<http://www.rasplay.org/?p=2049>

[***Home***](http://www.rasplay.org/)***»***[***산딸기 가지고 놀기***](http://www.rasplay.org/?cat=4)***»***[***산딸기 GPIO***](http://www.rasplay.org/?cat=256)***»***[***GPIO 기초강좌***](http://www.rasplay.org/?cat=291)***» 라즈베리파이 GPIO 강좌 : 01. GPIO 소개 및 핀배치***

# [라즈베리파이 GPIO 강좌 : 01. GPIO 소개 및 핀배치](http://www.rasplay.org/?p=2049)

# ****라즈베리파이 GPIO 강좌****

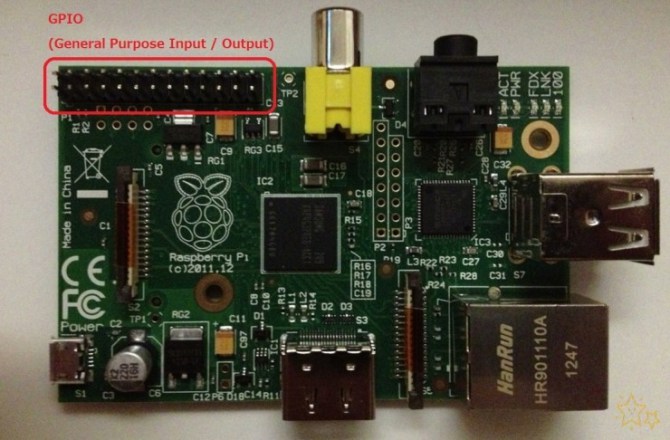
### 라즈베리파이 GPIO 강좌 : 01. GPIO 소개 및 핀배치

2013.03.01

##### ****1. 라즈베리파이 GPIO 소개****

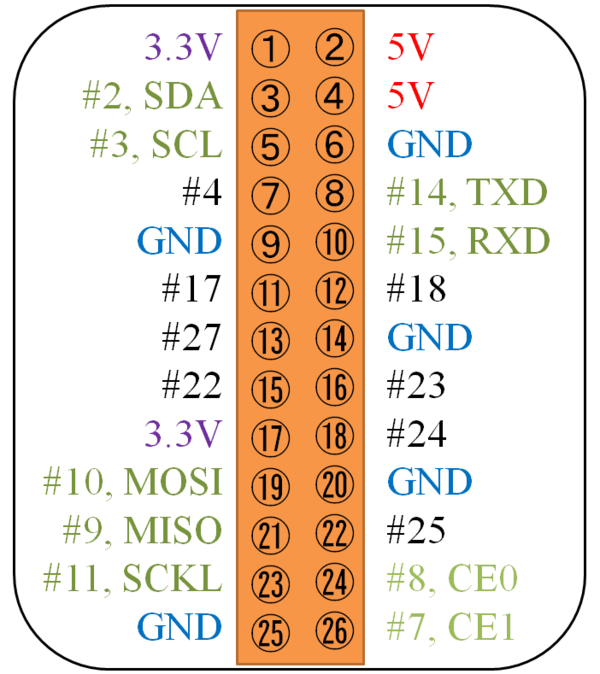
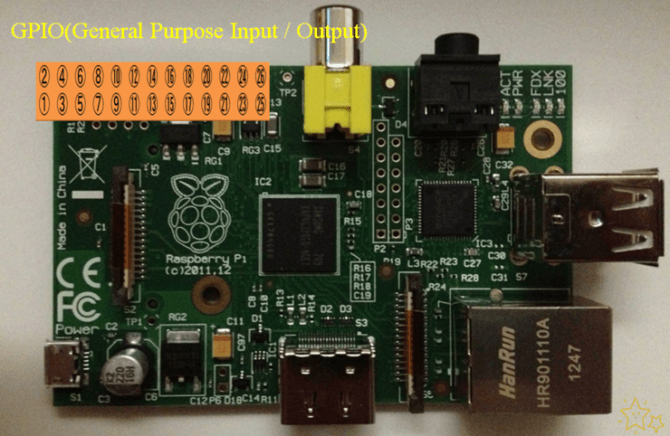
소형 PC 라즈베리파이는 USB, 이더넷, HDMI 출력, 영상출력, 사운드출력 등의 기능을 갖고 있는 하나의 작은 컴퓨터이다. 가격이 저렴하고, 작은 크기이지만 그 성능은 간단한 OS를 돌리고 동영상을 보고, 인터넷 서핑을 하는 목적에는 전혀 문제가 없는 완전한 컴퓨터인 것이다.

더욱이, 재미있는 점은 단일 마이크로세서와 같이 입출력 신호를 제어할 수 있는 포트가 있다는 것이다. 범용적인 목적으로 입/출력을 담당하는 GPIO (General Purpose Input / Output) 를 가지고 있으며, SPI 통신, I2C 통신, UART 통신 등도 갖추고 있다. 이는 마이크로프로세서를 다루는 사람들에게는 흥미로운 일이 아닐 수 없다. 왜냐하면 일반적으로 마이크로세서는 간단한 리얼타임OS(RTOS) 정도 올리던 것을 전부였는데 라즈베리파이는 데비안계열의 기능 축소판이긴 하지만 리눅스 OS를 올릴 수 있으며, 더욱이 자신이 원하는 포트 제어를 할 수 있기 때문이다. 물론, 제어전용 마이크로프로세서에 비해 리눅스를 기반으로 하고 있기에 시스템 부하에 따라 프로그램 실행 속도가 늦어지거나 실행 지점이 다르다는 등 실시간 제어가 안된다는 것이 문제 이기는 하지만 그런 제약이 없는 프로젝트라면 사용할 만 할 듯 싶다. 추가로 ADC, PWM, 인터럽트 기능이 없는 것도 조금 아쉽지만 제한된 범위에서의 제어에는 문제 없을 듯 싶다.

[](http://i1.wp.com/www.rasplay.org/wp-content/uploads/1.jpg)

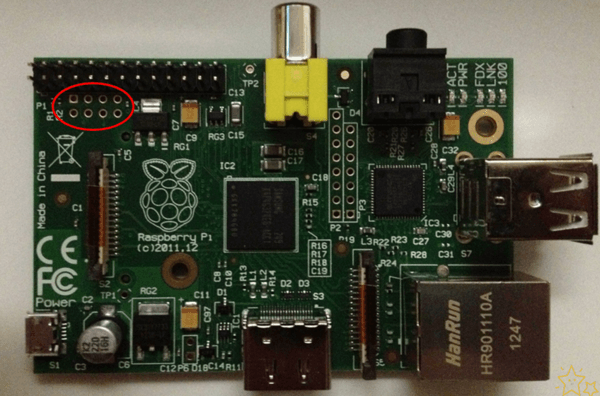
##### ****2. 라즈베리파이 외부핀****

현재 라즈베리파이 모델B (버전2, 512RAM) 의 경우, 외부핀은 26핀으로 구성되어 있다. (별도로 P5 헤더 및 카메라핀 등 헤더핀이 삽입되어 있지는 않지만 공간을 할당해 놓은것은 제외한다.) 이 중, UART, SPI, I2C 관련핀들은 GPIO 목적으로도 사용되기도 하고 각 관련된 특수 기능을 수행하기도 한다. 아래는 각 핀을 배치에 따라 나열해 본 것이다.

[](http://i1.wp.com/www.rasplay.org/wp-content/uploads/2.png) [](http://i1.wp.com/www.rasplay.org/wp-content/uploads/Capture.png)

위 그림에서 알 수 있듯이 라즈베리파이의 P1 헤더핀에는 전원과 관련하여 9핀이 사용되고 있고, UART 관련 총2핀, I2C 관련 총2핀, SPI 관련 총5핀이 사용되고 있다. 또한, GPIO 전용으로 총 8핀으로 총 26개 핀으로 구성되어 있다. 물론, UART 관련 총2핀, I2C 관련 총2핀, SPI 관련 총5핀 는 각 주어진 기능을 수행하기도 하지만, GPIO로도 이용가능한 핀들이기에 GPIO는 총 17핀이다.

