

Netzausfall-Detektor mit SMS-Alarm

Erkennt und meldet auch kurze Stromausfälle

Idee: **Horst van Bremen** (D)

Entwicklung und Text: **Mathias Claußen**

und **Luc Lemmens** (Elektor-Labor)

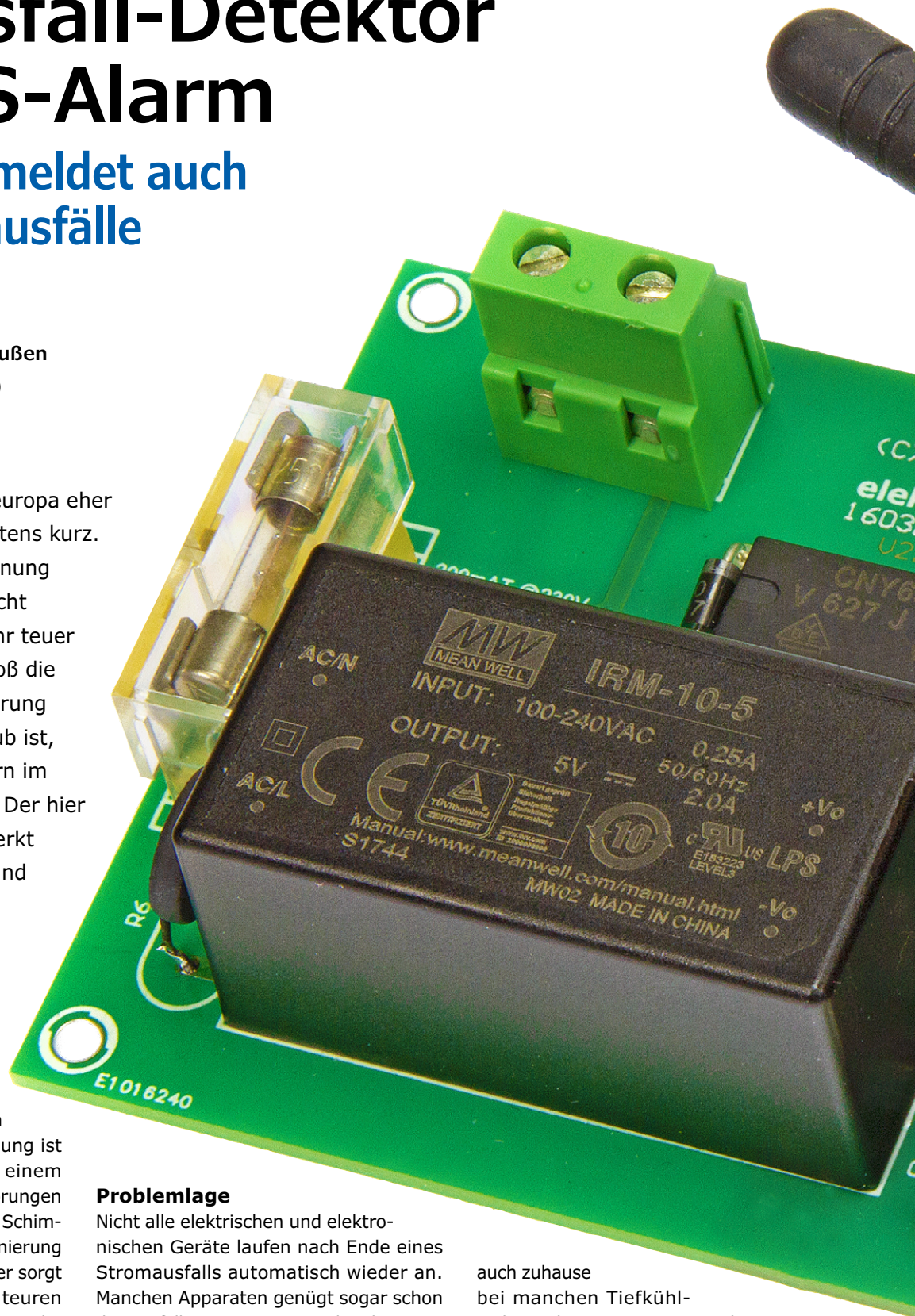
Stromausfälle sind in Mitteleuropa eher selten und wenn, dann meistens kurz. Doch falls bei der Ferienwohnung anschließend die Heizung nicht wieder anläuft, kann das sehr teuer werden. Und selbst wenn bloß die heimische Kühltruhe auf Störung geht, während man im Urlaub ist, „stinkt“ das den Heimkehrern im wahrsten Sinne des Wortes. Der hier beschriebene Detektor bemerkt selbst kurze Stromausfälle und alarmiert darüber per SMS.

Elektor-Leser Horst van Bremen ist genau das passiert: Die Heizung ist in seiner Ferienwohnung nach einem Stromausfall nicht wieder angesprungen und die folgenden Schäden durch Schimmel machten eine aufwendige Sanierung erforderlich. Der kluge Elektroniker sorgt vor, dachte er sich nach dieser teuren Erfahrung. Also machte er sich an die Entwicklung eines Geräts, das selbst sehr kurze Stromausfälle erkennt und den Status an eine Web-Plattform weiterleitet.

