

# Anleitung für die Einbindung der Zehnder ComfoAir 350 in IP-Symcon

## 1. Erstellen der I/O Instanz "Serial Port - ComfoAir 350)

### Instanzkonfiguration

[? DOKUMENTATION](#) | [INSTANZID KOPIEREN](#) | [INSTANZOBJEKT BEARBEITEN](#)

✓ Schnittstelle ist geöffnet

Seriellen Port öffnen

Port  
COM3

Gewöhnlich müssen diese Optionen nicht verändert werden

Baudrate  
9600

Datenbits  
8

Stopbits  
1

Parität  
Kein(e)

## 2. Erstellen Splitter Instanz "Cutter ComfoAir 350

### Instanzkonfiguration

[? DOKUMENTATION](#) | [SCHNITTSTELLE KONFIGURIEREN](#) | [SCHNITTSTELLE ÄNDERN](#)

Objektbaum: CUTTER COMFOAIR 350

Typ  
Benutze Zeichen für Schnitte

**Konfiguration (Zeichen für Schnitte)**

Linke Trennzeichen  
07 F3 07 F0  Hex

Rechtes Trennzeichen  
07 0F  Hex

**Konfiguration (Feste Schnitte)**

Eingabelänge  
0

Sync-Zeichen  Hex

## Benötigter Datenfluss:

**Serielle Schnittstelle → Cutter → Registervariable → RX-Lüftung.ips → ComfoAir.ips**

**Aus diesem Grund müssen daran anschließend folgenden Schritte durchgeführt werden:**

3. Anlegen der Kategorie "Lüftungsanlage" im Objektbaum
4. Anlegen der Register Variable "Lueftung"
5. Anlegen des Scripts RX-Lueftung

...

### **// Script RX-Lueftung.ips.php**

```
<?php
$debug=0;
Include('ComfoAir.ips.php'); // Funktionen für ComfoAir-Auslesung inkludieren
$IDRegVar=15168;

if ($_IPS['SENDER'] == "RegisterVariable") {
    $RawData = $_IPS['VALUE'];

    if ($debug){
        echo "\nAnkommende RohDaten:\n";
        for ($i = 0;
            $i < strlen($RawData); $i++) {
            printf("%02x ", ord($RawData[$i]));
        }
        echo "\n";
        echo "\nNutzdaten:\n";
        $RawData = str_replace("\x07\x07","\x07",$RawData);
        $rawDataArr = String2Arr($RawData);
        $dataArr = ExtractData($rawDataArr);
        print_r($dataArr);
    }

    if(startsWith($RawData,"\x00\x9C")) //RS232 Modus
    {
        Extract_C9C($RawData,$IDRegVar,"RS232 Modus",3);
    }
    else if(startsWith($RawData,"\x00\xCE")) //Ventilationsstufe abrufen
    {
        Extract_CCE($RawData,$IDRegVar,"Ventilationsstufen","Ventilatorstufe",3,1);
    }
    else if(startsWith($RawData,"\x00\xE0")) //Bypassregelung
    {

        Extract_CE0($RawData,$IDRegVar,"Bypassmodus","BypassStufe","Bypass",3,1,3);
    }
}
```

```

}
else if(startsWith($RawData,"\x00\xD2")) //Temperaturen abrufen
{
    Extract_CD2($RawData,$IDRegVar,"Temperaturen","Komfort-Temperatur",3,1 );
}
else if(startsWith($RawData,"\x00\x98")) //Sensordaten
{
    Extract_C98($RawData,$IDRegVar,"Sensordaten",3);
}
else if(startsWith($RawData,"\x00\xD6")) //Status abrufen
{
    Extract_CD5($RawData,$IDRegVar,"Status",3);
}
else if(startsWith($RawData,"\x00\xDA")) //Störungen und Filterstatus
abrufen
{
    Extract_CDA($RawData,$IDRegVar,"Fehler","Letzter Fehler","Filter",3);
}
else if(startsWith($RawData,"\x00\x68")) //Bootloader Version
{
    Extract_C68($RawData,$IDRegVar,"Bootloader Version",3);
}
else if(startsWith($RawData,"\x00\x6A")) //Abfrage des Gerättyps
{
    Extract_C6A($RawData,$IDRegVar,"Firmware Version",3);
}
else if(startsWith($RawData,"\x00\xA2")) //Abfrage Konnektorplatine
{
    Extract_CA2($RawData,$IDRegVar,"Konnektorplatine",3);
}
else if(startsWith($RawData,"\x00\x04")) //Eingänge abrufen
{
    Extract_C04($RawData,$IDRegVar,"Eingaenge","Badezimmerschalter",3,1);
}
else if(startsWith($RawData,"\x00\x0C")) //Ventilatorstatus, Drehzahl
{

Extract_C0C($RawData,$IDRegVar,"Ventilatorstatus","Ventilatorstufe","Zuluft%","Abluft%
","Zuluft U/Min","Abluft U/Min",1,3);
}
else if(startsWith($RawData,"\x00\x0E")) //Klappenstatus
{
    Extract_C0E($RawData,$IDRegVar,"Klappenstatus",3);
}
else if(startsWith($RawData,"\x00\x10")) //Temperaturstatus
{

