

Anleitung Nachbau HM-Sen-LI-O (mit BH1750)

•

Beitrag von [Bratmaxe](#) » 09.05.2018, 16:44

Ich habe hier mal ein Bild zusammengestellt, wie man den **HM-Sen-LI-O** Sensor nachbauen kann. Ich denke das das vielen helfen wird den Nachbau zu erstellen.

Anmerkung:

Ich habe die Software nicht erstellt, das waren viele andere fleißige Mitglieder dieses Forums und von Extern.

Der Dank geht daher an alle, die sich die Mühe machen und hier Ihre Freizeit "opfern" um uns den Nachbau zu ermöglichen bzw. zu erleichtern.

Ich habe die Beschreibung bewusst der Beschreibungen von jp112sdl angepasst, so das man sich schnell zurecht findet:

benötigte Hardware:

1x Arduino Pro Mini ATmega328P (3.3V / 8MHz)

1x CC1101 Funkmodul (868 MHz)

1x FTDI Adapter (zum Flashen)

1x Taster

1x LED (optional)

1x Widerstand 330 Ohm (optional)

1x Lichtsensor BH1750

[https://m.de.aliexpress.com/item/327228 ... 2722878671](https://m.de.aliexpress.com/item/327228...2722878671)

Draht

Platine:

Die Universalplatine von deimos habe ich in meinem Fall verwendet, um alles etwas aufgeräumter wirken zu lassen.

<https://github.com/alexreinert/PCB>

Akkulaufzeit:

Um die Batterielebensdauer zu erhöhen, ist es unbedingt notwendig, die grüne LED vom Arduino Pro Mini zu entfernen!

Die LED und den 330R Widerstand kann man ebenfalls weglassen, um noch ein wenig mehr Akku zu sparen, man sieht dann allerdings nicht, das der Arduino "lebt".

Um die Batterie/den Akku weiter zu schonen, kann die Messgenauigkeit runtergesetzt werden.

Standard ist eine Genauigkeit von 0,5Lx eingestellt, was ca 120ms Messzeit bedeutet.

Stellt man die Genauigkeit auf 4Lx, sinkt die Messzeit auf 16ms, was ganze 86% Reduktion der Messzeit bedeutet und somit auch Zeit, die der Arduino nicht aktiv sein muss. Da selbst ein bedeckter Wintertag ca. 3500Lux hergibt, sollten für den normalen Hausgebrauch die 4Lx Genauigkeit vollkommen ausreichend sein (entspricht 0,11% Abweichung).

Hierfür muss im Sketch die Zeile:

Code: [Alles auswählen](#)

```
Bh1750<>      bh1750;
```

durch die Zeile

Code: [Alles auswählen](#)

```
Bh1750<0x23, ::BH1750::Mode::CONTINUOUS_LOW_RES_MODE>      bh1750;
```

ersetzt werden. (Nochmals danke an jp112sdl für diese Info)

Der LDO auf dem Aduino Board kann/sollte ebenfalls entfernt werden.

Je nach Arduino Board (Version) kann man hierfür die Lötbrücke öffnen, oder muss das IC ablöten (wenn keine Lötbrücke vorhande).

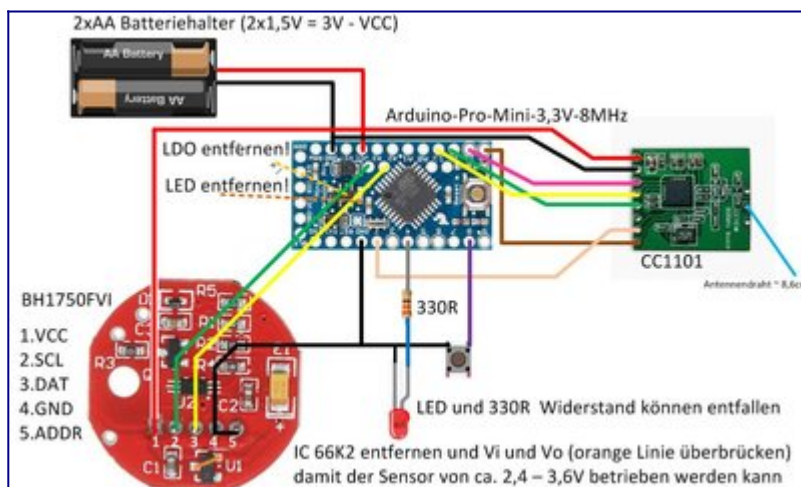
Ich habe die Batterielaufzeit mit 2xAA Batterien mit 2000mAh und 120 Sekunden Sendeabstand mal in einer Excel Tabelle durchgerechnet.

Arduino mit LED und LDO und BH1750 mit LDO - ca. 2,5 Monate

Arduino ohne LED, ohne LDO und BH1750 ohne LDO - ca. 2,2 Jahre

Man sieht also, das es sich durchaus lohnt die "Sparmaßnahmen" auch durchzuführen!

Anschaltung:



Schaltung

LDO 1750 brücken:



Beim BH1750 wurde ein LDO (662K) verwendet, damit der BH1750 mit bis zu 5 V betrieben werden kann.

Da wir diesen aber mit unter 3,3V betreiben wollen, müssen wir den LDO auf dem BH1750 Board entfernen und den Vin mit dem Vout überbrücken. (siehe Anschaltbild) dadurch kann der BH1750 auch mit 2,4V noch sicher arbeiten. Allerdings darf nun nicht mehr wie 3,3V an den BH1750 ankommen, Solange wir aber nur zwei Batterien verwenden und der FTDI Adapter auf 3,3V eingestellt ist, kann das nicht passieren! Zusätzlich sinkt der Stromverbrauch von ~68µA mit LDO auf ~4µA ohne LDO).

Code flashen:

WICHTIG! Da der LDO auf dem BH1750 entfernt wurde verträgt dieser nur noch maximal 3,6V. Der FTDI Adapter muss also auf 3,3V eingestellt werden.

Passiert das nicht, kann es sein, das der BH1750 Schaden nimmt und nicht mehr funktioniert!

AskSinPP Library in der Arduino IDE installieren

Achtung: Die Lib benötigt selbst auch noch weitere Bibliotheken, siehe README.

Projekt-Datei herunterladen. --> zu finden in den Beispielen von jp112sdl

([https://github.com/jp112sdl/Beispiel_As ... M-Sen-LI-O](https://github.com/jp112sdl/Beispiel_As...M-Sen-LI-O))

Arduino IDE öffnen

Heruntergeladene Projekt-Datei öffnen

Werkzeuge

- Board: Arduino Pro or Pro Mini einstellen

- Prozessor: ATmega328P (3.3V 8MHz) auswählen

- Port: entsprechend FTDI Adapter einstellen

- Im Sketch ist standardmäßig der Tsl2561-Sensor aktiviert. Um den BH1750 zu reaktivieren, müssen bei alle Zeilen wo BH1750 drin steht das // am Zeilenanfang entfernt werden und die Zeilen mit Tsl2561 alle mit // am Zeilenanfang ausgeklammert werden (oder gelöscht werden)

- Menü "Sketch" -> "Hochladen" auswählen.

Gerät anlernen:

Kann wie jedes HM-Gerät angelernt werden, CCU in den Anlernmodus bringen, Taster kurz drücken.. Fertig.

Sollten mir hier Fehler unterlaufen sein, so sagt mir bitte schnell bescheid, damit wir diesen beheben können.

Viel Erfolg

Gruß Carsten

Zuletzt geändert von [Bratmaxe](#) am 21.06.2018, 13:12, insgesamt 9-mal geändert.

Gruß Carsten

[Nach oben](#)