

RASPBERRY PI 3 Experiment-Set





Sehr geehrter Kunde,

Wir freuen uns, dass Sie mit Ihren Kauf uns Ihr Vertrauen und die Möglichkeit gegeben haben die Vielseitigkeit und Qualität eines renkforce Produktes zu beweisen. Wir wünschen Ihnen viel Freude mit dem Raspberry Pi 3 Experiment-Set

Das Experiment-Set bildet die perfekte Grundlage für Ihren Raspberry. Starten Sie mit diesem umfangreichen Zubehör-Paket Ihre Experimente und Projekte. Lassen Sie Ihrer Kreativität freien Lauf und führen Sie mit den Steckbrückenkabeln, Widerständen, LEDS und Schalter verschiedenste Versuche durch. Eine Verbindung zwischen Raspberry Pi[®] 3 und der Steckplatine lässt sich mit dem Cobbler-Verbindungskit realisieren. Mit dem USB TTL-Kabel können Sie Ihren Einplatinencomputer sogar mit einem PC verbinden.

Das Gehäuse bietet der Platine den nötigen Schutz und ermöglicht Zugriff auf die Schnittstellen. Die Stromversorgung übernimmt ein Mico-USB Nezteil mit 2000 mA Ausgangsstrom und einer Wechselclip-Funktion für den internationalen Einsatz. Mit dem vorinstallierten Noobs Betriebssystem auf der MicroSD-Karte können Sie sofort losstarten. Eine Verbindung mit einem Monitor oder sogar dem Fernseher kann über das beiliegende HDMI[™]-Kabel hergestellt werden.

Die absolute Vielseitigkeit zeichnen das Raspberry Pi[®] Set desweiteren aus. Setzen Sie es als Mini-PC in Ihrem Wohnzimmer ein und geben Sie Videos in Full HD über den HDMI[™]-Ausgang wieder oder verbinden Sie ihn mit weiteren Platinen über den GPI0-Port um Steuerungen zu realisieren.

In der folgenden Anleitung haben wir einzelne Beispiele aufgeführt, um Sie in den ersten Schritten mit dem Raspberry Pi 3 Experiment Set zu begleiten und einen reibungslosen Einstieg zu ermöglichen.



Lieferumfang

- Raspberry Pi[®] 3 Model B 1 GB
- Gehäuse
- Raspberry Pi Micro-USB-Netzteil 2.5 A mit Wechselclips für Europa, Großbritannien, Australien, USA
- Noobs Betriebssystem auf 8 GB MicroSD-Karte
- Kühlkörper-Set
- HDMI[™]-Kabel
- Cobbler-Verbindungskit
- USB TTL-Kabel
- Steckplatine
- Steckbrücken-Verbindungskabel
- 4x 10 mm LEDs
- 2x Schaltermodule
- Relais
- IR-Bewegungsmelder



Vorbereitung des Raspberry Pi

Um mit den Raspberry Pi die folgenden Beispiele nutzen zu können, muss dieser vorab vorbereitet werden. Im Lieferumfang finden Sie neben dem Raspberry Pi 3 auch ein passendes Gehäuse aus Acryl-Glas, welche eine eigene Anleitung für den Zusammenbau beinhaltet.



Nachdem Sie das Gehäuse mit eingesetzten Raspberry Pi zusammengebaut haben, setzten Sie die beiliegende microSD-Karte in den entsprechenden Slot an der Unterseite der Platine ein.

Verbinden Sie daraufhin per HDMI-Kabel den Raspberry Pi an einen HDMI-fähigen Bildschirm/Fernseher und schließen Sie an den USB-Buchsen eine handelsübliche Computermaus und Tastatur an (Alternativ: drahtlos Maus-/Tastaturempfänger) Das in diesem Set enthaltende microUSB Netzteil, hat die Möglichkeit mit verschiedenen Länderadaptern ausgestattet und somit rund um den Globus eingesetzt zu werden. Setzten Sie hier den zu Ihrer Wandsteckdose passenden Adapter auf das Netzteil auf und verbinden Sie das Netzteil auf der einen Seite an die microUSB-Buchse des Raspberry Pi und auf der anderen Seite mit der Wandsteckdose.

Der Raspberry Pi 3 sollte nun starten und ein Bild auf dem HDMI-Bildschirm ausgeben.