```

```

Extract_C10($RawData,$IDRegVar,"Aussenlufttemperatur","Zulufttemperatur","Ablufttemperatur","Fortlufttemperatur",3);
}
else if(startsWith($RawData,"\x00\x12")) //Tastenstatus
{
    echo "WRL :: Tastenstatus :: ";
    // Extract_C12($RawData,31671); //????????????????????????????????????????
}
else if(startsWith($RawData,"\x00\x14")) //Analoge Eingänge
{
    echo "WRL :: Analoge Eingaenge :: ";
    Extract_C14($RawData,51809);
}
else if(startsWith($RawData,"\x00\xCA")) //Zeitverzögerung
{
    Extract_CCA($RawData,$IDRegVar,"Zeitverzoeigerung",3);
}
else if(startsWith($RawData,"\x00\xDE")) //Betriebsstunden
{
    Extract_CDE($RawData,$IDRegVar,"Betriebsstunden","Stunden Stufe 1","Stunden Stufe 2","Stunden Stufe 3","Stunden gesamt","Filter Countdown Tage",3);
}
}
...

```

**6. Eintragen des RX--Lueftung Scripts in der Register Variablen "Lueftung" als Ziel.**

**7. Anlegen des Scripts "Lueftung auslesen"**

...

**// Script Lueftung auslesen.ips.php**

```

<?php
Include('ComfoAir.ips.php'); // Funktionen für ComfoAir-Auslesung inkludieren
$IDRegVar=50765; //Die ID der Schnittstelle
    $sleep=100;

        SendCommand(array(0x00,0xD1),0,$IDRegVar); //Temperaturen
    IPS_Sleep ($sleep);
    SendCommand(array(0x00,0x0F),0,$IDRegVar); //Temperaturstatus
    IPS_Sleep ($sleep);
    SendCommand(array(0x00,0x0D),0,$IDRegVar); //Klappenstatus
    IPS_Sleep ($sleep);
    SendCommand(array(0x00,0xDD),0,$IDRegVar); //Betriebsstunden
    IPS_Sleep ($sleep);

```

```

SendCommand(array(0x00,0x03),0,$IDRegVar); //Eingaenge
IPS_Sleep ($sleep);
SendCommand(array(0x00,0xD9),0,$IDRegVar); //Stoerungen
IPS_Sleep ($sleep);
SendCommand(array(0x00,0xDF),0,$IDRegVar); //Bypassregelung
IPS_Sleep ($sleep);
SendCommand(array(0x00,0x03),0,$IDRegVar); //Eingaenge
IPS_Sleep ($sleep);
SendCommand(array(0x00,0xD5),0,$IDRegVar); //Status
IPS_Sleep ($sleep);
SendCommand(array(0x00,0xC9),0,$IDRegVar); //Zeitverzögerung
IPS_Sleep ($sleep);
SendCommand(array(0x00,0xCD),0,$IDRegVar); //Ventilationsstufen
IPS_Sleep ($sleep);
SendCommand(array(0x00,0x0B),0,$IDRegVar); //Ventilatorstatus
IPS_Sleep ($sleep);
SendCommand(array(0x00,0x67),0,$IDRegVar); //Bootloader Version
IPS_Sleep ($sleep);
SendCommand(array(0x00,0x69),0,$IDRegVar); //Firmware Version
IPS_Sleep ($sleep);
//SendCommand(array(0x00,0xA1),0,$IDRegVar); //Konnektor Platine

```

...

## 8. kopieren des Scriptes "ComfoAir.ips.php" in das Script Verzeichnis unter IP-Symcon

...

```
// Script ComfoAir.ips.php
```

<?

