Dropbeats

DB2605 EV Charging Controller

Communication Protocol

Rev 1.0.4, August 2024

CONTENTS

1 OVERVIEW

The DB2605 EV Charging Controller is the core unit of a Supply Equipment Communication Controller (SECC) that manages Combined Charging System (CCS) AC high-level communication charging and basic charging. The DB2605 is equipped with a Qualcomm QCA700X and a powerful MCU that runs RTOS with a complete ISO 15118-2/20 software stack and an optional IEC 61851 stack.

In an EVSE architecture, the SECC, containing a DB2605, is the child in a parent-child relationship with a Charging Control Unit (CCU).

This document specifies the communication protocol between the DB2605 SECC and the CCU.

2 COMMUNICATION PROTOCOL

The DB2605 communicates with the CCU through a UART port.

UART serial port settings:

- Baud rate 115,200 bps
- 8 data bits
- 1 stop bit
- No parity bit

In addition to the serial port connection, Reset and Boot input pins are provided:

The Reset signal from the CCU triggers DB2605 reboot.

If the Boot signal is pulled high when Reset is triggered, the DB2605 reboots and enters the firmware upgrade process

There are three message types between the CCU and the DB2605:

- Charging message

  Charging messages manage the charge session.

- Data transfer message

  Data transfer adopts the Xmodem protocol. Both 128 byte and 1K message lengths are supported. Data transfer is initiated by the CCU, which sends or receives files by setting DataTransferReq bits in keep-alive messages; the DB2605 responds by setting the bit corresponding to DataTransferRes in its keep-alive message. After the handshake ends, data transmission occurs. During data transmission, charging messages between CCU and DB2605 are suspended.

- Firmware upgrade message

  Firmware upgrade is initiated by the CCU. The CCU triggers the DB2605 to enter serial port upgrade mode through Boot and Reset signals.

3 CHARGE MESSAGE

3.1 MESSAGE FRAME FORMAT

Start Bytes (2 bytes)	Length (2 bytes)	Message Contents	Checksum (1 byte)
0xDBAC (fixed)	Message length (Start Bytes to Checksum)	Message Parameters	Message content CRC-8 calculation (start byte to last data byte)

Frame ID (4 bytes)	Frame Parameters (various bytes)
Frame Identification	Payload

Field Descriptions

Start Byte: 2 bytes, fixed value 0xDBAC, used to mark the frame header.

Length: 2 bytes, the length of the message frame. The length starts with the Address byte and ends with the Checksum byte, inclusive.

Message Contents: 4-byte fixed frame ID. The Frame Parameters field depends on the payload to be transmitted.

Checksum: 1 byte. The value obtained by CRC-8 calculation of each byte of the message frame, from the Start byte to the last Data byte. The CRC-8 polynomial is $X^8 + X^2 + X + 1$.

Example:

DB	AC	00	11	18	B0	56	F4	02	24	64	90	00	FF	FF	00	CA

Start Byte		0xDBAC
Length		0x0011
Message	Frame ID	0x18B056F4
	Frame Parameters	0x02 0x24 0x64 0x90 0x00 0xFF 0xFF 0x00
Checksum		0xCA

3.2 MESSAGES

The following messages are enough to complete a simple charging session, outlined at a high level in the next section.

Message	ID	Direction	Description
SECC_Status	0x18B056F4	DB2605 → CCU	Status message, aka keepalive
SECC_SysInfo	0x18B156F4	DB2605 → CCU	Software version, parameters
SECC_EvChgLimits	0x18B556F4	DB2605 → CCU	EV charge limitation
SECC_EvEvccId	0x18BA56F4	DB2605 → CCU	EVCC ID
SECC_EvTargets	0x18BF56F4	DB2605 → CCU	EV targets
CCU_Status	0x18C0F456	DB2605 ← CCU	Status message, aka keepalive
CCU_EvseChgLimits	0x18C3F456	DB2605 ← CCU	EVSE charge limitation

The SECC_Status message from SECC to CCU and the CCU_Status message from CCU to SECC are keep-alive messages. Repeating every 100ms, they report status and do much of the work of managing charging sessions.

For additional messages and detailed information, refer to *DB2605 EV Charging Controller Communication Matrix*

4 CHARGE ELEMENTS

4.1 IEC 61851-1 SOURCE

The SECC_Status message includes the SeccIec61851Source signal, which indicates the source managing the IEC 61851-1 control and measurement processes.

- **Signal:** SeccIec61851Source
- **Message:** SECC_Status
- **Values:**

IEC61851_SOURCE_CCU	**Description:** The Charge Control Unit (CCU) handles the IEC 61851-1 control pilot, proximity pilot generation, and measurement, including circuits and PWM pulse. **Usage**: Pull DB2605 Pin 22 Low
IEC61851_SOURCE_SECC	**Description**: The DB2605 SECC itself manages the IEC 61851-1 control pilot, proximity pilot generation, and measurement, including circuits and PWM pulse. **Usage**: Pull DB2605 Pin 22 High

- **Purpose:** This setup allows the DB2605 EV charging controller to flexibly handle the control and measurement processes either internally (via SECC) or externally (via CCU).

4.2 Occupation

To initiate a charging session while the SECC is in an idle state, the Charge Control Unit (CCU) needs to occupy the SECC. This is done by sending the CcuChgPortOcpd signal with CHARGE_PORT_AC_SINGLE_PHASE_CORE or CHARGE_PORT_AC_THREE_PHASE_CORE_CASE_B.

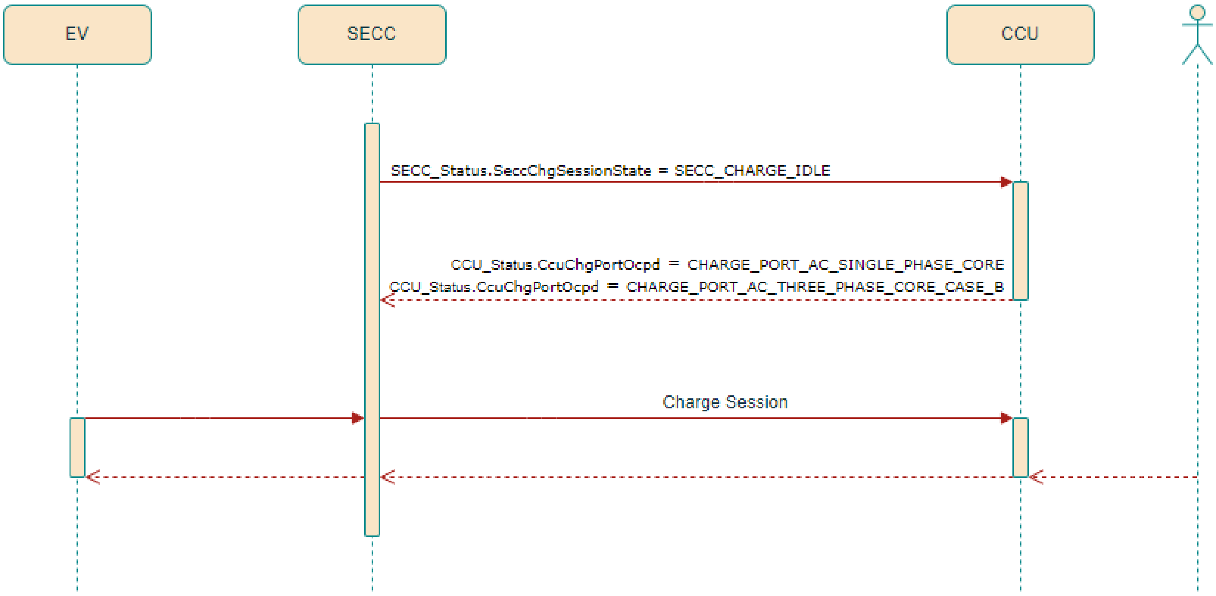
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_CHARGE_IDLE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_AC_SINGLE_PHASE_CORE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_AC_THREE_PHASE_CORE_CASE_B
Charge Session

Figure 1. Occupation

- **Signal:** CcuChgPortOcpd
- **Message:** CCU_Status
- **Values:**

CHARGE_PORT_AC_SINGLE_PHASE_CORE	A type of power distribution system that uses a single alternating current (AC) voltage phase.
CHARGE_PORT_AC_THREE_PHASE_CORE_CASE_B	three-phase AC power.

- **Purpose:** indicates that the CCU intends to occupy the SECC for a charging session.

4.3 RELEASE

To properly conclude a charging session, the CCU (Charge Control Unit) must release the SECC. This is achieved by sending the CcuChgPortOcpd signal with the CHARGE_PORT_UNOCCUPIED value.

