

Whitepaper: ein typisches IoT-Thing

Vom Cockpit via Cloud aufs Tablet

Batteriebetriebene Transponder verbinden Rennwagen mit dem IoT und liefern Livedaten

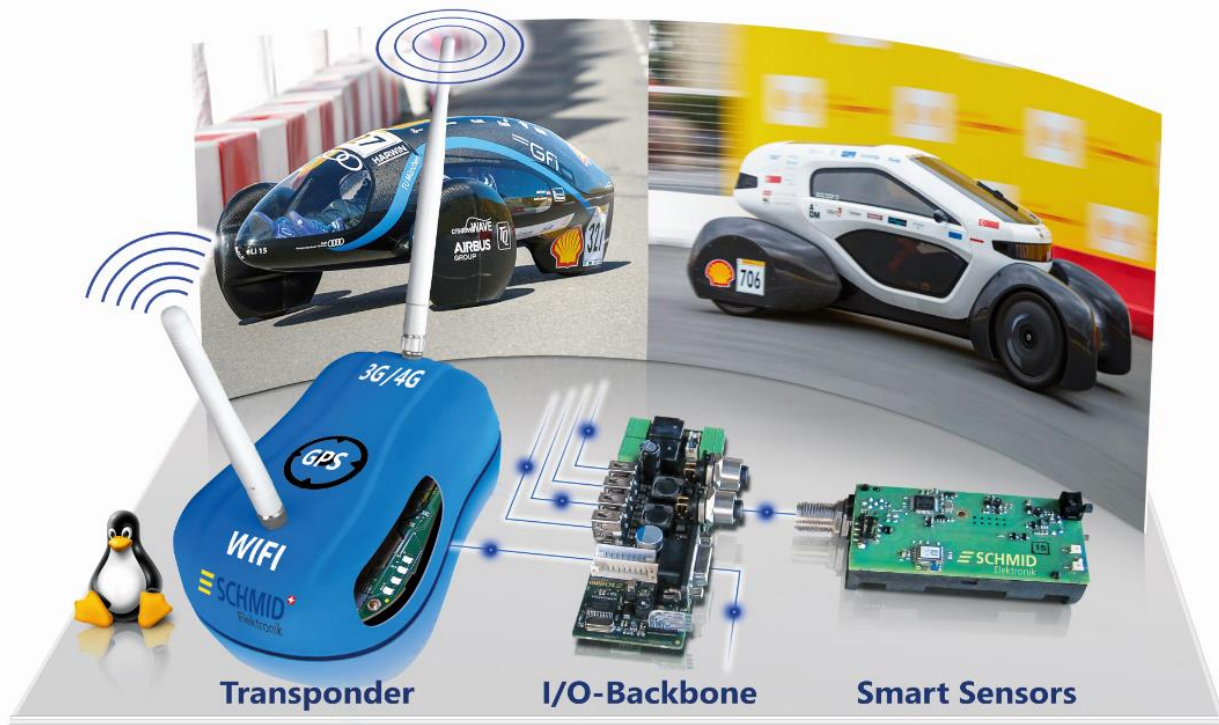


Bild 1: Ein Telemetriesystem besteht aus einem Transponder im 3D-gedruckten Gehäuse und einem I/O-Netzwerk mit smarten Sensoren und Aktoren, die im Rennfahrzeug verteilt sind.

Stichworte // Connected Cars, smarte Incar-Sensorik, Aktoren, live Daten, Telemetrie, IoT-Framework, Cloud-Computing, MQTT, JSON, GPS, 3G/4G-Kommunikation, WIFI, Messnetzwerk, Datenanalyse, Microservice Architektur, Agile Softwareentwicklung, internat.Teamwork, 3D-Design, 3D-Printing.

Live vom Sensor in die Cloud: ein batteriebetriebenes Telemetriesystem sammelt Sensordaten, kombiniert diese mit GPS-Koordinaten, sendet diese Information live über 3G/4G-Mobilfunk an eine IoT-Plattform und erhält von da Aktorbefehle. Dieser Praxisbericht gibt Einblick in einen Regelkreis; von der physikalischen bis in die informationstechnische Ebene und von da auf das Smartphone oder Tablet. Und das im Rahmen eines Öko-Rennens, welches umweltbewusstes Denken so richtig cool macht und unsere Ingenieure der Zukunft fördern soll. Die in diesem Projekt entwickelte Transponderlösung wird von Schmid Elektronik gegenwärtig industrialisiert. Sie soll für jeden, der ähnliche Aufgabenstellungen und Herausforderungen hat, als fertiges COTS-Modul verfügbar sein.

