

03-2009

dpc-Haustechnik

© Ulrich Piechnick – Juli 2008, 2009



in Arbeit

[BUS-PROTOKOLLE]

Technisches Handbuch

Übersicht

Requests (Befehle an den Controller)

Type	Sub-type	Bedeutung
,C'	,0'	Modul-Konfiguration
,C'	,1'	Tabellen-Konfiguration
,C'	,2'	Schaltuhr aktivieren/deaktivieren
,C'	,3'	EEP löschen
,C'	,4'	Loglevel einstellen
,C'	,5'	Sensorkonfiguration neu ins RAM lesen
,C'	,6'	Speichervariablen überschreiben (MemVarTab)
,C'	,D'	Datum und Uhrzeit stellen. Datenformat TT-MM-YY-WT-STD-MIN-Sek-DST
,G'	,1'	Status anfordern
,G'	,2'	Get Konfig-Tabelle (nächster Parameter entscheidet welche)
G	,3'	Einen Messwert vom Sensornetz anfordern
G	,4'	Alle Sensornamen und –Messwerte anfordern
G	,5'	Firmware-Version anfordern
,G'	,6'	Bootzeitpunkt anfordern
,G'	,7'	Speichervariablen (MemVarTab) anfordern
,G'	,8'	RemoteDevice-Daten vom Funkhub anfordern (DeviceTab)
,G'	9	Fühlerdaten vom Funkhub anfordern (FühlerTab)
,G'	,a'	DateTimeRequest, Datum und Uhrzeit anfordern
,G'	,b'	Aktuelle Dimmerhelligkeiten anfordern
,S'	,1'	Schalten Szene
,S'	,2'	Schalten Aktor
,S'	,3'	Automatisches Aufdimmen starten
,S'	,4'	Stopp aufdimmen
,S'	,5'	Dimmerhelligkeit einstellen
,S'	,r'	Softreset

Übersicht

Responses (Antworten vom Controller)

Type	Sub-Type	Bedeutung
,s'	,1'	Status. Format: Anz. I2C-Dev, je Dev. Ein Byte, PortA, PortB, PortC
,s'	,3'	Ein Messwert incl. Sensorname
,s'	,4'	Alle SensorNamen und Messwerte aus SensorTab
,s'	,5'	Firmware-Version
,s'	,6'	Fehlerstatus nach dem Speichern von Konfigurations-Daten
,s'	,7'	Boottimestamp
,s'	8	Speichervariablen (MemVarTab)
,s'	,9'	Fühlerdaten vom Funkhub
,s'	,a'	Remote-Device-Daten vom Funkhub schicken (DeviceTab)
,s'	,c'	Aktuelle Dimmer-Helligkeiten

Request C0

Modul-Konfigurationsdaten übertragen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	0	1	ASCII	Paket-Sub-Type
5	19	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Rel. Tabellenadresse MSB (\$00)
7	X	1	Hexwert	Rel. Tabellenadresse LSB (\$0-\$0F)
8	X	1	Hexwert	Eintrags-ID (0.15) innerhalb der Tabelle
9	X	1	Hexwert	Config-Byte 0
10	X	1	Hexwert	Config-Byte 1
11	X	1	Hexwert	Config-Byte 2
12	X	1	Hexwert	Config-Byte 3
13	X	1	Hexwert	Config-Byte 4
14	X	1	Hexwert	Config-Byte 5
15	X	1	Hexwert	Config-Byte 6
16	X	1	Hexwert	Config-Byte 7
17	X	1	Hexwert	Config-Byte 8
18	X	1	Hexwert	Config-Byte 9
19	X	1	Hexwert	Config-Byte 10
20	X	1	Hexwert	Config-Byte 11
21	X	1	Hexwert	Config-Byte 12
22	X	1	Hexwert	Config-Byte 13
23	X	1	Hexwert	Config-Byte 14
24	X	1	Hexwert	Config-Byte 15
25	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando werden Modul-Konfigurationsdaten an das adressierten Modul übertragen. Diese Daten werden auf den Stellen 0 bis 256 des EEPROMs abgelegt. Jeder Konfigurationsblock enthält 16 Nutzbytes.