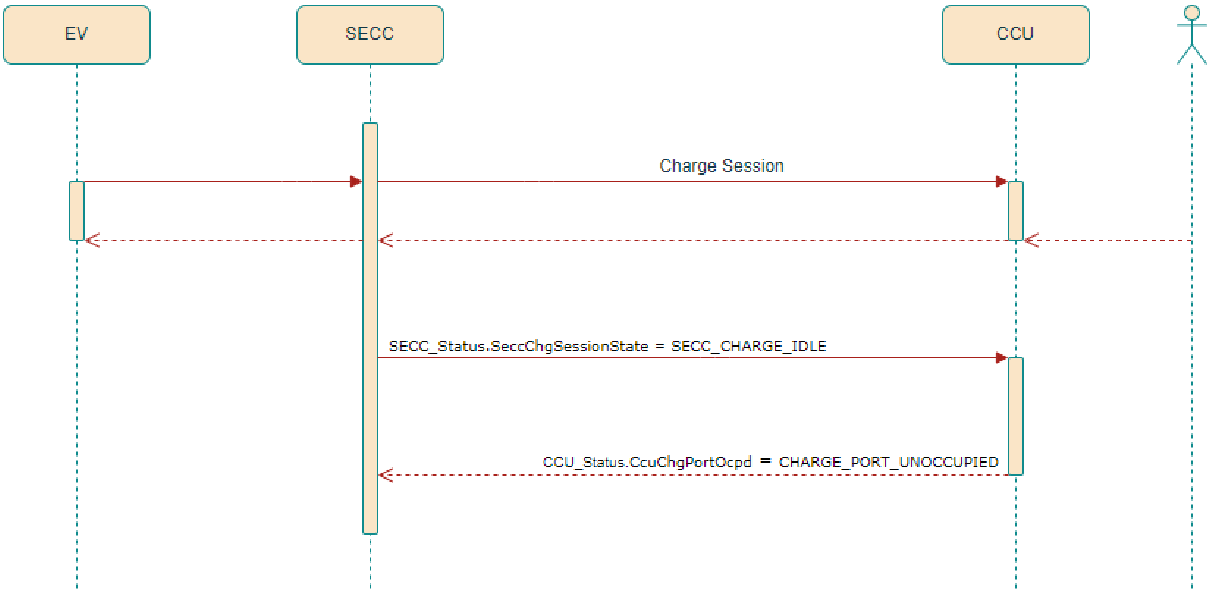
EV
SECC
CCU
Charge Session
SECC_Status.SeccChgSessionState = SECC_CHARGE_IDLE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_UNOCCUPIED

Figure 2. Release

- **Signal:** CcuChgPortOcpd
- **Message:** CCU_Status
- **Value:** CHARGE_PORT_UNOCCUPIED
- **Purpose:** indicates that the charge port is unoccupied and the SECC should release and end the charging session.

This signaling ensures that the SECC appropriately manages the charging session based on the status of the charge port as communicated by the CCU.

4.4 AUTHORIZATION

To start a charging session when a user presents an EIM (RFID), the CCU sends the CcuChgSessionAuth signal within the CCU_Status message. The possible values for this signal that start a charge session are EIM_AUTHORIZED or FREE_SERVICE.

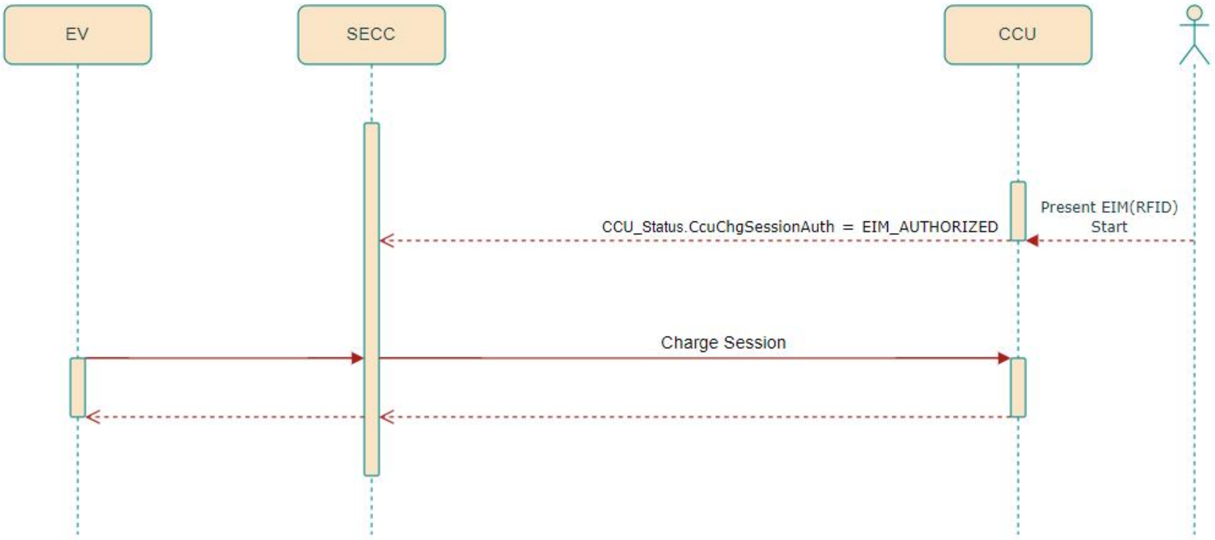

Figure 3. Authorization

- **Signal:** CcuChgSessionAuth
- **Message:** CCU_Status
- **Values:**

EIM_UNAUTHORIZED
Indicates that the user is not authorized to start a charging session.

EIM_AUTHORIZED
Indicates that the user is authorized to start a charging session.

FREE_SERVICE
Indicates that the charging connector is plugged into the vehicle, and the system is ready to start or is in the process of starting a charging session.

- **Purpose:** To start a charging session.

This mechanism ensures that the charging session can be initiated based on user authorization or free service availability.

4.5 CONTRACTORS - CLOSE

EV is ready to accept energy, SECC sends a signal to close the contactors.CCU receives the signal and proceeds to close the contactors and sends a confirmation signal back to the SECC indicating that the contactors is closed and the EV is ready to start charging.

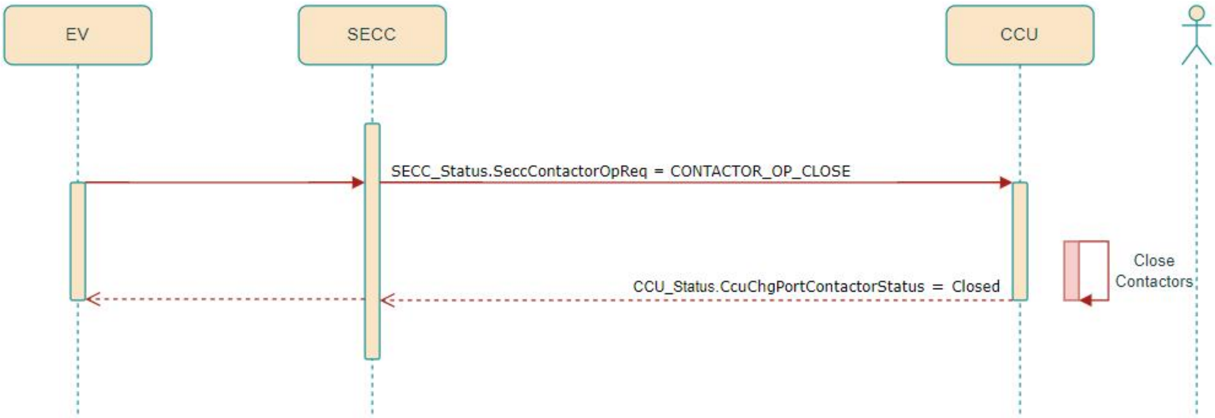
EV
SECC
CCU
SECC_Status.SeccContactorOpReq = CONTACTOR_OP_CLOSE
Close Contactors
CCU_Status.CcuChgPortContactorStatus = Closed

Figure 4. Contractors - Close

- **Signal:** SeccContactorOpReq
- **Message**: SECC_Status
- **Values**: CONTACTOR_OP_CLOSE
- **Purpose**: This signal is sent from the SECC to the CCU to request the closing of the contactors to start the energy transfer.

- **Signal:** CcuChgPortContactorStatus
- **Message:** CCU_Status
- **Value:** Closed
- **Purpose**: This signal is sent from the CCU back to the SECC to confirm that the contactors have been successfully closed and that the energy transfer can proceed.

4.6 CONTRACTORS - OPEN

EV is ready to stop charging, SECC sends a signal to open the contactors. CCU receives the signal and proceeds to open the contactors and sends a confirmation signal back to the SECC indicating that the contactors is opened.

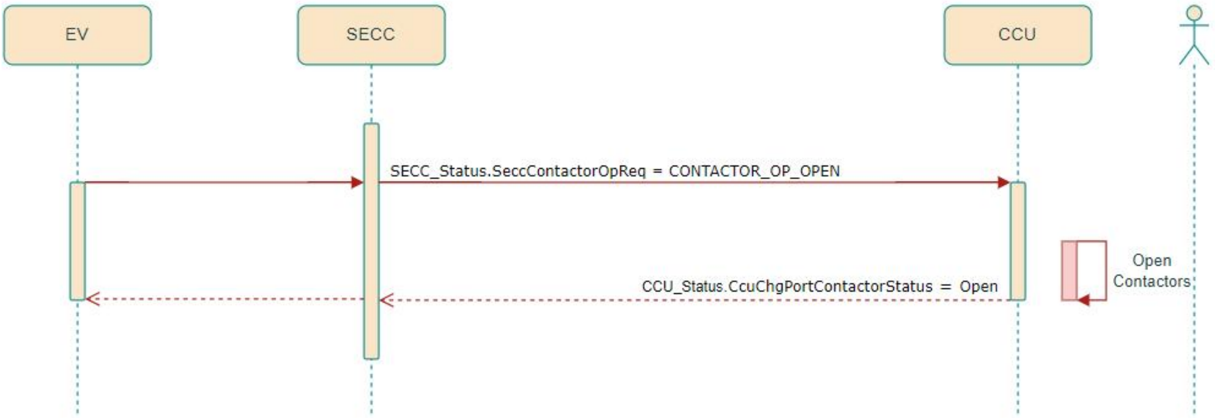
EV
SECC
CCU
SECC_Status.SeccContactorOpReq = CONTACTOR_OP_OPEN
CCU_Status.CcuChgPortContactorStatus = Open
Open Contactors

Figure 5. Contractors - Open

- **Signal:** SeccContactorOpReq
- **Message**: SECC_Status
- **Value:** CONTACTOR_OP_OPEN
- **Purpose**: Requests the CCU to open the contactors.