Im Shell Eco Marathon (SEM, siehe letzten Abschnitt dieses Whitepapers) sind die konkreten „Dinge“ 200 Rennfahrzeuge, die über das Internet der Dinge (IoT) miteinander verbunden sind (Bild 2). Sie messen und beurteilen Zustände und kontrollieren Situationen. Ähnlich wie wir Menschen versuchen sie, wenn immer möglich, online zu gehen, entweder über WIFI oder 3G/4G. Dazu nutzen sie eine Art „industrielles Händi“, bestehend aus einem Transponder und einem I/O-Netzwerk mit Backbone und smarten Sensoren und Aktoren (Bild 3).

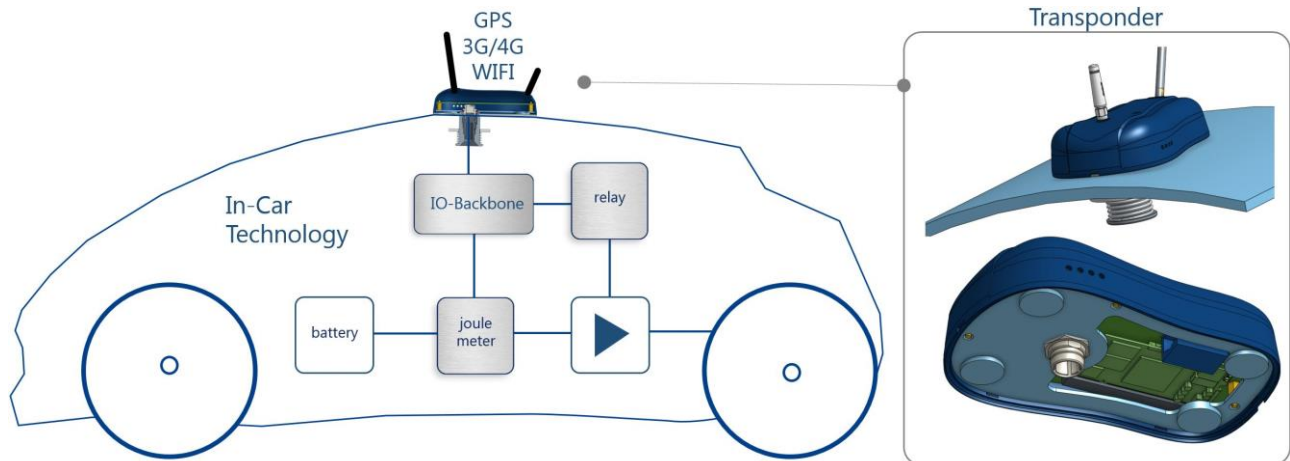


Bild 2: Ein typisches Setup eines batteriebetriebenen Rennfahrzeuges mit aufgestecktem Transponder, verbunden mit dem I/O-Backbone, der im Fahrzeug montiert ist.