Problemlage

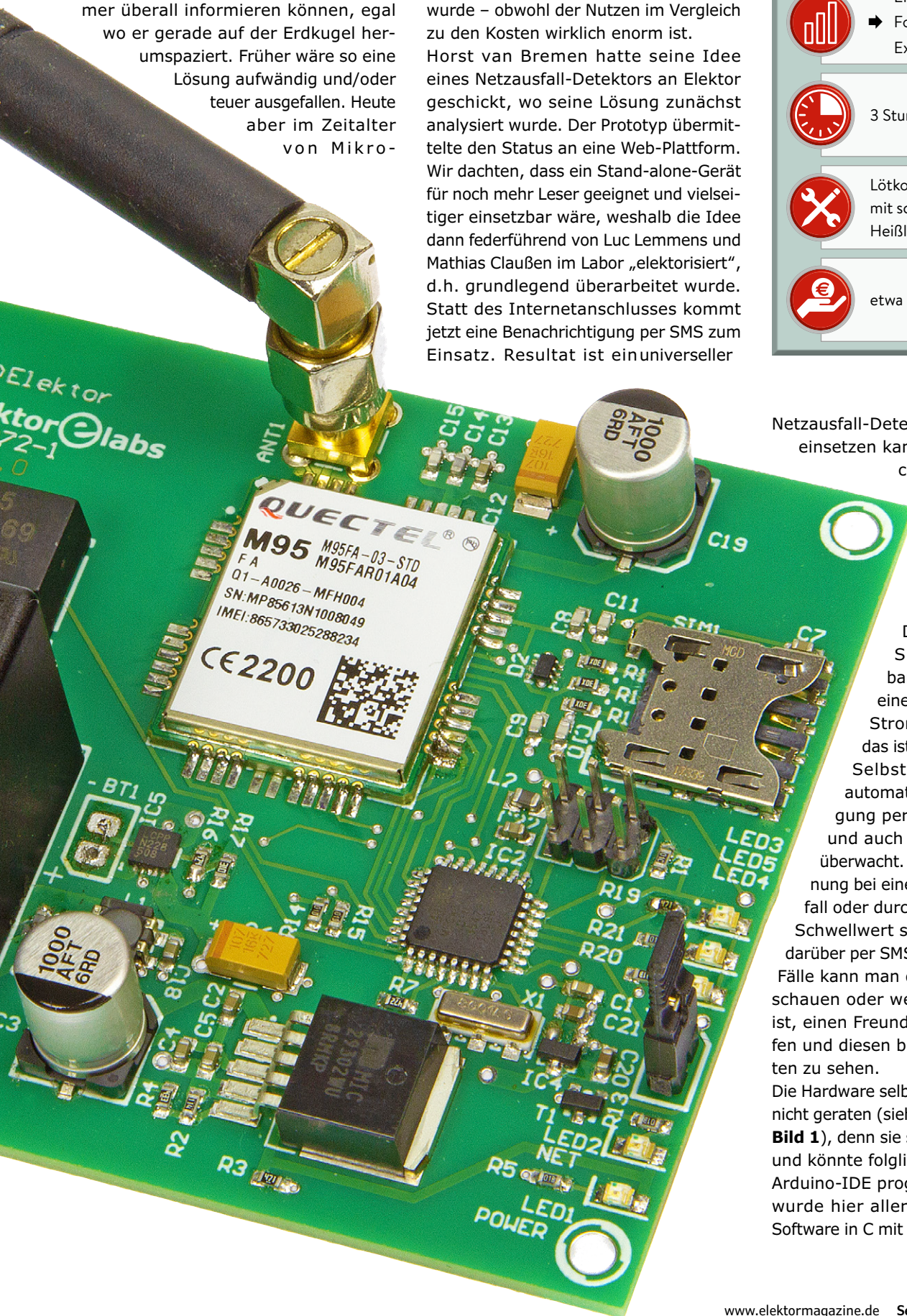
Nicht alle elektrischen und elektronischen Geräte laufen nach Ende eines Stromausfalls automatisch wieder an. Manchen Apparaten genügt sogar schon der Ausfall von einigen Perioden der Netzspannung und sie gehen auf Störung. Das ist nicht nur bei Heizungen in Ferienwohnungen problematisch, sondern

auch zuhause bei manchen Tiefkühltruhen oder Bewässerungsanlagen, während man den Sommerurlaub auf der anderen Seite des Globus verbringt. Und nicht immer kann ein Nachbar jeden Tag



vorbeischaun, oder er bemerkt vielleicht gar nicht, dass irgendetwas ausgefallen ist oder nicht mehr läuft. Dann ist der Schaden unter Umständen so groß, dass sich das elektronische Vorbeugen lohnt. Eine gute „Netzausfallschadensverhinderungslösung“ sollte ihren Eigentümer überall informieren können, egal wo er gerade auf der Erdkugel herumspaziert. Früher wäre so eine Lösung aufwändig und/oder teuer ausgefallen. Heute aber im Zeitalter von Mikro-

controllern, gut ausgebauten Handynetzen und preiswerten wie einfach zu handhabenden GSM-Modulen ist so ein Vorhaben gar nicht so schwierig umzusetzen. Also ist es kein Wunder, dass die passende Lösung erst jetzt kommt, und nicht schon 1975 in Elektor veröffentlicht wurde – obwohl der Nutzen im Vergleich zu den Kosten wirklich enorm ist. Horst van Bremen hatte seine Idee eines Netzausfall-Detektors an Elektor geschickt, wo seine Lösung zunächst analysiert wurde. Der Prototyp übermittelte den Status an eine Web-Plattform. Wir dachten, dass ein Stand-alone-Gerät für noch mehr Leser geeignet und vielseitiger einsetzbar wäre, weshalb die Idee dann federführend von Luc Lemmens und Mathias Claußen im Labor „elektorisert“, d.h. grundlegend überarbeitet wurde. Statt des Internetanschlusses kommt jetzt eine Benachrichtigung per SMS zum Einsatz. Resultat ist ein universeller



INFOS ZUM PROJEKT



Netzüberwachung

SMS GSM-Modem

ATmega



Einsteiger

➔ Fortgeschrittene
Experte



3 Stunden



Lötkolben

mit schmaler Spitze oder
Heißluftlötstation



etwa 100 Euro

Netzausfall-Detektor, den man überall einsetzen kann, wo ein zu überwachendes Stromnetz plus Mobilfunk vorhanden ist.

Features

Wird ein Netzausfall erkannt, schickt der Detektor sofort eine SMS an eine einstellbare Handy-Nummer und eine zweite SMS, wenn der Strom wieder da ist. Aber das ist noch lange nicht alles: Selbstverständlich ist eine automatische Notstromversorgung per Batterie vorgesehen, und auch deren Spannung wird überwacht. Sollte die Batteriespannung bei einem längeren Stromausfall oder durch Alterung unter einen Schwellwert sinken, wird man auch darüber per SMS informiert. Im Fall der Fälle kann man dann also selbst nachschauen oder wenn das nicht möglich ist, einen Freund oder Nachbarn anrufen und diesen bitten, nach dem Rechten zu sehen.