Vorbereitung des Betriebsystemes

Die beiliegende micro-SD Karte beinhaltet das Betriebssystem-Installationsprogramm "NOOBS", welches erlaubt mehrere verschiedene Betriebssysteme auf dem Raspberry Pi zu nutzen—unter anderem auch als Office-PC, als Programmierumgebung oder als Multimediacenter.

Sollten Sie den Raspberry Pi zusätzlich mit einem Netzwerkkabel an das Internet angeschlossen haben, sollten Sie eine aktuelle Auswahl an den installierbaren Betriebsystemen angezeigt bekommen.

Darunter befindet sich auch das Betriebsystem "Raspbian", welches eine für den Raspberry Pi angepasste Version der Linux-Distribution Debian ist (diese wird Ihnen auch ohne Online-Zugang angezeigt, da dieses auf dieser SD-Karte schon vorbereitet ist).

Für die folgenden Beispiele empfehlen wir auf Raspbian zu setzen und zeigen im folgenden auf, wie dieses zu installieren ist:



Wählen Sie zu allererst das Betriebssystem aus, indem Sie das Kreuz in das vorgesehene Kästchen mit einem Linksklick setzen—danach starten Sie die Installation mit einem Drücken auf den Punkt "Install (i)"



9					
stall (i)	Edit config (e)	Wifi networks (w)	Online help (h)	Exit (Esc)	
• (0	Raspbian [REC A community-	COMMENDED] created port of Del	bian jessie for the	Raspberry Pi	
-		_			
	Warning: this data on the S already instal	will install the sele D card will be over led.	nfirm ected Operating S written, including	ystem(s). All ex any OSes tha	kisting t are
<u>.</u>	Warning: this data on the S already instal	Co will install the sele D card will be over led. <u>Y</u> es	nfirm ected Operating S written, including <u>N</u> o	ystem(s). All ex any OSes tha	kisting t are
	Warning: this data on the S already instal	Co will install the sele D card will be over led.	nfirm ected Operating S; written, including	ystem(s). All ex a any OSes tha	kisting t are
Disk space	Warning: this data on the S already instal	Co will install the sele D card will be over led.	nfirm ected Operating S written, including	ystem(s). All ex	kisting t are

Die darauffolgende Meldung bestätigen Sie mit einen Drücken auf "Yes"



Danach wird die Installation gestartet; diese dauert in der Regel 10-15 Minuten.





Nach der Installation können wir diese mit einem Drücken auf OK beenden.



Danach bootet der Raspberry Pi in das Betriebssystem Raspbian.



Kommunikation mit dem PC über die serielle Schnittstelle

In diesem Set enthalten finden Sie ein USB-TTL Adapterkabel. Dieses ermöglicht Ihnen mit diversen Mikrokontroller– und Einplatinencomputersystemen zu kommunizieren—sei es die Überprüfung von Messwerten, Eingabe von Parametern oder die Steuerung des Betriebssystems über eine Terminalkonsole. Letztere Möglichkeit ist im Betriebsystem Raspbian des Raspberry Pi enthalten.



Über die serielle Schnittstelle auf den Pins 8 [TXD] und 10 [RXD] auf dem GPIO-Header des Raspberry Pi, lässt sich somit eine Verbindung zur Terminalkonsole aufbauen und Befehle für das Betriebsystem eingeben, ohne dass eine Verbindung per Netzwerk aufgebaut oder ein Bildschirm per HDMI angeschlossen werden muss.

Hierzu gehen Sie wie folgt vor:

Schließen Sie das große Datenkabel auf die GPIO-Leiste auf, wie es auf dem Bild auf der rechten Seite gezeigt wird. <u>Achten Sie</u> <u>hierbei darauf, dass die rot markierte Seite</u> <u>des Kabels nach außen zeigt.</u>





Hiernach setzen Sie die Cobbler Verbindungplatine auf das Steckbrett auf und verbinden das Datenkabel, welches vom Raspberry Pi kommt.



Setzen Sie nun in die Verbindungsstücke des USB-TTL Verbindungskabels die Pin-Stiftadapter ein; diese Sind im beiliegenden Kabel-Set enthalten.

Nun haben Sie die Möglichkeit die Verbindungsstücke des USB-TTL Kabels in das Steckbrett zu stecken. Auf der Cobbler Verbindungsplatine, sehen Sie die entsprechenden Pinbezeichnungen des Raspberry Pi's aufgedruckt.

