

swissbit®

Application Note

**AN2109de**

**Power Failure Testing**

© Swissbit AG 2022

  Creative-Commons-Lizenz<sup>1</sup>

---

<sup>1</sup>Dieses Werk steht unter der Creative-Commons-Lizenz „Namensnennung 4.0 International“. Um eine Kopie dieser Lizenz zu sehen, besuchen Sie <http://creativecommons.org/licenses/by/4.0/>

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Gefahren bei plötzlichem Stromausfall</b>	<b>2</b>
<b>3</b>	<b>Stromausfälle testen</b>	<b>3</b>
3.1	USB . . . . .	4
3.2	CompactFlash . . . . .	4
3.3	SD-Karten . . . . .	5
3.4	eMMC . . . . .	5
3.5	SATA und CFast . . . . .	5
3.6	NVMe und CFexpress . . . . .	5
<b>4</b>	<b>Test-Hardware</b>	<b>6</b>
<b>5</b>	<b>Zusammenfassung</b>	<b>7</b>

## 1 Einleitung

NAND-Flash ist eine Technologie, die gegenüber Festplatten eine deutlich höhere Datenrate ermöglicht. Insbesondere profitieren hier zufällige Lese- und Schreibzugriffe. Im Falle eines plötzlichen Stromausfalls sind Festplatten aber in der Lage, den begonnenen Schreibvorgang sauber abzuschließen, indem sie die Rotationsenergie des Plattenstapels zurückgewinnen und damit für einen kurzen Moment die Elektronik versorgen können. Bei SSDs kann man sich mit großen Kondensatoren behelfen, um noch für ein paar hundert Millisekunden die Versorgungsspannung zu stützen, bis die internen Puffer und der DRAM-Cache weggeschrieben wurden. Nachteilig sind hier die hohen Kosten für die Kondensatoren, für die aufgrund des industriellen Temperaturbereiches und des verfügbaren Raumes nur Tantal-Elektrolytkondensatoren in Frage kommen. Daher werden solche SSDs nur für Spezialanwendungen eingesetzt. Speichermedien, die sämtliche Daten im Cache bei einem Stromausfall noch auf den Flash schreiben können, tragen den Zusatz *PLP (Power Loss Protection)*.

Da PLP-Medien gegenüber einem Stromausfall geschützt sind, werden im Folgenden die Probleme und Testmöglichkeiten von Medien ohne PLP betrachtet. Die Methoden sind aber

auch auf PLP-Medien übertragbar, um die versprochenen Eigenschaften zu überprüfen.

## 2 Gefahren bei plötzlichem Stromausfall

Ein plötzlicher Stromausfall kann bei Flash-Medien zu verschiedenen Fehlern und Problemen führen, die im schlimmsten Fall einen Totalausfall zur Folge haben. Die Ursache liegt in der Technologie begründet: Der gesamte Flash ist in sogenannte *Blöcke* aufgeteilt. Diese Blöcke bestehen wiederum aus *Pages*. Während sich nur ganze Pages in einem Vorgang programmieren lassen, kann das Löschen nur für ganze Blöcke erfolgen. Dadurch ergibt sich, dass SSDs im Gegensatz zu HDDs keine feste Zuordnung zwischen den logischen Adressen (die der Host sieht) und den physikalischen Flash-Adressen haben. Zudem ist die Anzahl der Programmier- und Löschkzyklen beschränkt, weshalb das *Wear-Leveling* für eine gleichmäßige Abnutzung der Blöcke sorgt, was natürlich nur möglich ist, wenn die Adresszuordnung nicht statisch ist.

Da nur ganze Pages programmiert werden können, die je nach Flash eine Größe von bis zu 64 KiB haben, und diese Größe noch mit der Anzahl der parallel geschalteten Flashes multipliziert werden muss, wird mit jedem Schreibzugriff bis zu 1 MiB an Flashzellen programmiert – selbst wenn das Betriebssystem nur einen Verzeichniseintrag aktualisiert.