- **Signal:** CcuChgPortContactorStatus
- **Message:** CCU_Status
- **Value:** Opened
- **Purpose**: Confirms that the contactors are now open.

4.7 SHUTDOWN

When user shutdowns charge session, CCU sends request to stop charge session, After the charge session is end, user plugs out the charge cable.

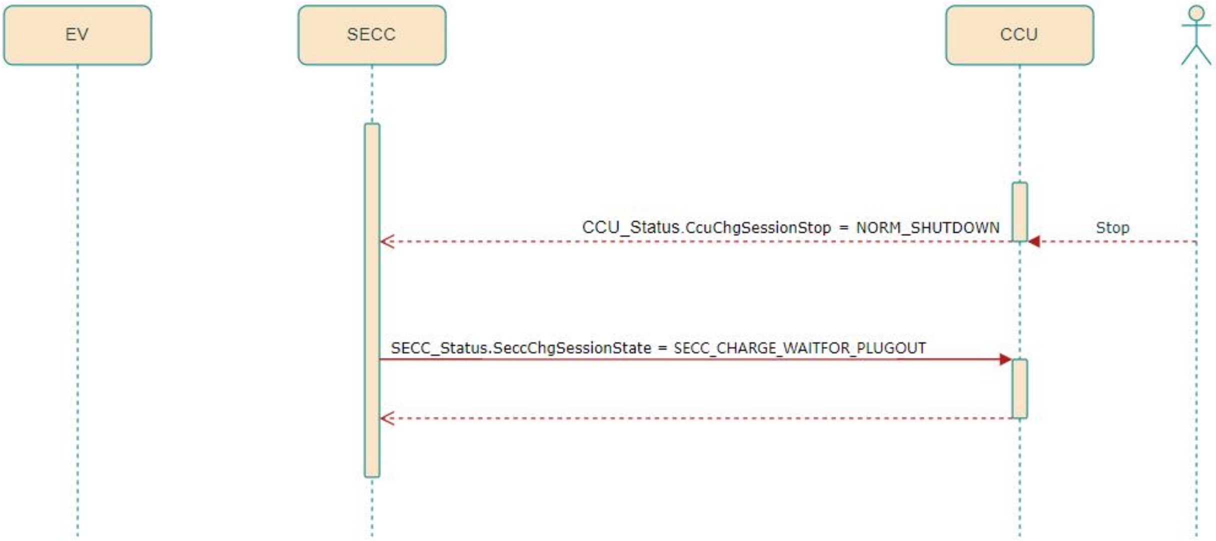
EV
SECC
CCU
CCU_Status.CcuChgSessionStop = NORM_SHUTDOWN
Stop
SECC_Status.SeccChgSessionState = SECC_CHARGE_WAITFOR_PLUGOUT

Figure 6. Shutdown

- **Signal:** CcuChgSessionStop
- **Message**: CCU_Status
- **Value:**

NORM_SHUTDOWN	Indicates normal shutdown.
EMGY_SHTUDWON	Indicates emergency shutdown.
OTHS_SHTUDWON	Indicates other types of shutdown.

- **Purpose**: Requests to stop the charging session with the reason for the shutdown.

4.8 Duty Cycle Setting

In case IEC 68151-1 source is CCU, the SECC is responsible for setting the PWM duty cycle using SeccCpPwmDutyCycleSet while engaged in high-level communication charging.

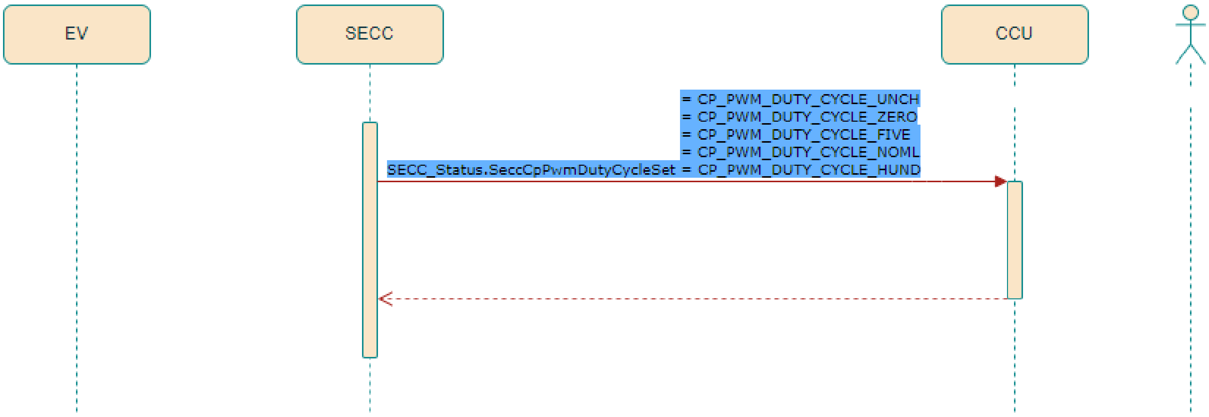

Figure 7. Duty Cycle Setting

- **Signal:** SeccCpPwmDutyCycleSet
- **Message:** SECC_Status
- **Value:**

CP_PWM_DUTY_CYCLE_UNCH	Indicates to keep the current duty cycle.
CP_PWM_DUTY_CYCLE_HUND	100%, the control pilot signal is fully high (no low period).
CP_PWM_DUTY_CYCLE_NOML	Nominal Duty Cycle, the standard duty cycle for regular charging conditions.
CP_PWM_DUTY_CYCLE_FIVE	5%, Indicates that high-level Power Line Communication is necessary.
CP_PWM_DUTY_CYCLE_ZERO	0%, the control pilot signal is fully low (no high period).

- **Purpose**: manages the PWM duty cycle.

4.9 Port Status Report

In case IEC 68151-1 source is CCU, the CCU manages the physical aspects of the control pilot and proximity pilot, and reports the status to the SECC.

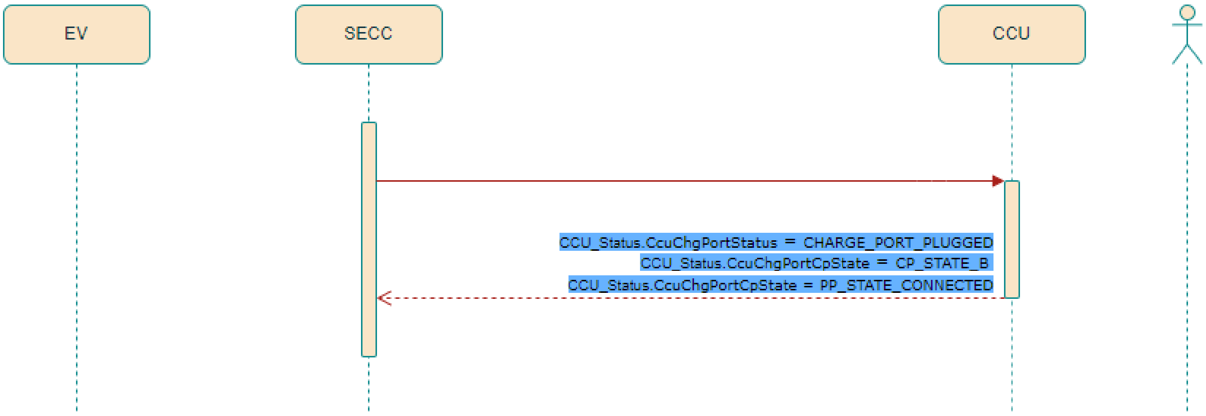
EV
SECC
CCU
CCU_Status.CcuChgPortStatus = CHARGE_PORT_PLUGGED
CCU_Status.CcuChgPortCpState = CP_STATE_B
CCU_Status.CcuChgPortCpState = PP_STATE_CONNECTED

Figure 8. Port Status Report

- **Signal:** CcuChgPortStatus
- **Message:** CCU_Status
- **Value:**

CHARGE_PORT_UNKNOWN Indicates that the system is unable to ascertain the current state of the charging port.

CHARGE_PORT_UNPLUGGED Signifies that the charging connector is not plugged into the vehicle, and no charging session can occur.

CHARGE_PORT_PLUGGED Indicates that the charging connector is plugged into the vehicle, and the system is ready to start or is in the process of starting a charging session.

CHARGE_PORT_DEPRESSED Indicates that the charging connector is plugged into the vehicle, and the system is not ready to charge due to Proximity circuit opening (pressing of S3).

- **Purpose**: Indicates the charge port status.

- **Signal:** CcuChgPortCpState
- **Message:** CCU_Status
- **Value:**

PILOT_ST_UNKNOWN Indicates that the status of the control pilot (CP) line is unknown

CP_STATE_A Indicates that the EV is not connected to the charging station or the connector is not properly seated. The CP voltage is typically at 12 volts.

CP_STATE_B Signifies that the vehicle and the charging station are in communication, but the conditions for charging are not met. The CP voltage is around 9 volts.

CP_STATE_C Indicates that the vehicle is currently actively charging.

The CP voltage is typically at 6 volts.