LabVIEW auf eigener Hardware

Wie schnell sich das Telemetriesystem in nur schon einem einzigen Jahr entwickelt hat, lässt sich am veränderten Aussehen der Hardware erkennen. Anfang März 2016 Jahr ging noch ein Funktionsmuster in der Grösse einer kompakten Schuhschachtel ans Rennen. Drei Monate später schrumpfte das Volumen um 60% auf einen handlichen Komplett-IPC im Standard-Alu-Gehäuse, und das trotz grösserem Funktionsumfang. Speziell die Fahrzeuge der Prototypenklasse (Siehe letzten Abschnitt in diesem Whitepaper) verlangten jedoch nach weiterer Miniaturisierung um 50%. Zusammen mit dem Wunsch nach weiteren Funktionen wie Kamera und IMU (Inertiale Messeinheit) führte das zur Designentscheidung, den Prozessrechner und das I/O räumlich voneinander zu trennen und mit einem robusten Kommunikationsbus zu verknüpfen. So entstand das heutige, in 3D gedruckte Transpondergehäuse in der Form einer Maus, welches etwa ein i.MX 6 ARM-Multicoremodul, Baseboard, 3G/4G-Modem, GPS-Modul und WIFI auf der Grösse einer Handfläche beherbergt (Bild 3 oben). Mit einem ausgeklügelten Mechanismus aus Kraft- und Formschluss wird die Transpondermaus an der Fahrzeugkarosserie fixiert. Über den Kommunikationsbus, bestehend aus I2C, CAN, USB und RS485 verbindet sie sich mit dem I/O-Backbone (Bild 3 unten). Dieser wird im Fahrzeuginnern montiert und dient als Verteiler für die verwendeten intelligenten Sensoren.

Gegenwärtig wird die Möglichkeit untersucht, die bestehende i.MX6-Lösung durch die Technologie «NI LabVIEW mit Linux auf ARM und FPGA» zu ersetzen, um grafische Programmierung mit LabVIEW zu ermöglichen. Damit liesse sich LabVIEW auf eigener Hardware nutzen und Anwender könnten eigene Embedded-Echtzeitanwendungen direkt auf den Transponder laden.

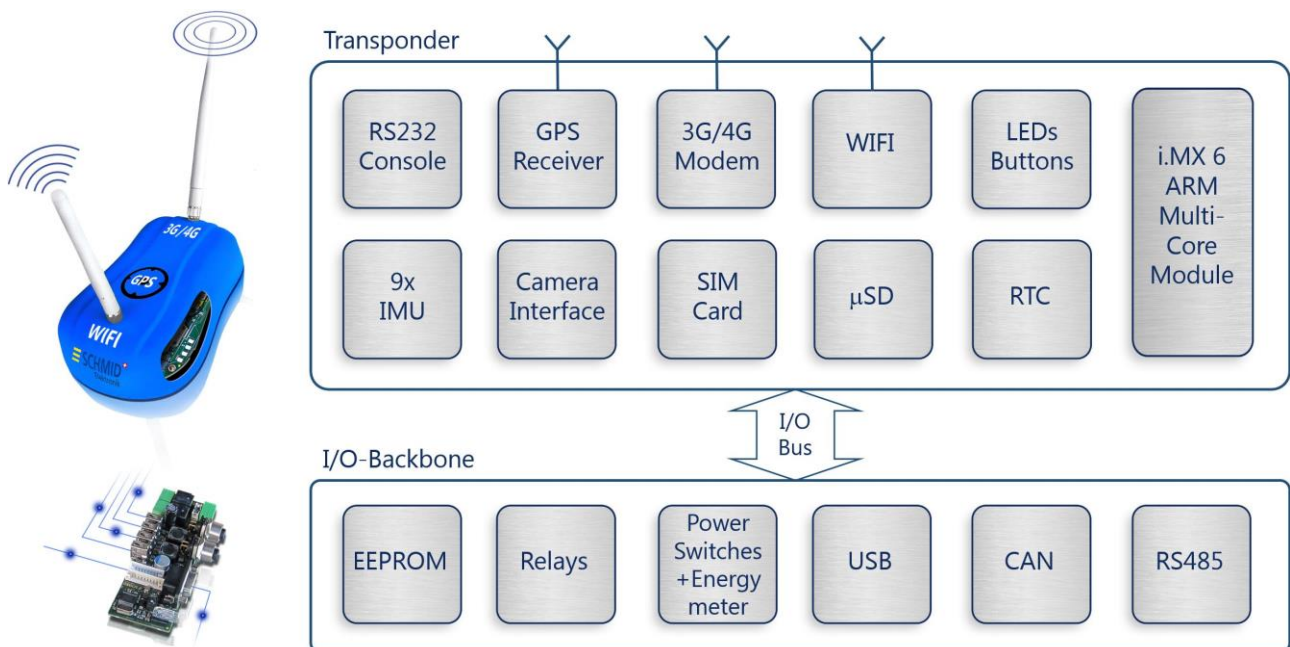


Bild 3: der I/O-Backbone (unten) fusioniert im Fahrzeug Sensoren und Aktoren und speichert fahrzeugspezifische Parameter ab, wie zB Teamnummer und IP-Adresse. Er kommuniziert mit dem Transponder (oben) über I2C, CAN, USB und RS485.

