Vehicle Charging Control Unit

ECU Development Experiences
Introduction

Interaction with CCS Type-2 Inlet
Power-Line Communication
Backend Communication
Summary
System Environment

- ECU is part of charging system for electric city-bus
- DC-charging via CCS2 standard
- Support of DIN70121 and ISO15118 (EIM / PNC)

ECU responsibilities:

- Interaction with CCS2 inlet
- Coordination of charging-related vehicle functions
- Coordination of HV-switches between inlet and DC-link
- Vehicle state-management according to charging schedule (wake-up / sleep)
- Communication with customer backend
Interaction with CCS Type-2 Inlet

Power-Line Communication
Backend Communication
Summary
Interaction with CCS Type-2 Inlet

**Sensors / Actuators**

**Input Signals:**
- Proximity Pin (PP)
- Control Pilot Signal according to IEC61851 (CP)
- Coupler Lock-State
- Temperature Sensors (AC, DC+, DC-)
- Push-Button (User-Interaction)

**Output Signals:**
- Coupler Locking Mechanism
- LEDs (User-Interaction)
Interaction with CCS Type-2 Inlet

Proximity Pin

- Evaluation of PP signal and electrical diagnostic based on ADC signal
- Voltage segments are influenced by three resistors:
  - Pullup-resistor in ECU (R4)
  - Resistor in inlet (R5)
  - Resistor in coupler (RC)

- Development according to IEC61851:2010 (Annex A – B.5)
- Resistor in inlet specified to 4k7 instead 2k7
- ECU pullup-resistor has to be adapted to optimize voltage segments (no overlapping)

<table>
<thead>
<tr>
<th>Short to GND</th>
<th>Plug present 100 Ohm</th>
<th>Plug present 220 Ohm</th>
<th>Plug present 680 Ohm</th>
<th>Plug present 1500 Ohm</th>
<th>Plug not present</th>
<th>Open-Load</th>
<th>Short to battery</th>
</tr>
</thead>
<tbody>
<tr>
<td>0V</td>
<td>0,3V</td>
<td>0,7V</td>
<td>1,6V</td>
<td>2,4V</td>
<td>3,8V</td>
<td>4,7V</td>
<td>...36V</td>
</tr>
</tbody>
</table>
Interaction with CCS Type-2 Inlet

Control Pilot Signal

**PWM signal encodes following information:**
- Duty-cycle: Maximum current (0 to 100%)
- Voltage: Charging state (-12V to +12V)

**Frequency and duty-cycle:**
- Captured via AUTOSAR MCAL module ICU
- Timer unit of microcontroller guarantees high resolution

**Voltage:**
- ADC signal
- Trigger of ADC conversion is challenging (maximum rise time 2us)
Interaction with CCS Type-2 Inlet

Control Pilot Signal – Voltage Measurement

Options for ADC conversion trigger (microcontroller-dependent):

- **Software trigger**
  - Delayed by interrupt locks and jitter

- **Hardware trigger**:
  - **Rising-Edge**
    - Not reliable if edge steepness is not high enough
  - **Streaming**
    - High CPU-load caused by interrupts and processing algorithm

- **Final solution**: Combination of streaming-mode and DMA unit

![Diagram showing 1 kHz / 1ms timing with 50us delay and 5% voltage level](image)
Interaction with CCS Type-2 Inlet

Coupler Lock Mechanism

- Not standardized interface

- 12V DC-motor controlled via ECU
  - Specific parameters may be supplier-specific

- Different solutions for lock-state detection already available:
  - Detection via light barrier or via micro-switches
  - Two states (open / closed) or three states (open / driving / closed)

Risk: Not all inlets provide same interfaces to ECU! Hardware re-design could be required to fit new inlet model
Interaction with CCS Type-2 Inlet

Extension of Product Portfolio – Split of IEC 61851 relevant components

1. I/O Hardware Abstraction:
   - Depends on microcontroller and ECU-hardware
   - Transforms input values into physical/logical value, e.g. CP frequency
   - Controls outputs (e.g. S2 state)

2. Sensor/Actuator components
   - Hardware-independent
   - Calculates overall state, e.g. combination of CP frequency and voltage
   - Provides services, e.g. BCB toggle to wakeup EVSE
   - Can be modelled as AUTOSAR software component