Die Hardware selbst ist so kompliziert gar nicht geraten (siehe die Blockschaltung in **Bild 1**), denn sie setzt auf eine AVR-MCU und könnte folglich mit der klassischen Arduino-IDE programmiert werden. Es wurde hier allerdings eine kompakte Software in C mit Hilfe von AtmelStudio 7

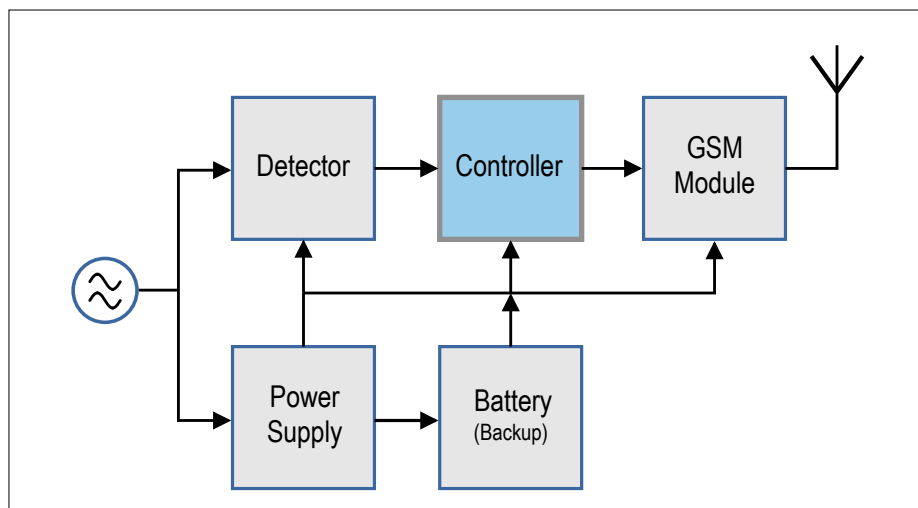


Bild 1. Die Blockschaltung des Netzausfall-Detektors besteht aus fünf funktional getrennten Modulen.

entwickelt, worauf später noch detailliert eingegangen wird. Die Möglichkeiten des Projekts erfordern einen gewissen „Preis“ in Form von ausreichenden Ressourcen des Mikrocontrollers. Statt der ursprünglichen MCU mit nur 8 KB Flash und 512 Byte RAM wurde der Arduino-Klassiker ATmega328P mit immerhin 32 KB Flash und 2 KB RAM ausgewählt. Jetzt ist also auch noch genug Platz für eigene Erweiterungen oder Anpassungen. Die Firmware bietet folgende Funktionen:

- SMS bei Stromausfall
- SMS bei Rückkehr der Netzspannung
- SMS, wenn Backup-Batterien ersetzt werden sollten (<3,9 V)

- SMS bei leeren Backup-Batterien (<3,7 V)
- Energiesparmodus bei Stromausfall nach dem Senden der SMS
- Konfiguration der Rufnummer für den Alarm per SMS
- Speicherung der Konfiguration im EEPROM

Als Kriterium für den Netzausfall wird sicherheitshalber zur Vermeidung von Fehlalarmen (z.B. durch Spikes oder Störungen der Kurvenform) geschaut, ob innerhalb einer Zeitspanne von 250 ms kein Nulldurchgang der Netzspannung erfolgte. Keine Wechselspannung für

mehr als 250 ms gilt daher als Netzausfall. Diese Art der Detektion hat neben der Einfachheit den Vorteil, dass es keinen Unterschied macht, ob ein Netz mit 50 Hz oder 60 Hz überwacht wird.

Hardware

Der allseits beliebte Mikrocontroller ATmega328P ist das zentrale Bauteil der Schaltung von **Bild 2** (rechts oben). Die positiven Halbwellen der Netzspannung werden durch den Optokoppler IC3 erfasst und gelangen so galvanisch getrennt an Pin 1 von IC 2. Da die Sendeleuchte eines Optokopplers nicht sehr spannungsfest ist, werden die negativen Halbwellen durch D1 unterdrückt. Für ausreichende Spannungsfestigkeit bei einem 230-V-Netz braucht es mit R8 und R9 eine Serienschaltung aus zwei Vorwiderständen. Für ein 115-V-Netz ist nur einer der beiden Widerstände erforderlich, und der andere kann durch eine Drahtbrücke ersetzt werden. Dieser Detektor-Konfiguration können die in Netzen auftretenden Spannungsspitzen praktisch nichts anhaben, wenn man mal von extrem seltenem direktem Blitzeinschlag in die Hauszuleitung absieht. Sehr praxisgerecht ist die Verwendung des vergossenen AC/DC-Konverters MOD1 [1] statt eines konventionellen Netzteils mit Trafo. Diese Lösung passt nicht nur gut auf eine Platine, sondern eignet sich dank der Wandlung von 115/230 VAC in 5 VDC für alle Netze in praktisch jedem Land der Welt. Dieses Modul ist mit F1 nicht nur durch eine Sicherung, sondern durch den NTC R6 auch gegen hohe Einschaltströme geschützt, denn bei MOD1 handelt es sich logischerweise um ein Schaltnetzteil mit Siebelko nach dem Eingangs-gleichrichter. Sekundärseitig wird der der Low-Drop-Spannungsregler IC1 von den über L1 und etliche Cs gefilterten 5 V von MOD1 versorgt. IC1 macht daraus 4,4 V, welche über „V+“ einerseits LED1 und LED2 versorgen und andererseits an den integrierten schnellen Umschalter IC5 gelangen. Dieses LTC4413 genannte IC [2] wird vom Hersteller mit „dual 2.6A, 2.5 to 5.5V fast ideal diodes“ beschrieben. Es kann also sehr schnell und unterbrechungsfrei auf die 4,5 V der drei Primärzellen von BT1 umschalten, falls das Netz ausfällt. Dies passiert dann, wenn die Spannung von IC1 unter 3,5 V fällt, was via Pin 2 von IC5 erfasst wird. Die 4,4 V vom Ausgang von IC5 versorgen einerseits das GSM-Modul

LEDs

Die gelbe LED1 ist direkt mit 5 V verbunden und leuchtet daher solange die Netzspannung verfügbar ist. Die grüne LED2 wird über T1 von MOD2 angesteuert und signalisiert die Einbuchung in ein Mobilfunk-Netz. Fehlen noch die drei an die MCU angeschlossenen LEDs: Die rote LED3 ist der Indikator für den Batterie-Status, die orange LED4 signalisiert das Vorhandensein der Netzwechselspannung und die grüne LED5 zeigt den Status des GSM-Modems an. Ihre Funktionen sind wie folgt:

LED	Status	Funktion
LED3	aus	Batterien in Ordnung oder Netzausfall
	blinkt	Schwache Batterie, baldiger Austausch erforderlich
	leuchtet	Batterie leer, kein SMS-Versand mehr möglich
LED4	aus	Netzausfall
	blinkt	Netzwechselspannung vorhanden
LED5	aus	Modem nicht bereit oder Netzausfall
	blinkt @ 1 Hz	Modem eingebucht, aber keine Nummer konfiguriert
	Blinkt @ 10 Hz	Modem initialisiert, aber nicht eingebucht
	leuchtet	Modem ins GSM-Netz eingebucht und betriebsbereit