Intelligentes Sensor-/Aktor-Netzwerk

Je nach Energietyp – Verbrennungsmotor, Elektromobilität oder Brennstoffzelle- wird die verbrauchte Energie über einen Durchfluss-, Strom- oder Gassensor gemessen. Alle am I/O-Backbone angeschlossenen Sensoren sind intelligent, digitalisieren die Messwerte vor Ort und übermitteln die Messwerte an die Transpondermaus. Sogar käufliche Standardsensoren mit analogen und digitalen Ausgängen wurden um ein Mixed-Signal-Mikrocontrollerboard in der Grösse eines 2-Euro-Geldstückes erweitert. Der instabile USB aus der früheren Version wurde komplett durch den robusten CAN-Bus ersetzt. Als Aktor dient ein Solidstaterelais mit NO- (Normally Open) und NC- (Normally Closed) Anschlüssen. Es wird entweder in Serie mit dem vorgeschriebenen Notaus-Schalter oder parallel zur Abschaltlogik direkt im Antrieb verdrahtet.

Microservices ermöglichen internationales Teamwork

Die Embedded-Anwendungssoftware greift über sogenannte Microservices auf die Messwerte zu (Bild 4). Unter dieser Architektur kann man sich einen einzelnen Prozess (Linux-Daemon) pro Sensor vorstellen, der über Lowlevel-Treiber auf die Hardware zugreift, Betriebssystemfunktionen nutzt und nach aussen ein TCP-Socket-Interface implementiert. Jeder dieser Prozesse liest beim Starten das Konfigurationsfile des Sensors, erfasst dann die Messwerte, speichert sie zusammen mit einem Zeitstempel in einem CSV-Logfile ab und sendet sie über einen spezifischen TCP-Port raus. Der grosse Vorteil dieser Architektur ist der hohe Abstraktionsgrad und das flexible Nutzen dieser Sensordienste. Ist der Transponder nämlich mit dem Internet verbunden, kann man aus einer anderen Firma, aus dem Ausland und sogar von einem fernen Kontinent darauf zugreifen. In einer ersten Projektphase simulierte das Schweizer Team die Sensordienste mit einen LabVIEW-Prototyp (Bild 5), während die „richtigen“ Dienste in C auf Linux entwickelt wurden. Das Team aus Indien kümmerte sich um die Embedded-Java-Applikation, die auf die Sensordienste zugreift. Die Cloudspezialisten befanden sich in den Niederlanden und das IoT-Team arbeitete in Frankreich. Alle arbeiteten parallel auf dem in der Schweiz bei Schmid Elektronik online gestellten Zielsystem!