Sensor/Actuator components are generic and could be part of future product versions.
Introduction
Interaction with CCS Type-2 Inlet

- **Power-Line Communication**
  Backend Communication
  Summary

Power-Line Communication
Integration of Qualcomm QCA7005

- Firmware consists of
  - Softloader
  - Firmware
  - Configuration-File (PIB)
- Maintained by Qualcomm
- Provided as binary

- Storage of QCA firmware
  1. Separated flash-chip
     - Additional costs
     - Firmware-update more complex
  2. Host CPU
     - Additional startup-time due to firmware download
     - Firmware-update straightforward
     - On-the-fly parameter update (e.g. MAC address)
QCA7005 – PIB File

- PIB File can be customized
  - Settings for functional behavior, e.g. SLAC as EVSE or EV
  - Parameters (e.g. MAC address)

- Tone map
  - Controls amplitude of all carriers used by Power-Line communication
  - Important parameter to optimize the EMC results
  - Required by ISO15118-3:
    > Calibration of transmission power must be performed at CCS2 inlet
    > Therefore individual for each vehicle platform
QCA7005 – Tone Map

- Carriers > 28Mhz notched
Power-Line Communication

Communication Stack

Charging Application

- RTE
- JSON/EXI
- DNS
- SEC
- SPI

Infra-structure Comm.

- V2GTP
- HTTP
- TLS
- TCP/IPv6
- EthIf
- Eth
- EthTrcv

Product  Project  3rd Party
TLS mandatory for PNC
Set of certificates must be stored in ECU

Installation of OEM provisioning certificate:
- Non standardized interface (UDS diagnostic routine, CAN, ...)
- Confidential data (private key)
- ECU-specific

Installation of V2G root certificate:
- Non standardized interface
- Can be customer specific

If certificate updates required during lifetime, updates must be supported by OEM aftersales process
- Encryption for confidential data
- Real-time tracking of certificates (vehicle ↔ customer/contract)
Introduction
Interaction with CCS Type-2 Inlet
Power-Line Communication

- **Backend Communication**
  
Summary
Backend Communication

Communication to Backend-Server

- **Intention**
  - Data exchange with fleet-operator
  - Coordination of vehicle fleet
  - Optimal preparation of vehicle for next drive

- **Requirements:**
  - Bi-directional data exchange with backend
  - Several data objects
  - Security
    - Authentication
    - Encryption

- **Possible solutions:**
  - Extension of V2G protocol (schema)
  - New message type for V2GTP
  - Proprietary protocol based on TCP/UDP
  - HTTP / HTTPs
  - External ECU (via CAN)
Extension of V2G Protocol

- **Variant A**: Extension of V2G schema
  - Usage of existing V2G communication
  - Adaption of EV- and EVSE-basic-software required
  - Original schema must be still supported

- **Variant B**: New message type in V2GTP
  - Usage of existing V2G communication
  - Message can be simply forwarded to backend
  - Adaption of EV- and EVSE-basic-software required

```
19
```
Backend Communication

Proprietary Protocol

- TCP or UDP used as transport protocol
- No adaption of EV- and EVSE-basic-software required
  - Communication based on proprietary protocol
  - DNS/DHCP for address resolution required
Communication via internet possible
- Well-known protocols / techniques
  - Authentication:
    - TLS Server- and/or Client-Authentication
    - Basic HTTP
  - Encryption via HTTPs
- Internet access requested as Value Added Service (VAS)
- DNS/DHCP for address resolution required
- JSON used as data format:
  - Small overhead
  - Easy to parse in ECU
- Entire communication stack available as MICROSAR components
Introduction
Interaction with CCS Type-2 Inlet
Power-Line Communication
Backend Communication

Summary
Summary

Lessons Learned

Interaction with CCS Type-2 Inlet
- Potential for new product components (Basic-SW)
- CCS2 interfaces could be incompatible to ECU hardware since not standardized

Power-Line Communication
- Entire communication stack available as Basic-SW components
- Storage of firmware on host controller
- PIB file offers many optimization parameters

Backend Communication
- HTTPs fulfills all requirements
- Low project-specific development effort
For more information about Vector and our products please visit www.vector.com

Author: Epping, Michael
Vector Germany