

# eBUS

- [Documentation](#)
- [Vaillant](#)
- [Commands](#)
- [System Overview](#)

# Documentation

## Datenaufbau

Byte	Richtung	Abkürzung	Beschreibung
1	->	QQ	Quelladresse
2	->	ZZ	Zieladresse
3	->	PB	Primärbefehl
4	->	SB	Sekundärbefehl
5	->	NN	Zahl der folgenden Bytes
6 bis 5 + NN	->	Data	Datenbytes
6 + NN	->	CRC	Prüfziffer
7 + NN	<-	ACK	Positive Bestätigung des Empfängers (ab hier nicht bei Broadcast-Nachrichten)
8 + NN	<-	NN2	Datenlänge der Antwort
9 + NN bis 8 + NN + NN2	<-	Data	Daten vom Slave an den Master
9 + NN + NN2	<-	CRC	Prüfziffer
10 + NN + NN2	->	ACK	
11 + NN + NN2	->	SYN	Kennung, dass Bus wieder bereit ist für andere Teilnehmer

Service/Befehl => Kombination von Primär und Sekundärbefehl => Darstellung als HEX-Wert

z.B. Primärbefehl 07 (Systemdatenbefehle) + Sekundärbefehl 00 (Datum/Zeit Meldung eines Masters) = Service 0700

Primärbefehle b5 sind von Vaillant, also nicht per Standard definiert

## Kommunikation

Master-Slave Telegramme und Broadcast Telegramme

Jeder Master hat auch eine Slave Adresse (Slave = Master + 5)

Broadcast-Adresse ist FE

# CRC

Muss mit expandiertem Datenstring erfolgen (<https://ebus-wiki.org/doku.php/ebus/ebuscrc>)

```
////////////////////////////////////
//
// CRC-Berechnung aus http://www.mikrocontroller.net/topic/75698
//
////////////////////////////////////

#ifdef USE_CRC_TAB
const UCHAR CRC_Tab8Value[256]
"/*****
/** Function for CRC-calculation with tab operations */
*****/
UCHAR crc8(UCHAR data, UCHAR crc_init)
{
    UCHAR crc;

    crc = (UCHAR) (CRC_Tab8Value[crc_init] ^ data);
    return (crc);
}

#else

/*****/
/** slower, but less memory */
*****/
unsigned char crc8(unsigned char data, unsigned char crc_init)
{
    unsigned char crc;
    unsigned char polynom;
    int i;

    crc = crc_init;
    for (i = 0; i < 8; i++)
    {
```

```

    if (crc & 0x80)
    {
        polynom = (unsigned char) 0x9B;
    }
    else
    {
        polynom = (unsigned char) 0;
    }
    crc = (unsigned char)((crc & ~0x80) << 1);
    if (data & 0x80)
    {
        crc = (unsigned char)(crc | 1) ;
    }
    crc = (unsigned char)(crc ^ polynom);
    data = (unsigned char)(data << 1);
}
return (crc);
}
#endif

```

```

UCHAR CalculateCRC( UCHAR**Data, int DataLen )
{
    UCHAR Crc = 0;
    for( int i = 0 ; i < DataLen ; ++i, ++Data )
    {
        Crc = crc8(**Data, Crc );
    }
    return Crc;
}

```

# Vaillant

## Datenaufbau

Byte	Richtung	Abkürzung	Beschreibung
1	->	QQ	Quelladresse
2	->	ZZ	Zieladresse
3	->	PB	Primärbefehl
4	->	SB	Sekundärbefehl
5	->	NN	Zahl der folgenden Bytes
6 bis 5 + NN	->	Data	Datenbytes
6 + NN	->	CRC	Prüfziffer
7 + NN	<-	ACK	Positive Bestätigung des Empfängers (ab hier nicht bei Broadcast-Nachrichten)
8 + NN	<-	NN2	Datenlänge der Antwort
9 + NN bis 8 + NN + NN2	<-	Data	Daten vom Slave an den Master
9 + NN + NN2	<-	CRC	Prüfziffer
10 + NN + NN2	->	ACK	
11 + NN + NN2	->	SYN	Kennung, dass Bus wieder bereit ist für andere Teilnehmer

Service/Befehl => Kombination von Primär und Sekundärbefehl => Darstellung als HEX-Wert

z.B. Primärbefehl 07 (Systemdatenbefehle) + Sekundärbefehl 00 (Datum/Zeit Meldung eines Masters) = Service 0700

## Vaillant Addresses

### Master

Adress	Description
10h	Main Control Unit: * VRS620 (auroMATIC 620)
3F	Burner

Slave

Adress	Description
23h	
25h	
26h	Outside temperature sensor (including DCF77 clock)
50h	
ECh	

# Commands

## 1. Service 03h

- Name: Service Data Commands Burner Automats
- Data: information regarding operating time, start counts and fuel consumption

### 1.1 Service 03h 04h

- Name: Complete Reading of Start Counts
- Data: number of starts of a burner control unit

### 1.2 Service 03h 05h

- Name: Complete Operating Time, Reading Level 1
- Data: operating time counter (in case of multiple level burners, level 1) of a burner control unit

### 1.3 Service 03h 06h

- Name: Complete Operating Time, Reading Level 2
- Data: operating time counter level 2 of a burner control unit

### 1.4 Service 03h 07h

- Name: Complete Operating Time, Reading Level 3
- Data: operating time counter level 3 of a burner control unit

### 1.5 Service 03h 08h

- Name: Complete Reading Fuel Quantity Counter
- Data: fuel quantity counter of a burner control unit

### 1.5 Service 03h 10h

- Name: Read Meter Reading
- Data: chosen meter reading
- Parameters: Type of meter, Type of fuel (oil/gas)

## 2. Service 05h

- Name: Burner Control Commands
- Data: communication between control unit, room sensor or control components

## **2.1 Service 05h 00h**

- Name: Operational Requirements between Burner Control Unit and Room Controller
- Data: Start/Stop data requirement from room controller
- Parameters: Start or Stop transmission

## **2.2 Service 05h 01h**

- Name: Operational Data of Room Controller to Burner Control Unit
- Data: one-time/cyclic room controller operation data

## **2.3 Service 05h 02h**

- Name: Operational Data of Room Controller to Burner Control Unit
- Data: one-time/cyclic room controller operation data



# System Overview

## **Scan.08 HMU00**

Display in basement

## **Scan.15 CTLV3**

VR720/3?Main controller in kitchen

## **Scan.76 VWZIO**

VWL 75/6? aka arotherm plus heat pump

## **Scan.f6 NETX3**

myVaillant connect (white box)