Beispiel:

Wird unterstützt von:

Buskoppler, C-2560

Request C1

Konfigurationsdaten übertragen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type
5	19	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Rel. Tabellenadresse MSB
7	x	1	Hexwert	Rel. Tabellenadresse LSB
8	x	1	Hexwert	Eintrags-ID (0.254) innerhalb der Tabelle
9	x	1	Hexwert	Config-Byte 0
10	x	1	Hexwert	Config-Byte 1
11	x	1	Hexwert	Config-Byte 2
12	x	1	Hexwert	Config-Byte 3
13	x	1	Hexwert	Config-Byte 4
14	x	1	Hexwert	Config-Byte 5
15	x	1	Hexwert	Config-Byte 6
16	x	1	Hexwert	Config-Byte 7
17	x	1	Hexwert	Config-Byte 8
18	x	1	Hexwert	Config-Byte 9
19	x	1	Hexwert	Config-Byte 10
20	x	1	Hexwert	Config-Byte 11
21	x	1	Hexwert	Config-Byte 12
22	x	1	Hexwert	Config-Byte 13
23	x	1	Hexwert	Config-Byte 14
24	x	1	Hexwert	Config-Byte 15
25	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando werden Konfigurationsdaten an das adressierten Controller übertragen. Dabei handelt es sich um die in den verschiedensten Tabellen abgelegten Daten. Gespeichert werden die Daten beginnend ab der EEPROM-Adresse 256. Jeder Konfigurationsblock enthält 16 Nutzbytes.

Beispiel:

Wird unterstützt von:

Buskoppler, C-2560

Request C2

Schaltuhr aktivieren bzw. deaktivieren

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	2	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	0=deaktivieren, 1=aktivieren
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando aktiviert oder deaktiviert die Schaltuhrenfunktion auf dem adressierten Controller

Beispiel:

Wird unterstützt von:

Buskoppler, CPU

Request C3

EEProm löschen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	3	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	1	1	Hexwert	Muss \$01 sein
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando kann das Löschen des gesamten EEPROMs im adressierten Modul angestoßen werden. Während dieser Zeit (bis zu einer Minute) werden im Modul alle Interrupts gesperrt. Es ist erst wieder nach dem Löschvorgang ansprechbar.

Beispiel:

Wird unterstützt von:

Buskoppler, C-2560

Request C4

Loglevel einstellen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	4	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	n	1	Hexwert	Entspricht dem LogLevel (0-3)
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando kann der Loglevel eingestellt werden. Zur Zeit werden die Level 0 bis 3 unterstützt. Der Logvel 1 gibt die wichtigsten Informationen auf dem Debug-Kanal aus. Der Loglevel 3 ist sehr gespächig. Der Loglevel 0 absolut stumm.

Beispiel:

Wird unterstützt von:
C-2560

Request C5

Sensorkonfiguration ins RAM lesen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	5	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	1	1	Hexwert	Muss \$01 sein
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando wird der Controller die Messfühler-Konfiguration neu ins RAM lesen. Dadurch wirken sich Änderungen direkt aus.

Beispiel:

Wird unterstützt von:

C-2560

Request C6

Speichervariablen im Controller überschreiben

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	Z	1	ASCII	Absender des Paketes, hier ‚z‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	C	1	ASCII	Paket-Type (kleines s)
4	6	1	ASCII	Paket-Sub-Type
5	33	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	MSB Variable 0
7	X	1	Hexwert	LSB Variable 0
8	X	1	Hexwert	MSB Variable 1
9	X	1	Hexwert	LSB Variable 1
10	X	1	Hexwert	MSB Variable 2
11	X	1	Hexwert	LSB Variable 2
12	X	1	Hexwert	MSB Variable 3
13	X	1	Hexwert	LSB Variable 3
14	X	1	Hexwert	MSB Variable 4
15	X	1	Hexwert	LSB Variable 4
16	X	1	Hexwert	MSB Variable 5
17	X	1	Hexwert	LSB Variable 5
18	X	1	Hexwert	MSB Variable 6
19	X	1	Hexwert	LSB Variable 6
20	X	1	Hexwert	MSB Variable 7
21	X	1	Hexwert	LSB Variable 7
22	X	1	Hexwert	MSB Variable 8
23	X	1	Hexwert	LSB Variable 8
23	X	1	Hexwert	MSB Variable 9
25	X	1	Hexwert	LSB Variable 9
26	X	1	Hexwert	MSB Variable 10
27	X	1	Hexwert	LSB Variable 10
28	X	1	Hexwert	MSB Variable 11
29	X	1	Hexwert	LSB Variable 11
30	X	1	Hexwert	MSB Variable 12
31	X	1	Hexwert	LSB Variable 12
32	X	1	Hexwert	MSB Variable 13
33	X	1	Hexwert	LSB Variable 13
34	X	1	Hexwert	MSB Variable 14
35	X	1	Hexwert	LSB Variable 14
36	X	1	Hexwert	MSB Variable 15
37	X	1	Hexwert	LSB Variable 15
38	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Wird unterstützt von:
Controller C-2560