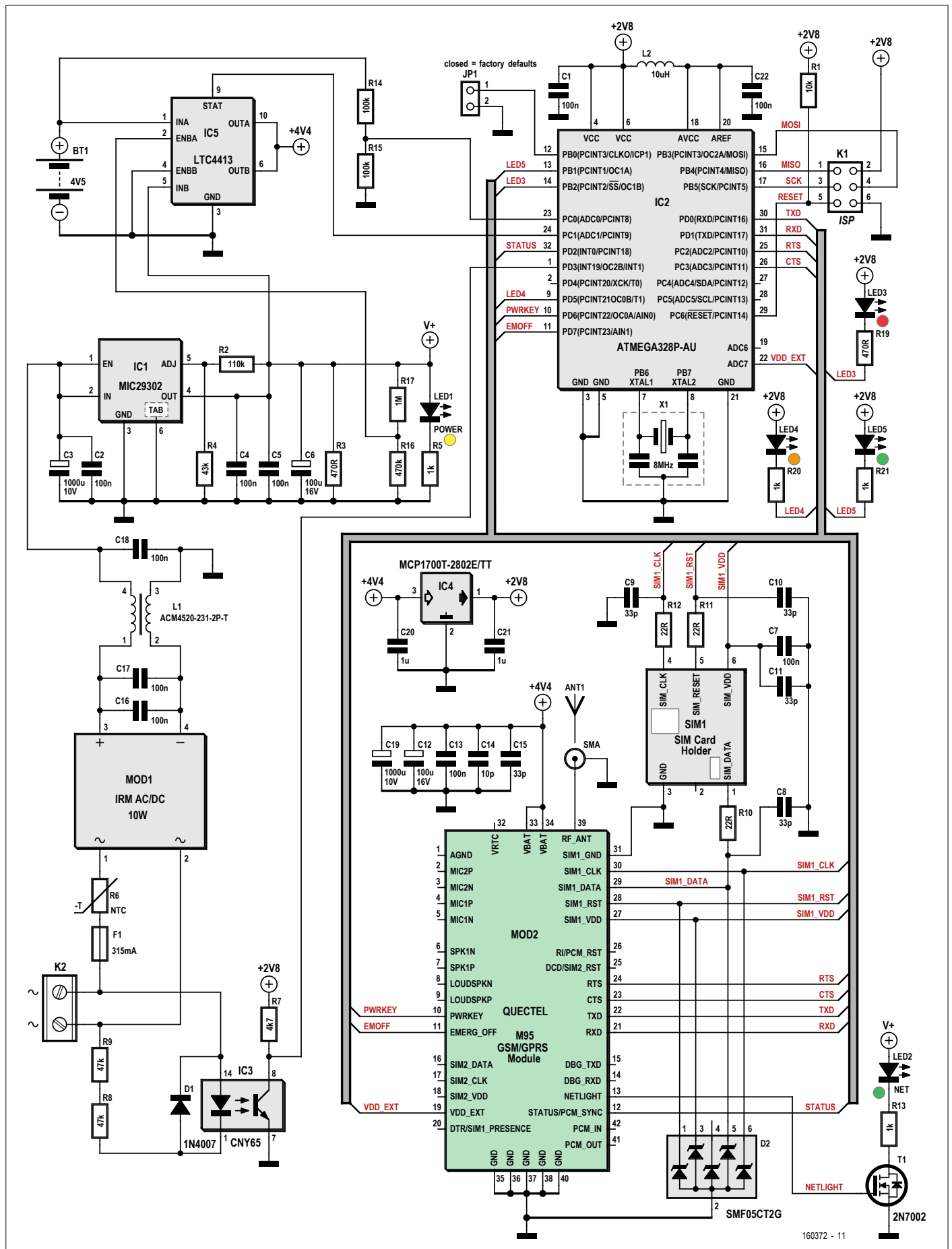


Bild 2. Dank des Blockschaltbilds ist auch die eigentliche Schaltung des Netzausfall-Detektors nicht schwer zu verstehen. Zentrales Bauteil ist der AVR-Mikrocontroller IC2.



STÜCKLISTE

Widerstände:

(wenn nicht anders angegeben:

SMD 0603, 1 %, 0,1 W, 50 V)

R1 = 10 k

R2 = 110 k

R3, R19 = 470 Ω

R4 = 43 k

R5, R13, R20, R21 = 1 k

R6 = NTC 80R B57236S0800M000, Epcos

R7 = 4k7

R8, R9 = 47 k, 5 %, 0,6 W, 250 V, axial,
bedrahtet

R10, R11, R12 = 22 Ω

R14, R15 = 100 k

R16 = 470 k

R17 = 1 M

Spulen:

L1 = Gleichtaktfilter ACM4520-231-2P-T

L2 = 10 μ / 600 Ω @ 100 MHz,
BLM12AJ601SN1D, SMD 0805

Kondensatoren:

(wenn nicht anders angegeben:

SMD 0603, 50 V, keramisch)

C1, C2, C4, C5, C7, C13, C16...C18, C22 = 100 n

C3, C19 = 1000 μ / 10 V, Elko, SMD, radial, $\varnothing$
8 mm

C6, C12 = 100 μ / 16 V, Tantal-Elko, SMD 2312

C8...C11, C15 = 33 p, COG/NP0

C14 = 10 p, COG/NP0

C20, C21 = 1 μ / 10 V

Halbleiter:

D1 = 1N4007, 1000 V / 1 A

D2 = SMF05CT2G, Schutzdioden-Array

LED1 = LED, gelb, SMD 0805

LED2, LED5 = LED, grün, SMD 0805

LED3 = LED, rot, SMD 0805

LED4 = LED, orange, SMD 0805

T1 = 2N7002, N-Kanal-MOSFET, SOT-23

IC1 = MIC29302WU TR, Low-Drop-

Spannungsregler, einstellbar

IC2 = ATmega328P-AU,

programmiert 160372-41

IC3 = CNY65, Optokoppler

IC4 = MCP1700T-2802E/TT, Low-Drop-

Spannungsregler, 2,8 V

IC5 = LTC4413, Spannungsumschalter

Außerdem:

MOD1 = AC/DC-Konverter-Modul 5 V / 2 A,
z.B. Meanwell IRM-10-5

MOD2 = M95, GSM-Modul von Quectel

X1 = CSTCC8M00G53-R0, Keramik-

Resonator, 8 MHz Murata

SIM1 = Micro-SIM-Slot, SMD, z.B. Molex

78723-1001

ANT1 = SMA-Buchse für Platine, z.B. Molex

73391-0070

Passende GSM-Antenne mit SMA-Stecker,

z.B. RF Solutions ANT-GHEL2R-SMA *

JP1 = 2-pol. Stiftleiste, 1x2, stehend, RM 1/10"

Jumper für JP1

F1 = Sicherung T, 5 x 20 mm,

315 mA @ 230 V oder 620 mA @ 115 V

Sicherungshalter für F1, z.B. Multicomp

MC000827

Sicherungsabdeckung für F1 (MC000833) *

K1 = 6-pol. Stiftleiste, 2x3, stehend, RM 1/10"

K2 = 2-pol. Schraubklemme, 630 V,

RM 7,68 mm

BT1 = Batteriehalter 3 x AA-Zellen

Clip für BT1

Platine 160372-1 V2.1

* siehe Text

MOD2 direkt und andererseits über den Low-Drop-Spannungsregler IC4 die MCU plus LED3...LED5 mit 2,8 V. Das ist niedriger als die maximale Versorgungsspannung von IC2 und kommt so der Energieaufnahme entgegen, ist aber auch ausreichend hoch für den mit X1 auf 8 MHz festgelegten Takt.

Falls Sie sich wundern, warum mit MOD1 eine 5-V-Version mit 2 A Belastbarkeit ausgewählt wurde: So ein GSM-Chip wie MOD2 benötigt beim Senden im Mobilfunknetz immer wieder Stromspitzen von locker bis zu 1,6 A. Diese (kurzfristige) Leistungsaufnahme ist notwendig, denn andernfalls wäre die Mobilfunkreichweite bei schlechten Bedingungen zu gering. Aus diesem Grund ist mit C19 auch noch ein extra großer Elko als Puffer vorgesehen, denn normale Alkali-Primärzellen brechen bei solchen Spitzenströmen ungepuffert ein Stück ein, und das GSM-Modul würde sich selbst deaktivieren, wenn die Versorgung die Schwelle von 3,3 V unterschreiten würde.

Die Beschaltung des GSM-Moduls M95 folgt ansonsten weitgehend den Empfehlungen von Quectel [3]. Sein SIM-Karten-Slot SIM1 ist mit R10...R12, D2 sowie C8...C11 gegen HF-Einstrahlung geschützt.

Die MCU überwacht über den Spannungsteiler R14/R15 mit ihrem analogen Eingang ADC0 (Pin 23) auch die Spannung der Backup-Batterie. Mit Pin 24 wird der Status von IC5 abgefragt. Was der Status der einzelnen LEDs zu bedeuten hat, ist extra im **Kasten LEDs** aufgeführt.

Aufbau

Für die Schaltung in Bild 2 wurde im Labor eine kompakte Platine (**Bild 3**) entwickelt, die man entweder im Elektor-Shop beziehen (dort gibt es übrigens auch einen fertig mit der Firmware programmierten Controller) oder sich mit Hilfe der Layout-Dateien von der Elektor-Webseite zu diesem Beitrag [4] selbst machen kann. An gleicher Stellen finden sich natürlich alle benötigten Code-Ressourcen und die HEX-Datei mit der Firmware.

Wie man in **Bild 4** beim fertigen Prototypen sehen kann, ist die Bestückung der Platine so schwierig auch wieder nicht. Zwar sind nur wenige Bauteile vom bedrahteten Typ, aber es sind zumindest keine superwinzigen SMDs oder Chips mit extrem feinen Pin-Abständen vorhanden. Von Hand löten sollte also trotz SMDs im

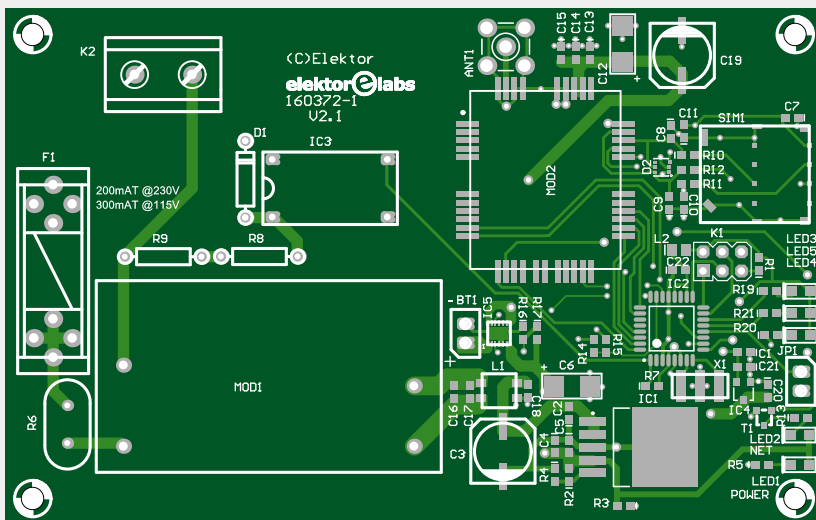
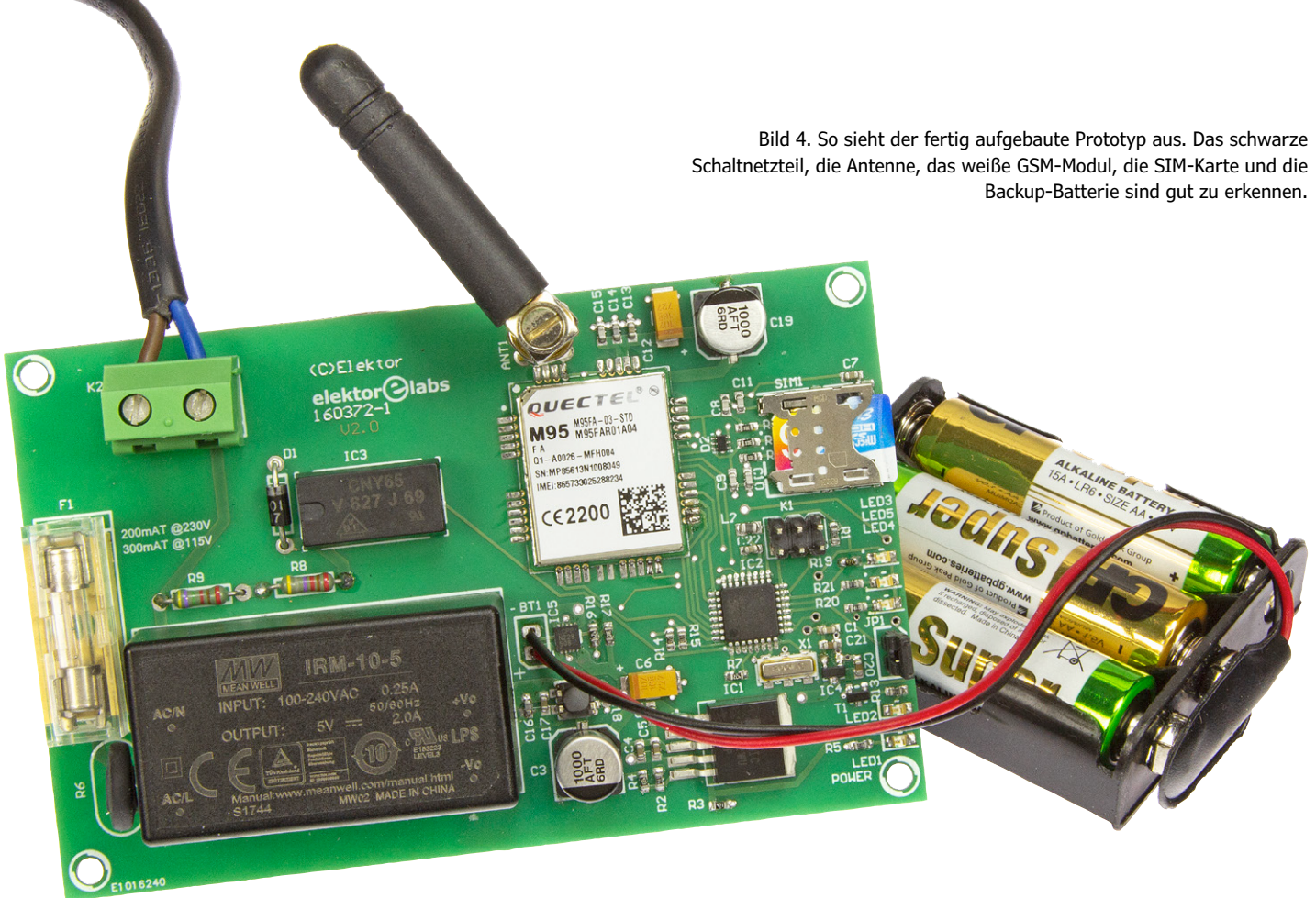