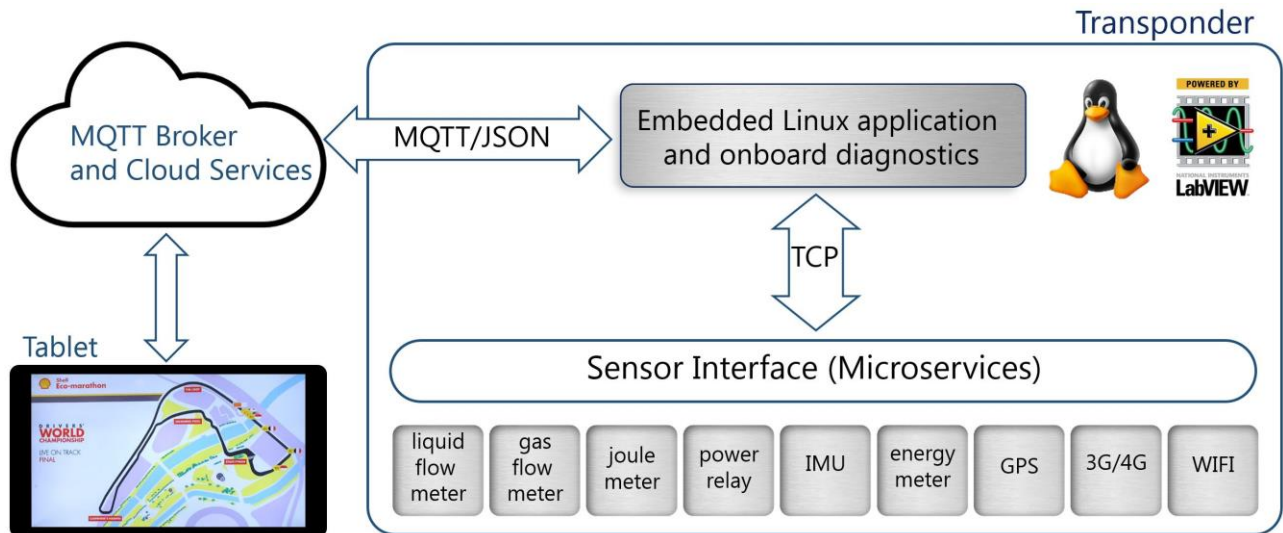


Bild 4: Microservices kapseln die Hardware inklusive Low-Level-Treiber und kommunizieren mit der Hauptanwendung über TCP-Sockets. Von da gelangen die Daten über MQTT an die IoT-Plattform und via Cloudservices aufs Smartphone oder Tablet.

„Anhängliche“ Anwendungssoftware

Die Embedded-Anwendungssoftware besteht im Groben aus drei Teilen: IOT-Link, Incar- I/O und Onboard-Diagnose. Sie läuft als Top-Level-Prozess und ist für folgende sechs Aufgaben verantwortlich:

1. Über Microservices die Sensoren einlesen und Aktoren steuern (Bild 4)
2. Übertragen der Sensordaten an die IoT-Plattform
3. Entgegennehmen und Ausführen von Kommandos aus dem IoT
4. Onboard Diagnose und Übertragen des Gesundheitszustandes an die IoT-Plattform
5. Abschalten des Fahrzeuges beim Überschreiten des zulässigen Energielevels
6. Ändern der Konfiguration oder Updaten der Applikationssoftware

Die Kommunikation zur IoT-Plattform und zurück basiert auf dem MQTT-Protokoll (Publish-Subscribe Messaging) im Takt von 1Hz. Die Daten selber sind als strukturierte JSON-Objekte formatiert. Nach dem Aufstarten liest die Anwendung verschiedene Konfigurationsfiles ein und wechselt von aussen gesteuert zwischen drei Zuständen:

1. **Sleep:** die Applikation ist in einem von der RTC kontrollierten Schlafmodus, führt Diagnosen durch und wartet auf IoT-Kommandos, z.B. „Wach auf!“ (2).
2. **Awake:** in diesem Low-Power-Wachzustand werden nur GPS Koordinaten und Diagnosewerte im 5s-Herzschlag ans IoT übermittelt. Jetzt kennt das IoT und damit die Rennleitung den genauen Standort und Zustand der Fahrzeuge.
3. **Active:** kurz vor dem Rennen wird nun auch das I/O aktiviert und Renn- und Diagnosedaten im 1Hz Takt live gesendet. Nach dem Rennen und Download der Logdateien über WIFI wird das System wieder schlafen gelegt (1).

Das Schema der Onboard-Diagnostik besteht aus drei Ebenen. Die Erste arbeitet statisch, sorgt sich um das Housekeeping der Sensoren und loggt Fehlerzustände. Auf der Zweiten finden Plausibilitätsprüfungen der dynamischen Signalverläufe statt, z.B. Filtern von Spikes. Die Dritte ist schliesslich für vorausschauende Analysen und Trends zuständig.