Request CD

Datum und Uhrzeit stellen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	C	1	ASCII	Paket-Type
4	D	1	ASCII	Paket-Sub-Type
5	8	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Tag (1-31)
7	x	1	Hexwert	Monat (1-12)
8	x	1	Hexwert	Jahr 2stellig (08=2008)
9	x	1	Hexwert	Wochentag (0-6, 0=Sonntag)
10	x	1	Hexwert	Stunde (0-23)
11	x	1	Hexwert	Minute (0-59)
12	x	1	Hexwert	Sekunde (0-59)
13	x	1	Hexwert	DST (1=Sommer, 0=Winterzeit)
14	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando stellt das Datum und die Uhrzeit und Sommer/Winterzeit im adressierten Controller. Die übertragenen Daten müssen vor dem Absenden validiert werden.

Beispiel:

Wird unterstützt von:

Buskoppler, C-2560, Funkhub

Request G1

Statusdaten anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	G	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	\$00
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert das angesprochene Modul auf, einen gesamten Staus zu schicken. Die Interpretation kann in den einzelnen Modulen unterschiedlich sein.

Beispiel:

Wird unterstützt von:

Buskoppler, C-2560

Request G2

Konfigurationstabellen anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	G	1	ASCII	Paket-Type
4	2	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Bestimmt die Tabelle
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert das angesprochene Modul auf, die in Byte 5 adressierte Konfigurationstabelle auf dem Debug-Kanal auszugeben.

\$01 Schaltuhrentabelle
unfertig

Beispiel:

Wird unterstützt von:
Controller C-2560

Request G3

Sensorwert anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	A	1	ASCII	Broadcast-Adresse als Empfänger
3	G	1	ASCII	Paket-Type
4	3	1	ASCII	Paket-Sub-Type
5	5	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	ASCII	Sensorname, 1. Zeichen
7	x	1	ASCII	Sensorname, 2. Zeichen
8	x	1	ASCII	Sensorname, 3. Zeichen
9	x	1	ASCII	Sensorname, 4. Zeichen
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert sendet an die Broadcast-Adresse die Anforderung an alle Module, den Messwert des im Byte 6 bis 9 zu senden. Es ist dafür zu sorgen, dass dieser Sensorname nur bei einem Modul bekannt ist. Dieses Modul wird anschließend den Messwert zurücksenden (Command s3)

Beispiel:

Wird unterstützt von:

Controller C-2560, Buskoppler SHT75

Request G4

Fühlerwert anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	A	1	ASCII	Broadcast-Adresse als Empfänger
3	G	1	ASCII	Paket-Type
4	4	1	ASCII	Paket-Sub-Type
5	\$2	1	Hexwert	Länge der Payload plus Checksumme
6	x	4	ASCII	Fühlername
10	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando kann an einen speziellen Controller oder per Broadcast (Adresse A) an alle Controller geschickt werden.

Der Controller, der den im Byte 6 bis 9 genannten Messfühler besitzt, wird auf den Request antworten. Alle anderen Controller schicken keine Antwort.

