

**kHome Protokollspezifikation**

Version 0.1

Peter Kappelt

letzte Änderung: 05.03.2017 18:30

# Inhalt

[1 Inhalt 2](#_Toc476436995)

[2 Aufbau eines kHome-Telegramms 3](#_Toc476436996)

[2.1 Standardtelegramm 3](#_Toc476436997)

[2.1.1 Protokolltyp 3](#_Toc476436998)

[2.1.2 Telegrammtyp 3](#_Toc476436999)

[2.1.2.1 REG\_W 3](#_Toc476437000)

[2.1.2.2 REG\_R 4](#_Toc476437001)

[2.1.2.3 REG\_B 4](#_Toc476437002)

[2.1.2.4 CNF\_W 5](#_Toc476437003)

[2.1.2.5 CNF\_R 5](#_Toc476437004)

[2.1.2.6 STS\_R 6](#_Toc476437005)

[2.1.2.7 ANS 6](#_Toc476437006)

[2.1.3 Adresse des Senders 7](#_Toc476437007)

[2.1.4 Adresse des Empfängers 7](#_Toc476437008)

[2.1.5 Nutzdaten 7](#_Toc476437009)

[2.1.6 CRC 8](#_Toc476437010)

[3 Versionsgeschichte 9](#_Toc476437011)

# Aufbau eines kHome-Telegramms

Ein kHome-Telegramm kann über verschiedene Protokolle und Hardwareschichten übertragen werden. Das bedeutet, dass der folgende Aufbau für den innersten Kern eines Telegramms steht. Je nach verwendetem Übertragungsmedium können neben aufgezeigtem Telegramm noch Präfixe und Suffixe folgen.

## Standardtelegramm

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 |  | 5+(n–1) | 6+(n–1) |
| Protokolltyp | Telegrammtyp | Adresse des Senders | Adresse des Empfängers | [Länge der Nutzdaten](#_Nutzdaten) | [Nutzdaten  Byte 0](#_Nutzdaten) | … | [Nutzdaten Byte n – 1](#_Nutzdaten) | CRC |

### Protokolltyp

Das Protokolltyp-Byte informiert über die Spezifikation, der die nachfolgenden Daten folgen. Momentan muss an dieser Stelle immer 0x01 übertragen werden.

Es dient für künftige Versionen des Protokolls, welche größere Änderungen im Telegrammaufbau einführen. Ein kHome-Gerät kann somit korrekt (mit einem Fehler) antworten, wenn ein Telegramm einer neueren Protokollspezifikation gesendet wird. Es wird nicht versucht ein Telegramm neueren Aufbaus zu interpretieren.

### Telegrammtyp

Der Telegrammtyp definiert das Kommando, welches ausgeführt werden soll. Neben den hier spezifizierten Telegrammtypen kann jedes kHome-Gerät auch eigene Telegrammtypen spezifizieren.

|  |  |  |
| --- | --- | --- |
| Wertigkeit | Typ | Kurz-Klartext |
| 0x01 | Schreiben in ein Datenregister | REG\_W |
| 0x02 | Lesen eines Datenregisters | REG\_R |
| 0x03 | Broadcast eines Datenregisters | REG\_B |
| 0x04 | Schreiben eines Konfigurationsbytes | CNF\_W |
| 0x05 | Lesen eines Konfigurationsbytes | CNF\_R |
| 0x06 | Lesen eines Statusbytes | STS\_R |
| 0x07 – 0x09 | Reserviert |  |
| 0x10 – 0x5F | Gerätespezifisch (falls nicht spezifiziert: reserviert) |  |
| 0x60 – 0xFE | Reserviert |  |
| 0xFF | Antwort auf ein Telegramm | ANS |

#### REG\_W

Schreiben eines Wertes in ein Register. Die Standard-Register enthalten Daten, die für die alltägliche Ausführung des Gerätes zuständig sind (Solltemperaturen, Schaltstatus, …). Ein Gerät kann bis zu 255 Register besitzen.

Je nach Geräteimplementierung können Register als Read-only definiert werden. Dies ist in der Gerätedokumentation hinterlegt.

Je nach Geräteimplementierung kann mit dem Schreiben eines Registers eine Funktion aufgerufen/ getriggert werden (z.B. Starten eines Sensor-Lesevorgangs, Schalten eines Ausgangs). Dies ist in der Gerätedokumentation hinterlegt.

Die Länge der Nutzdaten beträgt für einen REG\_W-Befehl immer zwei Byte. Sie werden folgendermaßen definiert:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4 | 5 | 6 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | Nutzdaten Byte 1 | … |
|  | 0x02 | Adresse des Registers | Wert, der in das Register geschrieben werden soll. |  |

Folgende Antworten sind möglich:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Weitere Daten in der Antwort |
| 0x00 | fehlerfrei | * #7: neuer Wert des Registers |
| 0xFC | der Wert, der in das Register geschrieben werden soll, ist ungültig | - |
| 0xFD | das CRC-Byte stimmte nicht mit dem Berechneten überein | - |
| 0xFE | Das Register ist als Read-only deklariert | - |
| 0xFF | Die Adresse des R ist unbekannt | - |

#### REG\_R

Auslesen eines Register-Wertes. Die Standard-Register enthalten Daten, die für die alltägliche Ausführung des Gerätes zuständig sind (Solltemperaturen, Schaltstatus, …). Ein Gerät kann bis zu 255 Register besitzen.