* 5v : 총 2핀
* 3.3v : 총 2핀
* GND : 총 5핀
* TXD, RXD : UART 관련 총2핀
* SDA, SCL : I2C 관련 총2핀
* MOSI, MISO, SCKL, CE0, CE1 : SPI 관련 총5핀
* GPIO 4, 17, 18, 22, 23, 24, 25, 27 : GPIO 전용 총 8핀

[](http://i1.wp.com/www.rasplay.org/wp-content/uploads/p5.png)

더욱이, 위의 그림의 빨간 동그라미의 부분은헤더핀이 삽입되어 있지는 않지만 헤더핀을 삽입하면 별도의 8개핀을 추가로 이용가능하다. 이는 핀번호 순서대로 5v,3.3v, GPIO28, 29, 30, 31, GND, GND 로 구성되어 있다. 본 강좌에서는 별도로 다룰 예정은 아니지만 필요한 사람은 헤더핀 납땜 후 사용하면 크기도 작도 간단히 사용하기 좋을 듯 싶다.

그 이외에도 리셋을 달 수 있는 P6 헤더, JTAG 과 관련된 P2, P3 헤더 등도 있지만 GPIO 강좌와는 관련이 없으니 설명은 생략하도록 한다.

마지막으로, 종합하면 라즈베리파이에서 사용할 수 있는 GPIO는 2, 3, 4, 7, 8, 9, 10, 11, 14, 15, 17, 18, 22, 23, 24, 25, 27, 28, 29, 30, 31 으로 총 21개의 핀이다. (라즈베리파이 모델B 버전2, 512RAM)

##### ****3. 라즈베리파이 주의사항****

1) 전원구성  
: 라즈베리파이의 외부핀의 전원관련으로는 3.3v, 5v 의 두 개의 전원핀이 존재한다. 허나~ GPIO 관련하여 사용할 때는 3.3v 의 회로만을 구성해야한다. 이는 라즈베리파이의 동작 전원이 3.3v 이고 그 이상의 전원에 보호하는 기능이 없다. 즉, 5v는 단순히 microUSB 선으로 들어오는 5v 전원을 연결했을뿐 실제 라즈베리파이는 3.3v 범위안에서 작업을 해야한다. 예를들어 스위치, LED를 달더라도 3.3v 에 회로를 맞춰야 한다.

2) 전류사용  
: 3.3v 의 핀은 최대 30mA이 전류를 사용할 수 있다. 그 이상의 전류가 필요한 회로를 구성하게 되면 라즈베리파이의 메인 회로에 문제가 있을 수 있다.

3) 과전압, 과전류  
: 라즈베리파이는 각 외부핀에 대해 전원 보호 회로를 구성하고 있지 않다. 즉, GPIO 핀에 과전류, 과전압을 주게 되면 내부 라즈베리파이 회로에 영향을 주고 동작하지 않을 수도 있다.

4) 버전별 핀배열  
: 라즈베리파이 모델 B 는 출시된 시점에 따라 버전이 1, 2로 나뉜다. 이 두가지 버전은 P1 헤더핀 26개가 서로 다르게 설정되어 있어서 사용함에 있어서 주의해야 한다. 본 강좌에서 설명한 핀 배열은 라즈베리파이 모델B 버전2의 512RAM 버전이다.

[ 참고자료 ]

[1] “라즈베리파이 공식 사이트”, http://www.raspberrypi.org/  
[2] “라즈베리파이 위키미디어 설명”, http://en.wikipedia.org/wiki/Raspberry\_Pi  
[3] “Embedded Linux Wiki”, http://elinux.org/  
[4] “공식 판매점 Element14”, http://downloads.element14.com/raspberryPi1.html?COM=raspi-group  
[5] “공식 판매점 RS Componts”, http://uk.rs-online.com/web/generalDisplay.html?id=raspberrypi  
[6] “공식 판매점 Allied Electronics”, http://www.alliedelec.com/lp/120626raso/  
[7] “element14의 레즈베리파이 커뮤니티”, http://www.element14.com/community/groups/raspberry-pi  
[8] Embedded Linux 커뮤니티의 “RPi Tutorials”, http://elinux.org/RPi\_Tutorials  
[9] “RASPBERRY PI INFORMATION AND VIDEOS”, http://raspi.tv/  
[10] “Raspberry Pi Spy”, http://www.raspberrypi-spy.co.uk/  
[11] “RaspiHub”, http://www.raspihub.com/  
[12] “Adafruit Blog”, http://www.adafruit.com/blog/category/raspberry-pi/  
[13] “Raspberry Pi Pod”, http://www.recantha.co.uk/blog/  
[14] “doctormonk”, http://www.doctormonk.com/2013/02/raspberry-pi-and-breadboard-raspberry.html  
[15] “bcm2835”, http://www.open.com.au/mikem/bcm2835/

End.

written by Yoonseok Pyo  
<http://robotpilot.net/>  
<http://cafe.naver.com/openrt>

[[http://i1.wp.com/i.creativecommons.org/l/by-sa/3.0/88x31.png?w=670](http://creativecommons.org/licenses/by-sa/3.0/deed.ko)](http://creativecommons.org/licenses/by-sa/3.0/deed.ko)이 저작물은 [크리에이티브 커먼즈 저작자표시-동일조건변경허락 3.0 Unported 라이선스](http://creativecommons.org/licenses/by-sa/3.0/deed.ko) 에  
따라 이용할 수 있습니다.

<http://m.clien.net/cs3/board?bo_table=lecture&bo_style=view&wr_id=188209>

라즈베리파이 활용 및 GPIO 강좌 목차

[http://www.rasplay.org](http://www.rasplay.org/)  
[http://wIKI.rasplay.org](http://wiki.rasplay.org/)

**라즈베리파이 활용 및 GPIO 강좌 목차**

**Created Date: 2013.10.01**

**Modified Date: 2013.12.05**

**revision 5**

**0. 개발자를 위한 라즈베리 기초(Special Post)**

[**01 개발을 위한 SSH 환경 꾸미기**](http://cafe.naver.com/openrt/2944)

[**02 개발을 위한 SAMBA 환경 꾸미기**](http://cafe.naver.com/openrt/2913)

[**03 개발을 위한 Notepad++ 환경 꾸미기**](http://cafe.naver.com/openrt/3128)

[**04 개발을 위한 Notepad++ 환경 꾸미기2**](http://cafe.naver.com/openrt/3155)

Tip : 클러스터링 크로스 컴파일 distcc (  [Mac](http://www.rasplay.org/?p=6964) , [Ubuntu](http://www.rasplay.org/?p=6996) )

**I. 라즈베리파이 사용법 기초강좌**: 중첩내용 관련 포스트 Tip으로 등록.

     1. [싱글 보드 컴퓨터 (SBC)](http://cafe.naver.com/openrt/160)

     2. [Raspberry Pi 소개](http://cafe.naver.com/openrt/161)

     3. [Raspberry Pi 구매](http://cafe.naver.com/openrt/162)

     4. [Raspberry Pi 악세사리](http://cafe.naver.com/openrt/163) :

     5. [Raspberry Pi 퀵스타트 (OS 설치 및 테스트 구동)](http://cafe.naver.com/openrt/165)

     6. 리눅스에서 라즈베리파이 개발환경 구축하기

         o 내용 : 우분투에서 라즈베리파이 개발환경 구축하기.