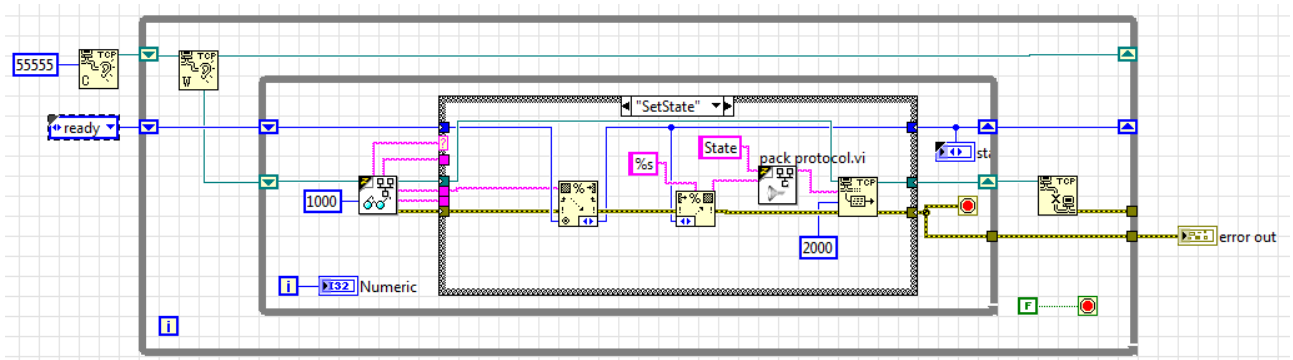


Bild 5: LabVIEW-Prototypen ermöglichen firmen- und länderübergreifende Teamarbeit über Microservices und Embedded-Applikation auf demselben, online gestellten Zielsystem.

Mehr Werkzeuge, bitte !

Eine Komplexität dieser Liga verlangt nach genügend Werkzeugen, die oft unter grossem Zeitdruck entwickelt werden müssen (Bild 6). Mit einer Reihe in LabVIEW entwickelte Tools lassen sich die Transponder etwa testen, konfigurieren, aufspüren, überwachen, die Renndaten extrahieren und sie auswerten. Via WLAN und TCP-Sockets verbinden sie sich über den Transponder direkt mit den einzelnen Microservices (Bild 4) und können so direkt auf Sensoren, Aktoren und Dateien zugreifen. Dank der hohen Abstraktion der grafischen Programmierumgebung LabVIEW liessen sich diese Tools schnell entwickeln und sogar vor Ort und während der Rennen anpassen und verbessern.

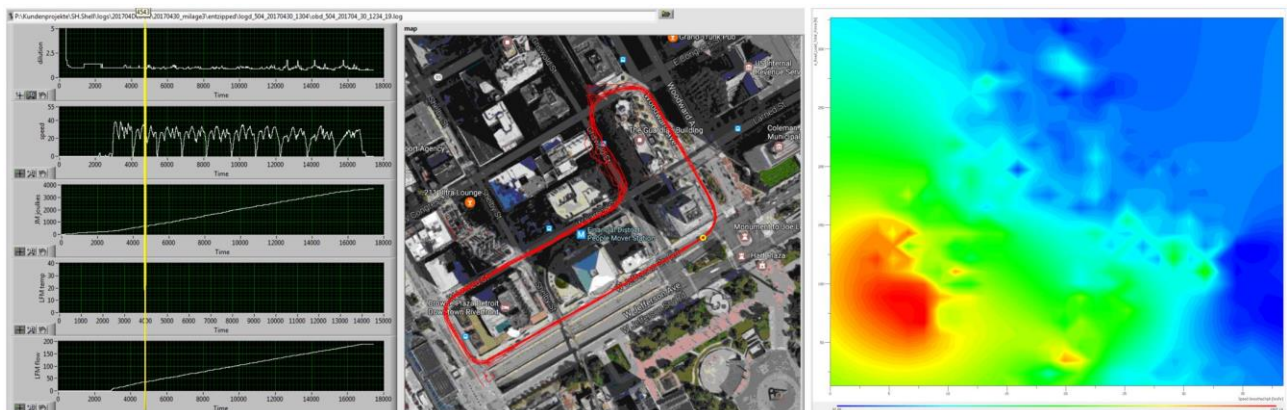


Bild 6: Verschiedene LabVIEW Tools ermöglichen die Konfiguration und Test der Rennfahrzeuge. Datenanalysetools ermöglichen etwa die Überprüfung der Tages-Renndaten (links). Den Teams wird am Ende des Rennens der spezifische Kraftstoffverbrauch (rechts) abgegeben, mit dem Ziel, an der Fahrzeugeffizienz und Rennstrategie zu feilen. Bestehende Rekorde sollen gebrochen werden!