Bild 3. Für die Schaltung in Bild 2 wurde im Elektor-Labor die abgebildete kompakte Platine entwickelt.

Bild 4. So sieht der fertig aufgebaute Prototyp aus. Das schwarze Schaltnetzteil, die Antenne, das weiße GSM-Modul, die SIM-Karte und die Backup-Batterie sind gut zu erkennen.



Format 0603 machbar sein.

Da die Platine (links) Netzspannung führt, sollte sie natürlich in ein isolierendes Plastikgehäuse eingebaut werden. Auch den Halter für die drei AA-Primärzellen sollte man gut befestigen, damit er nicht etwa durch Erschütterungen im Gehäuse herumfliegt oder sich gar eine Zelle aus dem Halter löst. Auch der links abgebildete transparente Schutz der Sicherung ist keine schlechte Idee. Der fertig aufgebaute und konfigurierte Netzausfall-Detektor kann jetzt überall hin mitgenommen und (per passendem Netzkabel) einfach in eine Steckdose gesteckt werden – fertig!

Betrieb & mehr

Da Primärzellen (Batterien) als Notstromversorgung bei Netzausfall verwendet werden, sollte der Energieverbrauch bei Batteriebetrieb so gering wie möglich sein. Wenn man den Mikrocontroller mit 2,8 V und mit einem externen 8-MHz-Resonator betreibt, braucht dieser rund 4 mA. Bei einer Kapazität der Batterien von 1.500 mAh kommt man dann auf nur 375 Stunden = 15,625 Tage Notbetrieb. Der größte Verbraucher ist das Modem. Es benötigt beim Senden einer SMS maximal 300 mA. So hoch wird der Stromverbrauch aber auch, wenn sich das Modem gerade im Netz einbucht oder

dies versucht. Bei permanentem (aber unrealistischen) Modem-Dauerbetrieb ergäbe sich somit ein Stromverbrauch von 304 mA – die Batterien wären dann nach rund fünf Stunden reif für die Tonne. Man müsste sie dann spätestens nach jedem Stromausfall austauschen. Der Stromverbrauch des Mikrocontrollers lässt sich reduzieren, indem man nicht benötigte Teile abschaltet. Beim ATmega328P gibt es hierfür das PRR (Power Reduction Register). Damit lassen sich schnell 10 % = 0,4 mA einsparen. Außerdem gibt es den IDLE-Mode, bei dem der Prozessorkern gestoppt wird, bis ein Interrupt auftritt. Die aktuelle Version der Firmware spart so etwa 50 % Strom, was nur noch 2 mA bedeutet, obwohl die

Software noch aktiv ist.

Bei einem Stromausfall wird aber das Modem zum Problem. Seine bis zu 300 mA sind nicht tolerabel. Daher wird das Modem nach erfolgreichem Versenden der Netzausfall-SMS komplett abgeschaltet. Im Mittel braucht das Modem dann nur noch wenige μ A, was den Batteriebetrieb deutlich verlängert. Der letzte Punkt ist die Deaktivierung der Status-LEDs bei Batteriebetrieb, denn diese würden ebenfalls 2...3 mA verbrauchen. Da der Stromausfall-Detektor in der Regel stationär an einem Ort eingesetzt wird, ist es auch nicht erforderlich, bei Stromausfall etwas mit LEDs zu signalisieren.



