# Protokoller

## X10.1

Til at kommunikere mellem Styreboks og enhed benyttes X10.1 kommunikation.

X10.1 kommunikation sendes 1 bit ad gangen hvorfor en kommando kan opdeles i flere pakker. Fast for hver kommando er 4 startbits, 4 huskode bits, 8 adressebits, 3 typebits, nul eller flere databits, en paritetsbit og 6 slutbits.

Startbits (STX): Fortæller at en besked er på vej. Forekommer altid som: '1110'  
Huskode: Indeholder koden for det hus hvor enheden befinder sig  
Adresse: Indeholder adressen på enheden  
Typebits: Definerer antal databit i beskeden og kommandoen/svaret i datapakken. Se afsnittet *typeliste* for uddybning.   
Databits: Indeholder bools eller fejlkoder  
Paritetsbits: Validerer om besked er modtaget intakt. Der benyttes even parity (se terminologi liste for uddybning). Forekommer altid som: '0' (true) eller '1' (false)  
Slutbits (ETX): Fortæller at en besked er slut. Forekommer altid som: '000000'

I X10.1 protokollen er alle lige typenumre hvor Styreboks tilgår Enhed, oftest kommandoer. Alle ulige typenumre er hvor Enhed tilgår Styreboks, alle svar.

I oversigten over protokollerne er startbits, paritetsbit og slutbits ikke synlige eftersom de altid er en del af beskeden, og ikke defineret af brugeren. Eks:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | Huskode | Adresse | Type | Data | Paritet | ETX |

Alle beskeder bliver altid bekræftet med en godkendelsesbesked eller en fejlmeddelelse.

### Tabeloversigt over X10.1 kommunikation.

Nedenstående tabeller beskriver de forskellige datapakker der sendes mellem Styreboks og Enhed.  
Tabellerne er opdelt efter følgende eksempler:

|  |  |  |
| --- | --- | --- |
| Use case der tages udgangspunkt i | | |
| Metode/funktion | | |
| Beskrivelse af type og metode/funktion | | |
| Retning på dataflow: Fra > Til | | |
| Huskode: | '4-cifret binært tal' | |
| Adresse: | '8-cifret binært tal' | |
| Type: | '3-cifret binært tal' | |
| Data: | Bitnummer: data | |
| UC7: Kør Simulering (**eksempel**) | | | |
| switchState(unitID) | | | |
| Sender ny status | | | |
| Styreboks > Enhed | | | |
| Huskode: | | '0001' | |
| Adresse: | | '0001-0001' | |
| Type: | | '010' | |
| Data: | | b0: true | |

”UC7: Kør Simulering (**eksempel**)” som datapakke i bits:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STX | Huskode | Adresse | Type | Data | Paritet | ETX |
| '1110' | '0001' | '0001-0001' | '010' | '1' | '0' | '000000' |

### Kommandoer

|  |  |
| --- | --- |
| UC2: Statusforespørgsel | |
| getUnitStatus(unitID) | |
| Anmoder om status på enheden | |
| Styreboks > Enhed | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '000' |
| Data: | - |

|  |  |
| --- | --- |
| UC7: Kør Simulering | |
| switchState(unitID) | |
| Sender ny status | |
| Styreboks > Enhed | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '010' |
| Data: | b0: bool |

### Svar

|  |  |
| --- | --- |
| UC2: Statusforespørgsel | |
| status | |
| Sender status på enheden | |
| Enhed > Styreboks | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '001' |
| Data: | b0: bool |

|  |  |
| --- | --- |
| UC7: Kør Simulering | |
| Return true | |
| Sætter enhed i ny tilstand | |
| Enhed > Styreboks | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '011' |
| Data: | b0: bool |

## Generelle svar

### Godkendt

|  |  |
| --- | --- |
| validated | |
| Besked modtaget og forstået | |
| Styreboks > Enhed | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '100' |
| Data: | - |

|  |  |
| --- | --- |
| validated | |
| Besked modtaget og forstået | |
| Enhed > Styreboks | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '101' |
| Data: | - |

