

DesignNews

MicroPython Embedded Applications

DAY 1 : BBC Micro:bit Overview: MicroPython Hello World

Sponsored by









Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click "Help" or submit a question asking for assistance.
- Participate in 'Group Chat' by maximizing the chat widget in your dock.









Don Wilcher

Visit 'Lecturer Profile' in your console for more details.



Course Kit: Keyestudio 37 in 1 Starter Kit with BBC micro:bit









- What is a BBC micro:bit
- BBC micro:bit Architecture
- Introduction MicroPython
- Educational Technology Research Literature
- Lab Activities
 - a) Programming Environments:
 - i. Mu
 - ii. Read-Eval-Print-Loop (REPL)
 - b) Cyber-Pet: ver.1
 - c) Cyber-Pet: ver.2







What is a BBC micro:bit?



The BBC micro:bit is a pocket-sized computer that:

- Introduces software and hardware integration.
- Has an LED 5x5 matrix display.
- Include three tactile pushbutton switches.
- Has multiple electronic sensors.
- Has many input/output features.

Source: https://microbit.org/get-started/first-steps/introduction/



What is a BBC micro:bit?...

BBC micro:bit current version







Source: https://tech.microbit.org/latest-revision/#using-the-new-features-in-python





Question 1

The BBC micro:bit has a LED matrix display?





What is a BBC micro:bit?



The new micro:bit (version 2) includes

- A built in Micro-Electro-Mechanical System (MEM)s microphone.
- A LED Indicator.
- A speaker.
- Extra tactile pushbutton switch.
- A power button switch.

Source: <u>https://microbit.org/get-started/first-steps/introduction/</u>



What is a BBC micro:bit?

Feature comparison

Overset (ed. E)		Lateral (19)
Current (v1.5)	Feature	Latest (V2)
Nordic Semiconductor nRF51822	Processor	Nordic Semiconductor nRF52833
256kB Flash 16kB RAM	Memory	512kB Flash, 128kB RAM
NXP KL26Z, 16kB RAM	Interface	NXP KL27Z, 32kB RAM
N/A	Microphone	MEMs microphone and LED indicator
N/A	Speaker	On board speaker
N/A	Logo touch	Touch sensitive logo pin
25 pins. 3 dedicated GPIO, PWM, i2c, SPI and ext. power. 3 ring pins for connecting crocodile clips/banana plugs.	Edge Connector	25 pins. 4 dedicated GPIO, PWM, i2c, SPI and ext. power. 3 ring pins for connecting crocodile clips/banana plugs. Notched for easier connection
Shared I2C Bus	12C	Dedicated I2C bus for peripherals
2.4Ghz Micro:bit Radio/BLE Bluetooth 4.0	Wireless	2.4Ghz Micro:bit Radio/BLE Bluetooth 5.0
5V via Micro USB port, 3V via edge connector or battery pack	Power	5V via Micro USB port, 3V via edge connector or battery pack, LED power indicator, Power off (push and hold power button)
90mA available for accessories	Available current	190mA available for accessories
ST LSM 303	Motion sensor	ST LSM 303
C++, MakeCode, Python, Scratch	Software	C++, MakeCode, Python, Scratch
5cm(w) x 4cm(h)	Size	5cm(w) x 4cm(h)

Source: <u>https://tech.microbit.org/latest-revision/#using-the-new-features-in-python</u>



Question 2



Using slide 10 what components are added to version 2?





What is a BBC micro:bit?...

BBC micro:bit version 2



Source: <u>https://tech.microbit.org/latest-revision/#using-the-new-features-in-python</u>







Source: https://tech.microbit.org/hardware/#hardware-block-diagram



Reviewing slide 13, write down the part number for the USB interface?













BBC micro:bit Architecture...

Microcontroller



	Features
item	details
Model	Nordic nRF52833
Core variant	Arm Cortex-M4 32 bit processor with FPU
Flash ROM	512KB
RAM	128KB
Speed	64MHz
Debug	SWD, J-Link/OB
More Info	Software, NRF52 datasheet

16





0

BBC micro:bit Architecture...

item	details	Features	
Stack	Bluetooth 5.1 with	Bluetooth Low Energy(BLE)	
Band	2. <mark>4</mark> GHz ISM (Indu	strial, Scientific and Medical) 2.4GHz2.41GHz	
Channels	50 2MHz channel	s, on <mark>ly 40 used (0 to 39), 3 ad</mark> vertising channels (37,38,39)	
Sensitivity	-93dBm in Blueto	oth low energy mode	
Tx Power	-40dBm to 4dBm		
Role	GAP Peripheral &	GAP Central	
Congestion avoidance	Adaptive Frequer	cy Hopping	
Profiles	BBC micro:bit pro	ofile	
More Info	Bluetooth		





BBC micro:bit Architecture...

item	details	Features	
Protocol	Micro:bit Ra	dio	
Freq band	2.4GHz		
Channel rate	1Mbps or 2Mbps		
Encryption	None		
Channels	80 (080)		
Group codes	255		
Tx power	Eight user co	onfigurable settings from 0(-30dbm) to 7 (+4dbm)	
Payload size	32 (standard) 255 (if reconfigured)	
More Info	Micro:bit Ra	dio	



Introduction to MicroPython

MicroPython is a small and efficient implementation of:

- Python 3 programming language.
- A subset of the Python standard library.
- An optimized Python standard library

Additional Notes:



- MicroPython can run on microcontrollers.
- Can run in constrained environments.



