

## HEG Open Hardware Designs

### MIT License

Copyright 2019, Joshua Brewster

Permission is hereby granted, free of charge, to any person obtaining a copy of this hardware, software, and associated documentation files (the "Product"), to deal in the Product without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Product, and to permit persons to whom the Product is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Product.

THE PRODUCT IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE PRODUCT OR THE USE OR OTHER DEALINGS IN THE PRODUCT.

### HEGduino kit

This design is based off the original, now-expired HEG patent 5995857 by Hershel Toomim and Bob Marsh. The original sensor design is simply an OPT101 spaced at 3cm from 2 50mA LEDs - one in the Red spectrum and one in the Infrared spectrum. The Red LED is set at 650nm and the Infrared LED is set at either 850nm or 950nm. 850nm provides more linear scaling for blood flow ratio, while 950nm is the industry standard for pulse oximetry.

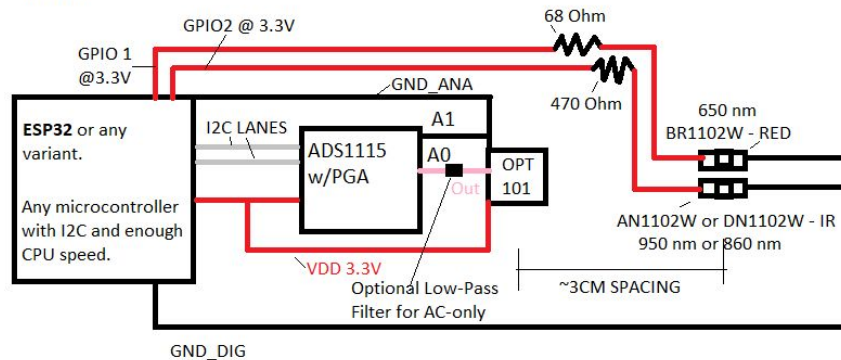
Our microcontroller of choice is the Espressif ESP32 using Arduino libraries, available for ~5 dollars on average, with our firmware available for free under the MIT license at [https://github.com/moorthyknight/HEG\\_ESP32](https://github.com/moorthyknight/HEG_ESP32). We also originally accomplished it with an Arduino Nano v3 which we purchased for 1 dollar apiece plus shipping. We are utilizing the ESP32 for its IoT capabilities which allows full interaction via a local webserver on the device itself. It also can connect online for remote data collection abilities or use bluetooth as well as traditional serial methods. None of our work is original on this in the sense that we used wholly open source libraries to develop the software solution, aside from our own simple method for driving the LEDs and timing the readings.

Continue below for multiple designs.

**Varied designs (thereby classifying them under the MIT License for open source use):**

Joshua Brewster

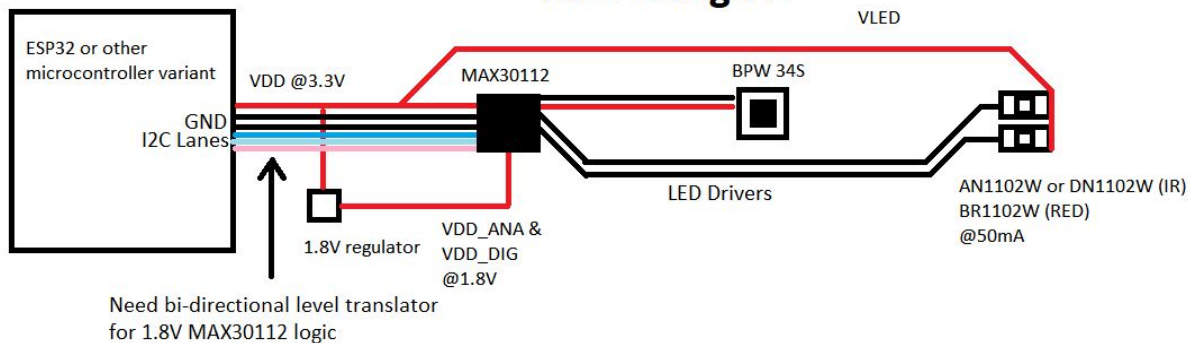
7/11/18



# HEG DESIGN 1

Design 1 as described above, in simplified form. The OPT101 has a built in transimpedance amplifier and the ADS1115 has up to 16X programmable gain.

## HEG Design 2



Experimental Design 2 with a MAX30112, enabling up to 3200sps via I2C. BPW34 is a highly sensitive photodiode. We followed the datasheet recommendations for decoupling capacitors and grounding. Drivers coming soon.

SPI needs bi-directional level translation for 1.8V MAX86141 Logic



The diagram illustrates the internal architecture and external connections of the MAX98140/MAX98141. Key internal blocks include:

- REFERENCE**: Connected to a 1.8V supply and a 10μF capacitor.
- DIE TEMP 12-BIT ADC**: Connected to the REF and VDD/DIG lines.
- AMBIENT CANCELLATION**: Two blocks, one for the main ADC and one for the MAX98141-only version.
- 19-BIT CURRENT ADC**: Two blocks, one for the main ADC and one for the MAX98141-only version.
- DIGITAL NOISE CANCELLATION**: Two blocks, one for the main ADC and one for the MAX98141-only version.
- 128-WORD FIFO**: Receives data from the ADCs and outputs to the SPI interface.
- SPI INTERFACE**: Connected to a host (AP) via SCL, SCK, SDI, SDO, CSB, and CSB lines.
- CONTROLLER**: Manages the device's operation, connected to the FIFO and LED drivers.
- LED DRIVERS**: Connected to the controller and the LED pins (LED1\_DRV, LED2\_DRV, LED3\_DRV).

External connections include:

- VDD ANA**: 1.8V supply for the analog section.
- VDD DIG**: 1.8V supply for the digital section.
- VDDIO**: I/O supply voltage, connected to the INT pin via a resistor R.
- GPIO1** and **GPIO2**: General-purpose I/O pins.
- RTC CLK**: Real-time clock input, connected to the RTC CLK pin via a resistor R.
- VBAT**: Backup battery supply, connected to the VBAT pin via a 10μF capacitor.
- POND**: Power-on delay pin, connected to ground.
- GND ANA** and **GND DIG**: Analog and digital ground connections.

NOTE 1: WHEN GPIO\_CTRL[3:0] = 0001 ONLY.

Above is the reference sheet provided by MAXIM, we followed this exactly and included ground planes wherever possible for improved isolation and decoupling. The MAX86141 should be as close to the photodiode site as possible.

As you can see, it's fairly easy to design these things, the key is finding the best and most affordable parts - which improve year to year. These are a solid foundation to begin from. And yes, I used Paint for these diagrams... but I recommend EAGLE or KiCAD for some of the better PCB design tools.

More design notes on next page...

[illegible]

There are several key design ideas to keep in mind when developing the sensor.

2nd: The analog and digital components need separate ground lines/planes from the sensor to prevent feedback from the LED voltages. The analog output should also be shielded with copper planes to minimize noise.

4th: You don't need more than 100 samples per second and 16 bits for good signal fidelity. We demonstrated this with a minimum-bar ADS1115, which includes PGA and differential modes to get pretty decent signal. We also recommend the MCP3424, an 18-bit, 240sps sigma-delta ADC which is more than enough for pulse oximetry, and simultaneous sensor data. These are both available as arduino kit-styled components.