>65.6%</td>
<td></td>
</tr>
</tbody>
</table>

https://insights.stackoverflow.com/survey/2018
and Python is **the** most wanted language

<table>
<thead>
<tr>
<th>Loved</th>
<th>Dreaded</th>
<th>Wanted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>Python</strong> 25.1%</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>JavaScript</strong> 19.0%</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Go</strong> 16.2%</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Kotlin</strong> 12.4%</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TypeScript</strong> 11.9%</td>
</tr>
</tbody>
</table>

Top Companies using Python

[Image of various company logos]

https://data-flair.training/blogs/python-career-opportunities/
MicroPython
MicroPython

“it is compact enough to fit and run within just 256k of code space and 16k of RAM”
MicroPython Code Size

< 80KB

http://micropython.org/resources/code-dashboard/
MicroPython

Created in 2013 by Damien George, Australian programmer and physicist
Kickstarter 2013: PyBoard + MicroPython

```python
# Add your Python code here. E.g.
from microbit import *
import music

notes = [
    'c4:1', 'e', 'g', 'c5', 'e5', 'g4', 'c5', 'e5', 'c4', 'e'
]

while True:
    display.scroll('Hello, World!')
    display.show(Image.HEART)
    sleep(2000)
    music.play(notes)
```

nRF51822, 16 MHz ARM Cortex-M0
Micro:bit Educational Foundation

“We enable and inspire all children to participate in the digital world, with particular focus on girls and those from disadvantaged groups”
BBC Acorn (Cambridge / UK)

1981 : BBC Micro. 6502 CPU

1987 : Archimedes. ARM CPU
QUALIFICATION OF MICROPYTHON VM
AND OBCP ENGINE
PYTHON IN SPACE

ESTEC – TEC-SW Final Presentation Day
Dec 11th 2018
Runs on Cortex M3 and Tensilica Xtensa CPUs
ESP8266

- Wifi 802.11 b/g/n
- 4MB Flash
- 32 KiB instruction RAM
- 32 KiB instruction cache RAM
- 80 KiB user-data RAM

US$ 5
ESP32

- Wifi 802.11 b/g/n
- Bluetooth v4.2
- BR/EDR(*) and BLE(*)
- Dual Core
- 4MB Flash
- 520 KiB SRAM

*) BR = Basic Rate, EDR = Enhanced Data Rate, BLE = Bluetooth Low Energy

US$ 8
PyBoard

- STM32F405RG
- 168 MHz Cortex M4 with hardware floating point
- 1MiB flash ROM and 192KiB RAM
- 3-axis accelerometer

US$ 44
PYBD-SFxW
pyboard D-series
micropython.org

US$ 58 - 90
PyBoard (clone)

- STM32F405RG
- 168 MHz Cortex M4 with hardware floating point
- 1MiB flash ROM and 192KiB RAM
- 3-axis accelerometer

US$ 17
Digi XBee3® Cellular LTE CAT 1

“Integrated MicroPython programmability for edge compute”

US$ 100

Edge computing

Pycom LoPy

MicroPython enabled development board (LoRa, WiFi, Bluetooth)

US$ 30

MicroPython on bare metal Raspberry Pi Zero

https://github.com/boochow/micropython-raspberrypi
MicroPython on servers and desktop

https://github.com/micropython/micropython/wiki/Getting-Started
Commercial support: Zerynth

The Middleware for IoT

https://www.zerynth.com/
Live Coding
Simple LED Blinker

https://gitlab.forge.hefr.ch/fribourg-linux-seminar/19.05/py-blink
Remote temperature sensor

https://gitlab.forge.hefr.ch/fribourg-linux-seminar/19.05/py-temperature
Remote temperature sensor and LED
“Sonar” to measure distances

https://gitlab.forge.hefr.ch/fribourg-linux-seminar/19.05/py-sonar
“Sonar” to measure distance
References

- https://micropython.org/
- https://www.python.org/
- GOTO 2016 • MicroPython & the Internet of Things by Damien George
- 35C3 - MicroPython – Python for Microcontrollers
- Pycon Ireland 2017: MicroPython: The Next Step to World Domination by Steve Holden
- https://io.adafruit.com
- MicroPython for Leon (ESA)
- Writing fast and efficient MicroPython
- The Early Days of MicroPython
Backup
Abstract

According to the 2018 survey of Stack Overflow, Python is one of the most loved programming languages and the most wanted one. Python is used in many world-class software companies as well as in high schools, colleges and universities. MicroPython is an implementation of the Python 3 programming language optimized for embedded systems. During this presentation, I will present MicroPython and its derivatives, and I will show you some typical use cases for MicroPython. In the second part, I will demonstrate how to install the system on a cheap microcontroller and during a live coding session, I will implement a functional IoT device.