

```
#!/usr/bin/env python
import RPi.GPIO as GPIO
import time

LED = 4

GPIO.setmode(GPIO.BCM)
GPIO.setup(LED, GPIO.OUT)

def writeBinary(data):
    for digit in data:
        if digit == '1':
            GPIO.output(LED, GPIO.HIGH)
            print "LED on"
        else:
            GPIO.output(LED, GPIO.LOW)
            print "LED off"
        time.sleep(0.5)

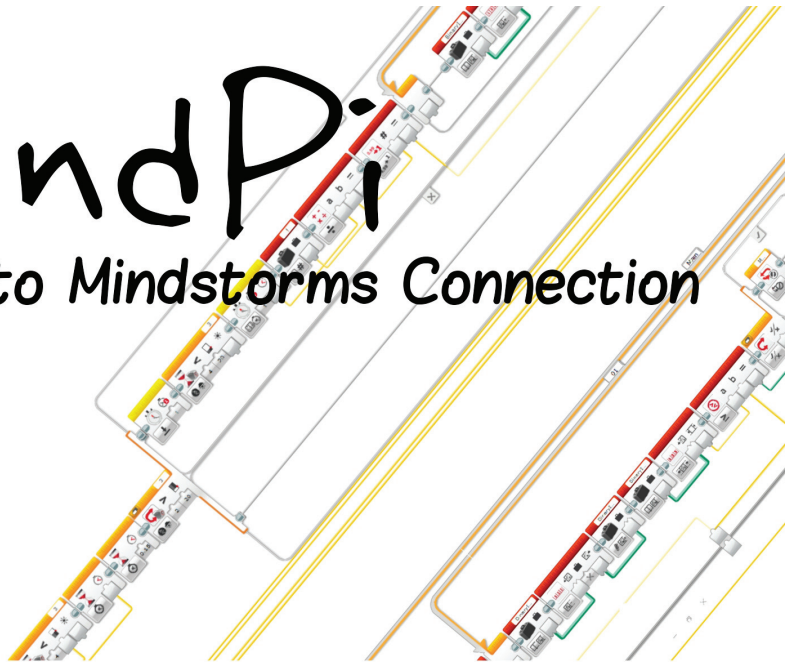
def main():
    data = "101100"
    writeBinary(data)

if __name__ == '__main__':
    main()

```

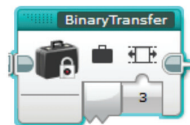
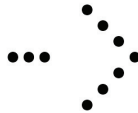
MindPi

Your Raspberry Pi to Mindstorms Connection



Por Boston Abrams

```
Com.Binary("101100")
```



LEGO® MINDSTORMS es una gran herramienta para poder construir y programar de forma fácil, pero esa conocida facilidad de uso de los productos de LEGO® significa que se limitan algunas funcionalidades, como un servidor web.

Seguro que dirás «¿Qué hay de BrickPi?» Bueno, BrickPi es otra manera de controlar motores y sensores de LEGO® MINDSTORMS con un Raspberry Pi (un pequeño ordenador hecho para los aficionados a la electrónica) mediante una placa que se conecta con él. Pero después de usar el BrickPi pensé que podría hacer algo diferente. Quería conseguir que el LEGO® MINDSTORMS EV3 se comunicara con el Raspberry Pi, no reemplazar el EV3, como hace el BrickPi. Además, otra ventaja de mi diseño es que si (como yo) ya tienes un EV3 y un Raspberry Pi, solamente te costará unos céntimos.

¿Cómo funciona? Con algo tan sencillo como un LED.

Como funciona:

Para conseguir que el EV3 y el Pi se comuniquen hace falta que puedan intercambiar información. Esto se logra conectando un LED rojo al Pi: la pata positiva se conecta con el pin GPIO del Raspberry Pi y la pata negativa a tierra. Ahora puedes colocar el LED dentro de una viga Technic y fijarlo delante del sensor de color (Imagen a la izquierda: Las piezas plateadas son las patas del LED que se conectarán al Pi) de modo que cuando parpadea el LED, el sensor de color del EV3 puede leer esa información. El programa que he escrito toma los datos y crea un patrón de destellos que el ladrillo MINDSTORMS lee con el sensor de color y convierte en datos.



Como ejemplo, imagina que quieres transmitir un número de rotaciones a un motor (un número de 1 a 10). Si por ejemplo quieres enviar el número 5 lo escribes en la función Com.Numeral(), que convertirá el número en binario y hará que el LED parpadee con el patrón correspondiente. (Un breve aparte: Elegí programar en Python porque viene preinstalado en el Raspberry Pi y ya tenía algunos conocimientos del lenguaje) El sensor de color del EV3 lee estos destellos con un «MiBloque» (lo que en EV3-G, el lenguaje de programación visual de LEGO® MINDSTORMS se llama una función) llamado NumeralTransfer que está esperando ese patrón en el ladrillo EV3. Cuando el mensaje ha llegado a su tamaño especificado, el bloque deja de grabar y convierte el número binario en decimal. En nuestro ejemplo sería 5. Con eso se concluye la transferencia de información de un sistema a otro. Ahora puedes usar este dato para hacer que el motor gire 5 rotaciones.

Esto en sí puede parecer poca cosa, pero si lo unes a las posibilidades del microordenador Raspberry Pi, se pueden conseguir cosas sorprendentes. Por ejemplo se puede transmitir información de una página web o hacer una estación de control con Pygame. (Pygame es una interfaz gráfica para hacer videojuegos en Python.)

Como ejemplo, una cosa que he hecho es hacer un robot que abre un cofre en el popular juego para teléfono Clash Royale. Metes algunos datos en una página web que se usan para mover un brazo que pulsa los botones en la pantalla del teléfono, abriendo así un cofre dentro del juego.

Algunos otros proyectos que se pueden construir con una interfaz similar: Un brazo robótico controlado por un servidor web, una aplicación para controlar un coche a distancia o una grúa controlada por un Pygame.

Ahora que MindPi funciona aún tengo algunos objetivos para el proyecto. Estos incluyen una interfaz que acepte letras, aumentar la velocidad de transmisión y conseguir una comunicación en ambas direcciones.

Para instrucciones detalladas y para probarlo tú mismo:

<https://github.com/BostonA/MindPi>

#