             - 버츄얼박스 에서 우분투 설치하기

             - 우분투에서 라즈비안 설치하기

             - 우분투에서 라즈베리파이 크로스 컴파일 환경 구축하기

             - 우분투에서 NFS 서버 설치 및 세팅하기

             - 우분투에서 Putty 및 NFS Client 설치 및 세팅하기

             - 우분투에서 코딩 및 크로스 컴파일 하기 ( EX: hello world )

         o 파일 다운로드 : [1장](http://cafeattach.naver.net/67f27bcbda8b835f769cf2c0ff1f6519b9eb18f4f6/20130219_19_cafefile/25572101_1361234909119_OCG0Bd_pptx/rpi1.pptx?type=attachment) , [2장](http://cafeattach.naver.net/b82da41700333282a34c2e1d26c0bfc66639c994/20130219_198_cafefile/25572101_1361235102798_5pWgBJ_pptx/rpi2.pptx?type=attachment)

    7. [Raspberry Pi 기본설정(raspi-config)](http://cafe.naver.com/openrt/191)

        o Tip : [OS 설치 후 초기 설정작업](http://cafe.naver.com/openrt/233) :

        o Tip : [원격접속을 위한 포트변경 및 서비스 재기동 방법](http://cafe.naver.com/openrt/246)

    8. [보안쉘(ssh)을 이용한 원격 접속](http://cafe.naver.com/openrt/193)

    9. [xrdp를 이용한 원격 접속](http://cafe.naver.com/openrt/195)

  10. [라즈베리파이 이미지 백업](http://cafe.naver.com/openrt/196)

  11. [사운드 테스트](http://cafe.naver.com/openrt/197)

  12. [무선랜 설정 (Wireless LAN, WLAN, WiFi)](http://cafe.naver.com/openrt/198)

  13. [유선랜(eth0) 고정 IP 설정 법](http://www.rasplay.org/?p=1516)

  14. [블루투스 키보드 연결하기](http://www.rasplay.org/?p=2328)

  15. [SD 메모리 용량 수동 늘리기](http://www.rasplay.org/?p=2837)

  16. [라즈베리파이 펌웨어 및 커널 업데이트](http://www.rasplay.org/?p=1054)

  17. [X윈도우 IBUS 한글자판(두벌식/세벌식) 변경하기](http://www.rasplay.org/?p=3816)

  18. [라즈베리파이를 무선공유기(AP)로 활용하기](http://www.rasplay.org/?p=6871)

**II. 라즈베리파이 GPIO제어 기초강좌**

     1. [라즈베리파이 GPIO 소개 및 핀배치](http://cafe.naver.com/openrt/224)

     2. 라즈베리파이 wiringPi 설치하기([자동설치(C언어, 파이썬)](http://www.rasplay.org/?p=4918) [수동설치(C언어)](http://cafe.naver.com/openrt/2706))

     3. [라즈베리파이 LED 출력, C언어(Output 테스트)](http://cafe.naver.com/openrt/228) , [라즈베리파이 LED 출력, 파이썬(Output 테스트)](http://cafe.naver.com/openrt/226)

     4. [라즈베리파이 스위치 입력, C언어(input 테스트)](http://cafe.naver.com/openrt/229) , [라즈베리파이 스위치 입력, 파이썬(input 테스트)](http://cafe.naver.com/openrt/227)

     5. [UART 테스트 (Windows PC <-> RaspberryPi)](http://cafe.naver.com/openrt/484)

         o Tip : [라즈베리파이 RS232 <-> UART 통신하기 1편](http://cafe.naver.com/openrt/249) , [2편](http://cafe.naver.com/openrt/276)

     6. [SPI 테스트 (ADC칩 MCP3208을 이용한 센서 값 읽기)](http://cafe.naver.com/openrt/494)

     7.[I2C 테스트 (RTC칩 DS1307 시간 저장 및 읽기, C)](http://cafe.naver.com/openrt/564) , [I2C 테스트 (파이썬)](http://www.rasplay.org/?p=6409)

**III. 라즈베리파이 활용법 강좌**

**III-1. 라즈비안 활용강좌**

      1.  [라즈베리파이 에뮬레이터 사용하기 : QEMU for Windows](http://www.rasplay.org/?p=1864)

      2. [라즈베리파이 멀티부팅 설치법](http://www.rasplay.org/?p=238)

      3. [라즈비안(Raspbian F2FS) 적용법](http://www.rasplay.org/?p=6007)

      4. [라즈베리파이 외부저장 장치(HDD,SSD)를 ROOFS사용하기](http://www.rasplay.org/?p=1779)

      5. [부팅 시, 프로그램 자동실행 등록 법](http://www.rasplay.org/?p=4854)

      6.  [원격제어(TIGERVNC) 설치 및 설정](http://www.rasplay.org/?p=749)

      7.  [X윈도우 절전모드 및 스크린세이버 기능 끄기](http://www.rasplay.org/?p=4828)

      8.  [X윈도우 MANAGER ENLIGHTENMENT 설치하기](http://www.rasplay.org/?p=4737)

      9.  [라즈비안 자동 로그인 하기](http://www.rasplay.org/?p=3161)

     10. [랜 케이블 하나로 라즈베리파이 접속하기](http://www.rasplay.org/?p=2804)

     11. [라즈베리파이로 테더링 무선 인터넷 접속하기 for iPhone](http://www.rasplay.org/?p=2045)

     12. [라즈베리파이 와 PC 간 USB 시리얼 통신하기](http://www.rasplay.org/?p=1617)

     13. [USB SPEAKERS 사용하기](http://www.rasplay.org/?p=1474)

     14. [라즈베리파이로 전자메일 보내기](http://www.rasplay.org/?cat=144)

     15. [라즈베리파이 인터넷 라디오 청취하기](http://www.rasplay.org/?p=1387)

     16. [넥서스7(NEXUS7) 과 라즈베리파이 터미널 통신하기](http://www.rasplay.org/?p=3321)

     17. [라즈베리파이에 소니 듀얼쇼크3(블루투스) 조이스틱 사용하기](http://www.rasplay.org/?p=5846)

     18. [라즈베리파이에 XBOX360 무선 조이스틱 사용하기](http://cafe.naver.com/openrt/1977)

     19. [라즈베리파이 부트 로고 변경 법](http://www.rasplay.org/?p=6371)

     20. [워커홀릭을 위한 휴대용 라즈베리파이 셋트](http://cafe.naver.com/openrt/589)

     21. [12V 어댑터 하나로 모니터 와 라즈베리파이 구동하기](http://www.rasplay.org/?p=6536)

     22. [파이북(Artrix) 제작 기](http://www.rasplay.org/?p=3111)

     23. [스크린 및 오디오 전송 강좌(넥서스7 → 라즈베리파이)](http://www.rasplay.org/?p=3516)

**III-2. 라즈베리파이 개발환경 활용강좌**

       1. [라즈베리파이에 WEBIDE 설치하기](http://www.rasplay.org/?p=3402) :

       2. [라즈베리파이에서 git server 구축하기](http://www.rasplay.org/?p=1325):

       3. [라즈베리파이에서 Java 환경 구축하기 : Open JDK 설치](http://www.rasplay.org/?p=4126)

       4. [라즈베리파이에서 Java 개발환경 구축하기 : Oracle JDK8 설치](http://cafe.naver.com/openrt/854)

       5. Node.js 라즈베리파이 GPIO 제어 개발환경 구축하기(Raspbian, Arch Linux)

           o [Raspbian에서 개발환경 구축하기](http://cafe.naver.com/openrt/323) , [Arch 에서개발환경 구축하기](http://www.rasplay.org/?p=4317)

**III-3. 라즈베리파이 서버 활용강좌**

       1. [라즈베리파이에 아파치 웹 서버 구축하기(Apache + php5 + mysql)](http://www.rasplay.org/?p=72)

       2. [라즈베리파이 웹 모니터링(RASPCONTROL)](http://www.rasplay.org/?p=1935)

       3. [엔진엑스(Nginx)에서 라즈컨트롤 사용하기](http://www.rasplay.org/?p=4194) :

       4. [개인용 워드프레스(wordpress) 블로그 만들기(Nginx + php5 + php-fpm + mysql)](http://www.rasplay.org/?p=1416)

       5. [FTP(VSFTP) 서버 구축하기](http://www.rasplay.org/?p=3051)

       6. [NAS(](http://www.rasplay.org/?p=1529)OPENMEDIAVAULT[) 서버 구축하기](http://www.rasplay.org/?p=1529)

       7. [클라우드 스토리지 서버(own Cloud) 구축하기](http://www.rasplay.org/?p=761) , [own Cloud 버그패치](http://www.rasplay.org/?p=1895)

       8. [MP3 스트리밍 서버(Cherry Music) 구축하기](http://www.rasplay.org/?p=4361)