Je nach Geräteimplementierung können Register als Read-only definiert werden. Dies ist in der Gerätedokumentation hinterlegt.

Je nach Geräteimplementierung kann mit dem Lesen eines Registers eine Funktion aufgerufen/ getriggert werden (z.B. Starten eines Sensor-Lesevorgangs, Schalten eines Ausgangs). Dies ist in der Gerätedokumentation hinterlegt.

Die Länge der Nutzdaten beträgt für einen REG\_R-Befehl immer ein Byte. Sie werden folgendermaßen definiert:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 4 | 5 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | … |
|  | 0x01 | Adresse des Registers |  |

Folgende Antworten sind möglich:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Weitere Daten in der Antwort |
| 0x00 | fehlerfrei | * #7: Wert des Registers |
| 0xFD | das CRC-Byte stimmte nicht mit dem Berechneten überein | - |
| 0xFF | Die Adresse des Konfigurationsregisters ist unbekannt | - |

#### REG\_B

Globales Senden/ Broadcast eines Register-Wertes. Im Gegensatz zu anderen Befehlen werden mit diesem Befehl keine anderen Geräte angesprochen, sondern ein Register-Wert wird durch ein Gerät selbst auf das Netzwerk gesendet. Je nach Geräteimplementierung kann dies beispielsweise zeitbasiert oder durch Werteänderung erfolgen.

Folgende Besonderheiten beim Telegrammaufbau sind zu beachten:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |  |
| … | Telegrammtyp | Adresse des Senders | Adresse des Empfängers | Länge der Nutzdaten | Nutzdaten Byte 0 | Nutzdaten Byte 1 | … |
|  | 0x03 | Jeweilige Geräteadresse | 0xFF | 0x02 | Adresse des Registers | Wert des Registers |  |

Eine Antwort auf dieses Telegramm ist nicht nötig.

#### CNF\_W

Schreiben eines Wertes in ein Konfigurationsregister. Konfigurationsregister enthalten Daten die das Verhalten eines Gerätes verändern (Timeouts, automatisches Senden, (de)aktivieren von Gerätefunktionen). Ein Gerät kann mit bis zu 255 Konfigurationswerten parametrisiert werden.

Mit der kHome-Protokollspezifikation werden folgende Registeradressen normiert:

|  |  |
| --- | --- |
| Konfigurations-registeradresse | Bedeutung |
| 0x00 | Geräteadresse. Standardmäßig auf 0x00 gesetzt. |

Sonstige Registeradressen und deren Bedeutungen sind in der jeweiligen Gerätedokumentation beschrieben.

Die Länge der Nutzdaten beträgt für einen CNF\_W-Befehl immer zwei Byte:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4 | 5 | 6 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | Nutzdaten Byte 1 | … |
|  | 0x02 | Adresse des Konfigurations-registers | Wert, der in das Konfigurations-register geschrieben werden soll. |  |

Folgende Antworten sind möglich:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Weitere Daten in der Antwort |
| 0x00 | fehlerfrei | * #7: neuer Wert des Registers |
| 0xFC | der Wert, der in das Register geschrieben werden soll, ist ungültig | - |
| 0xFD | das CRC-Byte stimmte nicht mit dem Berechneten überein | - |
| 0xFE | Das Register ist als Read-only deklariert | - |
| 0xFF | Die Adresse des Konfigurationsregisters ist unbekannt | - |

#### CNF\_R

Lesen eines Wertes aus einem Konfigurationsregister. Konfigurationsregister enthalten Daten die das Verhalten eines Gerätes verändern (Timeouts, automatisches Senden, (de)aktivieren von Gerätefunktionen). Ein Gerät kann mit bis zu 255 Konfigurationswerten parametrisiert werden.

Hinweise zur Belegung der Registeradressen sind unter 2.1.2.3 zu finden.

Die Länge der Nutzdaten beträgt für einen CNF\_W-Befehl beträgt immer ein Byte:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 4 | 5 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | … |
|  | 0x02 | Adresse des Konfigurations-registers |  |

Folgende Antworten sind möglich:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Weitere Daten in der Antwort |
| 0x00 | fehlerfrei | * #7: Wert des Registers |
| 0xFD | das CRC-Byte stimmte nicht mit dem Berechneten überein | - |
| 0xFF | Die Adresse des Konfigurationsregisters ist unbekannt | - |

#### STS\_R

Statusbytes enthalten Informationen zum Ablaufstatus der Software der Geräte. Da sie nur Status enthalten und vom Gerät selbst gesetzt werden sind sie Read-only-Register.