// Funktion ist nötig wenn Request via SeriellPort erfolgen soll

```

function SendCommand($byteArr,$dataArr,$SerialPortID) {
    $request = CreateRequest($byteArr,$dataArr);
    $requestStr="";
    foreach($request as $byte) {
        $requestStr = $requestStr . chr($byte);
    }
    COMPort_SendText($SerialPortID,$requestStr);
    return $requestStr;
}

```

/\*

// Funktion ist nötig wenn Request via LAN/TCP auf RS232- Adapter erfolgen soll

```

function SendCommandTCP($byteArr,$dataArr,$IDRegVar) {
    $request = CreateRequest($byteArr,$dataArr);
    $requestStr="";

```

```

        foreach($request as $byte) {
            $requestStr = $requestStr . chr($byte);
        }

        RegVar_SendText($IDRegVar,$requestStr);

        return $requestStr;
    }
*/

function CreateRequest($byteArr,$dataArr) {
    if($dataArr == null)
        array_push($byteArr,0);
    else {
        array_push($byteArr,dechex(count($dataArr)));
        $byteArr = array_merge($byteArr,$dataArr);
    }
    $crc = CalcCRC($byteArr);
    array_push($byteArr,$crc);
    $byteArr = array_merge(array(0x07,0xF0),$byteArr);
    $byteArr = array_merge($byteArr,array(0x07,0x0F));
    return $byteArr;
}

// Bootloader Version abrufen
function Extract_C68($rawDataStr,$id,$Bootloaderversion,$t1) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 13) {
        $Arr["Major"] = array_shift($dataArr);
        $Arr["Minor"] = array_shift($dataArr);
        $Arr["Beta"] = array_shift($dataArr);
        $Arr["Geraetename"] = Arr2String($dataArr);
    } else {
        $str = "Bootloader :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    //$Arr["RawData"] = String2Hex($rawDataStr);
    //$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    $varID=CreateVariableByName($id,$Bootloaderversion,$t1);
    SetValue($varID,$str);
    return $str;
}

// Firmware Version abrufen

```

```

function Extract_C6A($rawDataStr,$id,$Firmware,$t1) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 13) {
        $Arr["Major"] = array_shift($dataArr);
        $Arr["Minor"] = array_shift($dataArr);
        $Arr["Beta"] = array_shift($dataArr);
        $Arr["Geraetenname"] = Arr2String($dataArr);
    } else {
        $str = "Bootloader :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    // $Arr["RawData"] = String2Hex($rawDataStr);
    // $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    $varID=CreateVariableByName($id,$Firmware,$t1);
    SetValue($varID,$str);
    return $str;
}

```

// Konnektorplatine Version abrufen

```

function Extract_CA2($rawDataStr,$id,$Connectorplatine,$t1) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 14) {
        $Arr["Major"] = array_shift($dataArr);
        $Arr["Minor"] = array_shift($dataArr);
        $CCLuxeVer = array_pop($dataArr);
        $CCEaseVer = array_pop($dataArr);
        $Arr["Geraetenname"] = Arr2String($dataArr);

        $CCEaseMajor = ($CCEaseVer & bindec('11110000')) >> 4;
        $CCEaseMinor = $CCEaseVer & bindec('00001111');
        $CCLuxeMajor = ($CCLuxeVer & bindec('11110000')) >> 4;
        $CCLuxeMinor = $CCLuxeVer & bindec('00001111');

        $Arr["CC-Ease"] = $CCEaseMajor . "." . $CCEaseMinor;
        $Arr["CC-Luxe"] = $CCLuxeMajor . "." . $CCLuxeMinor;
    } else {
        $str = "Konnektorplatine :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
}