       9. [VPN SERVER(OPENVPN, Open SSL) 구축하기](http://www.rasplay.org/?p=3552)

      10. [라즈베리파이 irc 서버 구축법](http://www.rasplay.org/?p=3975)

      11. DB 서버 클러스터링 서버 구축법

              1. [RIAK CLUSTER 설치 - 1](http://www.rasplay.org/?p=1975)

              2. [RIAK CLUSTER 노드 설정 - 2](http://www.rasplay.org/?p=2085)

              3. [RIAK CLUSTER nginx 연동 - 3](http://www.rasplay.org/?p=2292)

**III-4. 라즈베리파이 오픈멀티미디어 XBMC 활용강좌**

       1. [XBMC(XBOX MEDIA CENTER) 란 ?](http://www.rasplay.org/?p=167) :

       2. [Openelec 외 2 종 XBMC 설치하기(Image Write)설치](http://www.rasplay.org/?p=206)

       3. [RASPBMC 설치하(Image Write)](http://www.rasplay.org/?p=5964)

       4. [라즈비안(Raspbian)에 XBMC 설치하기](http://www.rasplay.org/?p=1389)

       5. [한국에서 BBC 방송 시청하기](http://www.rasplay.org/?p=945)

       6. [리모컨으로 XBMC 제어하기](http://www.rasplay.org/?p=997)

**III-5. 라즈베리파이 게임 에뮬레이터 활용강좌**

       1. [PIMAME 설치하기](http://www.rasplay.org/?p=524)

       2. 네오지오 설치법 : [PiMAME 네오지오 설치하기](http://www.rasplay.org/?p=4931) ,  [UNIX버젼 컴파일 강좌](http://cafe.naver.com/openrt/1830)

       3. [세가세턴 에뮬(YABAUSE) 설치하기](http://www.rasplay.org/?p=658)

       4. [라즈베리파이 마인크래프트 설치하기](http://www.rasplay.org/?p=644)

       5. 도스(DOS) 에뮬레이터(R-Pix86) 설치법 : [터미널에서 설치하기](http://cafe.naver.com/openrt/1831) , [파이스토어에서 설치하기](http://www.rasplay.org/?p=4657)

**III-6. 라즈베리파이 카메라 활용강좌**

**ㄱ. 라즈베리파이 USB Camera**

            a. [USB캠을 이용한 웹 스트리밍(Motion)](http://www.rasplay.org/?p=4609)

            b. [아치리눅스 OpenCV 영상처리하기](http://www.rasplay.org/?p=3403)

**ㄴ. 라즈베리파이 전용 Pi-CAM**

            a. [라즈베리파이 카메라 연결 및 설정하기](http://cafe.naver.com/openrt/912)

            b. [파이카메라 촬영 H.264 를 MP4으로 변환하기](http://www.rasplay.org/?p=4595)

            c. [파이카메라 MOTION 적용법](http://www.rasplay.org/?p=4910)

            d. [G-streamer 를 이용한 영상 전송하기](http://www.rasplay.org/?p=5607)

            e. [RTMP를 이용한 웹으로 H264 영상전송](http://cafe.naver.com/openrt/2000)

            f. [안드로이드에서 라즈베리파이 스트리밍 서버 접속하기](http://www.rasplay.org/?p=4609) :

            g. [V4L2(Video4Linux2) 설치하기](http://www.rasplay.org/?p=6257)

            h. [PI-CAM V4L2(VIDEO4LINUX2) MOTION 설치](http://www.rasplay.org/?p=6287)

            i. [V4L2(VIDEO4LINUX) 와 VLC 를 이용한 RTSP STREAMING](http://www.rasplay.org/?p=6277)

            j. [V4L2(VIDEO4LINUX) 기반 OpenCV 설치 및 테스트](http://www.rasplay.org/?p=6330)

            k. [라즈베리파이 캠코더 제작하기](http://www.rasplay.org/?p=5897)

**IV. 라즈베리파이 GPIO 활용법 강좌**

       1. [스크래치(scrach)로 GPIO 제어하기](http://www.rasplay.org/?cat=291)

       2. 라즈베리파이 [WEBIOPI 설치하기](http://www.rasplay.org/?p=5061) , [WebIOPi 응용하기](http://cafe.naver.com/openrt/1846)

       3. [Node.js 라즈베리파이 GPIO 플러그인 설치하기](http://cafe.naver.com/openrt/372)

       4. [Node.js LED 출력(Output)](http://cafe.naver.com/openrt/864)

       5. [안드로이드 앱으로 GPIO 제어하기](http://www.rasplay.org/?p=5851)

       6. [libdc1394, ROS, FMVU 시리즈를 이용한 이미지 전송하기](http://cafe.naver.com/openrt/688)

       7. [라즈베리파이에서 허큘렉스 사용하기](http://cafe.naver.com/openrt/1298)

       8. [TTS 이용한 음성 출력](http://cafe.naver.com/openrt/1300)

       9. [안드로이드로 제어하는 라즈베리파이 무선 자동차](http://cafe.naver.com/openrt/1331)

     10. [라즈베리파이용 저가형 RC Car(SN754410) 제작하기](http://cafe.naver.com/openrt)

     11. [센서를 장착한 라즈베리파이 플라워](http://cafe.naver.com/openrt/1458)

     12. [환경센서보드 제작하기](http://cafe.naver.com/openrt/1594)

     13. [DB에 환경센서 정보 기록하기](http://cafe.naver.com/openrt/1605)

     14. [캐릭터 LCD 연결 및 테스트](http://www.rasplay.org/?p=3026)

     15. Nokia5110(PCD8544) 그래픽 LCD 연결 및 제어([C언어](http://www.rasplay.org/?p=4716) , [파이썬](http://www.rasplay.org/?p=5982))

     16. [라즈베리파이 멀티부팅 스위치 만들기](http://www.rasplay.org/?p=2604)

     17. [라즈베리파이에 GPS 모듈 연결 및 테스트](http://www.rasplay.org/?p=3560)

     18. [동작인식(PIR) 파이라이트](http://www.rasplay.org/?p=6457)

     19. [물달라고 트위트하는 라즈베리파이 플라워](http://cafe.naver.com/openrt/1458)

     20. [라즈베리파이와 알피노(아두이노) 연동하기](http://www.rasplay.org/?p=5075) , [참고 포스트](http://cafe.naver.com/openrt/2047)

     21. [라즈베리파이와 알피노(아두이노) 시리얼통신하기](http://www.rasplay.org/?p=5092)

     22. [라즈베리파이 터미널에서 알피노(아두이노) 컴파일 및 업로드 하기](http://www.rasplay.org/?p=5081)

     23. [라즈베리파이에서 아두이노(Atmega328) 다루기](http://cafe.naver.com/openrt/2776) ( [제1부](http://www.rasplay.org/?p=6801), 제2부 )

     24. [라즈베리파이에서 하드웨어 PWM 제어(C언어)](http://cafe.naver.com/openrt/2705)

     25. [라즈베리파이로 아두이노(멀티두이노) 스케치 사용하기](http://www.rasplay.org/?p=6801)

     26. [아두이노(멀티두이노) LED 켜기](http://www.rasplay.org/?p=6832)

     27. 라즈베리파이 미니모니터 제작하기(2.8" 터치 TFT-LCD) - [Preview](http://www.rasplay.org/?p=6871) [1부](http://www.rasplay.org/?p=6839) , [2부](http://www.rasplay.org/?p=6890)

     28. [라즈베리파이 위 ATMEGA328P(멀티두이노)로 서보모터 제어하기](http://www.rasplay.org/?p=6860)

**V. 라즈베리파이 공개 프로젝트 하나 : 멀티제어 RC카(LB1630)**

      1. [키보드로 RC카를 제어하기](http://www.rasplay.org/?p=5351)

      2. [웹(webiopi)으로 RC카 제어하기](http://www.rasplay.org/?p=5773) , [산딸기마을 깃 허브 소스](https://github.com/rasplay/clickpirc)