Beispiel:

Wird unterstützt von:

Controller C-2560, Buskoppler SHT75

Request G5

Firmware-Version anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	e	1	ASCII	Adresse des Controllers
3	G	1	ASCII	Paket-Type (kleines s)
4	5	1	ASCII	Paket-Sub-Type
5	8	2	Hexwert	Länge der Payload plus Checksumme
6	0	1	Hexwert	Immer 0x0
7	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Die aktuelle Programmversion vom Controller anfordern

Beispiel:

Wird unterstützt von:
Controller C-2560

Request G6

Boot-Timestamp anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	e	1	ASCII	Adresse des Controllers
3	G	1	ASCII	Paket-Type (kleines s)
4	6	1	ASCII	Paket-Sub-Type
5	8	2	Hexwert	Länge der Payload plus Checksumme
6	0	1	Hexwert	Immer 0x0
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Beim Start liest die Firmware Datum und Uhrzeit aus der Hardwareuhr und speichert die Informationen. Dieses Paket fordert diese Informationen vom adressierten Controller an.

Beispiel:

Wird unterstützt von:

Controller C-2560, FunkHub

Request G7

Inhalt der Speichervariablen anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	e	1	ASCII	Adresse des Controllers
3	G	1	ASCII	Paket-Type (kleines s)
4	7	1	ASCII	Paket-Sub-Type
5	8	2	Hexwert	Länge der Payload plus Checksumme
6	0	1	Hexwert	Immer 0x0
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Die Firmware des Controller C2560 besitzt 16 2-Byte große Speichervariablen. Diese Variablen werden von der Firmware zur Steuerung benutzt und auch verändert. Dieser Request fordert den adressierten Controller auf, den Inhalt dieser Variablen über den Bus zu senden.

Beispiel:

Wird unterstützt von:
Controller C-2560

Request G8

registrierte Funksensoren vom Funkhub anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	X	1	ASCII	Adresse des Funk-Hubs
3	G	1	ASCII	Paket-Type
4	8	1	ASCII	Paket-Sub-Type
5	X	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Immer \$1
7	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert die registrierten Funksensoren einschl. der Statistikzahlen vom Funkhub an. Der Hub antwortet mit dem Response ‚s8‘.

Beispiel:

Wird unterstützt von:

Funkhub

Request G9

empfangene Fühlerwerte vom Funkhub anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	X	1	ASCII	Adresse des Funk-Hubs
3	G	1	ASCII	Paket-Type
4	9	1	ASCII	Paket-Sub-Type
5	X	1	Hexwert	Länge der Payload plus Checksumme
6	a	1	ASCII	Sensorname (hier a oder * für alle)
7	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert die registrierten den Funksensor auf, die empfangenen Fühlerwerte einschl. der Namen zu schicken. Der Hub antwortet mit ‚s9‘.

Beispiel:

Wird unterstützt von:

Funkhub

Request Ga

Datum und Uhrzeit anfordern

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	e	1	ASCII	Adresse des Controllers
3	G	1	ASCII	Paket-Type (kleines s)
4	a	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	0	1	Hexwert	Immer 0x0
7	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Anfordern der aktuellen Uhrzeit und des Datums vom Zeitserver. Es wird der Controller antworten, der in den Konfigurationsdaten als Zeitserver gekennzeichnet ist.

Beispiel:

Wird unterstützt von:

Controller C-2560 (Zeitserver), Funkhub

Request Gb

Anfordern der aktuellen Dimmer-Helligkeiten

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	G	1	ASCII	Paket-Type
4	b	1	ASCII	Paket-Sub-Type
5	x	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Beliebig, vorzugsweise \$00
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando fordert den Dimmer auf, die aktuellen Helligkeitswerte aller 8 Kanäle zu übertragen

Beispiel:

Wird unterstützt von:

Dimmer als Empfänger

Request S1

Schalten Szene

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	S	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type
5	4	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Szenen-ID (0-254)
7	x	1	Hexwert	Ausgabewert für die gesamte Szene
8	x	1	Hexwert	Dauer. Ist sie > 0, überschreibt sie die in den Aktoren definierte Dauer
9	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando werden alle in der adressierten Szene hinterlegten Aktoren geschaltet. Die Interpretation des Wertes kann in den einzelnen Aktoren unterschiedlich sein. Bitte beachten Sie hier die Funktionsbeschreibung der Aktoren.