Die Verbindung muss nun wie folgt aufgebaut werden:





USB-TTL [TX] [Grün]- > Raspberry Pi [RX]USB-TTL [RX] [Weiß]- > Raspberry Pi [TX]USB-TTL [GND] [Schwarz]- > Raspberry Pi [GND]

*Das rote Kabel muss nicht angeschlossen werden



Hiernach können Sie das andere Ende des USB-TTL-Kabels in einen freien USB-Port Ihres PC stecken. Sollten Sie bislang den Raspberry Pi nicht vorbereitet haben, stecken Sie nun die beiliegende microSD Karte in den dazugehörigen Slot auf der Unterseite und verbinden Sie das microUSB-Netzteil an den microUSB Eingang des Raspberry Pi. Stecken Sie das Netzteil noch nicht in die Wandsteckdose an.

Die folgenden Schritte sind abhängig von Ihrem verwendeten Betriebsystem—im Folgenden zeigen wir auf, wie Sie die Terminal-Verbindung an einem Windows-PC durchführen können; die Vorgehensweise auf anderen Betriebsystemen sind jedoch in ihren Grundzügen ähnlich.

Für die Verbindung zur Terminalkonsole verwenden wir das Programm Putty. Dieses ist als Freeware unter http://www.putty.org/ zum Download verfügbar und wurde unter der MIT License veröffentlicht.

Laden Sie die "putty.exe" auf der o.g. Website herunter und starten Sie das Programm. Sie Sehen nun die folgende Maske—Dort müssen die beschriebenen Werte eingefügt bzw. geändert werden:

Session	Basic options for your PuTTY session
 Logging Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Connection Data Proxy Telnet Rlogin 	Specify the destination you want to connect to Serial line Speed COM7 1. 2. Connection type: Rlogin SSH Raw Telnet Rlogin SSH Load, save or delete a stored session 3. Saved Sessions Load Default Settings Load Save Save
E SSH Serial	Close window on exit: Always Never Only on clean exit



*Die unter Punkt 1 aufgezeigte COM-Nummer ist abhängig von Ihrem verwendeten System und kann variieren. Um die richtige COM-Nummer zu erhalten, gehen Sie in den Windows Gerätemanager und schauen nach, welche COM-Nummer dem USB-TTL Kabel vom Betriebsystem vergeben wurde. Hierzu gehen Sie im Gerätemanager unter "Anschlüsse"

🚔 Geräte-Manager —					
Datei Aktion Ansicht ?					
⇐ ⇒ 📰 🖺 🛛 🖬 🧶 🕼 😽					
V 📇 TECHNIK3					
 Anschlüsse (COM & LPT) 					
The Kommunikationsanschluss (COM1)					
Prolific USB-to-Serial Comm Port (COM7)					
> 4 Audio-, Video- und Gamecontroller					
> 🧃 Audioeingänge und -ausgänge					
> 🛄 Computer					
> 🛅 Druckwarteschlangen					
> 🕼 Eingabegeräte (Human Interface Devices)					

In unserem Fall ist dies COM7. Sollten Sie jedoch ein gelbes Ausrufezeichen neben dem "USB-to-Serial" Adapter-Symbol haben, wie im folgenden Bild…

🚔 Geräte-Manager
Datei Aktion Ansicht ?
V 🚔 TECHNIK3
✓ I Anschlüsse (COM & LPT)
Kommunikationsanschluss (COM1)
Trolific USB-to-Serial Comm Port (COM7)
> 4 Audio-, Video- und Gamecontroller
> 🐐 Audioeingänge und -ausgänge
> 💻 Computer
> 🚍 Druckwarteschlangen
> 🕼 Eingabegeräte (Human Interface Devices)
> 🌌 Grafikkarte
> 🥁 IDE ATA/ATAPI-Controller

... so ist die Treiber-Installation von Windows für den Adapter noch nicht vollständig. Um diese abzuschließen, folgen Sie bitte der Treiber-Installation Anleitung im folgenden Link.



Nach dieser Vorkonfiguration kann mit einem Drücken auf "Open" die serielle Verbindung aufgebaut werden.