Mithilfe der kHome-Protokollspezifikation sind folgende Statusbytes normiert:

|  |  |
| --- | --- |
| Statusbyte-adresse | Bedeutung |
| 0x00 | Globaler Ausführungsstatus. Soll dem Anwender eine grundlegende binäre Information zum Ausführungsstatus des Gerätes geben. (Im Sinne von „alles OK“ oder „Problem“).  Ein Wert von 0x00 steht für eine grundlegen problemlose Funktion.  Die anderen Werte sind in der jeweiligen Gerätedokumentation beschrieben, jedoch sollten alle Werte ungleich 0x00 für einen Fehler stehen. |
| 0x01 | Gerätetyp |

Der Gerätetyp wird folgendermaßen vergeben:

|  |  |
| --- | --- |
| Gerätetyp | Beschreibung |
| 0x01 | temperatureSensor V1 |

Sonstige Statusbytes sind in der jeweiligen Gerätedokumentation beschrieben.

Die Länge der Nutzdaten für einen STS\_R-Befehl beträgt immer genau ein Byte:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 4 | 5 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | … |
|  | 0x02 | Adresse des Statusregisters |  |

Folgende Antworten sind möglich:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Weitere Daten in der Antwort |
| 0x00 | fehlerfrei | * #7: Wert des Statusregisters |
| 0xFD | das CRC-Byte stimmte nicht mit dem Berechneten überein | - |
| 0xFF | Die Adresse des Statusregisters ist unbekannt | - |

#### ANS

Eine Antwort auf ein vorheriges Telegramm. Es muss vom ursprünglichen Telegrammsender auf ein Antworttelegramm gewartet werden, bevor ein neues gesendet wird.

Mit dem Antworttelegramm wird ein Status überliefert, der Auskunft über die Ausführung des im Telegramm spezifizierten Befehles gibt.

Die Länge der Nutzdaten beträgt mindestens zwei Byte:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 4 | 5 | 6 | 7 |  |
| … | Länge der Nutzdaten | Nutzdaten Byte 0 | Nutzdaten Byte 1 | Nutzdaten Byte 2 | … |
|  | mindestens 0x01 | Antwortcode | Typ des Telegramms, auf das geantwortet wird | spezifisch |  |

Folgende Antwortcodes sind spezifiziert:

|  |  |  |
| --- | --- | --- |
| Antwortcode | Beschreibung | Länge der Nutzdaten |
| 0x00 | OK. Befehl wurde problemlos ausgeführt | 2 + x  x ist abhängig vom Telegrammtyp, auf den geantwortet wird |
| 0x01 – 0x09 | Reserviert |  |
| 0x10 – 0x5F | Gerätespezifisch |  |
| 0x60 - | Reserviert |  |
| 0xFC | Fehler: Der Wert, der in das Register geschrieben wurde, ist ungültig. (siehe Gerätedokumentation) | 2 |
| 0xFD | Fehler: Das CRC-Byte stimmte nicht mit dem berechneten überein Telegrammbyte #6 (Typ des Telegramms, auf das geantwortet wird) ist in diesem Fall ebenfalls 0xFD | 2 |
| 0xFE | Fehler: Das Register ist als Read-only deklariert, es darf nur gelesen werden | 2 |
| 0xFF | Fehler: Die Adresse des (Konfigurations-, Status-) Registers ist unbekannt | 2 |

### Adresse des Senders

Der Sender kann Adressen von 1 bis 254 besitzen (inklusive 1 und 254) besitzen. Somit sind mit der aktuellen Protokollrevision bis zu 254 kHome-Geräte in einem Netzwerk möglich.

### Adresse des Empfängers

Zusätzlich zu den bei 1.1.3 beschriebenen Adressen kann die Empfängeradresse 255 betragen. Dabei handelt es sich um ein Broadcast-Paket, siehe 2.1.2.3

### Nutzdaten

Im Telegramm folgen die Länge der Nutzdaten und die Nutzdaten selbst. Es können 0 bis 255 Byte Nutzdaten übertragen werden. Jeweilige Anforderungen an die Nutzdaten und deren Bedeutung ist vom Telegrammtyp abhängig.

Zuerst wird das nullte Byte übertragen, als letzte das n-te Byte (wobei n = Länge der Nutzdaten – 1)

### CRC

Zur Verifizierung der korrekten Übertragung wird eine CRC-Wert über die Daten, vom Protokolltypbyte bis zum letzten Nutzdatenbyte.

Verwendet wird eine CRC-8-Wert mit dem Grundpolynom 0x07.

Informationen zur Berechnung, eine C#-Beispielimplementierung sowie ein Online-CRC-Rechner sind auf folgenden Webseiten zu finden:

<http://www.sunshine2k.de/articles/coding/crc/understanding_crc.html#ch44>

http://www.sunshine2k.de/coding/javascript/crc/crc\_js.html

# Versionsgeschichte

|  |  |  |
| --- | --- | --- |
| Datum | Version | Änderungen |
| 28. Februar 2017 | V0.1 | * Initiale Version |