Um den unnötigen Verschleiß zu reduzieren, puffert die SSD die Daten im Controller oder in einem externen DRAM für einige Millisekunden. Kommen in dieser Zeit weitere Daten, so können diese mit demselben Schreibzugriff in den Flash geschrieben werden. Der Flash wird im Allgemeinen in Segmenten von je 4 KiB verwaltet. In eine Page von 64 KiB passen somit 16 einzelne Schreibzugriffe, wenn jeder Schreibzugriff nicht mehr als 4 KiB umfasst. Die Effizienz steigt somit, wenn der Controller die Schreibzugriffe verzögert und noch weitere Daten eintreffen, bevor der Programmiervorgang im Flash gestartet wird. Der Nachteil ist natür-

lich, dass diese Daten verloren gehen, falls es zu einem plötzlichen Stromausfall kommt und es sich nicht um ein PLP-Medium handelt.

Für die meisten Anwendungen kann dieses Verhalten jedoch toleriert werden, da sich die betroffenen Daten einige Millisekunden zuvor noch im RAM vom Host befunden haben, und dort ebenso verloren gegangen wären, wenn der Stromausfall eben diese paar Millisekunden früher aufgetreten wäre.

Bei einem unerwarteten Stromausfall kann es zu folgenden Problemen kommen, hier entsprechend ihrer Schwere aufsteigend aufgezählt:

1. Benutzerdaten, die der Host an das Medium übertragen hat und deren Empfang vom Medium bereits bestätigt wurde, obwohl sie noch nicht auf den Flash geschrieben wurden, sind verloren. Dies ist, wie bereits geschildert, für viele Anwendungsfälle unproblematisch. Empfindliche Systeme, bei denen keinerlei Daten verloren gehen dürfen, verfügen über eine unterbrechungsfreie Stromversorgung, die dann auch die Speichermedien mit abdeckt.
2. Es sind nicht nur die gerade übertragenen Daten verloren sondern auch noch Daten, die mit den direkt vorangegangenen Schreibzugriffen geschrieben wurden, selbst wenn diese Schreibvorgänge bereits lange zurückliegen, aber seitdem kein Power-Cycle stattgefunden hat. Technologiebedingt kann dieser Effekt bei älteren MLC- und TLC-Speichern in Floating-Gate-Technologie auftreten.
3. Beim Stromausfall unvollständig geschriebene Daten können von der Fehlerkorrektur typischerweise nicht wiederhergestellt werden, da der Programmiervorgang nicht zu Ende geführt wurde. Das Versagen der Fehlerkorrektur führt in der Firmware fälschlicherweise zur Einschätzung, dass es sich um einen schlechten Flashblock handelt, was zum Ersetzen des Blocks mit einem Block aus dem Reservepool führt. Geschieht dies zu oft, werden alle Reservelöcke aufgebraucht, bis das Medium nicht mehr beschreibbar ist.
4. Es kommt zum sogenannten *Reordering*. Dabei gibt es anschließend keinen sauberen Übergang zwischen den neuen und den alten Daten beim Lesen der logischen Adressen in der identischen Reihenfolge wie zuvor beim Schreiben. Dies kann bei Datenbanken und Dateisystemen mit Journal zu massivem Datenverlust führen.
5. Es gehen statische Daten verloren, also Daten, die viele Schreibzugriffe zurückliegen oder aus einem vorangegangenen Power-Cycle stammen. Ein typisches Beispiel ist die Beschädigung von Betriebssystem-Dateien, wodurch das System plötzlich nicht mehr startet.
6. Die Firmware des Mediums oder ihre Meta-Daten wurden beschädigt. Dies kann sich durch verschiedene Symptome äußern: Massiver oder totaler Datenverlust, Meldung fehlerhafter Lifetime-Daten (S.M.A.R.T.) oder ein dauerhafter Geschwindigkeitsverlust, der möglicherweise nicht bemerkt wird. Im schlimmsten Fall meldet sich das Medium beim Host nicht mehr an.

Swissbit ist seit vielen Jahren marktführend bei der Härtung von Flash-Medien gegen plötzliche Stromausfälle. Entsprechend treten bei unseren Produkten nur die ersten beiden Fälle auf bzw. bei PLP-Medien keines der genannten Probleme.