Session	Basic options for your P	uTTY session
Logging Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Connection Data Draw	Specify the destination you want to Serial line COM7 Connection type: O Raw O Ielnet O Rlogin Load, save or delete a stored sess Saved Sessions Default Settings	o connect to Speed 115200 SSH Segial ion
Proxy - Telnet - Rlogin → SSH - Serial	Close window on egit: Always Never OC	Save Delete

Zu allererst wird Ihnen nur ein schwarzes Fenster gezeigt—nun können Sie das Netzteil in die Wandsteckdose stecken und somit den Raspberry Pi 3 starten. Nun sollte nach kurzer Zeit die Terminalausgabe beginnen

P C	COM7 - PuTTY		_		×
[2.878614]	<pre>[vc_sm_connected_init]: start</pre>			~
[2.879368]	<pre>[vc_sm_connected_init]: end - returning 0</pre>			
[2.879693]	of_cfs_init			
[2.879806]	of_cfs_init: OK			
[2.977397]	EXT4-fs (mmcblk0p7): INFO: recovery required on read	only	files	yst
em					
[2.989399]	EXT4-fs (mmcblk0p7): write access will be enabled du	ring	recov	ery
[3.153092]	usb 1-1: new high-speed USB device number 2 using dw	c_oto	1	
[3.164639]	Indeed it is in host mode hprt0 = 00001101			
[3.373395]	usb 1-1: New USB device found, idVendor=0424, idProd	uct=9	9514	
l	3.384905]	usb 1-1: New USB device strings: Mfr=0, Product=0, S	erial	Numbe	r=0
l	3.397820]	hub 1-1:1.0: USB hub found			
l	3.406430]	hub 1-1:1.0: 5 ports detected			
l	3.693011]	usb 1-1.1: new high-speed USB device number 3 using	dwc_c	otg	
l	3.803343]	usb 1-1.1: New USB device found, idVendor=0424, idPr	oduct	=ec00	
[3.814853]	usb 1-1.1: New USB device strings: Mfr=0, Product=0,	Seri	alNum	ber
=0					
Ļ	3.829756]	smsc95xx v1.0.4			
	3.897367]	<pre>smsc95xx 1-1.1:1.0 eth0: register 'smsc95xx' at usb-</pre>	3±980)000.u	sb-
1.1,	smsc95xx	USB 2.0 Ethernet, b8:27:eb:75:bc:82			
L	7.087085]	EXT4-fs (mmcblk0p7): orphan cleanup on readonly fs			
Ļ	7.098245]	EXT4-is (mmcblk0p/): 1 orphan inode deleted			
L	/.108321]	EXT4-is (mmcblk0p/): recovery complete			
					\sim



Nachdem das Betriebsystem komplett hochgefahren ist, wird als letzte Zeile "raspberrypi login:" angezeigt. Nun können Sie sich anmelden indem Sie "pi" als Benutzernamen und "raspberry" als passwort eingeben und die jeweiligen Eingaben mit Enter bestätigen (das Passwort wird bei der Eingabe nicht angezeigt; dieses muss man nach der Eingabe "blind" bestätigen)

2	COM7 - PuTTY	-		×
[[[9.842031] 9.893252] 9.905471] 9.922237]	<pre>systemd[1]: Starting Load Kernel Modules systemd[1]: Started Set Up Additional Binary Formats. systemd[1]: Starting Slices. systemd[1]: Reached target Slices.</pre>		^
[[]	9.943190] 9.944143] 9.960080]	systemd[1]: Mounted POSIX Message Queue File System. fuse init (API version 7.23) i2c /dev entries driver		
[[[9.979516] 9.997441] 10.028305] 10.047339]	<pre>systemd[1]: Mounted Debug File System. systemd[1]: Started File System Check on Root Device. systemd[1]: Started Increase datagram queue length. systemd[1]: Started Restore / save the current clock.</pre>		
[or [10.104581] the current 10.127820]	<pre>systemd[1]: Started Create list of required static device kernel. systemd[1]: Started Load Kernel Modules.</pre>	node	sf
[10.143540] 10.174795] 10.375912]	<pre>systemd[1]: Time has been changed systemd[1]: Started udev Coldplug all Devices. systemd[1]: Starting Apply Kernel Variables</pre>		
[[₽२ <i>व</i>	10.424142] 10.447125]	systemd[1]: Mounting Configuration File System systemd[1]: Mounting FUSE Control File System		
ras Pas	pbian GNO/L. pberrypi loo sword:	gin: pi		~

Danach sind Sie mit dem Raspbian Terminal per serieller Kommunikationsschnittstelle verbunden.