IM ELEKTOR-SHOP

→ 160372-1 Unbestückte Platine [4]

www.elektor.de/bare-pcb-160372-1

→ 160372-41 Programmierter Controller ATmega328P [4]

www.elektor.de/mains-outage-detector-controller

→ Mastech MS8301D Smart Digital Multimeter

www.elektor.de/mastech-ms8301d

→ Elektor 02/2012 mit dem Artikel „Belastungsanzeige für Notstrom-Aggregate“

www.elektor.de/elektor-02-2012-pdf-de

→ High Voltage Differential Probes DP10013

www.elektor.de/diff-probes-dp10013

Zur Software

Wie die in C geschriebene Firmware gestrickt ist, kann man dem Flussdiagramm in **Bild 5** entnehmen. Auf die Initialisierung folgt das Einlesen der Konfiguration und das Scharfschalten der Interrupts. Dann wird gewartet, bis die Spannung steht – sicherheitshalber im Sleep-Mode, damit die Batterie nicht leergesogen wird, solange das System nicht vom Netz versorgt wird. Anschließend werden das GSM-Modem konfiguriert und die Spannung der Batterie sowie der Status des Stromnetzes erfasst. Anhand der bisherigen Infos steuert der Controller die Indikator-LEDs passend an. Dann geht es ans Eingemachte: Eine SMS wird bei Netzausfall, Wiederkehr der Netzspannung, zu schwacher oder neuer Batterie verschickt. Solange sich dann nichts ändert, wird wieder geschlafen. Falls kein Netz vorhanden bzw. die Batterie leer ist, erfolgt ein Reset.

Die Software ist in mehrere aufgabenspezifische Module gegliedert. Diese lassen sich in vier Bereiche gliedern (**Bild 6**): Systemüberwachung, Parameterspeicher, Netzüberwachung und Kommunikation.

Zur Netzausfall-Detektion könnte der Abstand zwischen Nulldurchgängen ausgewertet werden, oder man definiert ein Zeitintervall, in dem zwei Halbwellen aufeinander folgen müssen. Auch wenn beide Ansätze so aussehen, als ob sie auf das Ermitteln der Netzfrequenz hinauslaufen, so gibt es doch etwas zu beachten: In Stromnetzen kommt es gelegentlich zu Umschaltvorgängen zur Behebung von Störungen. Dabei kann es zu kurzen Aussetzern und somit zu fehlenden Perioden = Nulldurchgängen

kommen. Damit nicht dann schon ein (falscher) Alarm ausgelöst wird, wird die Zeitspanne etwas größer gewählt und einfach geprüft, ob innerhalb von 250 ms mindestens ein Nulldurchgang erfolgt ist. Mit dieser Methode ist es nicht nötig, zwischen 50- und 60-Hz-Netzen zu unterscheiden.

Für die Netzüberwachung (**Bild 7**) sind

im C-Code der Software die Komponenten MAINS_MON (der eigentliche Monitor) und ein Zeitgeber im Modul TIMER verantwortlich, der als Software-Watchdog fungiert. Dabei wird der externe, auf fallende Flanken reagierende Interrupt 1 der MCU genutzt. Bei einer positiven Halbwelle zieht der Optokoppler den Eingang für Interrupt 1 nach „low“ und erzeugt damit eine Flanke. Innerhalb der Interrupt-Routine wird der Abstand zwischen dem Auftreten von zwei Interrupts ermittelt und der Software-Watchdog des Moduls TIMER zurückgesetzt. Solange der Reset innerhalb von 250 ms geschieht, löst der Watchdog nicht aus. Sollte aber das Netz ausfallen, unterbleibt dieser Reset und sorgt nach Watchdog-Ablauf dafür, dass ein Fehler gemeldet wird. Da TIMER wie Watchdog nur auf Interrupts reagieren, erfolgt die Netzausfallerkennung komplett unabhängig von der Hauptschleife.

Konfiguration

Zur Inbetriebnahme des Systems muss eine gültige SIM-Karte mit deaktivierter PIN verwendet werden. Wenn die SIM mit einer PIN gesichert ist, kann die Software nicht damit umgehen. Man kann die PIN sehr einfach deaktivieren, indem man die SIM-Karte in ein Handy steckt und die PIN in den Sicherheitseinstellungen deaktiviert.

Das GSM-Modul versucht sich ins Mobilfunknetz einzubuchen, sobald die Netzspannung in Ordnung ist oder die Batteriespannung ausreichend hoch ist. Es ist sinnvoll, die bestückte Platine bei der ers-

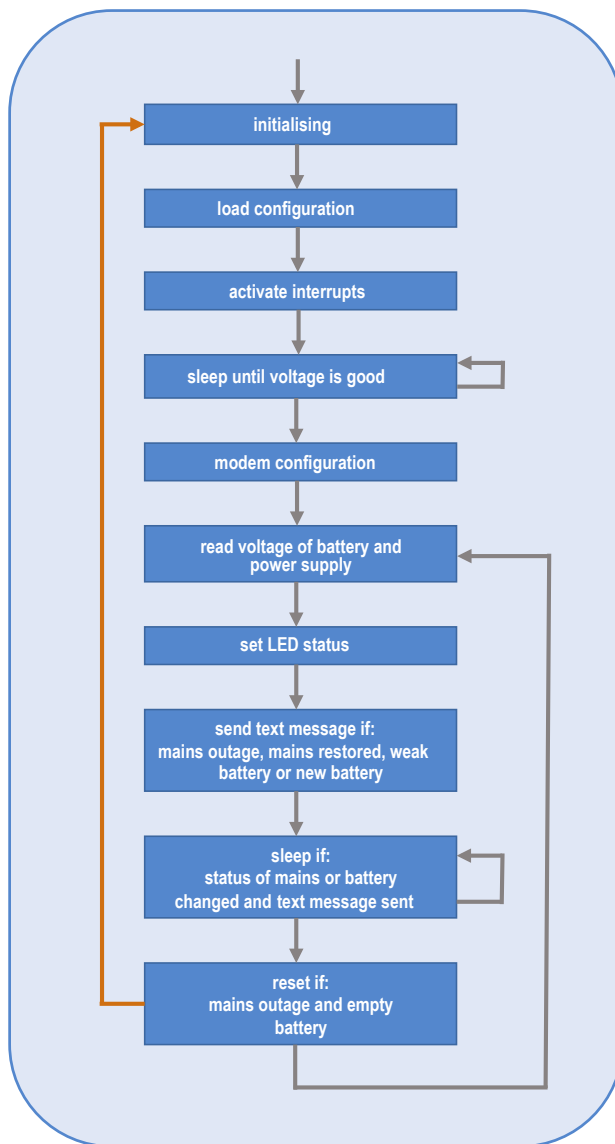


Bild 5. Das Flussdiagramm der in C geschriebenen Firmware.

