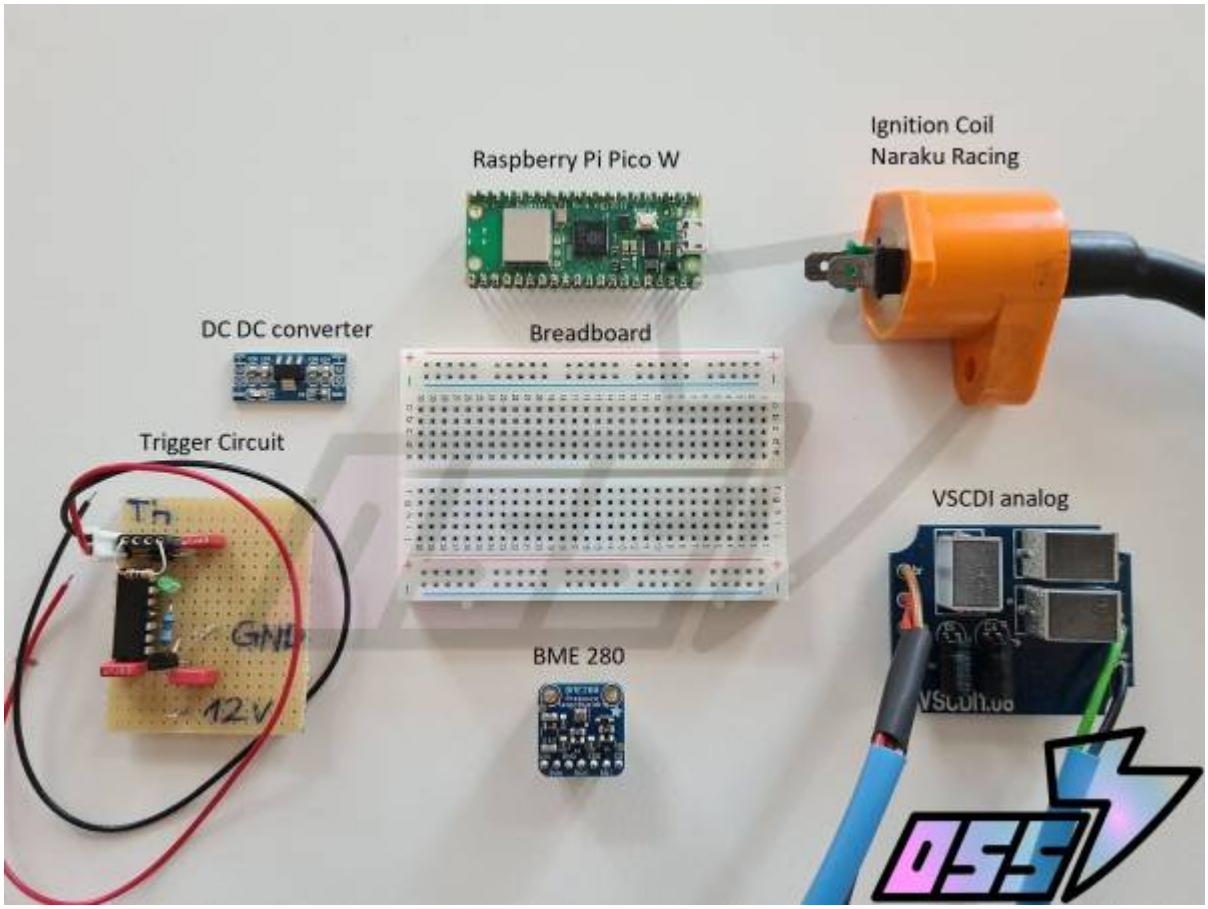
VSCDIP

Vape **S**port **CDI p**rogrammierbar

Konzept

Für die Entwicklung der VSCDIP werden folgende Komponenten eingesetzt

- Raspberry Pi Pico (W)
- DC-DC-Converter zur Spannungsversorgung
- Analoge CDI-Schaltung basierend auf der VSCDI-Erfahrungen
- Externe Zündspule
- Universal-Breadboard für den Prototypenaufbau
- Trigger-Schaltung für das Zündsignal
- Highlight: Temperatur-Luftfeuchte-Druck-Sensor