### 3 Stromausfälle testen

Soll im Rahmen der Produktqualifizierung die Robustheit gegen plötzliche Stromausfälle getestet werden, so kann der provozierte Ausfall für das ganze System oder nur für das Speichermedium erfolgen. Es bietet sich an, mit dem Speichermedium allein zu beginnen, da hier viel schnellere Zyklen möglich sind, um eine Vorauswahl zu treffen. Hat das Speichermedium erfolgreich die Vorgabe von z. B. 10.000 Zy-

klein erreicht, werden anschließend noch mindestens 1.000 Zyklen im Gesamtsystem absolviert. Diese dienen dazu, mögliche Kompatibilitätsprobleme wie Fehler im BIOS zu finden. Gerade bei extremen Testbedingungen kann es beim nächsten Einschalten zu einer verlängerten Diagnosephase des Mediums kommen, in der die Datenkonsistenz wiederhergestellt wird. Manche BIOS-Versionen haben jedoch mit der Ablösung von HDDs durch SSDs in den letzten Jahren die maximale Wartezeit entgegen der Empfehlung im SATA/ATAPI-Standard stark reduziert, was nun bei neueren SSDs mit TLC oder QLC und deren Komplexität in der internen Verwaltung zu einer Zeitüberschreitung führen kann, da in einem ungünstigen Fall die Startzeit wieder im Bereich von Festplatten liegen kann.

Wie oben beschrieben, sollte das Flash-Medium durch ein plötzliches Abschalten keinen Schaden nehmen. Eine gute Firmware muss in der Lage sein, permanente Fehler oder erhöhten Verschleiß zu verhindern. Allerdings gilt dies nur, solange sich die Anzahl und die Geschwindigkeit, mit der die Zyklen erfolgen, noch in einem realistischen Rahmen bewegen. Bei Qualifikationstests werden jedoch oft viele Jahre Feldeinsatz in wenigen Tagen simuliert. Kommt es dabei zu einer sehr schnellen Abfolge von extrem vielen Zyklen, in denen zwar ein Datenvolumen größer Null aber kleiner ein paar weniger Megabyte geschrieben wird, kann es – je nach Größe des Speichermediums – nach wenigen zehntausend Zyklen zu einem Ausfall des Speichermediums kommen. Das Problem tritt hauptsächlich auf, wenn die Zykluszeit nur einige Sekunden beträgt. Der Hintergrund ist, dass bei jedem Zyklus ein neuer Block beschrieben wird, aber keiner dieser Blöcke jemals vollständig gefüllt wird. Wird ein Block extrem oft und schnell nur teilweise programmiert, kann dies zu unzureichenden Löschvorgängen führen, infolge derer die erneute Programmierung eine immer weiter steigende Bitfehlerrate aufweist, bis es zu Datenverlust kommt. Dieses Problem lässt sich vermeiden, indem alle 3.000 Zyklen das Medium einmal komplett vollgeschrieben wird. Alternativ sollte die während eines Zyklus ge-

schriebene Datenmenge mindestens 0,05 % der Gesamtkapazität betragen.

### 3.1 USB

Bei USB-Geräten ist das Testen sehr einfach. Es muss lediglich die +5 Volt Versorgungsleitung aufgetrennt und mit einem Schalter versehen werden. Zwar existieren auch USB-Hubs, die schaltbare Ausgänge besitzen, allerdings sind diese schwer zu erhalten. Häufig ist es nur möglich, das Endgerät am Bus abzumelden; die Stromversorgung bleibt jedoch bestehen.

Wird die +5 Volt Leitung unterbrochen, wird das Endgerät vom Host abgemeldet. Sobald die Versorgung wiederhergestellt ist, wird das Gerät erkannt und vom Betriebssystem initialisiert.

### 3.2 CompactFlash

Bei allen Schnittstellen, die nicht mit differenziellen Signalen arbeiten, besteht die Gefahr, das Speichermedium über die Datenleitungen weiter mit Strom zu versorgen, selbst wenn die eigentliche Versorgung unterbrochen wurde. Der Strom fließt über die Datenleitungen zum Speichermedium und über die ESD-Schutzdioden zur Verteilung der Spannungsversorgung. Abbildung 1 zeigt rot gestrichelt den Stromfluss nach Auftrennung der positiven Versorgungsspannung.

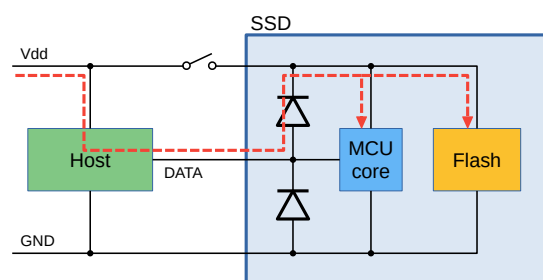


Abbildung 1: Stromfluss über ESD-Dioden

Die resultierende Versorgungsspannung im Speichermedium ist dann in der Regel zu gering für den normalen Betrieb. Der Controller erkennt die niedrige Versorgungsspannung

und pausiert alle weiteren Operationen. Kehrt die Versorgungsspannung auf Nominalniveau zurück, fährt der Controller mit seinen Operationen fort. Da die internen Zustände des Controllers dabei nicht verloren gehen, ist dies kein echter Stromausfall.