      3. [무선 조이스틱(XBOX360)로 RC카 제어하기](http://www.rasplay.org/?p=6429) , [산딸기마을 깃 허브 소스](https://github.com/rasplay/xbox360-wireless-360)

      4. 3[축 가속/3축 자이로 센서로 RC카 제어하기(산딸기마을 깃허브 소스)](https://github.com/rasplay/3axi3gyro)

      5. [스마트폰으로 RC카 제어하기 TouchPi RC](http://www.rasplay.org/?p=6000) , [산딸기마을 깃허브 소스](https://github.com/rasplay/touchpirc)

**VI. 라즈베리파이 오픈 프로젝트 하나 : RAS-POS**

      1. [오픈 프로젝트(RAS-POS)](http://www.rasplay.org/?p=6494) , [산딸기마을 깃 허브 소스](https://github.com/rasplay/RAS-POS)

**VI. 라즈베리파이 하드웨어정보 및 해킹**

      1. [라즈베리파이 BCM2580 데이터시트](http://www.rasplay.org/?p=3386)

      2. [라즈베리파이 전류제한](http://www.rasplay.org/?p=5052) , [전류제한 해제 폴리스위치 교체 및 테스트](http://cafe.naver.com/openrt/1967)

[청령](javascript:) (10-21 17:30)

안그래도 산딸기에 꽂혀 지르기 직전인데 딱 맞게 유용한 정보를 올려주셔서 감사합니다 :)

[늑대길이](javascript:) (10-21 17:52)

도움이 되신다니 다행입니다 .^^  
  
아직 정리가 안된 자료들이 많아 계속 업데이트를 해야할 것 같습니다. ㅠ.ㅠ

[Postklee](javascript:) (10-21 18:30)

ㅎㅎ 이번에 Node.js 입문하면서 라즈베리파이에 발을 담궜는데... 정말 방대한 정보가~ 감사합니다 :)

[늑대길이](javascript:) (10-21 19:38)

헙, node.js 가 파이에선 가볍고 좋죠 ^^

[http://m.clien.net/data/member/st/stan03.gif?dt=20130922161648](javascript:) (10-21 21:48)

와우 정리 정말 좋네요  
  
스크랩!  
via iOS

[늑대길이](javascript:) (10-22 21:01)

감사드립니다. ^^

[ppureo](javascript:) (10-21 22:46)

오오~ 잘 활용하겠습니다!

[늑대길이](javascript:) (10-22 21:01)

넵 많은 활용 부탁드립니다.

[구닷꼼](javascript:) (10-22 09:29)

와.. 대단하세요~   
  
스크랩 꽝!!! 추천 꽝!!!

[늑대길이](javascript:) (10-22 21:01)

헙 ! 감사합니다.

[진짜메뚜기](javascript:) (10-22 23:57)

이 분 글 보면서 참 대단하다고 생각했는데 말이죠  
  
  
  
안본 사이에 글이 엄청 늘어났네요 감사합니다

[꾼1982](javascript:) (10-23 10:52)

이런분들은 이런걸 즐기시는 거겠죠?? 부럽...  
  
시간은 어디서 나시는지 ㅎㅎㅎ

[블루프라그마](javascript:) (06-01 21:27)

뒤는게 보고 적습니다.  
  
감사합니다. ^^

새로고침

|  |  |
| --- | --- |
| [**이전 글**](http://m.clien.net/cs3/board?bo_table=lecture&bo_style=view&wr_id=188217) | 안드로이드 알림창 위젯 확장기능 |
| [**다음 글**](http://m.clien.net/cs3/board?bo_table=lecture&bo_style=view&wr_id=188190) | iOS 7 앱 아이콘 가이드.ai(R.. |

|  |  |  |  |
| --- | --- | --- | --- |
| **컨버젼스를 위한 node.js 강좌 - 2. 라즈베리파이 GPIO 다루기** | | | | [SBC:라즈베리파이](http://cafe.naver.com/ArticleList.nhn?search.clubid=25572101&search.menuid=43&search.boardtype=&userDisplay=) |
| 전체공개 | 2013.03.18. 13:01 |

양식의 맨 위

양식의 맨 아래

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어(gbox\*\*\*\*)**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=372&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | VIPhttp://cafeimgs.naver.net/levelicon/1/1_150.gif | [1:1대화](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=372&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | |
| <http://cafe.naver.com/openrt/372> [http://cafeimgs.naver.net/cafe4/btn-copy-add.gif](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=372&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | |
|  | | |

출처[도플광어의 구름기술연구소](http://cafe.naver.com/goorume) | [도플광어](http://cafe.naver.com/goorume/member/gbox2/article)

원문<http://cafe.naver.com/goorume/832>

이번시간에는 노드로 GPIO를 다뤄보도록 하겠습니다. GPIO에 대한 자세한 설명은 다른글들을 참고 해주세요

1. 하드웨어 세팅

4번 핀에 저항과 함께 led  를 연결합니다.

2. onoff 플러그인 설치하기

터미널에서 아래와 같이 명령어를 실행합니다.

sudo npm install -g onoff

onoff 라는 것으로 GPIO를 노드에서 다룰수 있도록 하는 플러그인을 설치합니다.

(참고 :<https://github.com/fivdi/onoff>)

3. 스크립팅

프로그램이 시작되면 led에 불이 5초간 들어왔다가 꺼지게 해보겠습니다.

먼저 편집기로 소스파일을 생성합니다.

sudo nano hellogpio.js

그리고 편집창이 열리면 다음과 같이 자바스크립트 소스를 작성합니다.

var Gpio = require('onoff').Gpio; // Constructor function for Gpio objects.

//첫번째 인자 : 핀번호는 4번으로 지정합니다.

// 두번째 인자: 핀모드를 출력(out) 모드로 지정합니다.

var    led = new Gpio(4, 'out');

//핀을 hight 상태로 만들어 불이 들어오게합니다.

led.writeSync(1);

// the LED and turn it off after 5 seconds.

setTimeout(function() {

//5초후에 이부분이 실행됩니다.

    led.writeSync(0);  // Turn LED off.

    led.unexport();    // Unexport GPIO and free resources

    console.log("finished 5sec..bye~ :)");

}, 5000);

4. 실행하기

항상 sudo가 앞에 붙습니다.

sudo node hellogpio.js

다음 시간에는 웹(http)을 이용해서 네트워크를 통해서 led를 켜고 끄는 것에 도전을 해보겠습니다.

|  |
| --- |
| **[출처]** [컨버젼스를 위한 node.js 강좌 - 2. 라즈베리파이 GPIO 다루기 (오픈소스 소프트웨어 & 하드웨어: 로봇 기술 공유 카페 (오로카))](http://cafe.naver.com/openrt/372)|**작성자** [도플광어](http://cafe.naver.com/openrt.cafe?iframe_url=/CafeMemberNetworkView.nhn%3Fm=view%26memberid=gbox2) |

|  |  |  |  |
| --- | --- | --- | --- |
| **컨버젼스를 위한 node.js 강좌 - 3. webapi 만들기** | | | | [SBC:라즈베리파이](http://cafe.naver.com/ArticleList.nhn?search.clubid=25572101&search.menuid=43&search.boardtype=&userDisplay=) |
| 전체공개 | 2013.05.18. 19:21 |

양식의 맨 위

양식의 맨 아래

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어(gbox\*\*\*\*)**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | VIPhttp://cafeimgs.naver.net/levelicon/1/1_150.gif | [1:1대화](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | |
| <http://cafe.naver.com/openrt/876> [http://cafeimgs.naver.net/cafe4/btn-copy-add.gif](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | |
|  | | |

출처[도플광어의 구름기술연구소](http://cafe.naver.com/goorume) | [도플광어](http://cafe.naver.com/goorume/member/gbox2/article)

원문<http://cafe.naver.com/goorume/912>

지난번에 예고 했던대로 이번에는 nodeJS를 사용하여 webapi 를 만들고 그걸사용해서 라즈를 인터넷을 통해 제어 해보도록 하겠습니다.