### Fejlmeddelelser

|  |  |
| --- | --- |
| error | |
| Besked modtaget men ikke forstået | |
| Styreboks > Enhed | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '110' |
| Data: | b0-b1: Fejlkode |

|  |  |
| --- | --- |
| error | |
| Besked modtaget men ikke forstået | |
| Enhed > Styreboks | |
| Huskode: | 'XXXX' |
| Adresse: | 'XXXX-XXXX' |
| Type: | '111' |
| Data: | b0-b1: Fejlkode |

## PSF (PC Styreboks Forbindelse)

Til at kommunikere mellem PC og Styreboks bruges PSF kommunikation som foregår over UART forbindelse.

PSF kommunikation sendes 1 byte ad gangen hvorfor en kommando kan opdeles i flere pakker. Fast for hver kommando er en startbyte, en typebyte, en længdebyte, nul eller flere databytes, en paritetsbyte og en slutbyte.

Startbyte (STX): Fortæller at en besked er på vej. Forekommer altid som: 'CC' i hexadecimal tal  
Typebyte: Definerer kommandoen/svaret i datapakken. Se afsnittet *typeliste* for uddybning  
Længdebyte: Definerer hvor mange databytes der sendes  
Databytes: Indeholder eksempelvis adresse, rum, navn på enhed/rum, tid mm.  
Paritetsbyte: Validerer om besked er modtaget intakt. Gennem en simpel checksum algoritme, som benytter XOR, tjekkes pakkens validitet (se softwaredesign for uddybning).   
Slutbyte (ETX): Fortæller at en besked er slut. Forekommer altid som: 'FF' i hexadecimal tal

I PSF protokollen er alle lige typenumre hvor PC tilgår Styreboks, oftest kommandoer. Alle ulige typenumre er hvor Styreboks tilgår PC, alle svar.

I oversigten over protokollerne er startbyte, paritetsbyte og slutbyte ikke synlige eftersom de altid er en del af beskeden, og ikke defineret af brugeren. Eks:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | Type | Længde | Data | Paritet | ETX |

Alle beskeder bliver altid bekræftet med en godkendelsesbesked eller en fejlmeddelelse.

### Tabeloversigt over PSF kommunikation.

Nedenstående tabeller beskriver de forskellige datapakker der sendes mellem PC og styreboks.

Tabellerne er opdelt efter følgende eksempler:

|  |  |
| --- | --- |
| Use case der tages udgangspunkt i | |
| Metode/funktion | |
| Beskrivelse af type og metode/funktion | |
| Retning på dataflow: Fra > Til | |
| Type: | 'Hexadecimal tal' |
| Længde: | 'Hexadecimal tal' |
| Data: | Bytenummer: data |

|  |  |
| --- | --- |
| UC2 – Statusforespørgsel (**eksempel**) | |
| getUnitStatus(unitID) | |
| Anmoder om status på enhed | |
| PC > Styreboks | |
| Type: | '04' |
| Længde: | '01' |
| Data: | B0: '0F' |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STX | Type | Længde | Data | Paritet | ETX |
| 'CC' | '04' | '01' | '0F' | 39 | 'FF' |

## Kommandoer

|  |  |
| --- | --- |
| UC1: Opstart af system | |
| validatePin() | |
| Validering af kodelås | |
| PC > Styreboks | |
| Type: | '00' |
| Længde: | '00' |
| Data: | - |