Es wäre sehr aufwendig, eine Testumgebung aufzubauen, die bei CompactFlash sämtliche Daten- und Steuerleitungen trennt. Daher bietet es sich an, die Schnittstelle zum Host zu ändern. Es existieren zwei Möglichkeiten: Entweder wird ein CF-Card-Reader verwendet, der per USB angeschlossen wird. Dann entspricht das Vorgehen dem mit einem USB-Speichermedium. Oder es kommt eine IDE-SATA-Bridge zum Einsatz, und das CF-Medium wird wie ein SATA-Medium verwendet. Hier ist darauf zu achten, dass die Bridge zusammen mit der CF-Karte von der Spannungsversorgung getrennt wird, da IDE im Gegensatz zu SATA nicht hot-plug-fähig ist.

### 3.3 SD-Karten

Bei SD-Karten tritt das gleiche Problem wie bei CF-Karten auf. Die Daten- und Steuerleitungen sind mit Pull-Up-Widerständen versehen, über die ausreichend Strom fließen kann, um den Controller zu versorgen, wenn die positive Spannungsversorgung getrennt wird. Daher ist es wichtig, bereits vor den Pull-Up-Widerständen zu trennen. Alternativ können auch hier SD-Card-Reader verwendet und wieder wie bei USB-Geräten verfahren werden.

### 3.4 eMMC

Das Interface von eMMC und SD ist weitestgehend identisch. Elektrisch sind sie vollständig kompatibel. Entsprechend besteht auch hier die Gefahr der Stromversorgung über die Pull-Up-Widerstände. Soll die Robustheit einer eMMC nicht erst im fertigen System überprüft werden, bieten sich hier Adapter von eMMC auf SD an. Bei diesen können die eMMC entweder aufgelötet oder in einen Klemmsockel gelegt werden. Der Adapter wird dann mittels SD-Card-Reader per USB verbunden, und im

USB-Kabel wird die positive Spannungsversorgung aufgetrennt.

### 3.5 SATA und CFast

CFast nutzt ebenfalls das SATA-Interface. Ebenso wie USB ist SATA hot-plug-fähig und verwendet differentielle Signalübertragung. Die Signale sind gleichspannungsfrei, da sowohl beim Host als auch im Speichermedium Serienkondensatoren in den Leitungen sitzen. Somit genügt es hier, nur die positive Versorgungsspannung zu schalten. Den verschiedenen SATA-Formfaktoren bietet der Host meist zwei Spannungen. Welche davon geschaltet werden muss, lässt sich dem Datenblatt entnehmen. Üblicherweise ist es die kleinere Spannung. Es gibt auch Medien, die mit zwei Spannungen versorgt werden können. In einem solchen Fall wird dann eine Leitung dauerhaft durchtrennt.

Es bietet sich an, für alle kleinen Formfaktoren einen Adapter auf 2,5" zu verwenden. Bei dem 12-poligen Stecker ist dann die rote 5 Volt Leitung zu schalten und zur Sicherheit die gelbe 12 Volt Leitung zu trennen.

Sollte das Medium nicht erkannt werden, muss im BIOS möglicherweise die Hot-Plug-Option für SATA aktiviert werden.

### 3.6 NVMe und CFexpress

Bei Speichermedien mit PCIe-Schnittstelle ist das Testen etwas aufwändiger. Dies liegt an den hohen Übertragungsraten, die das räumliche Absetzen des Speichermediums zum Auftrennen der Spannungsversorgung schwierig machen. Zudem haben teilweise die Hosts oder die Betriebssysteme noch Fehler in der Implementierung der Hot-Swap-Funktion.