Sponsored By



Introduction to MicroPython...

MicroPython is a small and efficient implementation of:

- Python 3 programming language.
- A subset of the Python standard library.
- An optimized Python standard library

Additional Notes:



- MicroPython can run on microcontrollers.
- Can run in constrained environments.

Source: https://micropython.org/





Introduction to MicroPython...

Creator of MicroPython





Damien P. George PhD



Additional Notes:

- Initial Release Date: May 3, 2014.
- Stable Release Date: September 2, 2020.
- Sources: https://en.wikipedia.org/wiki/MicroPython

https://micropython.org/



Question 4



Who is the creator of MicroPython?



Educational Technology Research Literature (ETRL):

Creating MicroPython based micro:bit prototypes

The ETRL will allow gems of product creation to be discovered.

- The ETRL will uncover historical perspectives of the target product.
- The ETRL will provide product technical descriptions.
- The ETRL will provide product equivalent comparisons.
- The ETRL will provide future insights on emerging technologies.



Sponsored By



Educational Technology Research Literature (ETRL): . . . Creating MicroPython based micro:bit prototypes



BBC micro:bit **INSPIRES A NEW GENERATION** (Continued)



Sources: Hodges, S., Sentence, S., Finney, J., & Bell, T. (2020). *Physical computing: A key element of modern computer science education*. https://www.microsoft.com/en-us/research/uploads/prod/2020/04/physical-²⁴ computing.pdf





Educational Technology Research Literature (ETRL): ...

Creating MicroPython based micro:bit prototypes



	TABLE 1. Com categorizatior	mercially available n builds on that prop	ohysical computing products. This posed by Przybylla and Romeike. ¹³
Categorization	Type of product	Examples	
1. Packaged electronics; no programing	Kits of packaged components and modules	Snap Circuits, basic LittleBits, Circuit Stickers	
2. Packaged programmable products (not boards);	Robot turtles	Sphero, Ozobot, Kibo, Dash and Dot, BeeBot, Cubetto	A 20000000 (2)
programmable via PC or phone; often battery-powered	Programmable construction sets	Lego WeDo, Lego Mindstorms, Pico Cricket, Vex Robotics	
 Board-level peripheral devices; need PC during use 	Integrated input/ output devices for PCs	Makey Makey, PicoBoard, BlinkM, Sense Board	
	Modular input/ output devices for PCs	Phidgets	

Sources: Hodges, S., Sentence, S., Finney, J., & Bell, T. (2020). *Physical computing: A key element of modern computer science education*. https://www.microsoft.com/en-us/research/uploads/prod/2020/04/physical-²⁵ computing.pdf



Educational Technology Research Literature (ETRL): . . . Creating MicroPython based micro:bit prototypes

Hodges et al. (2020) noted: "It is worth noting that the benefits of physical computing aren't limited to computer science education. There are diverse connections to other science, technology, engineering, and mathematics subjects"

Sources: Hodges, S., Sentence, S., Finney, J., & Bell, T. (2020). *Physical computing: A key element of modern computer science education*. https://www.microsoft.com/en-us/research/uploads/prod/2020/04/physical-²⁶ computing.pdf





Educational Technology Research Literature (ETRL): ...

micro:bit Timer Pneumatic Controller prototype





Sponsored By



Educational Technology Research Literature (ETRL): ...

micro:bit Timer Pneumatic Controller prototype



Video Source: https://youtu.be/0yrzmuqg1zY



Sponsored By



Lab Activities





What two environments can be used to program the BBC micro:bit?





Lab Activities What is Mu



Download About Tutorials How to ? Discuss Developers Language -

About Mu

Mu is a Python code editor for beginner programmers based on extensive feedback given by teachers and learners.







Lab Activities . . .

The Mu Code Editor needs to be installed!

Download About Tutorials How to .? Discuss Developers Language -

Download Mu

TRY THE ALPHA OF THE NEXT VERSION OF MU! Feeling brave? Don't mind reporting bugs? Enjoy giving feedback? Then we'd love you to take a sneak peak at the (unfinished work in progress) next version of Mu. These are unsigned installers:

If you're using Mu at EuroPython's beginners' day, this is the version you should install.

· Windows 32-bit

- Windows 64-bit
- Mac OSX

There are many ways to install Mu. The simplest is to download the official installer for Windows or Mac OSX. If you find you cannot install Mu because the computer you are using is locked down, you should try out Portal/U: a method of running Mu from a pendrive on Windows or OSX. You can also use Python's built-in psp tool. Some Linux distributions come with Mu packaged already (and you should use your OS's package manager to install it). Finally, if you're on Raspbian (the version of Linux for Raspberry Pi) you can install Mu as a package.