CP_STATE_D indicates that the EV is currently charging, and the vehicle’s cooling or ventilation system is required. The control pilot (CP) voltage is around 3 volts.

CP_STATE_E Indicates an error condition, the CP voltage is typically at 0 volts.

CP_STATE_F Indicates an fault condition, the CP voltage is typically at -12 volts.

CP_STATE_ERROR Indicates that the status of the control pilot (CP) line is error.

- **Purpose**: Indicates the charge port control pilot state.

- **Signal:** CcuChgPortPpState
- **Message:** CCU_Status
- **Value:**

PP_STATE_UNKNOWN Indicates that the status of the proximity pilot (PP) line cannot be determined.

PP_STATE_DISCONNECTED Indicates that the connector is not connected.

PP_STATE_CONNECTED Indicates the proximity pilot (PP) line is connected.

PP_STATE_DEPRESSED Indicates that the proximity pilot (PP) line is connected, but proximity circuit opening (pressing of S3).

- **Purpose**: Indicates the charge port proximity pilot state.

4.10 BASIC CHARGING OFFLINE

In case IEC 68151-1 source is CCU, CCU takes over a charge session from SECC to switch to a basic charging if an error occurs on the high-level communication, even if a HLC-C mode is already launched.SECC sends signal, SeccChgSessionState with SECC_CHARGE_BC_OFFLINE of message SECC_Status to CCU if an error occurs.CCU takes over charging session, and sends CHARGE_PORT_UNOCCPIED status back in signal, CcuChgPortOcpd of message CCU_Status, EIM_UNAUTHORIZED status in signal, CcuChgSessionAuth to release SECC.

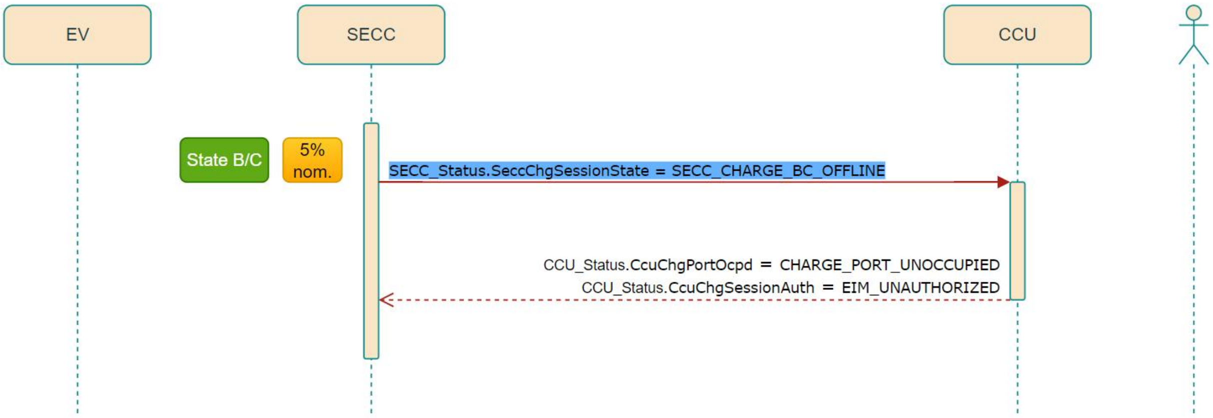
EV
SECC
CCU
State B/C
5% nom.
SECC_Status.SeccChgSessionState = SECC_CHARGE_BC_OFFLINE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_UNOCCUPIED
CCU_Status.CcuChgSessionAuth = EIM_UNAUTHORIZED

Figure 9. Basic Charging Offline

4.11 SUMMARY

In this communication protocol, the reactor pattern is used. CCU only needs to listen to 1 event(SeccContactorOpReq)and react, except of presenting RFID cards, plug-in and plug-out, to complete a charge session. In case IEC 68151-1 source is CCU, CCU listens one more signal(SeccCpPwmDutyCycleSet) and reporting charge port status, control pilot status, and proximity pilot status are needed.

5 CHARGE SEQUENCE

5.1 FIRST AUTHORIZE, THEN PLUG IN

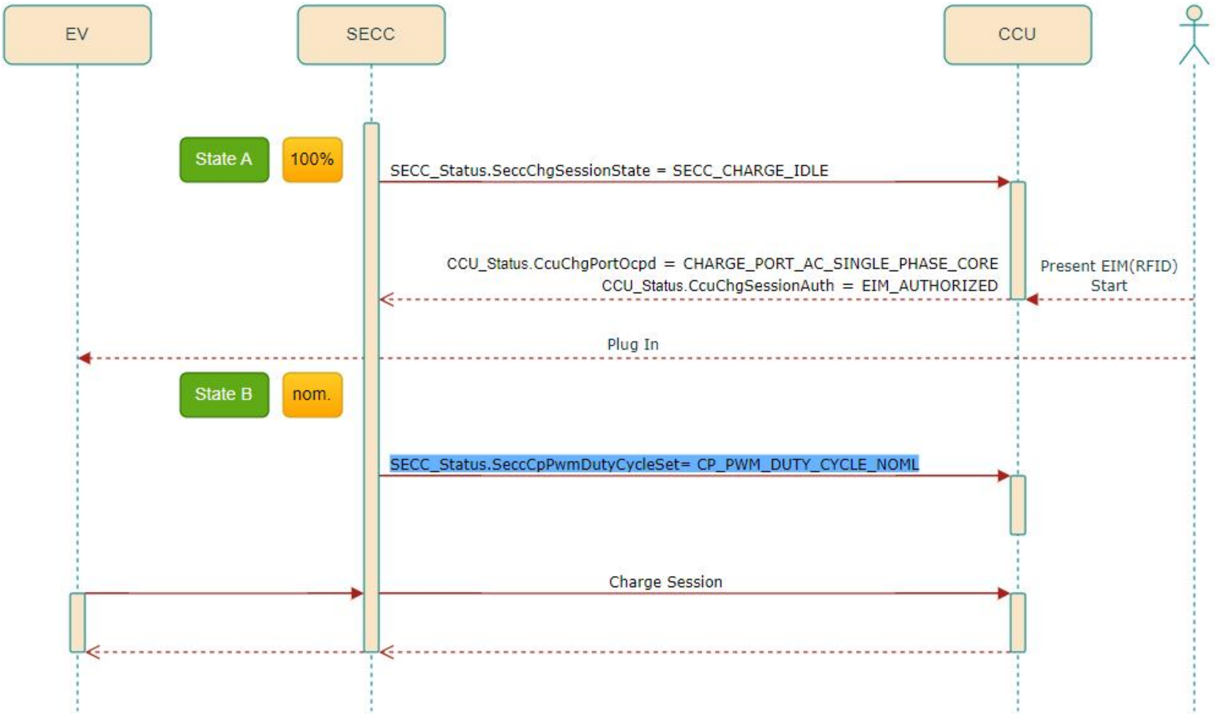
EV
SECC
CCU
State A
100%
SECC_Status.SeccChgSessionState = SECC_CHARGE_IDLE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_AC_SINGLE_PHASE_CORE
CCU_Status.CcuChgSessionAuth = EIM_AUTHORIZED
Present EIM(RFID)
Start
Plug In
State B
nom.
SECC_Status.SeccCpPwmDutyCycleSet= CP_PWM_DUTY_CYCLE_NOML
Charge Session

Figure 10. First authorize, Then plug in

When a Charging Station needs to charge an EV, it needs to authorized the EV drive to start charging. The authorization means could be a RFID, a start button, credit/debit card, PIN-code, or others. In this scenario, AC single phase core and RFID card is used as an example.

1. The EV driver wants to start charging the EV and presents an RFID card.
2. CCU sends signal, CcuChgPortOcpd of message CCU_Status as CHARGE_PORT_AC_SINGAL_PHASE_CORE to occupy SECC, and signal, CcuChgSessionAuth of message CCU_Status as EIM_AUTHORIZED.
3. Then, EV driver plugs the charge gun into EV inlet in.
4. SECC initiates high-level communication with EV, and sends signal, SeccCpPwmDutyCycleSet with CP_PWM_DUTY_CYCLE_NOML of message SECC_Status back to CCU in case that IEC 61851-1 source is CCU.

   CCU shall generate PWM with nominal duty cycle to control pilot wire.
5. Charging is starting.