Zum Absetzen des Speichermediums existieren fertige Lösungen, die mit einer hochwertigen Kabelverbindung den entfernten Anschluss von m.2 oder PCIe-Karten erlauben. Der Vorteil ist, dass hiermit keine Geschwindigkeitseinbußen einhergehen. Allerdings ist diese Lösung für das Testen von Stromausfällen nicht zu empfehlen, da der Aufwand zum Einrichten einer zuverlässigen Testumgebung sehr

hoch ist. Grundsätzlich ist NVMe hot-swap-fähig. Das bedeutet, dass ein Speichermedium, welches beim Systemstart angeschlossen war und für das somit die nötigen Systemressourcen reserviert wurden, zu einem späteren Zeitpunkt entfernt und auch wieder angeschlossen werden kann. Was ist der Theorie funktionieren soll, zeigt aber in der Praxis noch eine Reihe von Problemen. So hängt es vom verwendeten Betriebssystem, dem Host-Chipsatz, dem BIOS und sogar vom verwendeten PCIe-Slot auf dem Mainboard ab, ob überhaupt und wenn ja wie oft das Speichermedium nach einem Hot-Swap wieder erkannt wird. Unsere Erfahrung ist, dass die Kombination aus Intel-Chipsätzen, Linux-Betriebssystem und Server-Mainboards am besten funktioniert. Aber auch damit kann es nach ein paar tausend Zyklen dazu kommen, dass das Speichermedium nicht mehr erkannt wird und das ganze System ein Neustart benötigt.

Es bietet sich daher an, für NVMe ebenfalls USB-Bridges zu verwenden. Allerdings arbeiten auch diese nicht perfekt. So trat auch im Swissbit-Labor schon mehrmals der Fall auf, dass alle paar tausend Zyklen das Speichermedium nicht erkannt wurde, während es beim nächsten Zyklus wieder problemlos erkannt wurde ohne dass ein Neustart des Host erforderlich war. Dieses Verhalten müsste dann lediglich in der Testsoftware berücksichtigt werden.

Einzig die Stromversorgung per USB vom Host kann bei dieser Lösung ein Problem darstellen, da das Speichermedium mehr (Spitzen-)Strom ziehen kann, als der Host bereitstellt. Um dieses Problem zu umgehen, kann die +5 Volt Leitung aus dem Netzteil direkt mit der USB-Bridge verbunden bzw. geschaltet werden.

## 4 Test-Hardware

Fertige Hardware zum Test von Stromausfällen ist kaum verfügbar. Es existieren per USB steuerbare Relais, die aber meist über komplizierte und veraltete APIs verfügen, was die Einbindung in aktuelle Betriebssysteme schwierig macht. Daher wird im Folgenden ein einfacher

Bauvorschlag als Lösung vorgestellt.

Kernelement ist ein Atmel-Mikrocontroller mit USB-Anschluss. In diesem Beispiel mit einem Atmel-ATmega32U4, verbaut auf einem Arduino-kompatiblen Board mit der Bezeichnung „ProMicro“. Dieses Board kann mit der Arduino IDE programmiert werden, wenn als verwendetes Board „Leonardo“ ausgewählt wird. Prinzipiell kann jedes Arduino-Board verwendet werden, das über eine echte USB-Schnittstelle verfügt, so z. B. der Arduino Uno, Arduino Nano oder Arduino Micro. Gegebenenfalls muss dann die Initialisierung der USB-Schnittstelle modifiziert werden. Varianten, die die USB-Schnittstelle lediglich in Software emulieren, machen häufig Probleme wie z. B. Boards mit einem ATtiny.

Abbildung 2 zeigt den Schaltplan. Das Board wird per USB mit dem Host verbunden, von diesem mit Strom versorgt und gesteuert.

Der Mikrocontroller empfängt die Steuerbefehle vom Host und schaltet entsprechend Pin 17 (intern „Ao“) auf 5 Volt oder 0 Volt. Der Vorwiderstand R1 begrenzt den Basisstrom von Q1. Der Widerstand R3 sperrt den P-Kanal-MOSFET Q2 bis Q1 leitet. R2 dient lediglich dazu, Q1 zuverlässig zu sperren, solange die USB-Verbindung zum Mikrocontroller noch nicht hergestellt bzw. der Port Ao noch nicht initialisiert ist. Für Q1 kann ein Standard-NPN-Transistor verwendet werden. Für Q2 wurde IRLML6402 gewählt, da dieser Typ mit  $V_{GS(th),max} = -1,2V$  eine Versorgungsspannung von 3,3V sicher schaltet.

Die positive Spannungsversorgung der SSD wird aufgetrennt und entsprechend dem Schaltplan an J1 angeschlossen. Hängen die SSD und der Mikrocontroller an verschiedenen Hosts, so muss ein gemeinsames Massepotential hergestellt werden, damit Q2 schalten kann.