Beispiel:

Wird unterstützt von:

Dimmer, Buskoppler, C-2560

Request S2

Schalten Aktor

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	S	1	ASCII	Paket-Type
4	2	1	ASCII	Paket-Sub-Type
5	4	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Szenen-ID (0-254)
7	x	1	Hexwert	Ausgabewert für den Aktor
8	x	1	Hexwert	Dauer. Ist sie > 0, überschreibt sie die im Aktore definierte Dauer
9	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando wird der adressierte Aktoren geschaltet. Die Interpretation des Wertes kann in den einzelnen Aktoren unterschiedlich sein. Bitte beachten Sie hier die Funktionsbeschreibung der Aktoren.

Beispiel:

Wird unterstützt von:

Dimmer, Buskoppler, CPU als Empfänger

Request S3

Starten des automatischen Aufdimmens

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	S	1	ASCII	Paket-Type
4	3	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Beliebig, vorzugsweise \$00
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando startet das automatische Erhöhen der Helligkeit im Dimmer. Der Wert beginnt bei 0.

Beispiel:

Wird unterstützt von:

Dimmer als Empfänger

Request S4

Stoppen des automatischen Aufdimmens

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	S	1	ASCII	Paket-Type
4	4	1	ASCII	Paket-Sub-Type
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Beliebig, vorzugsweise \$00
7	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando stoppt das automatische Erhöhen der Helligkeit im Dimmer. Der Dimmer behält seinen aktuellen Wert.

Beispiel:

Wird unterstützt von:

Dimmer als Empfänger

Request S5

Einstellen der Dimmerhelligkeit

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	S	1	ASCII	Paket-Type
4	5	1	ASCII	Paket-Sub-Type
5	X	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Dimmerkanal (0-7)
7	X	1	Hexwert	Helligkeit (0-100)
8	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Kommando setzt in dem angegebenen Dimmerkanal den geforderten Helligkeitswert.

Beispiel:

Wird unterstützt von:

Dimmer als Empfänger

Request SR

Softreset ausführen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	E	1	ASCII	Adresse des Controllers
3	S	1	ASCII	Paket-Type
4	R	1	ASCII	Paket-Sub-Type (kleines r)
5	2	1	Hexwert	Länge der Payload plus Checksumme
6	1	1	Hexwert	Dummy, hier 0x1
7	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Durch dieses Kommando wird der Controller in einen Endlosloop geschickt. Der Watchdogtimer wird nach 2 Sekunden einen Reset ausführen.

Beispiel:

Wird unterstützt von:
Controller C-2560

Response s1

Stausdaten des Controller C-2560

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	s	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type
5	x	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Angeschlossene 7311-Module. Bit = 1 bedeutet angeschlossen, 0=nc. Bit 0 steht für Adresse 0, Bit 1 für Adresse 1 usw.
7	x	1	Hexwert	Wert Output.Modul 0, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
8	x	1	Hexwert	Wert Output.Modul 0, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
9	x	1	Hexwert	Wert Output.Modul 1, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
10	x	1	Hexwert	Wert Output.Modul 1, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
11	x	1	Hexwert	Wert Output.Modul 2, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
12	x	1	Hexwert	Wert Output.Modul 2, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
13	x	1	Hexwert	Wert Output.Modul 3, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
14	x	1	Hexwert	Wert Output.Modul 3, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
15	x	1	Hexwert	Wert Output.Modul 4, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
16	x	1	Hexwert	Wert Output.Modul 4, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
17	x	1	Hexwert	Wert Output.Modul 5, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
18	x	1	Hexwert	Wert Output.Modul 5, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
19	x	1	Hexwert	Wert Output.Modul 6, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
20	x	1	Hexwert	Wert Output.Modul 6, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
21	x	1	Hexwert	Wert Output.Modul 7, Port 0 (0-255) wenn nicht angeschlossen wird \$00 geschickt
22	x	1	Hexwert	Wert Output.Modul 7, Port 1 (0-255) wenn nicht angeschlossen wird \$00 geschickt
23	x	1	Hexwert	Temperatur auf der Trägerplatine
24	x	1	Hexwert	Frei, zur Zeit 0
25	x	1	Hexwert	Frei, zur Zeit 0

26	x	1	Hexwert	Checksumme
----	---	---	---------	------------

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando sendet der Controller C-2560 den aktuellen Status. Dieser Status wird im Laufe der Entwicklung noch erweitert.