COM7 - PuTTY		_		\times
[9.948430] sy [10.005033] sy or the current ke	ystemd[1]: ystemd[1]: ernel.	Started Restore / save the current clock. Started Create list of required static devic	e nodes	f
[10.028233] sg [10.044603] sg [10.075177] sg [10.278575] sg [10.334262] sg	ystemd[1]: ystemd[1]: ystemd[1]: ystemd[1]:	Started user changed Started udev Coldplug all Devices. Starting Apply Kernel Variables Mounting Configuration File System		
[10.357715] Sy Raspbian GNU/Linu	ystema[1]: ux 8 raspbe n: pi	mounting FOSE Control File System		
Password: Last login: Fri M Linux raspberrypi	May 27 12:1 i 4.4.11-v7	7:05 UTC 2016 on tty1 + #888 SMP Mon May 23 20:10:33 BST 2016 armv	71	
The programs incl the exact distrik individual files	luded with bution term in /usr/sh	the Debian GNU/Linux system are free softwar s for each program are described in the are/doc/*/copyright.	e;	
Debian GNU/Linux permitted by app]	comes with licable law	ABSOLUTELY NO WARRANTY, to the extent		
pi@raspberrypi:~3	Ş 📕			\sim

Seite 13



Abfrage eines Buttons / Leuchten einer LED mittels Python



Schließen die beiliegenden Komponenten, wie auf dem folgenden Bild an

Pin-Verbindungsübersicht:

Button-Module:

- S1 ——> GPIO4 / #4 [Raspberry Pi]
- S2 ——> GPIO17 / #17 [Raspberry Pi]
- V ---> 3V3
- G ——> GND [Raspberry Pi]

LED-Module:

- S1 ——> GPIO22 / #22 [Raspberry Pi]
- S2 > nicht verbunden
- V ——> nicht verbunden
- G ——> GND [Raspberry Pi]



Nachdem alles angeschlossen ist, muss nun die Datei mit dem ausführenden Programm generiert werden. Dazu verbinden Sie sich mit dem Terminal des Raspbian Betriebssystems und geben den folgenden Befehl ein...

nano LED_Button_Test.py

... um die Datei "LED_Button_Test.py" zu erstellen. Nun befinden Sie sich im Editor. Im folgenden sehen Sie ein Code-Beispiel geschrieben in der Code-Sprache Python. Dieses muss nun in den Editor kopiert werden.

```
# Benoetigte Module werden importiert und eingerichtet
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
# Hier wird der Eingangs-Pin deklariert, an dem der Button angeschlossen ist.
Button S1 PIN = 4
# Hier wird der Ausgang-Pin deklariert, an dem die LED angeschlossen ist.
LED PIN = 22
GPIO.setup(Button S1 PIN, GPIO.IN)
GPIO.setup(LED PIN, GPIO.OUT)
print "Button-LED-Test [druecken Sie STRG+C, um den Test zu beenden]"
# Hauptprogrammschleife
try:
  while True:
    if (GPIO.input(Button S1 PIN) = = 1):
      GPIO.output(LED PIN,1)
    else:
       GPIO.output(LED PIN,0)
# Aufraeumarbeiten nachdem das Programm beendet wurde
except KeyboardInterrupt:
  GPIO.cleanup()
```

* Vorsicht beim kopieren/abschreiben: Bei der Programmiersprache Python sind die Einrückungen von Bedeutung; dadurch wird eine Programmstruktur aufgebaut*

Nachdem man den Code kopiert hat, kann man den Editor mit der Tastenkombination "Strg+X" beenden. Man wird dann gefragt, ob man die Datei speichern möchte, was man mit "Y" bestätigen muss.



Nachdem man nach dem Schließen des Editors wieder im Terminal gelangt, kann man nun das Programm mit folgenden Befehl starten:

sudo python LED_Button_Test.py

Wird nun der rechte Button1 gedrückt, so leuchtet die LED auf.



Das Programm kann man danach mit Drücken der Tastenkombination "Strg+C" beenden.