Unter Linux erfolgt das Schalten mit folgenden Kommandos:

```
echo '1' > /dev/ttyACMo
echo '0' > /dev/ttyACMo
```

Eine '1' wird die Stromversorgung einschalten, eine '0' wird sie ausschalten. Zuvor muss jedoch noch mit der Arduino-IDE das Programm in den Mikrocontroller geschrieben

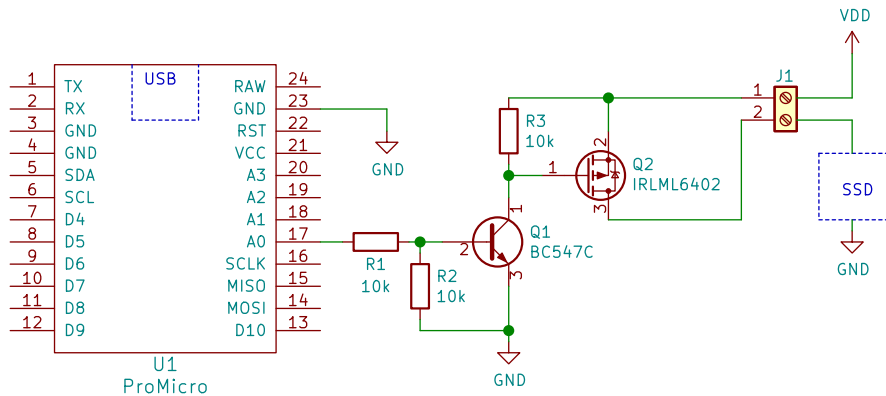


Abbildung 2: Schaltplan

werden. Das Programm ist sehr einfach gehalten:

```

void setup() {
  pinMode(Ao, OUTPUT);
  digitalWrite(Ao, false);
  SerialUSB.begin();
}

void loop() {
  if (SerialUSB.available()) {
    char in = SerialUSB.read();
    if (in == '0') {
      digitalWrite(Ao, false);
    } else if (in == '1') {
      digitalWrite(Ao, true);
    }
  }
}

```

Zu Beginn wird der Port Ao als Ausgang definiert und auf 0 Volt geschaltet. Danach wird die USB-Schnittstelle initialisiert. Anschließend wartet der Mikrocontroller auf den Empfang eines Zeichens über USB. Wird eine Null empfangen, wird der Ausgang auf 0 Volt geschaltet, wodurch Q1 und Q2 sperren und die SSD von der Versorgungsspannung getrennt wird. Bei einer Eins wird der Ausgang auf 5 Volt geschaltet, Q1 und Q2 leiten, die SSD startet. Der fertige Aufbau ist in Abbildung 3 gezeigt.

Auf einen konkreten Bauvorschlag für eine Erweiterung zum Schalten der Netzspannung des Hosts wird hier verzichtet. Dies ist mit einem geeigneten Relais einfach durchzuführen,

sollte aber nur von einer Elektrofachkraft erfolgen.

## 5 Zusammenfassung

Die vorgestellten Methoden und die Schaltung stellen eine einfache Lösung zur Implementierung von Ausfalltests der Spannungsversorgung dar. Mit ihnen lassen sich Ausfälle während der Schreib- und Lesezugriffe der Zielapplikation zeitlich gut steuern, um eine breite Abdeckung während verschiedener Betriebszustände zu erreichen. Die Robustheit und somit die Eignung eines Mediums kann für den jeweiligen Anwendungsfall überprüft werden, um nur Hersteller für das Endprodukt freizugeben, bei denen kein unkalkulierbares Ausfallrisiko im Feld besteht.