5.2 First Plug in, then Authorize

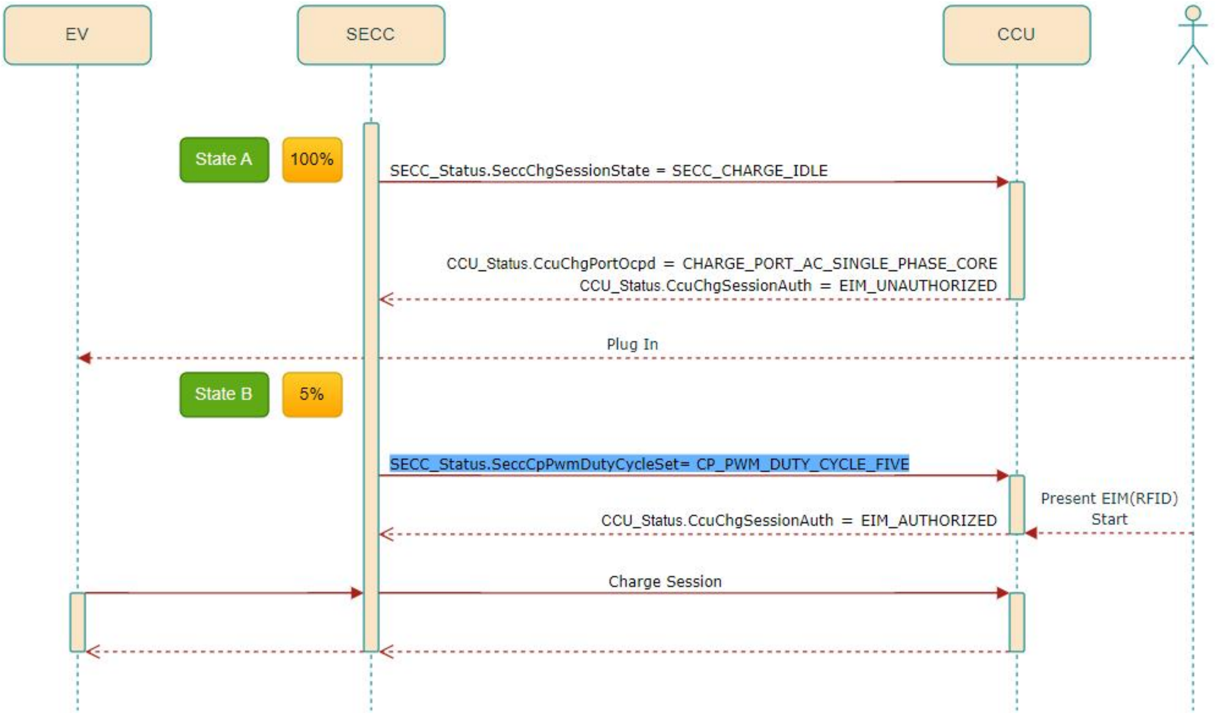
EV
SECC
CCU
State A
100%
SECC_Status.SeccChgSessionState = SECC_CHARGE_IDLE
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_AC_SINGLE_PHASE_CORE
CCU_Status.CcuChgSessionAuth = EIM_UNAUTHORIZED
Plug In
State B
5%
SECC_Status.SeccCpPwmDutyCycleSet= CP_PWM_DUTY_CYCLE_FIVE
CCU_Status.CcuChgSessionAuth = EIM_AUTHORIZED
Present EIM(RFID) Start
Charge Session

Figure 11. First plug in, Then authorize

1. The EV driver wants to start charging the EV and plugs the charging gun in to EV inlet in.
2. CCU sends signal, CcuChgPortOcpd of message CCU_Status as CHARGE_PORT_AC_SINGAL_PHASE_CORE to occupy SECC, and signal, CcuChgSessionAuth of message CCU_Status as EIM_UNAUTHORIZED.
3. SECC initiates high-level communication with EV, and sends signal, SeccCpPwmDutyCycleSet with CP_PWM_DUTY_CYCLE_FIVE of message SECC_Status back to CCU in case that IEC 61851-1 source is CCU.

   CCU shall generate PWM with nominal duty cycle to control pilot wire.
4. Then, EV driver presents an RFID card.
5. Charging is starting.

5.3 EV INITIATES THE END OF ENERGY TRANSFER PROCESS

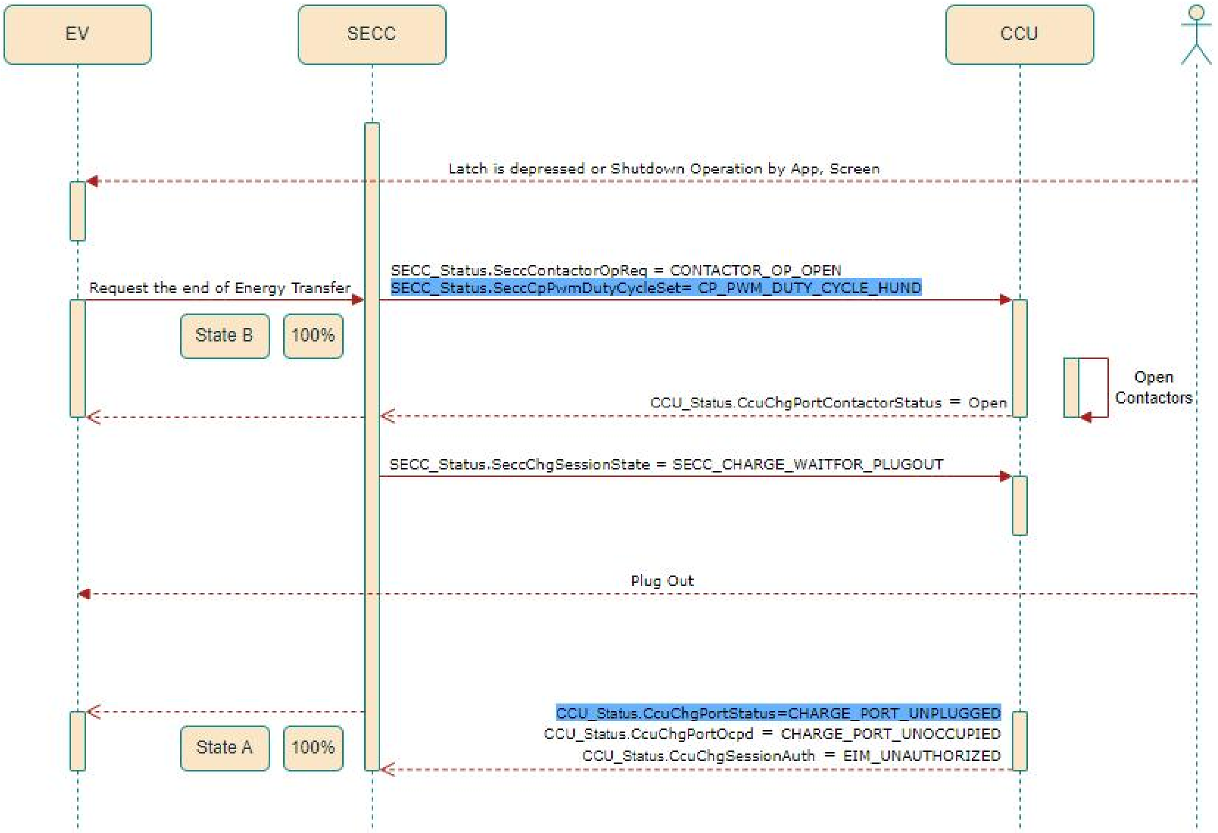
EV
SECC
CCU
Latch is depressed or Shutdown Operation by App, Screen
Request the end of Energy Transfer
SECC_Status.SeccContactorOpReq = CONTACTOR_OP_OPEN
SECC_Status.SeccCpPwmDutyCycleSet= CP_PWM_DUTY_CYCLE_HUND
State B
100%
Open Contactors
CCU_Status.CcuChgPortContactorStatus = Open
SECC_Status.SeccChgSessionState = SECC_CHARGE_WAITFOR_PLUGOUT
Plug Out
CCU_Status.CcuChgPortStatus=CHARGE_PORT_UNPLUGGED
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_UNOCCUPIED
CCU_Status.CcuChgSessionAuth = EIM_UNAUTHORIZED
State A
100%

Figure 12. EV initiates the end of energy transfer process

1. The EV driver wants to stop charging by EV screen or mobile apps, and operate.
2. SECC sends signal, SeccCpPwmDutyCycleSet with CP_PWM_DUTY_CYCLE_HUND of message SECC_Status, and signal, SeccContactorOpReq as CONTACTOR_OP_OPEN to request CCU to generate 100% duty cycle PWM in case that IEC 61851-1 source is CCU, and open contactors.
3. CCU sends Open status back in signal, CcuChgPortContactorStatus of message CCU_Status after it closes contactors.
4. SECC sends signal, CcuChgSessionState of message SECC_Status as SECC_CHARGE_WAITFOR_PLUGOUT to notify plug-out.
5. The EV drive plugs the charge gun out.
6. CCU sends CHARGE_PORT_UNPLUGGED status back in signal, CcuChgPortStatus of message CCU_Status, CHARGE_PORT_UNOCCPIED status in signal, CcuChgPortOcpd, EIM_UNAUTHORIZED status in signal, CcuChgSessionAuth after plug-out.
7. Charging is stopped.