If you're a developer, you can find the source code on GitHub.



https://codewith.mu/en/download





Lab Activities

The Mu Code Editor installed





33



Lab Activities

micro:bit attached to laptop computer's USB port

 \sim



5 V

Size

1 KB

1 KB



3/22/2016 5:30 PM Text Document 3/22/2016 5:30 PM Chrome HTML Do...

Туре

micro:bit USB Drive

Date modified



Lab Activities Read-Eval-Loop-Print (REPL)



According to pythonprogramminglanguage:

REPL is the language shell, **the Python Interactive Shell**. The REPL acronym is short for Read, Eval, Print and Loop.

The Python interactive interpreter can be used to easily check Python commands. To start the Python interpreter, type the command python without any parameter and hit the "return" key. The process is:

- 1.Read: take user input.
- **2. Eval:** evaluate the input.
- **3. Print:** shows the output to the user.
- 4. Loop: repeat.









Lab Activities

Type "help()" for more information. >>> help() Welcome to MicroPython on the micro:bit!

Coding with REPL:

Type help()

Try these commands: display.scroll('Hello') running_time() sleep(1000) button_a.is_pressed() What do these commands do? Can you improve them? HINT: use the up and down arrow keys to get your command history. Press the TAB key to auto-complete unfinished words (so 'di' becomes 'display' after you press TAB). These tricks save a lot of typing and look cool!









Lab Activities

Coding with REPL:





Lab Activities

Sponsored By



Coding with REPL: Button_a.is_pressed()



BBC_micro:bitREPL
>>> running_time()
805563
>>> button_a.is_pressed()
False
>>> button_a.is_pressed()
True
>>>

button_a.is_pressed() > False — Button A not pressed	
button a.is pressed() > True _	
Button A press and hold	0 1 2 3V GND



Lab Activities



Coding with REPL:

Explore:

Type 'help(something)' to find out about it. Type 'dir(something)' to see what it can do. Type 'dir()' to see what stuff is available. For goodness sake, don't type 'import this'.





Lab Activities

Coding with REPL:

PL:

Type dir()
>>> dir()
['pin2', 'pin0', 'pin1', 'pin3', 'pin6', 'pin13', 'pin4', 'uart', 'pin5', 'pin7', 'temperature'
'sleep', 'pin8', 'pin9', 'button_a', 'button_b', 'reset', '__name__', 'i2c', 'pin11', 'pin10',
'spi', 'panic', 'Image', 'running_time', 'compass', 'pin12', 'pin14', 'pin15', 'accelerometer',
'display', 'pin16', 'pin19', 'pin20']
>>>

Example:

Type dir(pin0)

>> dir(pin0)
'write_digital', 'read_digital', 'write_analog', 'read_analog', 'set_analog_period',
<pre>set_analog_period_microseconds', 'is_touched', 'PULL_UP', 'PULL_DOWN', 'NO_PULL', 'get_pull',</pre>
set_pull', 'get_mode']
>>





Coding with MicroPython:

Lab Activities



1	from microbit import *
2	
3	while True:
4	<pre>if button_a.is_pressed ():</pre>
5	display.show(Image.HAPPY)
6	<pre>elif button_b.is_pressed():</pre>
7	break
8	else:
9	display.show(Image.SAD)
10	
11	display.clear()
12	

Cyber-pet v1.0



Cyber-pet Micropython code source: BBC micro:bit Micropython Documentation Release 1.01, pg 12.





Coding with MicroPython:

Lab Activities



from microbit import *
display.show('Cyber-pet')
while True:
 if button_a.is_pressed ():
 display.show(Image.RABBIT)
 elif button_b.is_pressed():
 break
 else:
 display.show(Image.TORTOISE)
display.clear()

Cyber-pet v2.0



Images: BBC micro:bit Micropython Documentation Release 1.01, pp 4-6.



Question 6



On slide 44, modify the MicroPython Code whereby the Cyber-Pet can be converted into a simple Coin Toss.







Thank you for attending

Please consider the resources below:

- BBC micro:bit architecture
 <u>https://tech.microbit.org/hardware/</u>
- BBC micro:bit <u>https://microbit.org/get-started/first-steps/introduction/</u>
- Hodges, S., Sentence, S., Finney, J., & Bell, T. (2020). *Physical computing: A key element of modern computer science education*. <u>https://www.microsoft.com/en-us/research/uploads/prod/2020/04/physical-computing.pdf</u>
- MicroPython <u>https://micropython.org/</u>
- Micro:bit Timer Pneumatic Controller video <u>https://youtu.be/0yrzmuqg1zY</u>
- Mu Code Editor <u>https://codewith.mu/en/download</u>



DesignNews

Thank You

Sponsored by