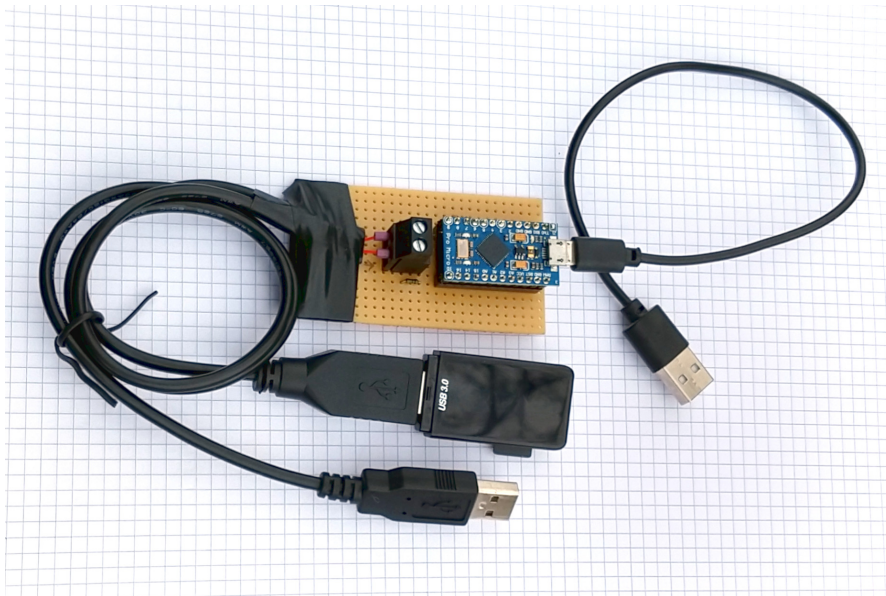


Abbildung 3: Aufbau mit (μ)SD-USB-Reader



## Kontaktieren Sie uns

<b>Hauptsitz</b>	<b>Swissbit AG</b> Industriestraße 4 9552 Bronschhofen Schweiz	Tel. +41 71 913 03 03 sales@swissbit.com
<b>Deutschland (Berlin)</b>	<b>Swissbit Germany AG</b> Bitterfelder Straße 22 12681 Berlin Deutschland	Tel. +49 30 936 954 0 sales@swissbit.com
<b>Deutschland (München)</b>	<b>Swissbit Germany AG</b> Leuchtenbergring 3 81677 München Deutschland	Tel. +49 30 936 954 400 sales@swissbit.com
<b>Nord- und Südamerika</b>	<b>Swissbit NA Inc.</b> 238 Littleton Road, Suite 202B Westford, MA 01886 USA	Tel. +1 978-490-3252 salesna@swissbit.com
<b>Japan</b>	<b>Swissbit Japan Co., Ltd.</b> CONCIERIA Tower West 2F 6-20-7 Nishishinjuku Shinjuku City, Tokyo 160-0023 Japan	Tel. +81 3 6258 0521 sales-japan@swissbit.com
<b>Taiwan</b>	<b>Swissbit Taiwan</b> 3F., No. 501, Sec.2, Tiding Blvd. Neihu District, Taipei City 114 Taiwan, R.O.C.	Tel. +886 912 059 197 salesasia@swissbit.com
<b>China</b>	<b>Swissbit China</b>	Tel. +886 958 922 333 salesasia@swissbit.com

**Disclaimer:**

The information in this document is subject to change without notice. Swissbit AG ("SWISSBIT") assumes no responsibility for any errors or omissions that may appear in this document, and disclaims responsibility for any consequences resulting from the use of the information set forth herein. SWISSBIT makes no commitments to update or to keep current information contained in this document. The products listed in this document are not suitable for use in applications such as, but not limited to, aircraft control systems, aerospace equipment, submarine cables, nuclear reactor control systems and life support systems. Moreover, SWISSBIT does not recommend or approve the use of any of its products in life support devices or systems or in any application where failure could result in injury or death. If a customer wishes to use SWISSBIT products in applications not intended by SWISSBIT, said customer must contact an authorized SWISSBIT representative to determine SWISSBIT willingness to support a given application. The information set forth in this document does not convey any license under the copyrights, patent rights, trademarks or other intellectual property rights claimed and owned by SWISSBIT.

ALL PRODUCTS SOLD BY SWISSBIT ARE COVERED BY THE PROVISIONS APPEARING IN SWISSBIT'S TERMS AND CONDITIONS OF SALE ONLY, INCLUDING THE LIMITATIONS OF LIABILITY, WARRANTY AND INFRINGEMENT PROVISIONS. SWISSBIT MAKES NO WARRANTIES OF ANY KIND, EXPRESS, STATUTORY, IMPLIED OR OTHERWISE, REGARDING INFORMATION SET FORTH HEREIN OR REGARDING THE FREEDOM OF THE DESCRIBED PRODUCTS FROM INTELLECTUAL PROPERTY INFRINGEMENT, AND EXPRESSLY DISCLAIMS ANY SUCH WARRANTIES INCLUDING WITHOUT LIMITATION ANY EXPRESS, STATUTORY OR IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

© 2022 SWISSBIT AG