5.4 EVSE INITIATES THE END OF ENERGY TRANSFER PROCESS

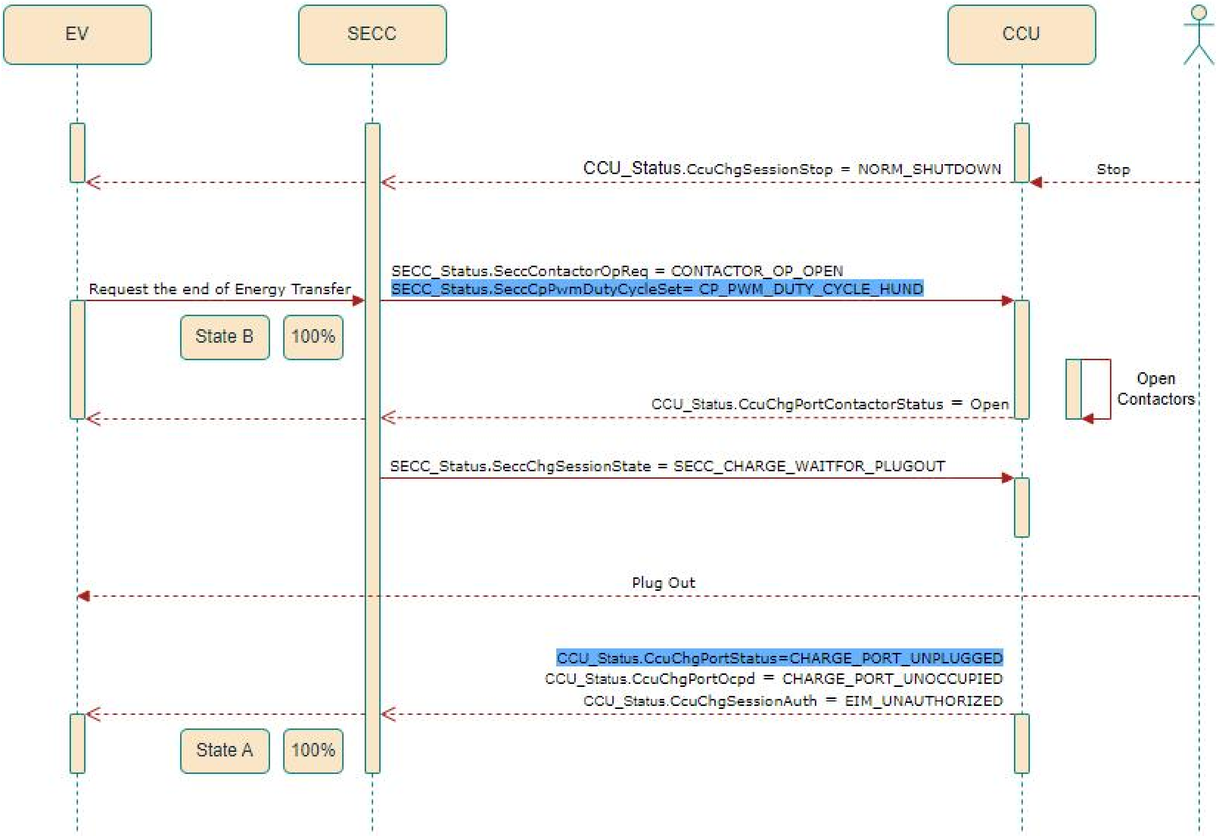
EV
SECC
CCU
CCU_Status.CcuChgSessionStop = NORM_SHUTDOWN
Stop
SECC_Status.SeccContactorOpReq = CONTACTOR_OP_OPEN
SECC_Status.SeccCpPwmDutyCycleSet= CP_PWM_DUTY_CYCLE_HUND
Request the end of Energy Transfer
State B
100%
Open Contactors
CCU_Status.CcuChgPortContactorStatus = Open
SECC_Status.SeccChgSessionState = SECC_CHARGE_WAITFOR_PLUGOUT
Plug Out
CCU_Status.CcuChgPortStatus=CHARGE_PORT_UNPLUGGED
CCU_Status.CcuChgPortOcpd = CHARGE_PORT_UNOCCUPIED
CCU_Status.CcuChgSessionAuth = EIM_UNAUTHORIZED
State A
100%

Figure 13. EVSE initiates the end of energy transfer process

1. The EV driver wants to stop charging by EVSE button or others, and operate.
2. CCU sends NORM_SHUTDOWN status in signal, CcuChgSessionStop of message CCU_Status.
3. SECC sends signal, SeccCpPwmDutyCycleSet with CP_PWM_DUTY_CYCLE_HUND of message SECC_Status, and signal, SeccContactorOpReq as CONTACTOR_OP_OPEN to request CCU to generate 100% duty cycle PWM in case that IEC 61851-1 source is CCU, and open contactors.
4. CCU sends Open status back in signal, CcuChgPortContactorStatus of message CCU_Status after it closes contactors.
5. SECC sends signal, CcuChgSessionState of message SECC_Status as SECC_CHARGE_WAITFOR_PLUGOUT to notify plug-out.
6. The EV drive plugs the charge gun out.
7. CCU sends CHARGE_PORT_UNPLUGGED status back in signal, CcuChgPortStatus of message CCU_Status, CHARGE_PORT_UNOCCPIED status in signal, CcuChgPortOcpd, EIM_UNAUTHORIZED status in signal, CcuChgSessionAuth after plug-out.
8. Charging is stopped.

6 SMART CHARGING

6.1 PROVISIONING

After SECC enters state SECC_CHARGE_INIT, the message, SECC_SysInfo is sent 3 times in a row according to the 250ms cycle.

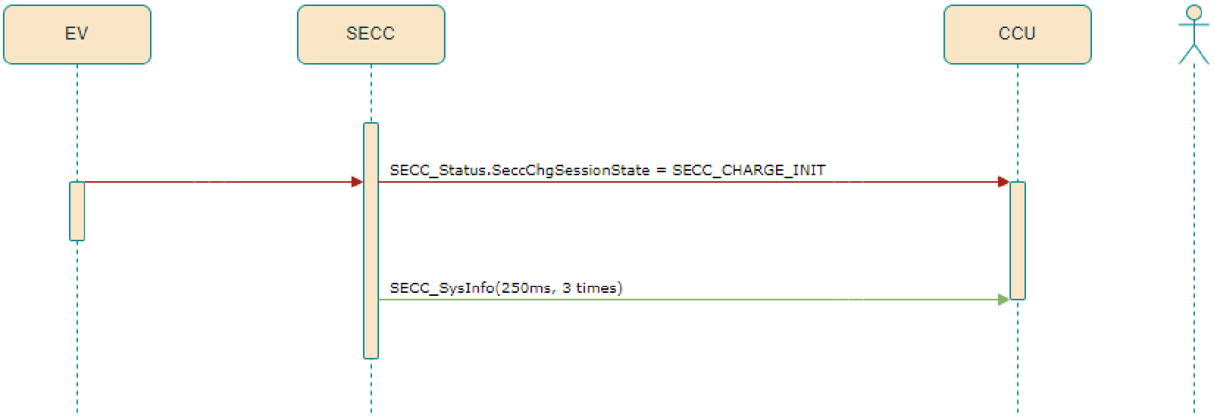
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_CHARGE_INIT
SECC_SysInfo(250ms, 3 times)

Figure 14. Provisioning

6.2 EVCCID

After SECC enters state SECC_ISO2_SESSION_SETUP, the message, SECC_EvEvccID is sent 3 times in a row according to the 250ms cycle.

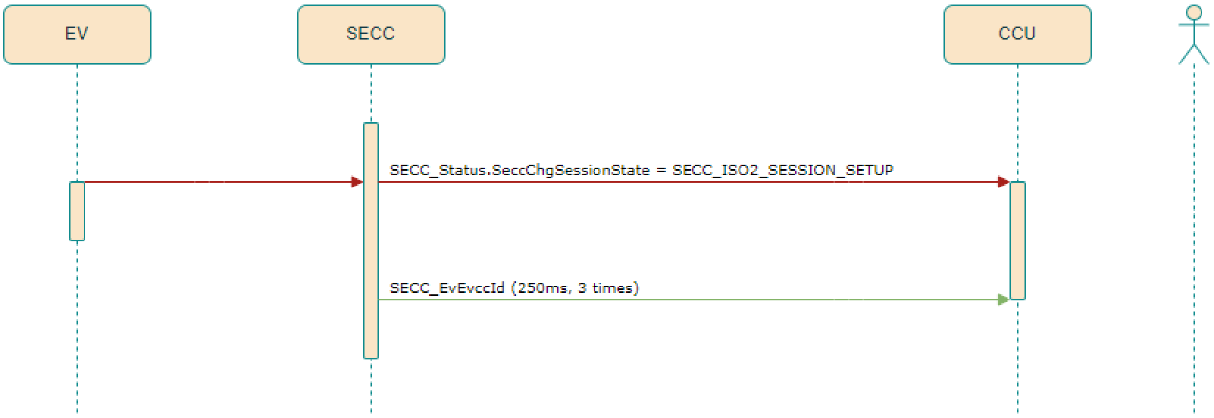
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_ISO2_SESSION_SETUP
SECC_EvEvccId (250ms, 3 times)

Figure 15. EVCCID

6.3 EV Charging Limits

After SECC enters state SECC_ISO2_CHARGE_PARAMETER_DISCOVERY, the message, SECC_EvChgLimits is sent 3 times in a row according to the 250ms cycle.

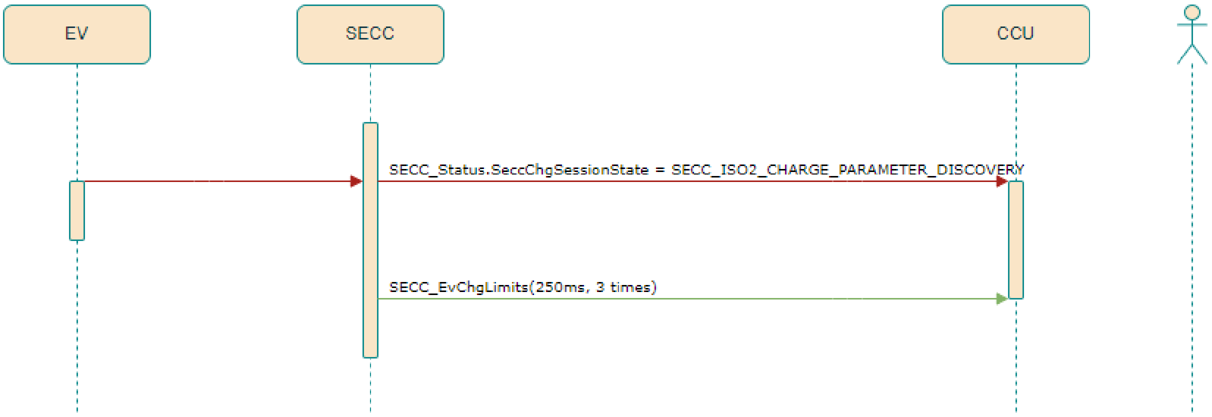
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_ISO2_CHARGE_PARAMETER_DISCOVERY
SECC_EvChgLimits(250ms, 3 times)

Figure 16. EV Charging Limits

6.4 EV Charging Targets

After SECC enters state SECC_ISO2_CHARGE_PARAMETER_DISCOVERY, the message, SECC_EvRessTargets is sent 3 times in a row according to the 250ms cycle.