Erweiterung des Beispiels um eine zweite LED



Schließen die beiliegenden Komponenten, wie auf dem folgenden Bild an

Pin-Verbindungsübersicht:

Button-Module:

- S1 ——> GPIO4 / #4 [Raspberry Pi]
- S2 ——> GPIO17 / #17 [Raspberry Pi]
- V ---> 3V3
- G ——> GND [Raspberry Pi]

LED-Module-Blau:

- S1 ——> GPIO22 / #22 [Raspberry Pi]
- S2 ——> nicht verbunden
- V ——> nicht verbunden
- G ——> GND [Raspberry Pi]

LED-Module-Rot:

- S1 ——> GPIO24 / #24 [Raspberry Pi]
- S2 ——> nicht verbunden
- V ——> nicht verbunden
- G ——> GND [Raspberry Pi]



Das vorherige Beispiel kann, da auf dem Button-Modul zwei Knöpfe vorhanden sind, um eine zweite LED erweitert werden. Hierzu geben Sie folgenden Befehl...

nano 2LED_2Button_Test.py

... in die Konsole ein, um die Datei "2LED_2Button_Test.py" zu erstellen. Das Code-Beispiel von vorhin, ist im Folgenden in einer abgeänderten Fassung, in der der zweite Knopf und die zweite LED integriert sind.

Benoetigte Module werden importiert und eingerichtet import RPi.GPIO as GPIO import time GPIO.setmode(GPIO.BCM) # Hier wird der Eingangs-Pin deklariert, an dem der Button angeschlossen ist. Button S1 PIN = 4Button S2 PIN = 17# Hier wird der Ausgang-Pin deklariert, an dem die LED angeschlossen ist. LED BLAU PIN = 22LED ROT PIN = 24 GPIO.setup(Button S1 PIN, GPIO.IN) GPIO.setup(Button S2 PIN, GPIO.IN) GPIO.setup(LED BLAU PIN, GPIO.OUT) GPIO.setup(LED ROT PIN, GPIO.OUT) print "Button-LED-Test [druecken Sie STRG+C, um den Test zu beenden]" # Hauptprogrammschleife try: while True: if (GPIO.input(Button S1 PIN) = = 1): GPIO.output(LED BLAU PIN,1) else: GPIO.output(LED BLAU PIN,0) if (GPIO.input(Button S2 PIN) = = 1): GPIO.output(LED ROT PIN,1) else: GPIO.output(LED_ROT_PIN,0) # Aufraeumarbeiten nachdem das Programm beendet wurde except KeyboardInterrupt: GPIO.cleanup()



Nachdem man nach dem Schließen des Editors wieder im Terminal gelangt, kann man nun das Programm mit folgenden Befehl starten:

sudo python 2LED_2Button_Test.py

Wird nun der rechte Button1 gedrückt, so leuchtet die blaue LED auf.



Wird der linke Button2 gedrückt, so leuchtet nun die rote LED auf.





Knopf durch Bewegungsmelder ersetzen



Schließen die beiliegenden Komponenten, wie auf dem folgenden Bild an

Pin-Verbindungsübersicht:

PIR-Modul:

- S1 ——> GPIO4 / #4 [Raspberry Pi]
- S2 > nicht verbunden
- V ---> 3V3
- G ——> GND [Raspberry Pi]

LED-Module-Rot:

- S1 ——> GPIO24 / #24 [Raspberry Pi]
- S2 > nicht verbunden
- V ——> nicht verbunden
- G ——> GND [Raspberry Pi]



Anstatt, dass ein Knopf gedrückt werden muss um die LED leuchten zu lassen, kann auch das PIR-Modul verwendet, was die Funktion eines Bewegungsmelders beinhaltet. Geben Sie folgenden Befehl...

nano PIR_LED_Test.py

... in die Konsole ein, um die Datei "PIR_LED_Test.py" zu erstellen und kopieren Sie das unten stehende Codebeispiel in den Editor.





Nachdem man nach dem Schließen des Editors wieder im Terminal gelangt, kann man nun das Programm mit folgenden Befehl starten:

sudo python PIR_LED_Test.py

Wird nun vor dem PIR-Modul eine Bewegung erkannt, so leuchtet die LED auf.



Die maximale Reichweite der Bewegungserkennung kann über den Empfindlichkeitsregler an der Unterseite des PIR-Moduls mittels eines Kreuzschraubenziehers zwischen 3m und 6m eingestellt

werden. (Das Einstell-Potentiometer hat hierbei kein Anfangs- und Endpunkt)