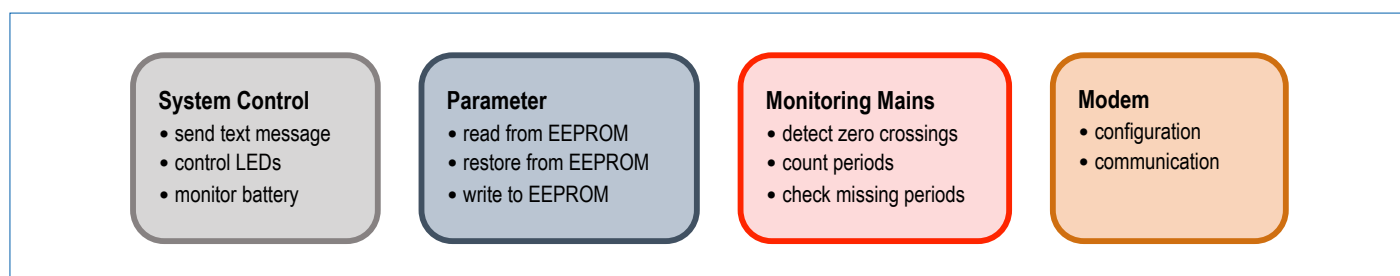


Bild 6. Die Firmware besteht aus vier funktional getrennten Modulen.

ten Inbetriebnahme mit Netzspannung zu versorgen. Wenn keine Netzspannung erkannt und die Karte aus den Batterien gespeist wird, bleiben die LEDs aus Energiespargründen dunkel, was die Konfiguration erschwert.

Ist die Anmeldung am Mobilfunknetz komplett gelungen, leuchtet LED5. Ab jetzt könnte man die Telefonnummer der SIM-Karte anrufen und so prüfen, ob die SIM-Karte korrekt angemeldet ist und nicht vom Netz blockiert wird. Die Schaltung gibt allerdings noch keine Warnung aus, denn es ist ja noch keine gültige Rufnummer im EEPROM des Mikrocontrollers eingetragen. Auf diese Weise wird verhindert, dass das System unbeabsichtigt SMS an eine unbekannte Nummer sendet. Um diesen Status zu signalisieren, sollte die grüne LED zunächst mit 1 Hz blinken.

Die Rufnummer wird per SMS eingestellt. Dazu muss man mit einem Handy (mit anderer Nummer) eine SMS mit einem bestimmten Text an die Rufnummer der eingesteckten SIM-Karte senden. Hierzu wird der Text „Config Remote XXX“ verwendet – „XXX“ steht für die SIM-Rufnummer. Nur die Zeichen „+,0,1,2,3,4,5,6,7,8,9“ sind hierfür erlaubt. Wurde die SMS empfangen und erfolgreich verarbeitet, antwortet das System mit einer SMS des Inhalts „Saved new number in EEPROM“. Das System ist nun einsatzbereit. Die grüne LED hört nach dem Senden der Antwort-SMS auf zu blinken und leuchtet jetzt dauerhaft. Ab jetzt kann der Status des Systems per SMS abgefragt werden. Dazu muss eine SMS mit dem Inhalt „Request Status“ an die Rufnummer der verwendeten SIM-Karte gesendet werden. Das System antwortet dann mit einer SMS des Inhalts „Mains frequency is <STATUS>

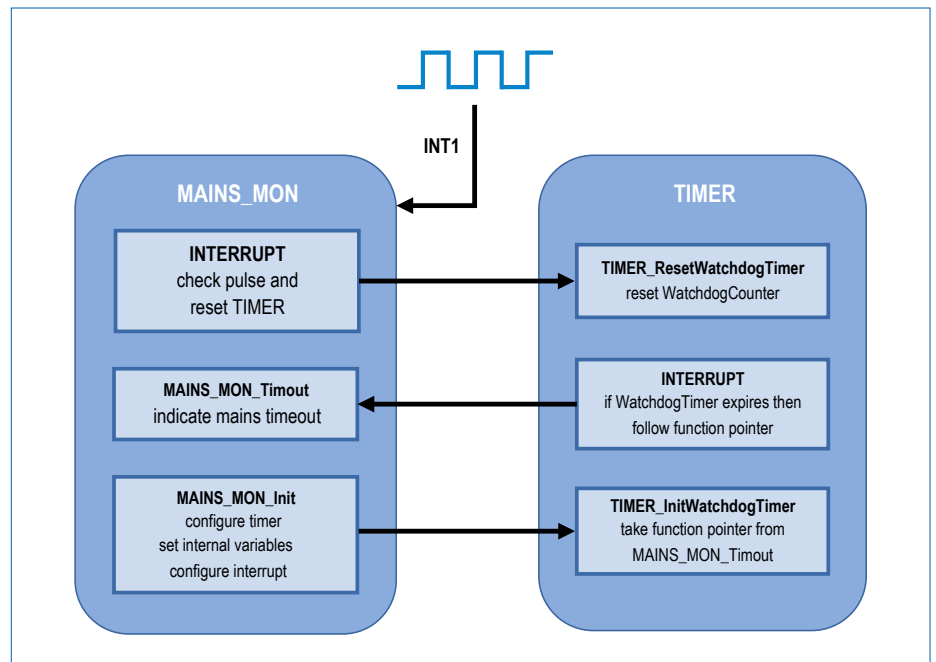


Bild 7. Watchdog in Software.

Weblinks

- [1] Meanwell IRM-10-5: <https://bit.ly/2KsAsYr>
- [2] LTC4413: <https://bit.ly/2Kfsiq6>
- [3] Quectel M95: <http://www.quectel.com/product/m95.htm>
- [4] Download: <http://www.elektormagazine.de/180344-01>

with <XX>Hz. Battery has <VBat>mV and is <BAT_STATUS>“.

Für einen Reset der aktuellen Konfiguration muss man Jumper JP1 stecken und das System aus und wieder einschalten – es muss also kurz stromlos sein, damit der Status von JP1 beim Booten erfasst werden kann. Nach diesem Neustart sollte die grüne LED wieder blinken. Der Jumper sollte dann wieder entfernt wer-

den, damit das System beim nächsten Start nicht schon wieder auf die Werkseinstellungen zurückgesetzt wird. Aus Sicherheitsgründen (**Netzspannung auf der Platine!**) sollte dies nur bei abgeschalteter Netzspannung erfolgen! Das System kann nun wie beschrieben neu konfiguriert werden. ◀

180344-01

Anzeige



**Hand Held Gehäuse
Standard und Wasserdicht**

www.hammondmfg.com/1553.htm

www.hammondmfg.com/1553W.htm

+ 44 1256 812812

sales@hammondmfg.eu