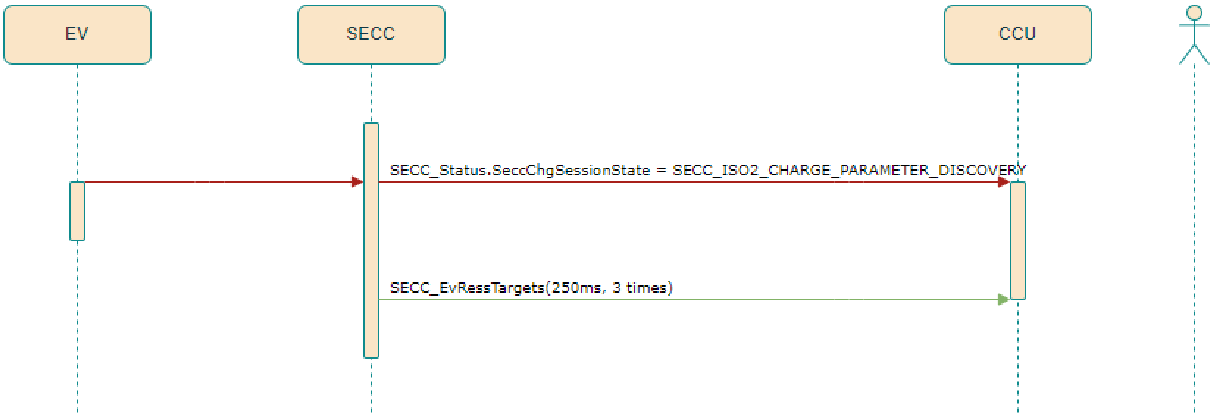
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_ISO2_CHARGE_PARAMETER_DISCOVERY
SECC_EvRessTargets(250ms, 3 times)

Figure 17. EV Charging Targets

6.5 EVSE Charging Limits

After SECC enters state SECC_ISO2_SESSION_SETUP, the message, CCU_EvseChgLimits is started to send according to 100ms cycle; After SECC enters state SECC_ISO2_SESSION_STOP_TERMINATE, it is stopped to send.

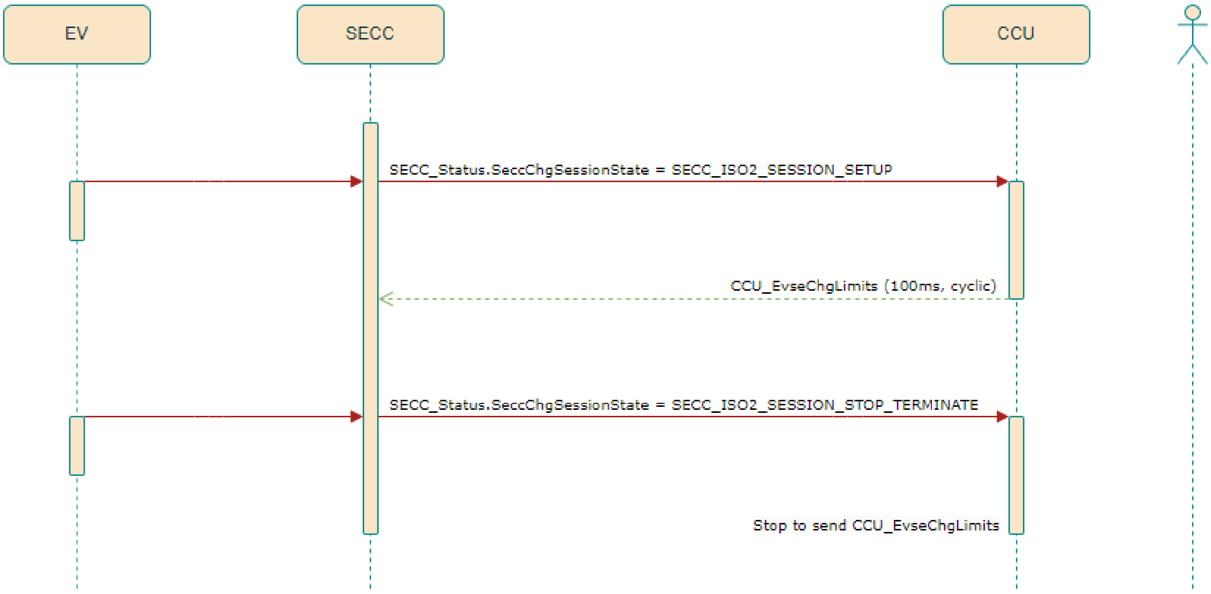
EV
SECC
CCU
SECC_Status.SeccChgSessionState = SECC_ISO2_SESSION_SETUP
CCU_EvseChgLimits (100ms, cyclic)
SECC_Status.SeccChgSessionState = SECC_ISO2_SESSION_STOP_TERMINATE
Stop to send CCU_EvseChgLimits

Figure 18. EVSE Charging Limits

7 Data Transfer

The data transfer operation is used to get or set a configuration file, calibration file, etc. It is also used to download or upload a configuration file, calibration file, PLC modem image, log file, etc.

CCU sends True status in signal, CcuDataTransferReq of message CCU_Status to notify a data transfer; SECC sends signal, SeccDataTransferRes with True or False Status of message SECC_Status to CCU to accept or not. If Accept, SECC_Status and CCU_Status keep-alive messages will be stopped to support data transfer in case of download and upload operations.

To support data transfer, a pair of messages are involved.

Message	ID	Direction	Description
CCU_DataTransferReq	0x18CAF456	DB2605 ← CCU	Data transfer request
SECC_DataTransferRes	0x18B256F4	DB2605 → CCU	Data transfer response

CCU sends CCU_DataTransferReq to SECC to request a data transfer operation; SECC sends SECC_DataTransferRes back to report status to accept or reject data transfer.

For additional messages and detailed information, refer to *DB2605 EV Charging Controller Communication Matrix*

7.1 DATA TRANSFER - GET/SET

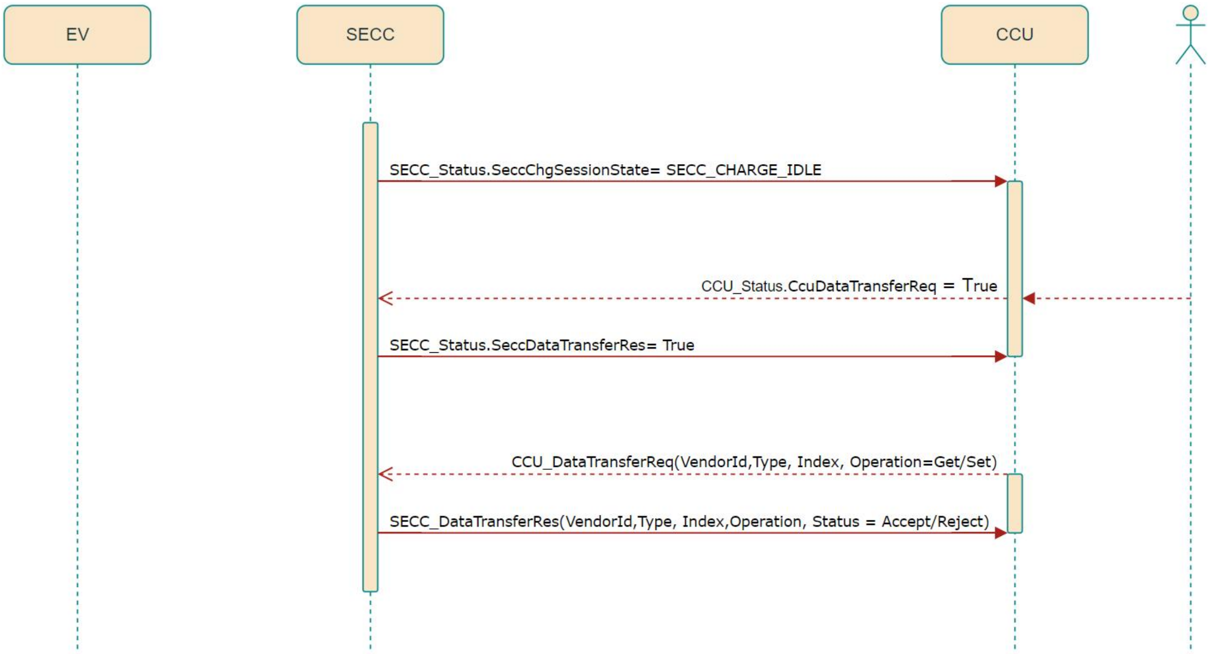
EV
SECC
CCU
SECC_Status.SeccChgSessionState= SECC_CHARGE_IDLE
CCU_Status.CcuDataTransferReq = True
SECC_Status.SeccDataTransferRes= True
CCU_DataTransferReq(VendorId,Type, Index, Operation=Get/Set)
SECC_DataTransferRes(VendorId,Type, Index,Operation, Status = Accept/Reject)

Figure 19. Data Transfer - Get/Set

1. User or remote controller notifies data transfer.
2. CCU sends True status in signal, CcuDataTransferReq of message CCU_Status to notify a data transfer; SECC sends signal, SeccDataTransferRes with True or False Status of message SECC_Status to CCU to accept or not.
3. If accepted, CCU sends CCU_DataTransferReq to SECC to request a data transfer operation with Operation as Get/Set; SECC sends SECC_DataTransferRes back to report.