1. gpio

먼저 지난 시간에 gpio는 모두 세팅해보셨으리라 생각하고 바로 gpio 를 사용해서 led를 깜박이는 코드를 작성해보겠습니다.

var Gpio = require('onoff').Gpio; // Constructor function for Gpio objects.

var led = [new Gpio(4, 'out'),new Gpio(17, 'out'),new Gpio(27, 'out')];

var iv;

// 0.5초 마다 깜박이기

iv = setInterval(function() {

    led[0].writeSync(led[0].readSync() === 0 ? 1 : 0); // 1 = on, 0 = off :)

}, 500);

일단 포트를 4,17,27 번을 사용하려고 포트를 할당 받았습니다.

그리고 4번포트를 0.5초간격으로 깜박이도록했습니다.(노드가 동작중인것을 알기위해서..)

나머지 17번 27번 포트는 사용자가 원하면 켜고 끄도록 하려고합니다. 그것도 인터넷을 통해서 말이죠

2. netwoking

그럼 인터넷서버를 만들어야겠죠?

c나 자바 같은것으로 socket 어쩌고 TCP어쩌고 해서 만드는방법도 있겠지만요. 저는 노드를 사용해서 초간단하게 서버를 만들어 보도록하겠습니다.

var http = require('http');

var gPort = 9010;

http.createServer(function (req, res) {

// 이부분에서 사용자 요청을 처리합니다.

}).listen(gPort);

이렇게 하면 사용자 요청을 받아서 처리해줄수 있는 아주 간단한 서버가 만들어 집니다.

기억하시는분들은 아시겠지만 이것은 첫시간에 했던 내용입니다.

3. web api

웹브라우져의 주소창에 url 경로를 지정해줄 수 있습니다.

http://192.168.0.77/go

요철 주소는 192.168.0.77이고 경로명은 /go 가 됩니다.

여기서 잠깐 생각을 넓혀봅시다.

인터넷을 하나의 큰 가상머신으로 본다면 go를 마치 함수처럼 사용할 수 도 있겠죠?

그래서 다음과 같이 요청처리 코드를 완성해줍니다.

var http = require('http');

var gPort = 9010;

http.createServer(function (req, res) {

    var ip = req.headers['x-forwarded-for']|| req.connection.remoteAddress;

    console.log('remote address:' + ip);

    res.writeHead(200, {'Content-Type': 'text/plain'});

    if(req.url == '/favicon.ico') {

    }

    else {

        console.log(req.url);

        switch(**req.ur**l) {

            case '/go':

                led[1].writeSync(1);

                led[2].writeSync(0);

                res.write('success call api : '+ req.url);

                break;

            case '/back':

                led[1].writeSync(0);

                led[2].writeSync(1);

                res.write('success call api : '+ req.url);

                break;

            case '/stop':

                led[1].writeSync(0);

                led[2].writeSync(0);

                res.write('success call api : '+ req.url);

                break;

            default:

                res.write('error api : '+ req.url);

                break;

        }

    }

    res.end();

}).listen(gPort);

console.log('Server running at : ' + gPort);

포트번호는 9010을 사용했습니다.

req.url 이라는 변수에 네트워크상에서 사용자가 요청한 경로 이름이 넘어 옵니다.

go 명령어는 17번 포트가 들어오고 back 은 27번 그리고 stop 모두 꺼버립니다.

전체 소스 첨부합니다.

실행은 터미널 창에서 아래와 같이 해주시면 됩니다.

sudo node 3.webapi.js

시연 동영상 주소:

<https://www.youtube.com/watch?v=cq1WzTh0zDI>

[전체소스]

//파일명 : 3.webapi.js

//작성자 : 도플광어

///////////////////////////////////////////

// GPIO

var Gpio = require('onoff').Gpio; // Constructor function for Gpio objects.

var led = [new Gpio(4, 'out'),new Gpio(17, 'out'),new Gpio(27, 'out')];

var iv;

// 0.5초 마다 깜박이기

iv = setInterval(function() {

    led[0].writeSync(led[0].readSync() === 0 ? 1 : 0); // 1 = on, 0 = off :)

}, 500);

////////////////////////////////////////////////////////////////

//서버

var http = require('http');

var gPort = 9010;

http.createServer(function (req, res) {

    var ip = req.headers['x-forwarded-for']|| req.connection.remoteAddress;

    console.log('remote address:' + ip);

    res.writeHead(200, {'Content-Type': 'text/plain'});

    if(req.url == '/favicon.ico') {

    }

    else {

        console.log(req.url);

        switch(req.url) {

            case '/go':

                led[1].writeSync(1);

                led[2].writeSync(0);

                res.write('success call api : '+ req.url);

                break;

            case '/back':

                led[1].writeSync(0);

                led[2].writeSync(1);

                res.write('success call api : '+ req.url);

                break;

            case '/stop':

                led[1].writeSync(0);

                led[2].writeSync(0);

                res.write('success call api : '+ req.url);

                break;

            default:

                res.write('error api : '+ req.url);

                break;

        }

    }

    res.end();

}).listen(gPort);

console.log('Server running at : ' + gPort);

////////////////////////////////////////////////////////////////////

|  |
| --- |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 이 작성자의 게시글  [구독하기](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)  |[더보기](http://cafe.naver.com/CafeMemberNetworkView.nhn?m=view&clubid=25572101&memberid=gbox2)   |  |  | | --- | --- | |  | | | [**gbox3d.tistory.com**](http://blog.naver.com/gbox2) | [http://blogpfthumb.phinf.naver.net/data18/2007/1/24/218/061_1-4505-gbox2.jpg?type=s40](http://blog.naver.com/gbox2) | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **댓글 9** | | | [**등록순http://cafeimgs.naver.net/cafe4/hidden.gif**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | | | **조회수 1504** | | | | **좋아요** [**0**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |
| * [블로그](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) * [카페](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) * [북마크](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) * [보내기](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | | | [인쇄](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) | | | 신고 |

|  |  |
| --- | --- |
| http://cafeimgs.naver.net/img/emot/emo11.gif | [**지능도시**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.18. 21:02[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 혹시   
  리부팅 시에 자동으로 node.js http server를 실행하는 것을   
  성공하셨나요?   
    
  자체적으로는 잘 실행되는데요.   
  부팅시에 자동으로 실행하기가 쉽게 될 것 같았는데   
  실제로 해보니 잘 안되네요.

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.18. 22:13[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 다음강좌로 준비해 보겠습니다^^

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/59/56/985659.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=985659) | [**표윤석**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.18. 21:16[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 완전 초 간단이네요. 이렇게 된다는게 놀랍네요!! 이렇게 심플하면서 직관적이라니... 멋집니다! 저도 진행중인 tcp/ip 제어가 있는데 이쪽으로 검토해봐야겠네요!! 좋은 정보 감사합니다!

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.18. 22:18[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 감사합니다^^;  
  요즘 일이 바뻐서 얻어가기만하고 강좌를 좀 쉬었네요. 이번주부터 가급적 한주에 하나씩 올려보도록할께요^^;

|  |  |
| --- | --- |
| http://cafeimgs.naver.net/img/emot/emo11.gif | [**한풀**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.23. 07:26[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 기대됩니다. 도플광어님! ^^ 그리고 감사합니다. ^^

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.24. 18:57[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 도움이 되신다면 기쁘죠^^;
* 삭제된 댓글입니다.

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.24. 18:57[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 감사합니다^^;
* 삭제된 댓글입니다.