## Svar

|  |  |
| --- | --- |
| UC1: Opstart af system | |
| pinValidated | |
| Svar på validering | |
| Styreboks > PC | |
| Type: | '01' |
| Længde: | '01' |
| Data: | B0: bool |

|  |  |
| --- | --- |
| UC1: Opstart af system | |
| getUnit() | |
| Henter tidstabel og ID | |
| PC > Styreboks | |
| Type: | '02' |
| Længde: | '00' |
| Data: | - |

|  |  |
| --- | --- |
| UC1: Opstart af system | |
| unit(ID, timeTable) | |
| Sender enheds ID og tidsplan | |
| Styreboks > PC | |
| Type: | '03' |
| Længde: | 'XX' |
| Data: | B0: bool |
|  | B1 – BX: Timetable |

|  |  |
| --- | --- |
| UC2 - Statusforespørgsel | |
| getUnitStatus(unitID) | |
| Anmoder om status på enhed | |
| PC > Styreboks | |
| Type: | '04' |
| Længde: | '01' |
| Data: | B0: Adresse |

|  |  |
| --- | --- |
| UC2 - Statusforespørgsel | |
| status | |
| Sender status på enhed | |
| Styreboks > PC | |
| Type: | '05' |
| Længde: | '01' |
| Data: | B0: Status |

|  |  |
| --- | --- |
| UC3 - Tilføjelse af enhed | |
| sendUnit(unitID) | |
| Giver styreboks besked om at ny enhed er tilføjet | |
| PC > Styreboks | |
| Type: | '06' |
| Længde: | '01' |
| Data: | B0: Adresse |
| UC3 - Tilføjelse af enhed | |
| Unit received | |
| Godkender at ny enhed er tilføjet til systemet | |
| Styreboks > PC | |
| Type: | '07' |
| Længde: | '01' |
| Data: | B0: bool |

|  |  |
| --- | --- |
| UC4 - Fjern enhed | |
| deleteUnit(unitID) | |
| Fjerner enhed fra system | |
| PC > Styreboks | |
| Type: | '08' |
| Længde: | '01' |
| Data: | B0: Adresse |
| UC4 - Fjern enhed | |
| Unit deleted | |
| Enhed fjernet | |
| Styreboks > PC | |
| Type: | '09' |
| Længde: | '01' |
| Data: | B0: bool |

|  |  |
| --- | --- |
| UC5 - Ret enhed | |
| modifyUnit(unit, prevUnitID) | |
| Retter enhed | |
| PC > Styreboks | |
| Type: | '0A' |
| Længde: | '02' |
| Data: | B0: Adresse |
|  | B1: Rum |

|  |  |
| --- | --- |
| UC6 - Ændring af tidsplan | |
| updateTime(unitID, schedule) | |
| Ændrer tidsplan for valgte enhed(er)/rum | |
| PC > Styreboks | |
| Type: | '0C' |
| Længde: | 'XX' |
| Data: | B0: Adresse |
|  | B1: Starttid |
|  | B2: Sluttid |
|  | B3: Dage |

|  |  |
| --- | --- |
| UC5 - Ret enhed | |
| Unit modified | |
| Enhed rettet | |
| Styreboks > PC | |
| Type: | '0B' |
| Længde: | '01' |
| Data: | B0: bool |

|  |  |
| --- | --- |
| UC6 - Ændring af tidsplan | |
| Time updated | |
| Tidsplanen er opdateret | |
| Styreboks > PC | |
| Type: | '0D' |
| Længde: | '01' |
| Data: | B0: bool |

## Generelle svar

### Godkendt

|  |  |
| --- | --- |
| validated() | |
| Godkender at besked er modtaget og forstået | |
| PC > Styreboks | |
| Type: | '0E' |
| Længde: | '00' |
| Data: | - |

|  |  |
| --- | --- |
| validated() | |
| Godkender at besked er modtaget og forstået | |
| Styreboks > PC | |
| Type: | '0F' |
| Længde: | '00' |
| Data: | - |

### Fejlmeddelelser

|  |  |
| --- | --- |
| error() | |
| Giver besked om fejl i data | |
| PC > Styreboks | |
| Type: | '10' |
| Længde: | '01' |
| Data: | B0: Fejlkode |

|  |  |
| --- | --- |
| error() | |
| Giver besked om fejl i data | |
| Styreboks > PC | |
| Type: | '11' |
| Længde: | '01' |
| Data: | B0: Fejlkode |

I2C/IIC/TWI

I2C protokollen benyttes til at kommunikere med vores Real Time Clock modul som er baseret på en DS3231 Real Time Clock, standarden for I2C er veldefineret ved 100kHz og 400 kHz, men i forbindelse med DS3231 anvendes der en 400kHz clock-frekvens. Ønskes der yderlige information om I2C standarden kan dette findes i I2C specifikationen der er vedlagt som bilag. For mere specifik information om brugen af I2C til kommunikation med DS3231 kan dette findes i DS3231 datasheet der ligeledes er vedlagt som bilag.

## SPI

SPI protokollen benyttes til kommunikation med vores SD-kort, der bliver i vores projekt anvendt direkte adgang til de enkelte datablokke på sd-kortet i stedet for at implementere et fat filsystem. For yderligere information om SPI i forbindelse med sd-kort se SD-Card simplified spec kapitel 7 som er vedlagt som bilag. For yderligere infomation om SPI generelt kan dette findes i the avr microcontroller and embeded system kapitel 17.

# Typeliste

Type er defineret ved at det er typen af besked der sendes. Ved dette forstås både hvilken retning beskeden går, f.eks fra PC til Styreboks, og hvilken handling der udføres. Se nedenstående tabeller for sammenhæng mellem type, retning på dataflow og handling:

## PSF

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Navn** | **Fra > Til** | **Handling** |
| 00 | validatePin() | PC > Styreboks | Validering af kodelås |
| 01 | pinValidated | Styreboks > PC | Svar på validering |
| 02 | getUnitStatus(unitID) | PC > Styreboks | Henter tidstabel og ID |
| 03 | unit(ID, timeTable) | Styreboks > PC | Sender enheds ID og tidsplan |
| 04 | getUnitStatus(unitID) | PC > Styreboks | Anmoder om status på enhed |
| 05 | status | Styreboks > PC | Sender status på enhed |
| 06 | sendUnit(unitID) | PC > Styreboks | Giver styreboks besked om at ny enhed er tilføjet |
| 07 | Unit recieved | Styreboks > PC | Godkender at ny enhed er tilføjet til systemet |
| 08 | deleteUnit(unitID) | PC > Styreboks | Fjerner enhed fra system |
| 09 | Unit deleted | Styreboks > PC | Enhed fjernet |
| 0A | modifyUnit(unit, prevUnitID) | PC > Styreboks | Retter enhed |
| 0B | Unit modified | Styreboks > PC | Enhed rettet |
| 0C | updateTime(unitID, schedule) | PC > Styreboks | Ændrer tidsplan for valgte enhed(er)/rum |
| 0D | Time updated | Styreboks > PC | Tidsplanen er opdateret |
| 0E | validated() | PC > Styreboks | Godkender at besked er modtaget og forstået |
| 0F | validated() | Styreboks > PC | Godkender at besked er modtaget og forstået |
| 10 | error() | PC > Styreboks | Giver besked om fejl i data |
| 11 | error() | Styreboks > PC | Giver besked om fejl i data |

## X10.1

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Navn** | **Fra > Til** | **Handling** |
| 000 | getUnitStatus(unitID) | Styreboks > Enhed | Anmoder om status på enhed |
| 001 | status | Enhed > Styreboks | Sender status på enhed |
| 010 | switchState(unitID) | Styreboks > Enhed | Sender ny status |
| 011 | Return true | Enhed > Styreboks | Sætter enhed i ny tilstand |
| 100 | validated | Styreboks > Enhed | Besked modtaget og forstået |
| 101 | validated | Enhed > Styreboks | Besked modtaget og forstået |
| 110 | error | Styreboks > Enhed | Besked modtaget, men ikke forstået |
| 111 | error | Enhed > Styreboks | Besked modtaget, men ikke forstået |