This operation is usually used in getting applied file or setting a file to apply into system. For example, PLC modem PIB operation.

Set operation:

Type is “PLC Modem PIB file”, Index is “5”, Operation is “Set”; The PLC Modem PIB file, which the index equals “5” will be applied in system to use.

Get operation:

Type is “PLC Modem PIB file”, Operation is “Get”; The applied file index “5” will be reported in Index.

7.2 Data Transfer - Download/Upload

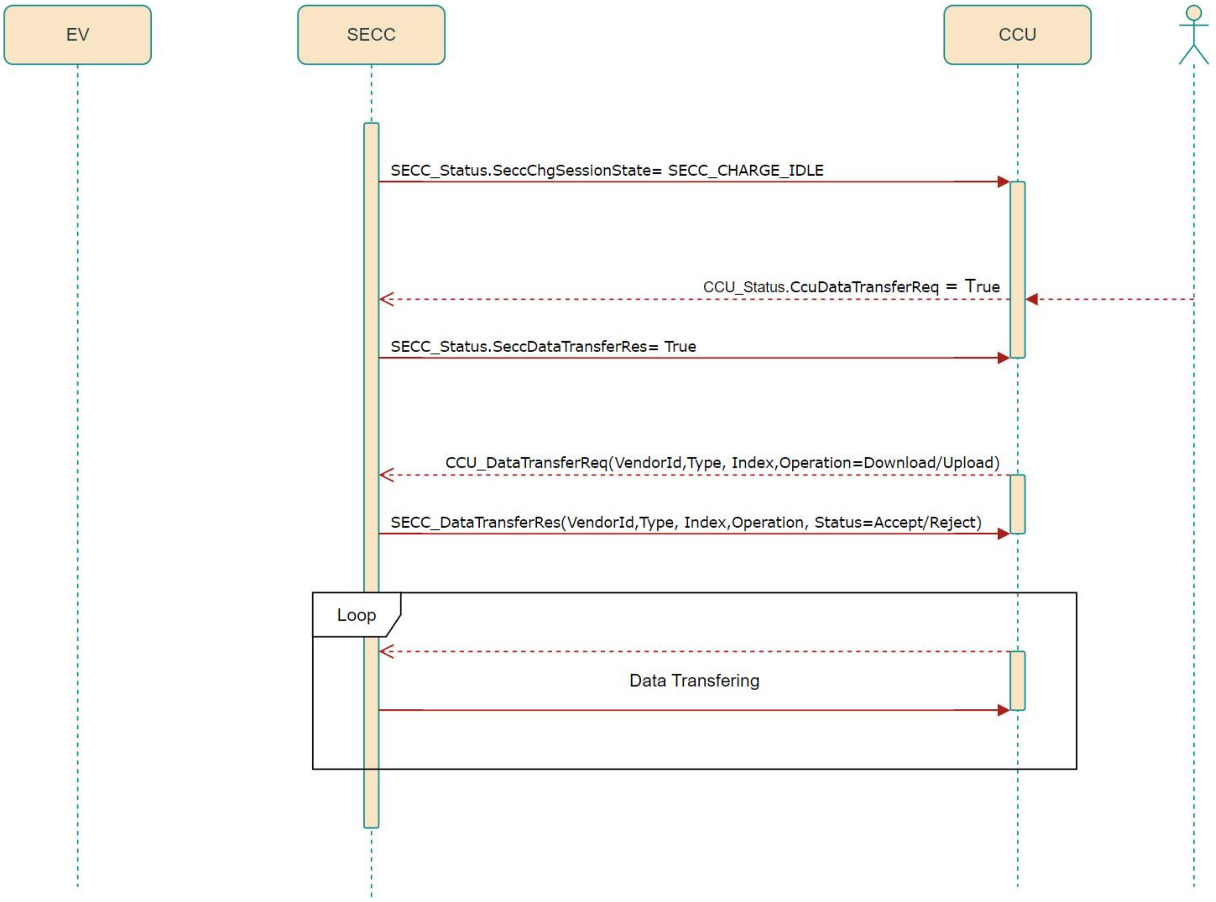
EV
SECC
CCU
SECC_Status.SeccChgSessionState= SECC_CHARGE_IDLE
CCU_Status.CcuDataTransferReq = True
SECC_Status.SeccDataTransferRes= True
CCU_DataTransferReq(VendorId,Type, Index,Operation=Download/Upload)
SECC_DataTransferRes(VendorId,Type, Index,Operation, Status=Accept/Reject)
Loop
Data Transfering

Figure 20. Data Transfer Download/Upload

1. User or remote controller notifies data transfer.
2. CCU sends True status in signal, CcuDataTransferReq of message CCU_Status to notify a data transfer; SECC sends signal, SeccDataTransferRes with True or False Status of message SECC_Status to CCU to accept or not.
3. If accepted, CCU sends CCU_DataTransferReq to SECC to request a data transfer operation with Operation as Download/Upload; SECC sends SECC_DataTransferRes back to report.
4. If accepted, CCU and SECC stops keep-alive messages.
5. Data transfer starts; After data transfer, Keep-alive messages will be sent again.

8 Firmware Upgrade

For detailed information, refer to

DB2605 EV Charging Controller Firmware Upgrade Programming Guide

LEGAL INFORMATION

Copyright

Disclaimer

None of the information contained in this document constitutes an express or implied warranty by Dropbeats Technology Co., Ltd. as to the sufficiency, fitness or suitability for a particular purpose of any such information or the fitness, or suitability for a particular purpose, merchantability, performance, compatibility with other parts or systems, of any of the products of Dropbeats Technology Co., Ltd., or any portion thereof, referred to in this document. Dropbeats Technology Co., Ltd. expressly disclaims all representations and warranties of any kind regarding the contents or use of the information, including, but not limited to, express and implied warranties of accuracy, completeness, merchantability, fitness for a particular use, or non-infringement.

In no event will Dropbeats Technology Co., Ltd. be liable for any direct, indirect, special, incidental or consequential damages, including, but not limited to, lost profits, lost business, or lost data resulting from any use of or reliance upon the information, whether or not Dropbeats Technology Co., Ltd. has been advised of the possibility of such damage.

CONTACTING DROPBEATS TECHNOLOGY

Dropbeats Technology
123 Juli Road, Building 4, Shanghai, China
Tel: +86 (21) 5085-0752
Fax: +86 (21) 5085-0753
Document Information: *document@drop-beats.com*
Corporate Information: *info@drop-beats.com*
Technical Support: *apps@drop-beats.com*
Web Site: *https://www.drop-beats.com*

REVISION HISTORY

Revision	Date	Descriptions
1.0.0	2024.5.31	Initial.
1.0.1	2024.6.7	Changed signal SECC_EvChgCrntLimits to SECC_EvChgLimits.
1.0.2	2024.6.18	Added IEC61851-1 source; Added SECC_EvEvccId, SECC_EvChgLimits, SECC_EvRessTargets, CCU_EvseChgLimit Usage.
1.0.3	2024.7.18	Split Charge element "Contactors" to "Contactors - Close" and "Contactors - Open"
1.0.4	2024.8.2	Updated SeccContactorOpReq

Annex A. CRC Code Examples

A.1. CRC8

```
/*************************************************************************
 * Name:    CRC-8     polynomial: 0x07 （x8+x2+x+1）
 * Width:      8
 * Poly:    0x07
 * Init:    0x00
 * Refin:   False
 * Refout:  False
 * Xorout:  0x00
 * Alias:   CRC-8
 ************************************************************************/
unsigned char CRC8(unsigned char *data, unsigned int datalen)
{
        unsigned char wCRCin = 0x00;
        unsigned char wCPoly = 0x07;

        while (datalen--)
        {
                wCRCin ^= (*(data++));
                for(int i = 0;i < 8;i++)
                {
                        if(wCRCin & 0x80)
                                wCRCin = (wCRCin << 1) ^ wCPoly;
                        else
                                wCRCin = wCRCin << 1;
                }
        }
        return (wCRCin);
}
```

A.2 CRC16

```
/*************************************************************************
 * Name:    CRC-16/XMODEM       x16+x12+x5+1
 * Width:       16
 * Poly:    0x1021
 * Init:    0x0000
 * Refin:   False
 * Refout:  False
 * Xorout:  0x0000
 * Alias:   CRC-16/ZMODEM,CRC-16/ACORN
 ***********************************************************************/
unsigned short CRC16_XMODEM(unsigned char *data, unsigned int datalen)
{
        unsigned short wCRCin = 0x0000;
        unsigned short wCPoly = 0x1021;

        while (datalen--)
        {
                wCRCin ^= (*(data++) << 8);
                for(int i = 0;i < 8;i++)
                {
                        if(wCRCin & 0x8000)
                                wCRCin = (wCRCin << 1) ^ wCPoly;
                        else
                                wCRCin = wCRCin << 1;
                }
        }
        return (wCRCin);
}
```