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/43/15/1131543.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1131543) | [**도플광어**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2013.05.24. 18:56[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* GPS 관련 수학 알고리즘때문이신건가요?

|  |  |
| --- | --- |
| [http://itemimgs.naver.net/personacon/9/38/1443809.gif](http://item.naver.com/personacon/PersonaconShop.jsp?Redirect=PersonaconSub.jsp?type=itemdetail%26itemseq=1443809) | [**water is life**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1) |

* 2014.02.20. 11:24[답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* [신고](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)
* 아호~ 감사합니다 ^^ 잘되네요

|  |  |  |
| --- | --- | --- |
| http://itemimgs.naver.net/personacon/64/40/1114064.gif |  |  |
| [스티커 댓글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)▲ NEW 이제 PC에서도 스티커로 댓글을 남겨보세요! | | |

[**[http://cafeimgs.naver.net/cafe4/ico-btn-write.gif](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)글쓰기**](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)

[[http://cafeimgs.naver.net/cafe4/ico-btn-check.gif](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)답글](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)

[목록](http://cafe.naver.com/ArticleRead.nhn?clubid=25572101&articleid=876&networkMemberId=gbox2&networkSearchKey=Article&networkSearchType=7&networkSearchPage=1)

**[출처]** [컨버젼스를 위한 node.js 강좌 - 3. webapi 만들기 (오픈소스 소프트웨어 & 하드웨어: 로봇 기술 공유 카페 (오로카))](http://cafe.naver.com/openrt/876) |**작성자** [도플광어](http://cafe.naver.com/openrt.cafe?iframe_url=/CafeMemberNetworkView.nhn%3Fm=view%26memberid=gbox2)

<http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/>

# Raspberry Pi, Node.js and a LED – Emit Morse Code

This post is part 2 of 9 in the series [Raspberry Pi](http://thejackalofjavascript.com/series/rpi/)

[](http://twitter.com/share?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&text=Raspberry+Pi%2C+Node.js+and+a+LED+%E2%80%93+Emit+Morse+Code+From+The+Jackal+of+Javascript+by+%40arvindr21)8[](http://www.linkedin.com/shareArticle?mini=true&url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)1[](https://plus.google.com/share?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)0[](http://reddit.com/submit?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&title=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code)0[](https://bufferapp.com/add?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&text=From%20The%20Jackal%20of%20Javascript%20Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code)[](https://flattr.com/submit/auto?user_id=arvindr21&title=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code&url=https://flattr.com/profile/arvindr21)[](mailto:?subject=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%26%238211;%20Emit%20Morse%20Code&body=From%20The%20Jackal%20of%20Javascript%20http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)[](http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)

In this post, we will get our hands a bit dirty with Raspberry Pi, Node.js and a LED. We will be writing a simple program in Node.js to turn a LED on and off. Then we will extend this logic to build a simple piece of embedded system that can take a piece of string and emit its morse code.

Interesting right? Below is a quick video as what we will be building

The above LED is emitting the Morse code for SOS. That is (… \_\_\_ …) [Dot Dot Dot, Dash Dash Dash, Dot Dot Dot]. Nice right?

You can get the completed code [here](https://github.com/arvindr21/Pi_MorseCode).

So, let us get started.

## Prerequsites

If you are new to Raspberry pi and have not yet installed Node.js on it, I would recommend going through [Getting Started with Raspberry pi and Node.js](http://thejackalofjavascript.com/getting-started-raspberry-pi-node-js/).

If you are new to electronics devices and circuits, I would recommend going through the[video lectures](http://www.allaboutcircuits.com/videos/index.html) from All About Circuits.

### Components needed

1. 1 – Raspberry pi B+
2. 1 – Breadboard
3. 1 – 68 Ohms resistor
4. 1 – LED
5. 2 – Female to male wires

## What is GPIO?

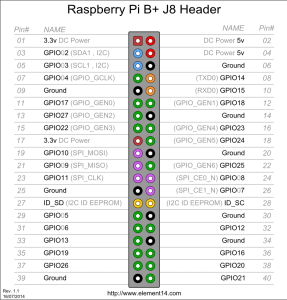
GPIO stands for General Purpose Input Output pins.

These pins are a physical interface between the Pi and the outside world. At the simplest level, you can think of them as switches that you can turn on or off (input) or that the Pi can turn on or off (output).

Inputs don’t have to come from a physical switch; it could be input from a sensor or a signal from another computer or device, for example. The output can also do anything, from turning on an LED to sending a signal or data to another device. If the Raspberry Pi is on a network, you can control devices that are attached to it from anywhere\*\* and those devices can send data back.

You can read more about Raspberry Pi GPIO [here](http://www.raspberrypi.org/documentation/usage/gpio/).

Raspberry pi B+ has 26 GPIO pins.

[](http://i2.wp.com/thejackalofjavascript.com/wp-content/uploads/2014/10/GPIO.png)

The remaining ones are either power or ground pins.

Below is the superimposed image of the pin layout on a pi

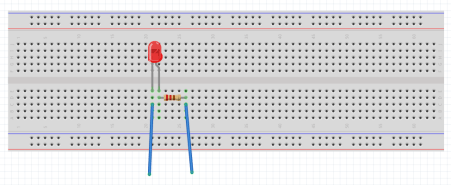
[](http://i0.wp.com/thejackalofjavascript.com/wp-content/uploads/2014/10/pi.png)

## Turn On Turn Off

Now that we have a basic understanding of GPIO, we will leverage one of pi’s GPIO pins to turn a LED on and off.

There are 2 parts to getting this task done. Part 1 is building a circuit that has a LED connected to the pi. Part 2 is the Javascript code that triggers the state of the LED.

So, let us focus on Part I. We will build the circuit. For this we will assemble the following schematic on a breadboard. 

[](http://i0.wp.com/thejackalofjavascript.com/wp-content/uploads/2014/10/Screen-Shot-2014-10-03-at-5.33.02-pm.png)The left end of the above circuit will be connected to Pin 9 – GND pin of pi (Left Column, 5th pin). And the right end of the above circuit will be connected to Pin 11(Left Column, 6th pin).

This completes our circuit. Now to the application logic. Connect the pi to your computer. And ssh from terminal/putty into pi. We will be using command line to tools to write the code.

So, as soon as you ssh into pi, you will be landing inside the /home/pi folder. We will create a new folder here named node\_programs. And inside this folder, we will be maintaining all our programs. Run

mkdir node\_programs

To step inside that folder, run

cd node\_programs

For this post, we will create a new folder named ledBlink and will step inside this folder. Run

mkdir ledBlink && cd ledBlink

Note : You can run multiple commands separated by a &&.

First we will initialize a new node project here. Run

npm init

Fill it up as applicable.

Now, we will use a node module named [*onoff*](https://www.npmjs.org/package/onoff) to interact with the GPIO pins from inside our Node.js code. This is the simplest way to interact with the pi from our node program. Run

npm install onoff --save

Next, we will create a new file named index.js inside the ledBlink folder. To create a new file, you can follow any of the 2 ways. First

touch index.js

This will create an empty file named index.js. Second

nano index.js

This will create a new file named index.js and will open the file for editing in the built in nano editor. Even though you are following the first approach, to edit the file, you need to run step 2.

Once the file is opened in the nano editor, paste the below.

~/node\_programs/ledBlink/index.js

JavaScript



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | var Gpio = require('onoff').Gpio,      led = new Gpio(17, 'out');    var iv = setInterval(function(){  led.writeSync(led.readSync() === 0 ? 1 : 0)  }, 500);    // Stop blinking the LED and turn it off after 5 seconds.  setTimeout(function() {      clearInterval(iv); // Stop blinking      led.writeSync(0);  // Turn LED off.      led.unexport();    // Unexport GPIO and free resources  }, 5000); |

**Things to notice**

**Line 1** : We include the GPIO module of onoff to interact with pi’s GPIO pins

**Line 2** : We tell our program that we are going to use a GPIO pin and it is of the type output. That is pi provides the input to that pin based on the logic we write. We will name the output of this pin as led, as logically we have a led placed here.

**Line 4** : We start a new setInterval()  to run the code inside it every 500 milliseconds.

**Line 5** : For every Gpio pin, we have a method named  writeSync(). This method takes in a 1 or a 0. If it is 1, it will command the pi to supply 3.3v to this pin and if it is 0, it will command the pi to stop the supply. This way, we control the pin.

And  led.readSync() tells us if this pin is in state 1 or 0. depending on that, we will toggle the values. That is if the value is 0, we set the value to 1 and if 1, we set the value to 0. Thus making the LED “blink” every 500ms. Simple right?

**Line 13** : After 5 seconds, we terminate the   setInterval() and clear the state of the pin and release all the resources.

Let us save the file and run the program. To save the program, press (ctrl+x or cmd+x). This will ask you to save the file. Press Y and press enter key to complete the operation.

