

Universeller Baukasten für Entwickler

Sensorik für IoT-Anwendungen

Auf der Basis eines Mikrocontrollers mit integriertem Funktransceiver von Nordic Semiconductor entsteht mit Sensoren und Erweiterungsmodulen ein Baukasten für IoT-Geräte. Ergänzt mit leistungsfähigen Tools lassen sich damit Prototypen und Proof of Concepts schnell realisieren.

Von Dr. Claus Kühnel



Die Entwicklung von Geräten für das Internet der Dinge (IoT) stellt Firmen oftmals vor eine Reihe von neuen Herausforderungen. IoT-Geräte zeichnen sich durch unterschiedliche Arten der Kommunikation untereinander oder mit einem Gateway aus. In der Regel wird diese Kommunikation per Funk erfolgen. Sensoren erfassen unterschiedliche Daten – Aktoren wirken auf zu steuernde Prozesse. Die Sensoren und Aktoren sind in ein IoT-Gerät zu inte-

grieren, was weitgehend über standardisierte Interfaces, wie zum Beispiel SPI, I²C, I²S, UART oder GPIO, erfolgt.

Die meisten IoT-Geräte kommunizieren nicht nur per Funk, sondern sollen vollständig autonom, also ohne externe Spannungsversorgung im Batteriebetrieb arbeiten. Stromsparende Schaltungen auf der Basis von Low-Power-Mikrocontrollern ermöglichen heute den Batteriebetrieb über Jahre hinaus. Auch der Einsatz von Akkus,

die per Solarzelle geladen werden, ist eine Option.

Dem reduzierten Strombedarf entgegen steht bei einer bestimmten Klasse von IoT-Geräten die Forderung nach erhöhter Rechenleistung (Edge-Computing), d. h. digitale Vorverarbeitung direkt im IoT-Gerät – z. B. hinsichtlich Security- und Safety-Aspekten, Datenkonzentration u.a.m.

Für dieses Aufgabenfeld stellt die Dresdner Firma Sensry mit Kallisto

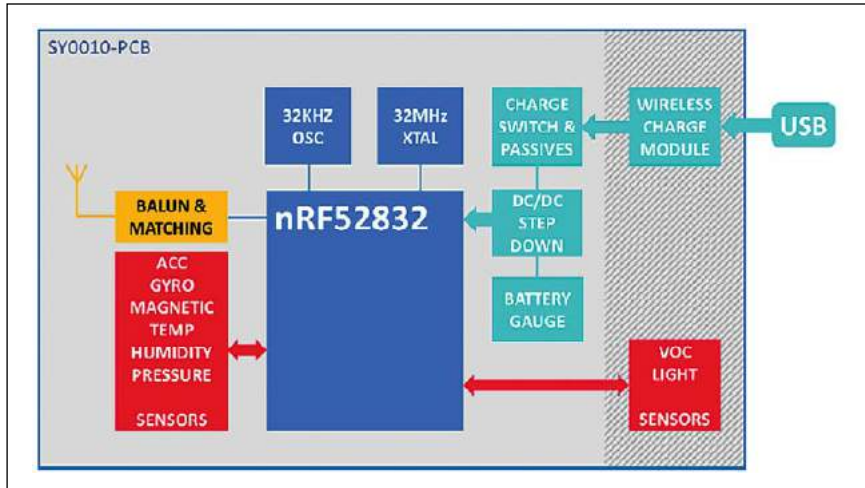


Bild 1. Blockdiagramm des Kallisto Starterkits mit dem Mikrocontroller nRF5282 als zentralem Baustein, Sensoren, Stromversorgung, Takterzeugung und HF-Antenne für Bluetooth Low Energy (BLE). (Bild: Sensry)

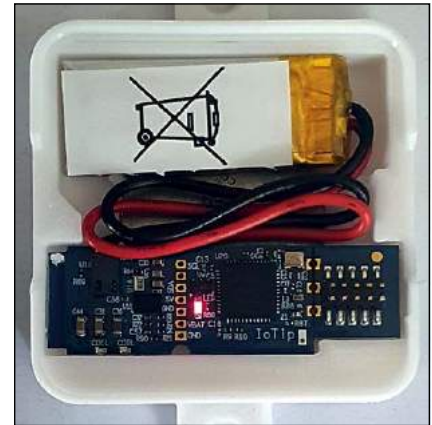


Bild 2. Wird das Gehäuse des Kallisto Starterkits geöffnet, ist eine Platine mit der Elektronik und der Akku sichtbar, die Wireless-Power-Spule befindet sich darunter. (Bild: Claus Kühnel)

eine universelle Sensorplattform für IoT-Anwendungen bereit, die einen nahtlosen Übergang vom Konzept zum Massenprodukt ermöglicht. Kallisto bietet Entwicklungsunterstützung über das geschilderte Spektrum von IoT-Anwendungen.

Kallisto ist eine auf einem Prozessor Cortex-M4F von ARM basierende

Smart-Node-Plattform für IoT-Sensorkanwendungen. Sie ist in verschiedenen Versionen erhältlich, wird ständig erweitert und zeichnet sich durch die komplette Integration aller notwendigen Komponenten als ein autarkes und energieeffizientes IoT-System aus, sodass sich Anwendungen »out of the box« umsetzen lassen.

Mit einer solchen Sensorplattform für IoT-Anwendungen steht ein effektives Werkzeug zur Verfügung, das es auch dem nicht auf IoT-Anwendungen spezialisierten Entwickler aus dem Bereich der Maker oder des Mittelstandes die Umsetzung seiner Produktidee oder eines damit einhergehenden neuen Geschäftsmodells ermöglicht [1–3].



DATA MODUL

DIE DISPLAY EXPERTEN

Unsere Erfahrung, unsere Branchenexpertise und unser Marktwissen in allen Bereichen der Displaytechnik machen uns zum perfekten Partner für professionelle Displaylösungen. Mit unserem umfangreichen Portfolio aller führenden Hersteller sowie vielfältigen Individualisierungsmöglichkeiten können wir nahezu jede Kundenanforderung effizient bedienen.

Für weitere Informationen
besuchen Sie uns auf
www.data-modul.com



Funktion	Sensor	Hersteller	Typ
Bewegungserfassung	3-Achsen-Beschleunigungssensor	Bosch	BMI160
	3-Achsen-Gyroskop		
	3-Achsen-Magnetometer	Bosch	BMM150
Umgebungserfassung	Temperatursensor	Bosch	BME280
	Rel. Luftfeuchtigkeitssensor		
	Luftdrucksensor		
	Helligkeitssensor	Texas Instruments	OPT3002
	Luftqualitätssensor	AMS	CC811

Tabelle: Fünf Sensoren auf dem Kallisto-Modul erfassen acht verschiedene Messwerte. (Quelle: Sensry)

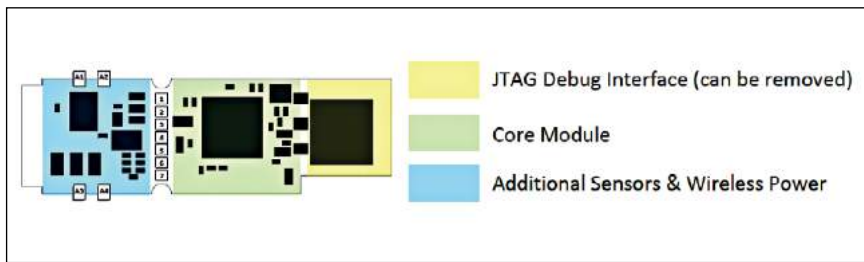


Bild 3. Die Platine des Kallisto Starterkits besteht aus drei Teilen: Sensorik und Stromversorgung, Mikrocontroller und die JTAG-Schnittstelle. (Bild: Sensry)

Beginnen mit einem Starterkit

Das Kallisto-SY010-Starterkit ist ein IoT-Gerät, mit dem anfangs das Konzept der Smart-Node-Plattform als Basis kundenspezifischer IoT-Entwicklungen gezeigt werden sollte.

Basis der Schaltung ist ein Mikrocontroller von Nordic, der nRF52832. Der Cortex-M4F-Prozessorkern – mit Hardware für Gleitkommaarithmetik – ist mit 64 MHz getaktet und weist mit 512 KB Flash und 64 KB SRAM genügend Ressourcen für viele IoT-Anwendungen auf. **Bild 1** zeigt die Ausstattung des Starterkitmoduls mit zahlreichen Sensoren sowie BLE-Antenne auf der Leiterplatte und Schaltung für die induktiven Energieübertragung (Qi) zum Laden des internen LiPo-Akkus. Beim Blick in das geöffnete Gehäuse des Kallisto SY010 (**Bild 2**) ist die dreigeteilte Kallisto-Platine und der darüber platzierte 110-mAh-LiPo-Akku zu erkennen. Die Spule für den Qi-Wireless-Power-Empfänger ist verdeckt. Die drei in einer Platine vereinten Module sind in **Bild 3** dargestellt.

Das Kallisto-Modul ist mit den in der Tabelle aufgeführten Sensoren ausgerüstet. Der Mikrocontroller ist von Hause aus mit einer Standard-Firmware programmiert, die den Zugriff auf alle

Sensoren gestattet. Die Entwicklung eigener Firmware für das Kallisto-Modul kann unter Windows mit einer der bekannten Entwicklungsumgebungen, wie Segger Embedded Studio (SES), MDK (Microcontroller Development Kit) von ARM, Keil mit µVision IDE (Integrated Development Environment), GCC (GNU Compiler Collection) oder der Embedded Workbench von IAR, erfolgen. Die Programmierung selbst erfolgt über das JTAG-Interface.

Mit dem Kallisto-Modul sind zwei Entwicklungsszenarien denkbar:

- Programmentwicklung für ein IoT-Gerät auf dem Kallisto-Modul
- Verwendung des Kallisto-Moduls als IoT-Gerät und Entwicklung einer Anwendung auf einem Host (Companion Board)

Im ersten Szenario können beispielsweise andere Sensoren integriert und der vorhandene Bluetooth-LE-Protokoll-Stack (BLE Stack) zur Kommunikation mit einem zentralen BLE-Knoten (Central Device) verwendet oder ein anderes Kommunikationsmodul, z. B. LoRaWAN, eingesetzt werden.

Im zweiten Szenario wird das Kallisto-Modul als Sensorknoten genutzt, der einem zentralen BLE-Knoten Daten zur Verarbeitung bzw. Weiterleitung über-

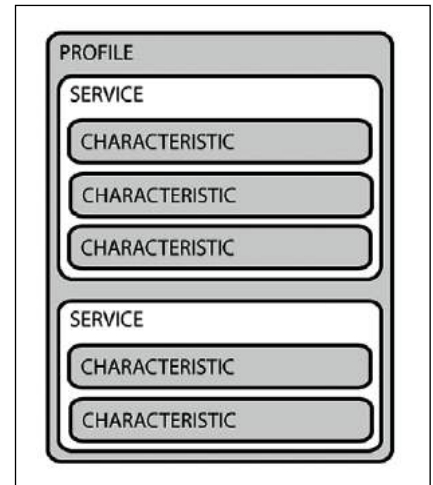


Bild 4. BLE definiert für die energiesparende Übertragung von kleinen Datenmengen wie Sensordaten ein generisches Datenprotokoll (GATT, Generic Attribute Profile) bei dem Services und die zugehörigen Characteristics in einer einfachen Tabelle gespeichert werden. (Bild: Claus Kühnel)

mittelt. Sensry stellt ein API für den Zugriff auf den vorhandenen BLE-Stack zur Verfügung.

Zugriff auf Sensordaten über BLE

GATT ist ein Akronym für das Generic Attribute Profile und definiert die Art und Weise, wie zwei BLE-Knoten Daten mithilfe von Services und Characteristics bidirektional übertragen. Ein generisches Datenprotokoll namens Attribute Protocol (ATT) wird verwendet, um Services, Characteristics und zugehörige Daten in einer einfachen Tabelle (Look-up Table) zu speichern (**Bild 4**). GATT kommt ins Spiel, sobald eine dedizierte Verbindung zwischen zwei Geräten hergestellt wurde. Das bedeutet, dass der Advertising-Prozess bereits durchlaufen ist.

Verbindungen über BLE sind exklusiv, d. h. dass ein BLE-Peripheriegerät (BLE Peripheral) immer nur an ein zentrales Gerät (BLE Central, Mobiltelefon etc.) angeschlossen werden kann. Sobald sich ein Peripheriegerät mit einem zentralen Gerät verbindet, stoppt es sein Advertising und ist damit für andere Geräte nicht mehr sichtbar, bis die bestehende Verbindung unterbrochen wird.

Dieser Verbindungsaufbau ist die einzige Möglichkeit, eine Zwei-Wege-Kommunikation zu ermöglichen, bei der ein zentraler BLE-Knoten aussagekräftige Daten an ein BLE-Peripheriegerät senden bzw. von diesem empfangen kann.

Zugriff per Smartphone

Mit nRF Connect, einem leistungsstarken Tool von Nordic Semiconductor, können vom Smartphone aus BLE-Geräte gescannt werden, es ist damit auch möglich, sich mit einem BLE-Gerät zu verbinden und zu kommunizieren.

Bild 5 zeigt die Verbindung mit einem Kallisto-Modul (Kallisto_Sensing_E6_ED). Farbig markiert sind im Ambient-Service mit der Base UUID 0x1120 die Characteristics mit der UUID 0x1133 (Temperature Data) sowie mit der UUID 0x1134 (Temperature Configuration).

Durch das Schreiben des Wertes 0x41 in die Characteristic mit UUID 0x1134 wird der Temperatursensor aktiviert und sendet daraufhin im Sekundentakt Messwerte (Notification).

Die Temperaturdaten werden nun im Sekundentakt in der Characteristic mit der UUID 0x1133 aktualisiert und stehen hier beispielsweise in den ob-



Bild 5. Mit der Smartphone-APP nRF Connect von Nordic Semiconductor ist es möglich, eine Verbindung zu einem BLE-Gerät herzustellen und Daten in beide Richtungen zu übertragen. (Bild: Claus Kühnel)

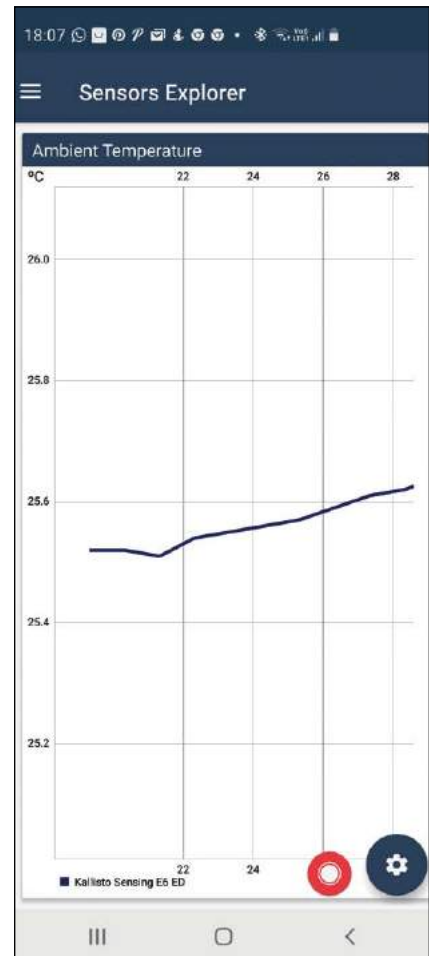


Bild 6. In der Kallisto Toolbox von Sensry lassen sich die empfangenen Sensordaten speichern und in Diagrammform anzeigen, wie in diesem Beispiel der Temperaturverlauf. (Bild: Claus Kühnel)

Keep it simple!
Sense with MEMS.



MEMS Sensor Portfolio & Kundenservice

Sensoren sind ein integraler Bestandteil jeder zukünftigen Anwendung. Das Messen von Temperatur, Feuchtigkeit, Druck oder Beschleunigung war noch nie einfacher. Nutzen Sie die Vorteile unseres Software Development Kits und der ab Werk erhältlichen Evaluierungsboards. Ausführliche Dokumentationen sowie der direkte Support durch geschulte Ingenieure lassen keine Fragen offen. Mit exzellenter Messgenauigkeit und Langzeitstabilität liefern die Sensoren hochpräzise und akkurate Ausgangswerte mit intelligenten On-Chip-Interrupt-Funktionen.

Kombinieren Sie Sensoren und Funk – starten Sie Ihre IoT-Anwendung noch heute:

www.we-online.de/sensors

- Support durch Ingenieure innerhalb von 24 h
- Exzellente Messgenauigkeit
- Ab Werk kalibriert und direkt einsatzbereit
- Kundenspezifische Interrupt Einstellungen
- Implementierte Algorithmen
- Digitale SPI- & I²C-Schnittstellen

#SensewithMEMS



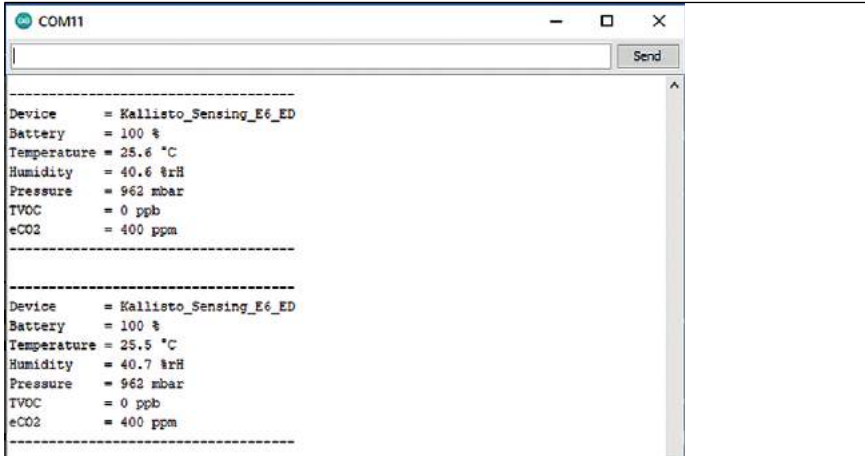


Bild 7. Die an der seriellen Schnittstelle des Arduino Uno Wi-Fi Rev2 ausgegebenen Sensordaten des Kallisto-Moduls werden etwa im Minutentakt aktualisiert. (Bild: Claus Kühnel)

ren vier Bytes als Wert 0x000009F5 zur Verfügung. Decodiert ist das ein Wert von 2549 dezimal, d. h. 25,49 °C. Auf diese Weise lässt sich jede Characteristic in der GATT Look-up Table untersuchen.

Wichtig für die Interpretation der Daten ist die Kenntnis der GATT-Tabelle, die der Hersteller des BLE-Gerätes zur Verfügung stellt.

Grafische Darstellung per Kallisto-Toolbox

Sensry bietet mit der Kallisto-Toolbox für Android Devices ein Tool zur vollständigen Konfiguration und Überprüfung des Kallisto-Moduls. Damit lassen sich die Sensorsignale in Echtzeit überprüfen und der Anwender bekommt auf diese Weise einen Eindruck von der Leistungsfähigkeit dieses kleinen IoT-Geräts. Die Kallisto-Toolbox (Bild 6) basiert auf der Kallisto-Android-API und zeigt einige ihrer Hauptfunktionen.

Zugriff über eine Arduino-APP

Mindestens für Maker, aber auch für am raschen Prototyping interessierte Entwickler, ist der Zugriff von einem Arduino – oder einem ähnlichen Mikrocontroller-Entwicklungsmodul – von Interesse. Die beim Arduino Uno Wi-Fi Rev2 vorhandene BLE-Funktion lässt sich mithilfe der Library ArduinoBLE für den Zugriff auf die vom Kallisto erhobenen Daten nutzen.

Das für den Zugriff auf das Kallisto-Modul geschriebene Programm *KallistoTestv1.ino* erlaubt es den Battery-Service und den Ambient-Service sowie die dazugehörigen Characteristics zu lesen.

Ohne hier auf die Einzelheiten eingehen zu wollen, zeigt Bild 7 die Datenausgabe des Programms über die Konsole der Arduino IDE. Etwa im Minutentakt werden die Sensordaten des Ambient-Service abgefragt.

Mit diesen Sensormesswerten ist es beispielsweise möglich, die Luftqualität und das Klima in einem Innenraum zu bewerten. Zusammen mit dem Arduino lässt sich also sehr schnell eine Messstation erstellen. Durch die Verwendung eines Arduino Uno Wi-Fi Rev2 können die Messwerte dann auch gleich

über Wi-Fi an einen Router und so ins Internet übertragen werden.

Linux-API für den Datenaustausch

Für Linux-Anwender interessant ist die Linux API, die Sensry per GitLab, erreichbar unter der URL: <https://gitlab.sensry.net/kallisto/kallistoapi>, bereitstellt. Alle Hauptfunktionen der API und deren Verwendung in einer Linux-Anwendung sowie die erforderlichen Interaktionen werden mit Codeschnipseln detailliert erklärt. Die Bibliothek wird als Linux Shared Library geliefert und muss auf dem System installiert werden.

Kleiner aber leistungsfähiger: Kallisto SY020

Das Modul Kallisto SY020 baut auf den Entwicklungserfahrungen mit dem Kallisto SY010 auf und bietet nicht nur ein weiterentwickeltes IoT-Gerät, sondern mit den Kallisto Extension Boards auch gleich das Bindeglied zur eigentlichen Anwendung.

Das briefmarkengroße Kallisto-Modul SY020 weist den leistungsstärkeren Mikrocontroller nRF52840 – mit einem Cortex-M4F-Prozessorkern von ARM, 256 KB RAM und 1 MB Flash-Speicher – und wieder zahlreiche Sensoren auf (Bild 8). Einen Vergleich von nRF52840 und nRF52832 hat Nordic Semiconduc-

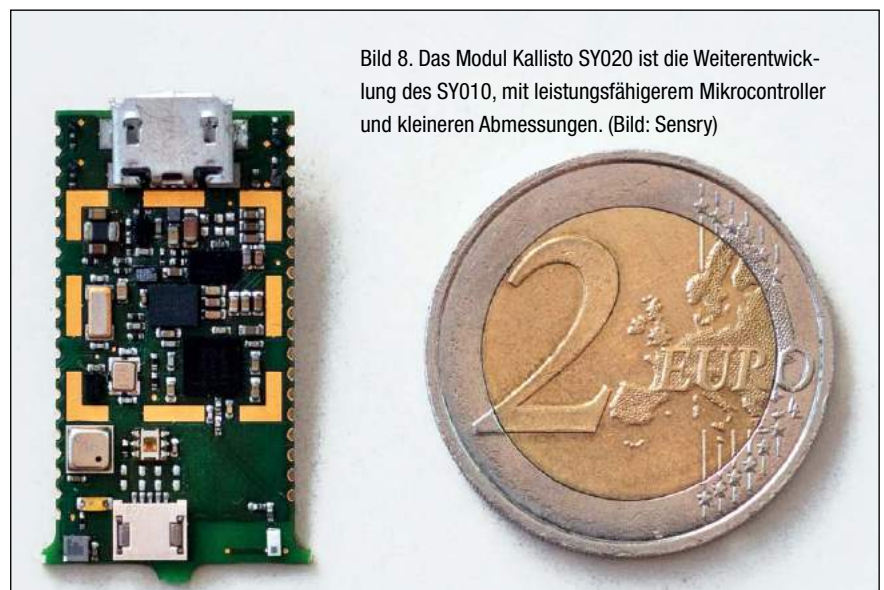


Bild 8. Das Modul Kallisto SY020 ist die Weiterentwicklung des SY010, mit leistungsfähigerem Mikrocontroller und kleineren Abmessungen. (Bild: Sensry)

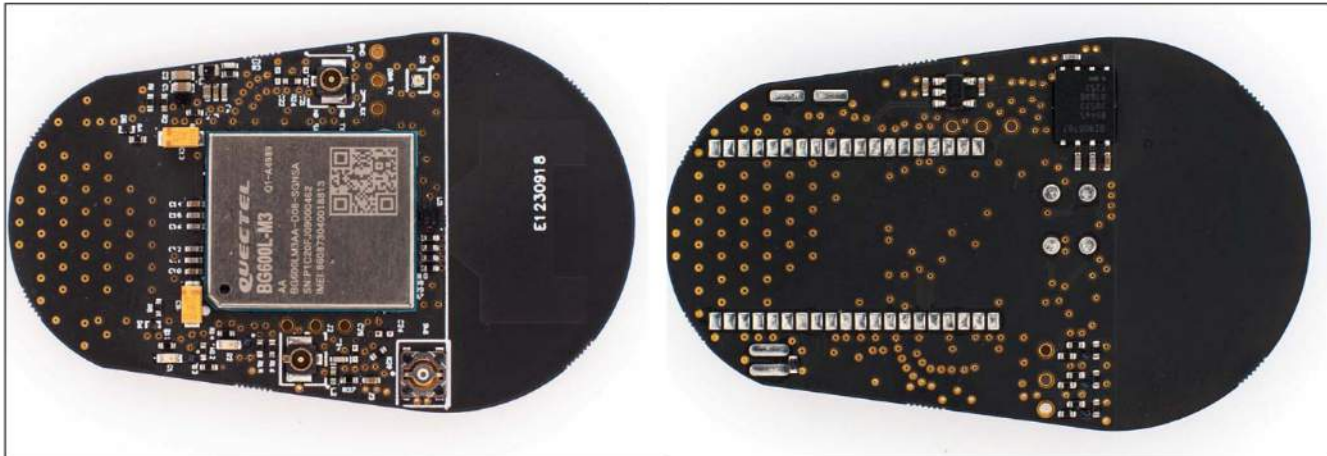


Bild 9. Kallisto-Erweiterungsmodul SY020-M01 (Vorder- und Rückseite) mit dem Mobilfunkmodul, Antennenanschlüssen und den Kontakten zum Verlöten des Sensormoduls Kallisto SY020 (siehe Bild 8). (Bild: Sensry)

tor auf seiner Website veröffentlicht [4]. In der Mitte des SY020-Moduls ist der Bereich zu erkennen, der mit einer Abschirmung versehen werden kann. Die Anschlüsse am Platinenrand ermöglichen eine einfache Montage auf einer Baugruppe.

Der Racemap-Tracker aus [3] ist ein sehr gutes Beispiel für eine IoT-Entwicklung auf Basis des Kallisto-SY020-Moduls, das die Grundlage für das Tracking-Modul bildet. Eine Reihe von optionalen Sensoren ermöglicht die effiziente Erfassung von Bewegung und Umweltparametern. Durch ein striktes Energiemanagement ist es Sensry gelungen, die Leistungsaufnahme zu minimieren. Auf dem Kallisto-Erweiterungsmodul SY020-M01 (Bild 9) sorgt ein Funkmodul von Quectel (BG600) für eine Mobilfunkverbindung per LTE-Cat NB2 (NB-IoT), LTE-Cat M1 und per EGPRS (2G) als Rückfallebene. Für die Positionserkennung ist im Funkmodul ein GNSS-Empfänger integriert. Im Erweiterungsmodul ist zusätzlich ein RFID-Transceiver verbaut, der die Erkennung und Zählung der Module im Nahfeld ermöglicht [5].

Durch die Kombination eines kompakten Controller-Moduls mit einem, die Besonderheiten der jeweiligen Anwendung aufnehmendem Extension-Modul, lässt sich baukastenartig eine anwendungsspezifische Schaltung erstellen. Andere Funktechniken wie z. B. LoRa als Kommunikationsschnittstelle werden bei Sensry bereits entwickelt. Dann können mit dem Kallisto über

BLE-konfigurierbare LoRaWAN-Knoten realisiert werden.

Entwicklungsunterstützung

Zur Entwicklungsunterstützung stellt Sensry ein Software-Repository für Kallisto zur Verfügung (<https://gitlab.sensry.net>). Nach der Registrierung stehen alle Quellen für Kallisto – SDK, Standard-Firmware, Android-API, Linux-API sowie Anwendungsbeispiele – für die eigene Produktentwicklung zur Verfügung. Auf der Repository-Website befinden sich:

- NordicSDK – wird vom KallistoSDK verwendet
- KallistoSDK – Software Development Kit für die Kallisto-Knoten (Sensor-Treiber, HAL, BLE-HAL)
- KallistoBleFw – vorinstallierte Standard-Firmware
- KallistoFwTemplate – Template zur Erstellung eigener Firmware
- KallistoAPI – Linux API (Library)
- KallistoLinuxExample – Beispiel für Linux
- KallistoAndroidAPI – Android API (Library)
- KallistoAndroidToolbox – Toolbox (Android APP)

Außerdem baut Sensry eine Support-Seite (<https://docs.sensry.net>), mit entwicklungsbegleitenden Informationen zu Hard- und Software sowie Entwicklertools auf, die zum Erstellen eigener vernetzter Sensry Produkte benötigt werden. hs

Literatur

- [1] Kühnel, C.: MAKER TOOLS – Entrepreneurs der Stunde: Maker – Startups – Unternehmen. DESIGN&ELEKTRONIK 2016, H. 1. S. 16–19.
- [2] Kühnel, C.: Entrepreneurs der Stunde, Teil 2: Maker – Startups – Unternehmen. DESIGN&ELEKTRONIK 2016, H. 3, S. 36–38.
- [3] Sachsen schmieden virtuelle Fabriken. Wirtschaft-in-sachsen.de, 7. Juli 2021, www.wirtschaft-in-sachsen.de/news/sachsen-schmieden-virtuelle-fabriken-gid-5199
- [4] NRF52832, NRF52840 and NRF52810 comparison. Nordic Semiconductor, Nordic Dev-Zone, <https://devzone.nordicsemi.com/f/nordic-q-a/24919/nrf52832-nrf52840-and-nrf52810-comparison>
- [5] Project Live-Tracking Plattform with Kallisto. Sensry, Website, <https://sensry.net/live-tracking-plattform/>



Dr. Claus Kühnel

studierte und promovierte an der Technischen Universität Dresden auf dem Gebiet der Informationselektronik und bildete sich später in Biomedizintechnik weiter. Von 2004 bis zu seiner Pensionierung war er bei QIAGEN Instruments in Hombrechtikon (CH) als Director Electronic Engineering & Embedded Systems für die Entwicklung von Elektronikhardware und hardwarenaher Software verantwortlich. Aktuell arbeitet er als Autor und Consultant. info@ckuehnel.ch

Grafik-ICs mit GUI-Simulator

HMI-Entwicklung vereinfachen



(Bild: wee dezing | Shutterstock)

Das Graphical User Interface (GUI) wird in der Regel erst beim Testlauf auf dem Zielsystem bewertet und dann iterativ optimiert. Soll ein Iterationsschritt getestet werden, muss ihn ein Softwareentwickler umsetzen. Neue Ansteuer-ICs stellen GUI-Simulatoren bereit, mit denen einige Testläufe eingespart werden können. Von Norbert Erhart