Raspberry Pi Pico

Der Pico ist ein günstiger und sehr potenter Microcontroller. Als Produkt der Raspberry Pi Foundation ist er langzeitverfügbar. Selbst in vielen Jahren kann der Microcontroller weiterhin bestellt werden.

Die Programmierung erfolgt mit Micropython. Python hat den Vorteil, dass es eine einfache

Hochsprache ist. Der wesentliche Nachteil ist, dass die Sprache interpretiert wird. Dadurch ist Python per se langsamer als beispielsweise ein Arduino gleicher Leistungsklasse und der Programmiersprache C.

Jedoch hat der Pico mit 133 Mhz genug Leistung für die VSCDIP. Falls es nicht reichen sollte, können zeitkritische Funktionen in C-Module ausgelagert werden. Auch das kann der Pico.

Optional gibt es den Pico W mit W-LAN und Bluetooth. Das könnte als Programmierschnittstelle verwendet werden.

https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html

DC DC Converter

Zur Spannungsversorung wird ein AMS117 verwendet. Der Eingangsspannungsbereich reicht bis 15 V. Die Ausgangsspannung von 3,3 V ist für den Pico und die Triggerschaltung.

https://eckstein-shop.de/MiniAMS1117-32C3VDC-DCSpannungsreglerVoltageRegulatorConvertor

Triggerschaltung

Die Triggerschaltung formt das Signal des Pickups in ein Rechtecksignal um, sodass der Microcontroller den Zündimpuls erkennt. Das ist bei der Vape-Zündung einer der schwierigsten Teile, da das Pickup-Signal sehr „hässlich“ ist.

VSCDI analog

Die Leistungsschaltung zur Zündfunkenerzeugung wird wie bei den bisherigen analogen VSCDI-Analogmodulen erfolgen. Als Bonbon wird die positiven und negativen Welle verarbeitet und damit die doppelte Zündenergie im Vergleich zur Standard-Vape-Zündung erzeugt.

Zündspule

In der externen Zündspule wird die Spannung für den Zündfunken von mehreren 10 000 V erzeugt. Prinzipiell sollte nahezu jede Zündspule ohne CDI funktionieren. Gute Erfahrung wurde mit der Naraku-Racing-Zündspule gemacht:

https://www.racing-planet.de/zuendspule-naraku-racing-v2-high-output-2-pins-p-72194-1.html

BME280

Eine CDI, die Temperatur, Luftdruck und Luftfeuchtigkeit berücksichtigt wäre revolutionär und ein Traum. Vorerst wird das auch ein Traum bleiben, denn der BME280 wird vorerst nicht integriert. Aber die zukünftige Integration wird berücksichtigt. Das ist insofern wichtig, damit der Programmcode für den Microcontroller nicht komplett unmgestrickt werden muss, sondern nur geringfügig erweitert.

Features

Anzahl der Programmierkurven

Prinzipiell lässt sich eine Vielzahl an Programmierkurven speichern. Es stellt sich nur die Frage des Handlings. Denkbar sind ein Schalter am Lenker und/oder ein Dip-Schalter an der CDI.

https://www.distrelec.de/de/dip-drehschalter-rnd-rnd-210-00089/p/30090545?trackQuery=cat-DNAV_PL_050101&pos=20&origPos=20&origPageSize=50&track=true&sid=6583f6fdf4aaf934804f6dab7779ca4bb7a95388&itemList=category

From:
https://opensimspark.org/ - **OpenSimSpark**

Permanent link:
https://opensimspark.org/vscdip

Last update: **2024/03/03 14:01**