To run the program, execute

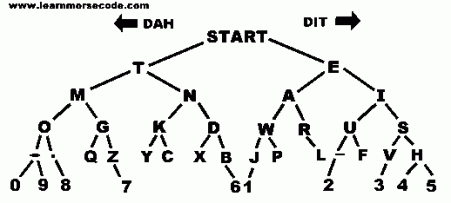
sudo node index.js

And you should see the LED blinking for 5 seconds before the program exits. Simple right. Now you can see the power of pi. How it can interact with the outside world and control stuff.

## Morse Code

**Morse code** is a method of transmitting text information as a series of on-off tones, lights, or clicks that can be directly understood by a skilled listener or observer without special equipment.

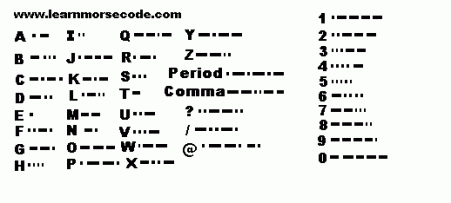
If you are new to Morse code, checkout [learnmorsecode.com](http://www.learnmorsecode.com/).

[](http://i0.wp.com/thejackalofjavascript.com/wp-content/uploads/2014/10/morse_code.gif)

This is how you read the above diagram

Place your pencil where it says START and listen to morse code.  
Move down and to the right every time you hear a DIT (a dot).  
Move down and to the left every time you hear a DAH (a dash).

So this will create a unique code for each of the alphabets, numbers and a few special characters. Like

[](http://i1.wp.com/thejackalofjavascript.com/wp-content/uploads/2014/10/morse_code_final.gif)

A simple and efficient language!

Now, we will write a program that will emit the Morse Code for a given string. To keep the blinking at a constant rate and in our control, we will use a node module named sleep. Run

npm install sleep --save

Now, we will create a new file named morseCode.js, inside the ledBlink folder. Run

nano morseCode.js

Now, paste the below code to morseCode.js

~/node\_programs/ledBlink/morseCode.js

JavaScript



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87 | var Gpio = require('onoff').Gpio,     led = new Gpio(17, 'out'),     sleep = require('sleep'),     \_baseTime = 128000, //micro seconds     sleepTime = \_baseTime,     btwCodes = \_baseTime \* 2,     btwLetters = \_baseTime \* 4,     btwWords = \_baseTime \* 8;      var text = (process.argv[2] ? process.argv[2] : '').toLowerCase();    var MorseCode = {     pattern: {        'a': '.\_',        'b': '\_...',        'c': '\_.\_.',        'd': '\_..',        'e': '.',        'f': '..\_.',        'g': '\_\_.',        'h': '....',        'i': '..',        'j': '.\_\_\_',        'k': '\_.\_',        'l': '.\_..',        'm': '\_\_',        'n': '\_.',        'o': '\_\_\_',        'p': '.\_\_.',        'q': '\_\_.\_',        'r': '.\_.',        's': '...',        't': '\_',        'u': '..\_',        'v': '...\_',        'w': '.\_\_',        'x': '\_..\_',        'y': '\_.\_\_',        'z': '\_\_..',        '1': '.\_\_\_\_',        '2': '..\_\_\_',        '3': '...\_\_',        '4': '....\_',        '5': '.....',        '6': '\_....',        '7': '\_\_...',        '8': '\_\_\_..',        '9': '\_\_\_\_.',        '0': '\_\_\_\_\_'     },     active: function(t) {        led.writeSync(1);     },     inactive: function() {        led.writeSync(0);     }  }    var \_t = text.split('');    for(var i = 0; i < \_t.length; i++) {     var \_l = \_t[i];       if(\_l == ' ') { // if the char is a space        sleep.usleep(btwWords);     }     else {        var \_c = MorseCode.pattern[\_l].split('');        sleep.usleep(btwLetters);        console.log('Letter Starts >> ', \_l);        for(var j = 0; j < \_c.length; j++) {           console.log("code >> ", \_c[j]);           MorseCode.active();           if(\_c[j] == '.') {              sleep.usleep(sleepTime);              MorseCode.inactive();              sleep.usleep(btwCodes);           }           else {              sleep.usleep(sleepTime \* 3);              MorseCode.inactive();              sleep.usleep(btwCodes);           }        }     }  }; |

**Things to notice**

**Line 1,3** : require node modules

**Line 2** : Declare the pin to be a out.

**Line 4-8** : A base time, based on which the speed of transmission/blinking can be controlled.

sleepTime – Time before which we reset the state of the LED to 0

btwCodes – Time between 2 dits or dahs or a dit and a dah for a given letter.

btwLetters – Time between each letter

btwWords – Time between each word

This way, we can control the speed of transmission/LED blinking.

**Line 11** : We read the text from command line

**Line 14-50** : We build an object with the mapping of character and its morse code.

**Line 53** : A helper function to make the the LED active

**Line 56** : A helper function to make the the LED inactive

**Line 60/61** : We will split the input string and iterate over it

**Line 65/66** : If the current character is a space, we will sleep for “ btwWords”

**Line 69** : We split the Morse code for a character and check if it is a dit or a dah.

**Line 76** : If the code is a dit, we sleep for “ sleepTime”. This is time that the LED stays lit.

**Line 81** : If the code is a dah, we sleep for “ sleepTime \* 3”. This is time that the LED stays lit.

**Line 77/82** : We reset the state of the LED

**Line 78/83** : We wait for “ btwCodes” before we start processing the next code.

Simple and easy right! Save the file by pressing ctrl + x and then Y and then enter key.

To run the program execute

sudo node morseCode.js "Hello World"

And you should see the below log in the console.

sudo node morseCode.js

Shell



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43 | pi@raspberrypi ~/node\_programs/ledBlink $ sudo node morseCode.js "Hello World"  Letter Starts >>  h  code >>  .  code >>  .  code >>  .  code >>  .  Letter Starts >>  e  code >>  .  Letter Starts >>  l  code >>  .  code >>  \_  code >>  .  code >>  .  Letter Starts >>  l  code >>  .  code >>  \_  code >>  .  code >>  .  Letter Starts >>  o  code >>  \_  code >>  \_  code >>  \_  Letter Starts >>  w  code >>  .  code >>  \_  code >>  \_  Letter Starts >>  o  code >>  \_  code >>  \_  code >>  \_  Letter Starts >>  r  code >>  .  code >>  \_  code >>  .  Letter Starts >>  l  code >>  .  code >>  \_  code >>  .  code >>  .  Letter Starts >>  d  code >>  \_  code >>  .  code >>  . |

And the LED will blink according to this.

Below is a recording of the program with “Morse Code” as input

Hope this post gave you a decent idea on GPIO and how you can control them.

Thanks for reading! Do comment.  
@arvindr21

Series Navigation[<< Getting Started with Raspberry pi and Node.js](http://thejackalofjavascript.com/getting-started-raspberry-pi-node-js/)[Raspberry pi, a Buzzer and Node.js – An IoT Doorbell >>](http://thejackalofjavascript.com/rpi-buzzer-node-iot-doorbell/)

[](http://twitter.com/share?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&text=Raspberry+Pi%2C+Node.js+and+a+LED+%E2%80%93+Emit+Morse+Code+From+The+Jackal+of+Javascript+by+%40arvindr21)8[](http://www.linkedin.com/shareArticle?mini=true&url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)1[](https://plus.google.com/share?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)0[](http://reddit.com/submit?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&title=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code)0[](https://bufferapp.com/add?url=http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/&text=From%20The%20Jackal%20of%20Javascript%20Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code)[](https://flattr.com/submit/auto?user_id=arvindr21&title=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%E2%80%93%20Emit%20Morse%20Code&url=https://flattr.com/profile/arvindr21)[](mailto:?subject=Raspberry%20Pi,%20Node.js%20and%20a%20LED%20%26%238211;%20Emit%20Morse%20Code&body=From%20The%20Jackal%20of%20Javascript%20http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)[](http://thejackalofjavascript.com/raspberry-pi-node-js-led-emit-morse-code/)

### Like this: