Sensoranbindung unter Embedded-Linux: Wie Industrial-I/O den Unterschied macht

Wie lassen sich Sensoren effizient und flexibel an Embedded-Linux-Systeme anbinden? Im MicroConsult-Webinar gab Embedded-Linux-Experte Andreas Klinger einen praxisnahen Einblick in eines der wichtigsten Subsysteme dafür: Industrial I/O (IIO) im Linux-Kernel.

Und wie so oft bei Andreas beginnt alles mit einem realen Projekt – in diesem Fall mit einer **smarten Bienenstock-Waage**.

Warum Industrial-IO?

Der Linux-Kernel bietet grundsätzlich zwei Subsysteme für Sensorik:

- **HWmon:** etabliert, aber eher für klassische Systemüberwachung
- **Industrial I/O:** moderner, flexibler, schneller und mit einer großen Zahl unterstützter Sensoren

IIO verfolgt ein klares Ziel: ein einheitliches Userspace-Interface für unterschiedlichste Sensoren.

Egal ob Temperatur, Luftdruck, Feuchtigkeit oder analoge Eingangsspannungen – im IIO-Subsystem sieht die Schnittstelle immer gleich aus. Das bedeutet:

- einfachere Sensorwechsel
- standardisierte Dateistrukturen im SysFS
- klare Skalierungsmodelle (SI-Einheiten über feste Skalierungsfaktoren)
- Unterstützung für eine große Bandbreite an Sensoren

Über 300 Sensoren und Aktoren werden heute bereits im Mainline-Kernel unterstützt – Tendenz steigend.

Praxisbeispiel: Die vernetzte Bienenstock-Waage

Um die Funktionsweise von IIO greifbar zu machen, stellte Andreas sein privates Projekt vor: eine Waage, die Gewicht, Temperatur, Feuchtigkeit und Luftdruck seiner Bienenstöcke misst und daraus wichtige Informationen ableitet:

- Wie viel Honig ist eingetragen?
- Hat ein Volk geschwärmt?
- Gibt es Räuberei zwischen Völkern?
- Wie verändert sich das Flugverhalten abhängig von Temperatur und Luftdruck?

©MicroConsult 02-26

Herzstück des Systems sind zwei Komponenten:

1. **HX711 – ein 24-Bit-ADC, der feinste Spannungsänderungen der Wägezelle erfasst**

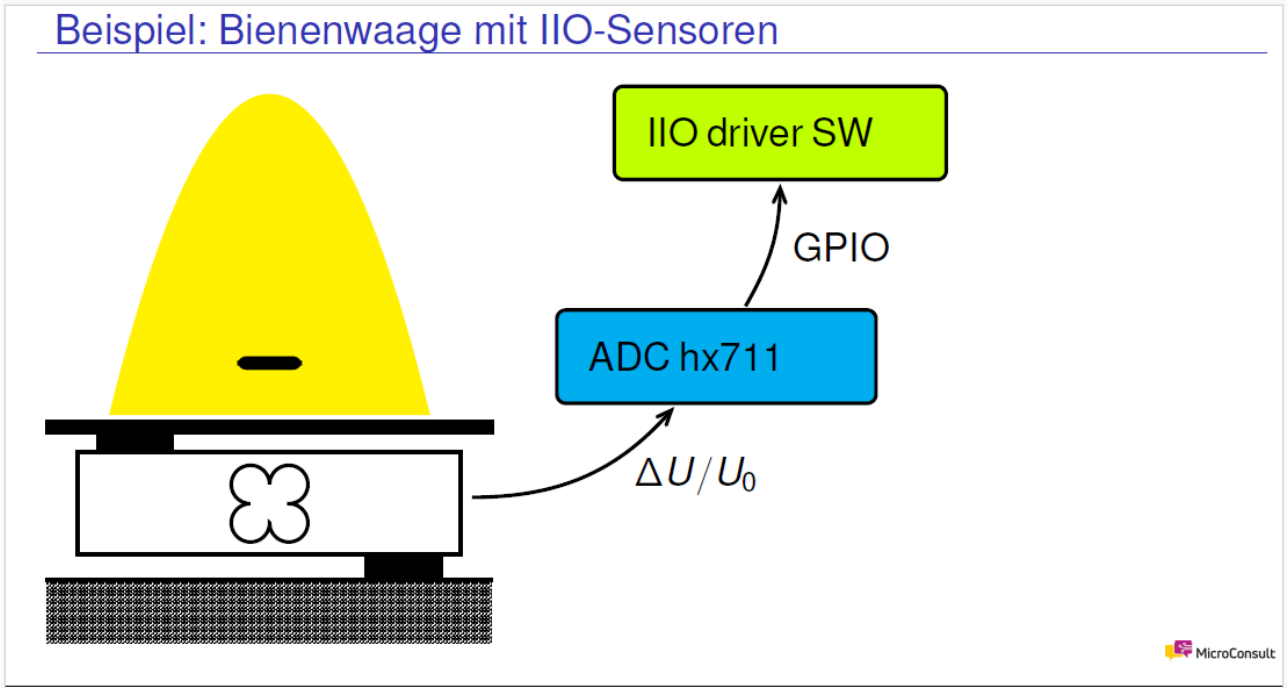


Bild 1: Sensor HX711 (Bild: Andreas Klinger)

2. **Bosch BME280 – ein Umweltsensor für Temperatur, Feuchte und Luftdruck**

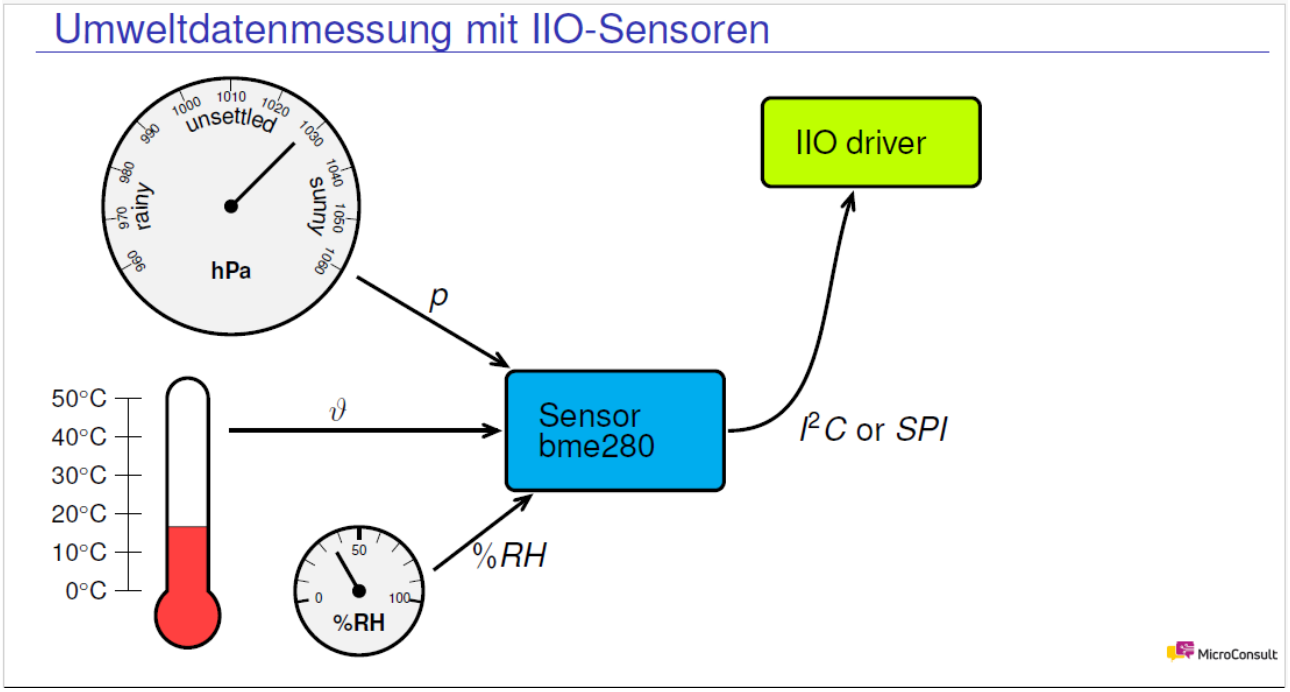


Bild 2: Sensor BME280 (Bild: Andreas Klinger)

Beide Sensoren sind vollständig über Industrial-IO angebunden – der HX711 dank eines Treibers, den Andreas selbst zum Mainline-Linux beigesteuert hat.

©MicroConsult 02-26

Die drei Betriebsmodi von Industrial-IO

1. **Direct Mode – Einfach abfragen**

Im Direct Mode fragt der Userspace den Wert ab, Linux liest den Sensor *in diesem Moment* aus:

- über SysFS-Dateien wie `in_voltage0_raw`
  mit zugehörigem Skalierungsfaktor (`in_voltage_scale`)
- ideal für sporadische Messungen

Ein Beispiel:

Das Gewicht eines Bienenstocks wird über die Rohwerte des ADC und den Scale-Faktor in Millivolt umgerechnet – Standardisierung ist einer der großen Vorteile von IIO.

2. **Buffered Mode – Präzise und kontinuierliche Erfassung**

Im Buffered Mode übernimmt ein **HR-Timer** die Triggerung des Sensors. Daten werden im Kernel in einem Ringpuffer gesammelt, mit präzisen Zeitstempeln ausgestattet und effizient an den Userspace durchgereicht.

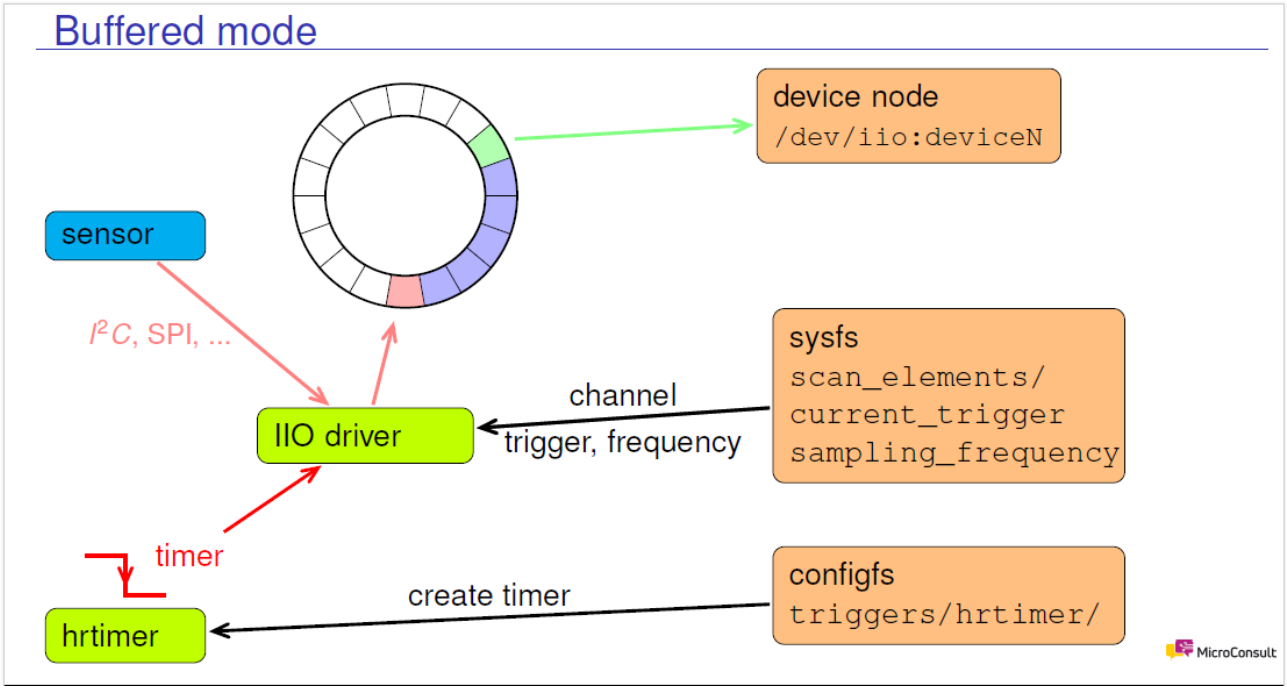


Bild 3: Buffered Mode (Bild: Andreas Klinger)

Dieser Modus ist perfekt für schnelle, regelmäßige Messungen, latenzkritische Anwendungen und große Datenmengen.

Andreas demonstrierte im Webinar live, wie per iio_generic_buffer mehrere Kanäle binär ausgelesen werden – inklusive Nanosekunden-Timestamps.

©MicroConsult 02-26

3. **Event-Triggered Mode – Reagieren statt Abfragen**

Der Benutzerprozess schläft, bis ein Sensorinterrupt ausgelöst wird – z. B. bei Schwellwertüberschreitungen. Der Modus eignet sich, wenn ein Sensor „von selbst" relevante Ereignisse meldet, energiesparendes Arbeiten nötig ist oder interruptfähige Sensorik vorhanden ist.

Da weder HX711 noch BME280 solche Interrupts liefern, wurde dieser Modus im Webinar nicht praktisch demonstriert.

Warum Industrial-IO für Entwickler so wertvoll ist

Andreas brachte es im Webinar auf den Punkt: **IIO sorgt für ein einheitliches Interface und abstrahiert die Hardware, ohne Flexibilität zu verlieren.**

Das bedeutet:

- Treiber implementieren alle Hardwaredetails direkt im Kernel (inkl. DMA, Interrupts usw.).
- Der Userspace hat überall dieselbe Zugriffslogik.
- Sensoren sind einfach austauschbar.
- Die Performance ist hoch und reproduzierbar.
- Der Entwicklungsaufwand für eigene Applikationen sinkt deutlich.

Auch die Zahl an verfügbaren Treibern zeigt, wie etabliert das Subsystem inzwischen ist.

Für alle, die tiefer einsteigen möchten

Im Webinar teilte Andreas hilfreiche Quellen:

- **Kernel-Konfiguration unter Device Drivers → IIO** – vollständige Liste aller Sensoren
- **Kernel-Dokumentation unter Documentation/ABI** – genaue Beschreibung des SysFS-Interfaces
- **Kernel-Quellcode** – Grep nach Sensortyp findet meist sofort die passende Treiberdatei
- **IIO-Mailingliste** – für Entwickler:innen oder komplexere Fragen

Und für Neugierige: Die Live-Daten seiner Bienenstöcke stellt Andreas öffentlich bereit.

Fazit: Industrial-IO ist der Schlüssel zu moderner Sensorik

Das Webinar zeigte eindrucksvoll, wie effizient, flexibel und praxisorientiert Industrial-IO Sensoranbindung unter Embedded Linux macht.

Mit einem einheitlichen Interface, klaren Standards und der Fähigkeit, auch komplexe Messaufgaben performant umzusetzen, gehört IIO zu den wichtigsten Werkzeugen für Embedded-Entwickler.

Und dank praxisnaher Beispiele wie der smarten Bienenstock-Waage wird deutlich: Embedded Linux kann viel – man muss nur wissen, wie man es nutzt.

©MicroConsult 02-26

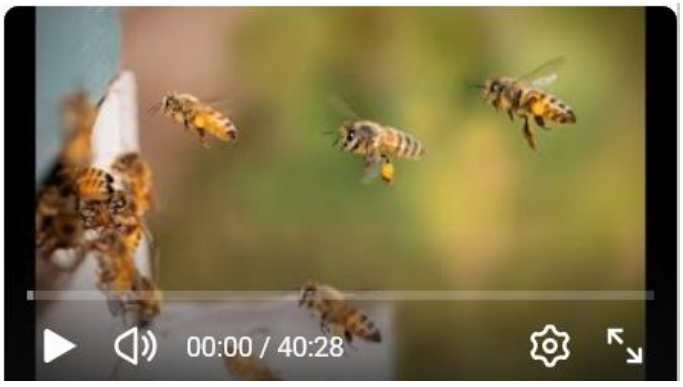
Weiterführende Trainings bei MicroConsult

Wer professionell mit Embedded-Linux, Kernel-Treibern oder Yocto arbeitet, findet bei MicroConsult das passende Programm:

- Embedded-Linux-Softwareentwicklung
- Embedded-Echtzeit-Linux vom Bootloader bis zum Realtime-System mit Yocto
- Embedded-Linux-Architektur: Kernel-Treiberentwicklung
- Embedded-Linux für Test und Support
- Yocto: Grundlagen und Anwendung

Alle Trainings bauen aufeinander auf – ideal für Entwicklerinnen und Entwickler, die ihr Linux-Knowhow systematisch auf- und ausbauen möchten.

Neugierig geworden? Im Webinar zeigt Andreas Klinger Schritt für Schritt, wie Industrial I/O in der Praxis funktioniert.



Weiterführende Informationen

MicroConsult Training & Coaching zu Embedded- und Echtzeitbetriebssystemen
MicroConsult Fachwissen Embedded- und Echtzeit-Softwareentwicklung

Der MicroConsult-Newsletter



Wir informieren Sie mehrmals jährlich über Trends und Best Practices im Embedded Systems Engineering.

Erhalten Sie wertvolles Fachwissen und Tipps aus erster Hand von unseren Embedded-Experten!

Jetzt abonnieren!

©MicroConsult 02-26