Beispiel:

Wird unterstützt von:

Buskoppler, CPU

Response s3

Sensorwert

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	A	1	ASCII	Broadcast-Adresse als Empfänger
3	s	1	ASCII	Paket-Type (kleines s)
4	3	1	ASCII	Paket-Sub-Type
5	8	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Alter des Messwertes
7	X	1	Hexwert	Messwert, MSB
8	X	1	Hexwert	Messwert, LSB
9	X	1	ASCII	Sensorname, 1. Zeichen
10	X	1	ASCII	Sensorname, 2. Zeichen
11	X	1	ASCII	Sensorname, 3. Zeichen
12	X	1	ASCII	Sensorname, 4. Zeichen
13	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dies ist die Antwort auf den Request G3 (Sensorwert anfordern)

Der Response wird von dem Modul geschickt, dem der Sensor bekannt ist. Dabei wird als Empfänger der Absender des Requests benutzt.

Beispiel:

Wird unterstützt von:

Controller C-2560, Buskoppler SHT75

Response s4

SensorNamen und -Werte

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	X	1	ASCII	Empfängeradresse
3	s	1	ASCII	Paket-Type (kleines s)
4	4	1	ASCII	Paket-Sub-Type
5	X	1	Hexwert	Länge der Payload plus Checksumme
6-9	X	4	ASCI	Name des Sensors
8	X	1	Hexwert	Messwert, MSB
9	X	1	Hexwert	Messwert, LSB
10	X	1	Hexwert	Alter des Messwertes in Minuten (max 255)
11	X	1	ASCII	- Bindestrich
n	X	x	x	Nächster Sensor
n	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dies ist die Antwort auf den Request G3 (Sensorwerte anfordern)
Alle in der SensorTab des Controllers gespeicherten Sensornamen und -Werte werden geschickt.

Beispiel:

Wird unterstützt von:

Controller C-2560

Response s5

Firmware-Version schicken

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	Z	1	ASCII	Absender des Paketes, hier ‚z‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	S	1	ASCII	Paket-Type (kleines s)
4	5	1	ASCII	Paket-Sub-Type
5	X	x	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Die aktuelle Programmversion des Controllers wird übertragen

Beispiel:

Wird unterstützt von:
Controller C-2560

Response s6

Fehlerstatus schicken

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	Z	1	ASCII	Absender des Paketes, hier ‚z‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	s	1	ASCII	Paket-Type (kleines s)
4	6	1	ASCII	Paket-Sub-Type
5	3	x	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	MSB des Fehlerstatus
7	x	1	Hexwert	LSB des Fehlerstatus
8	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Den aktuellen Fehlerstatus nach dem Speichern von Konfigurationsdaten an den Absender des Requests schicken

Beispiel:

Wird unterstützt von:

Controller C-2560

Response s7

Boot-Timestamp schicken

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	Z	1	ASCII	Absender des Paketes, hier ‚z‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	s	1	ASCII	Paket-Type (kleines s)
4	7	1	ASCII	Paket-Sub-Type
5	7	1	Hexwert	Länge der Payload plus Checksumme
6	X	1	Hexwert	Jahr, 2stellig
7	X	1	Hexwert	Monat
8	X	1	Hexwert	Tag
9	X	1	Hexwert	Stunde
10	X	1	Hexwert	Minute
11	X	1	Hexwert	Sekunde
12	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Beim Start liest die Firmware Datum und Uhrzeit aus der Hardwareuhr und speichert die Informationen. Dieses Response-Paket schickt die Information an den Absender des Request-Pakets.

Beispiel:

Wird unterstützt von:

Controller C-2560, FunkHub

Response s8

Speichervariablen des Controllers

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	Z	1	ASCII	Absender des Paketes, hier ‚z‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	s	1	ASCII	Paket-Type (kleines s)
4	8	1	ASCII	Paket-Sub-Type
5	33	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	MSB Variable 0
7	x	1	Hexwert	LSB Variable 0
8	x	1	Hexwert	MSB Variable 1
9	x	1	Hexwert	LSB Variable 1
10	x	1	Hexwert	MSB Variable 2
11	x	1	Hexwert	LSB Variable 2
12	x	1	Hexwert	MSB Variable 3
13	x	1	Hexwert	LSB Variable 3
14	x	1	Hexwert	MSB Variable 4
15	x	1	Hexwert	LSB Variable 4
16	x	1	Hexwert	MSB Variable 5
17	x	1	Hexwert	LSB Variable 5
18	x	1	Hexwert	MSB Variable 6
19	x	1	Hexwert	LSB Variable 6
20	x	1	Hexwert	MSB Variable 7
21	x	1	Hexwert	LSB Variable 7
22	x	1	Hexwert	MSB Variable 8
23	x	1	Hexwert	LSB Variable 8
23	x	1	Hexwert	MSB Variable 9
25	x	1	Hexwert	LSB Variable 9
26	X	1	Hexwert	MSB Variable 10
27	X	1	Hexwert	LSB Variable 10
28	X	1	Hexwert	MSB Variable 11
29	X	1	Hexwert	LSB Variable 11
30	X	1	Hexwert	MSB Variable 12
31	X	1	Hexwert	LSB Variable 12
32	X	1	Hexwert	MSB Variable 13
33	X	1	Hexwert	LSB Variable 13
34	X	1	Hexwert	MSB Variable 14
35	X	1	Hexwert	LSB Variable 14
36	X	1	Hexwert	MSB Variable 15
37	X	1	Hexwert	LSB Variable 15
38	X	1	Hexwert	Checksumme

Wird unterstützt von: Controller C-2560

Response s9

Registrierte Fühlerwerte schicken

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	H	1	ASCII	Absender des Paketes, hier ‚H‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	s	1	ASCII	Paket-Type (kleines s)
4	9	1	ASCII	Paket-Sub-Type
5	X	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Anzahl der folgenden Fühlerdaten
7	X	1	Hexwert	Name des Funksensors (z. B. a)
8-11	X	4	ASCII	Name des Fühlers, max. 4 Stellen
12-13	X	2	16Bitwert	Zuletzt empfangener Wert
14-15	X	2	16Bitwert	Alter des Wertes in Sekunden
16	#	1	ASCII	Trennzeichen
17				Beginn des nächsten Sensors
...	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Der Funkhub schickt die angeforderten Fühlerwerte an den Anforderer zurück. Im Request kann der Name des Funksensors oder ein Stern * vorgegeben werden.

Beispiel:

Wird unterstützt von:

Funkhub

Response sa

Registrierte Remote-Device-Daten schicken

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	Y	1	ASCII	Start of Text, kleines y
1	H	1	ASCII	Absender des Paketes, hier ‚H‘
2	9	1	Hexwert	Empfängeradresse hier ‚9‘
3	s	1	ASCII	Paket-Type (kleines s)
4	a	1	ASCII	Paket-Sub-Type
5	x	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Anzahl der Funksensoren
7	X	1	Hexwert	Name des Funksensors (z. B. a)
8	X	1	Hexwert	Tag
9	X	1	Hexwert	Monat
10	X	1	Hexwert	Jahr
11	X	1	Hexwert	Sekunden
12	X	1	Hexwert	Minuten
13	X	1	Hexwert	Stunden
14-15	X	2	16Bitwert	Sekunden seit letztem Kontakt
16-17	X	2	16Bitwert	Anzahl der ok-Pakete
18-19	X	2	16Bitwert	Anzahl der CRC-Fehler
20-21	X	2	16Bitwert	Anzahl der Pakete mit Retry-Counter 0
22-23	X	2	16Bitwert	Anzahl der Pakete mit Retry-Counter > 0
24-25	X	2	16Bitwert	Anzahl der verlorenen Pakete
26	#	1	ASCII	Trennzeichen
27				Beginn des nächsten Sensors
...	X	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Der Funkhub schickt die aktuellen Informationen zu den registrierten Funksensoren an den Anforderer

Beispiel:

Wird unterstützt von:

Funkhub

Response sc

aktuelle Dimmer-Helligkeiten

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	9	1	ASCII	Absender des Paketes, hier 9
2	B	1	ASCII	Empfänger des Pakets, hier B
3	s	1	ASCII	Paket-Type
4	c	1	ASCII	Paket-Sub-Type
5	9	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Helligkeit Dimmerkanal 0
7	x	1	Hexwert	Helligkeit Dimmerkanal 1
8	x	1	Hexwert	Helligkeit Dimmerkanal 2
9	x	1	Hexwert	Helligkeit Dimmerkanal 3
10	x	1	Hexwert	Helligkeit Dimmerkanal 4
11	x	1	Hexwert	Helligkeit Dimmerkanal 5
12	x	1	Hexwert	Helligkeit Dimmerkanal 6
13	x	1	Hexwert	Helligkeit Dimmerkanal 7
14	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Mit diesem Kommando sendet der Dimmer die eingestellten Helligkeitswerte an den Anforderer.

Beispiel:

Wird unterstützt von:

Dimmer als Absender

Funkprotokolle

Request r1

Schaltbefehl an Funkaktor

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	H	1	ASCII	Absender des Paketes, hier H
2	b	1	ASCII	Adresse des Funkaktors
3	r	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type (kleines r)
5	6	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Paketnummer (Zufallszahl)
7	x	1	Hexwert	Retrycounter (0 beim ersten Paket)
8	x	1	Hexwert	Bitnummer (0 bis 7)
9	x	1	Hexwert	Gewünschter Zustand des Aktors
10	x	1	Hexwert	Gewünschte Einschaltdauer in 2Sekunden (0=immer, 1=zwei Sekunde)
11	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Funkpaket schaltet einen Funkaktor

Beispiel:

Wird unterstützt von:

Controller Funkhub, Funkaktor

Response s1

Acknowledge (pos. und negativ)

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	b	1	ASCII	Absender des Paketes, hier b
2	H	1	ASCII	Absender des Requests als Empfänger
3	S	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type (kleines r)
5	5	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Paketnummer des Requests
7	x	1	Hexwert	Retry-Counter, meist 0
8	x	1	Hexwert	0=Ack, ff=nak
9	x	1	Hexwert	Verschiedener Inhalt. Der Funkator sendet hier den aktuellen Zustand des Ausgabeports
10	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Es handelt sich um ein positive (Byte8=0) oder negatives Acknowledge (Byte8=0xff).

Beispiel:

Wird unterstützt von:

Controller Funkhub, Funkaktor

Response A1

alle Fehlerwerte an Hub übertragen

[Zurück zum Inhalt](#)

Byte	Daten	Länge	Format/Wert	Anmerkung
0	y	1	ASCII	Start of Text, kleines y
1	H	1	ASCII	Absender des Paketes, hier H
2	b	1	ASCII	Adresse des Funkaktors
3	A	1	ASCII	Paket-Type
4	1	1	ASCII	Paket-Sub-Type
5	x	1	Hexwert	Länge der Payload plus Checksumme
6	x	1	Hexwert	Retry-Counter
7	x	1	ASCII	Fühlername, 1 Stelle
8	x	1	ASCII	Fühlername, 2 Stelle
9	x	1	ASCII	Fühlername, 3 Stelle
10	x	1	ASCII	Fühlername, 4 Stelle
11	x	1	ASCII	Aktueller Wert, 1. Stelle
12	x	1	ASCII	Aktueller Wert, 2. Stelle
13	x	1	ASCII	Aktueller Wert, 3. Stelle
14	x	1	ASCII	Aktueller Wert, 4. Stelle
...				Nächster Fühler und Wert
x	x	1	Hexwert	Checksumme

x = beliebig bzw. unterschiedlich

Aufgabe

Dieses Funkpaket sendet alle Fühlernamen und deren Werte an den Hub. Die Werte sind in 10tel des Messwertes und werden 4stellig in ASCII übertragen. Beispiel für 3,1 Grad → 0031. Die Darstellung in 10tel vermeidet Dezimalstellen.

Beispiel:

Wird unterstützt von:

Controller Funkhub, Funkaktor