Technology as a Service (TAAS)

Der Shell Eco Marathon (siehe nächster Abschnitt) ist eine internationale, multikulturelle Community mit einem ganz speziellen Spirit. Mit Leidenschaft, Kompetenz und Teamwork verfolgen alle gemeinsam das noble Ziel, machbare Lösungen zum schonenden Umgang mit unseren Energieressourcen zu finden. Hochmotivierte Rennteams aus aller Welt rüsten ihre Fahrzeuge mit Technologie vom Feinsten aus und sind der Dreh- und Angelpunkt des Marathons. Vor Ort werden sie von einem Team aus technischen Spezialisten betreut, dem seit 2016 auch Schmid Elektronik angehört. Schmid liefert Shell das Telemetriesystem als komfortablen „Technology as a Service (TAAS)“, von der Logistik über das Deployment und Support bis zur Überwachung und Datenextraktion. Am Ende des Tages erhält Shell die Renndaten (Bild 6), welche analysiert und im Sinne kontinuierlicher Verbesserung an die Teams abgegeben werden.

Der «SEM»: Von London nach Rom und zurück mit nur 1 Liter Benzin!

Seit 30 Jahren gewinnt beim Shell Eco Marathon (www.shell.com/energy-and-innovation/shell-ecomarathon) derjenige, der mit möglichst wenig Energie die längste Strecke zurücklegt. Dabei treten zwei Fahrzeugklassen gegeneinander an. Die praxisorientierte Urban-Concept-Klasse (Bild 1 rechts) unterstützt Ideen für Ökofahrzeuge für den Einsatz in Städten. Die durchgestylte Prototypenklasse (Bild 1 links) in Form überdimensionierter Zigarren zielt auf maximale Energieeffizienz hin. Beide werden je nach verwendetem Kraftstoff wiederum in drei Kategorien unterteilt. Zu den Verbrennungsmotortypen gehören Superbenzin, Diesel, synthetischer Kraftstoff, Biodiesel und Ethanol. Der Kategorie Elektromobilität gehören Photovoltaik und Akkumulatoren an. Und last but not least gibt's die Fahrzeuge mit Brennstoffzellen (Wasserstoff). Damit die verschiedenen Typen vergleichbar werden, wird die Energiemenge auf Superbenzin zurückgerechnet. Im Rahmen des Shell-Eco-Marathons liegt der aktuelle Rekord bei 3'771 km pro Liter Superbenzin, also von London nach Rom und zurück. Die TU München hingegen schaffte im letzten Jahr in einem eigens durchgeführten Testlauf sogar 11'000km mit nur einem Liter Benzin, was etwa der Distanz von Gibraltar nach Kapstadt entspricht. Dem etwas eintönigen Marathon fehlten aber noch die in Rennen übliche Aufregung und der Nervenkitzel. Also hat Shell 2016 eine neue Komponente eingeführt, die Drivers World Championship (DWC). Hier geht es zu und her wie bei der Formel 1. Die Besten aus dem Marathon, unabhängig von der Energiekategorie, treten am letzten Renntag direkt gegeneinander an. Jedem Fahrzeug wird eine bestimmte Energiemenge zugeteilt und ist diese verbraucht, stoppt ein Relais das Fahrzeug. Die Strategie im sehr engen Kopf-an-Kopf- Rennen besteht nun darin, einerseits Gas zu geben, andererseits aber den Energieverbrauch genau im Auge zu behalten, um so mit dem allerletzten Tropfen über die Ziellinie zu fahren. In den Fahrzeugen eingebaute Transponder übermitteln die Restenergie zusammen mit GPS-Koordinaten an eine IoT-Plattform und die von Cloudservices berechneten Ergebnisse finden ihren Weg über ein live Leaderboard und Karte zum mitfiebernden Publikum.

Literatur

- [1] Shell Eco-Marathon on the Shell website: www.shell.com/energy-and-innovation/shell-ecomarathon
- [2] Shell Eco-Marathon on WIKIPEDIA : https://de.wikipedia.org/wiki/Shell_Eco-Marathon