```

```

//$Arr["RawData"] = String2Hex($rawDataStr);
//$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varID=CreateVariableByName($id,$Connectorplatine,$t1);
SetValue($varID,$str);
return $str;
}

// Eingänge abrufen
function Extract_C04($rawDataStr,$id,$Eingaenge,$Badezimmerschalter,$t3,$t1) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 2) {
        $Schalteingaenge = array_pop($dataArr);
        $Stufenschalter = array_pop($dataArr);
        $Arr["Stufenschalter-L1"] = $Stufenschalter & bindec('00000001');
        $Arr["Stufenschalter-L2"] = ($Stufenschalter & bindec('00000010')) >> 1;
        $Arr["Schalteingang-Kueche"] = ($Schalteingaenge & bindec('00000010')) >>
1;
        $Arr["Schalteingang-Bad1"] = $Schalteingaenge & bindec('00000001');
        $Arr["Schalteingang-Bad2"] = ($Schalteingaenge & bindec('00010000')) >> 4;
        $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
        $varID=CreateVariableByName($id,$Badezimmerschalter,$t1);
        SetValue($varID,(int)$Arr["Schalteingang-Bad1"]);
    } else {
        $str = "Eingänge :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    // $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);

    $varID=CreateVariableByName($id,$Eingaenge,$t3);
    SetValue($varID,$str);
    return $str;
}

//Ventilatorstatus
function
Extract_C0C($rawDataStr,$id,$Ventilatorstatus,$Ventilatorstufe,$IDvHZuluft,$IDvHAbluft,$IDZZuluft,$IDZAbluft,$t1,$t3) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();

```



```

if(count($dataArr) == 6) {
    $Arr["Zuluft"] = array_shift($dataArr) ." %";
    $Arr["Abluft"] = array_shift($dataArr) ." %";
    $Arr["Drehzahl-Zuluft"] = round(1875000 / ((array_shift($dataArr) * 255 +
array_shift($dataArr) -4),2) . " U/Min";
    $Arr["Drehzahl-Abluft"] = round(1875000 / ((array_shift($dataArr) * 255 +
array_shift($dataArr) -4),2) . " U/Min";
    if (($Arr["Zuluft"]==0){$Arr["Drehzahl-Zuluft"]=0;}
    if (($Arr["Abluft"]==0){$Arr["Drehzahl-Abluft"]=0;}
    $IDvHZuluft=CreateVariableByName($id,$IDvHZuluft,$t1);
    $IDvHAbluft=CreateVariableByName($id,$IDvHAbluft,$t1);
    $IDDZZuluft=CreateVariableByName($id,$IDDZZuluft,$t1);
    $IDDZAbluft=CreateVariableByName($id,$IDDZAbluft,$t1);
    //$varIDstufe=CreateVariableByName($id,$Ventilatorstufe,$t1);

    //SetValue($varIDstufe, $AktiveStufe);
    SetValue($IDvHZuluft,(int)$Arr["Zuluft"]);
    SetValue($IDvHAbluft,(int)$Arr["Abluft"]);
    SetValue($IDDZZuluft,(int)$Arr["Drehzahl-
Zuluft"]);
    SetValue($IDDZAbluft,(int)$Arr["Drehzahl-
Abluft"]);

    //if ((int)$Arr["Zuluft"]<=5)
    SetValue($varIDstufe,0);
    //if ((int)$Arr["Zuluft"]>=10)
    SetValue($varIDstufe,1);
    //if ((int)$Arr["Zuluft"]>=30)
    SetValue($varIDstufe,2);
    //if ((int)$Arr["Zuluft"]>=55)
    SetValue($varIDstufe,3);
    } else {
        $str = "Eingaenge :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    //$Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    $varID=CreateVariableByName($id,$Ventilatorstatus,$t3);
    SetValue($varID,$str);
    return $str;
}

//Klappenstatus
function Extract_C0E($rawDataStr,$id,$klappenstatus,$t3) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);

```

```

$Arr = array();
if(count($dataArr) == 4) {
    $BypassInProzent = array_shift($dataArr) . "%";
    $Arr["Bypass"] = $BypassInProzent;
    $vorheizung = array_shift($dataArr);
    $Arr["Vorheizung"] = $vorheizung . " ";
    switch($vorheizung) {
        case 0:
            $Arr["Vorheizung"] .= " - Offen";
            break;
        case 1:
            $Arr["Vorheizung"] .= " - Zu";
            break;
        case 2:
            $Arr["Vorheizung"] .= " - Unbekannt";
            break;
    }
    $Arr["Motorstrom-Bypass"] = array_shift($dataArr);
    $Arr["Motorstrom-Vorheizung"] = array_shift($dataArr);
} else {
    $str = "Klappenstatus :: no valid Data received";
    $Arr["ERROR"] = $str;
    LogMsg(__FUNCTION__, $str);
}
// $Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s', time());
$str = Array2HtmlTable($Arr);
$varID = CreateVariableByName($id, $klappenstatus, $t3);
SetValue($varID, $str);
return $str;
}

```

//Temperaturstatus abrufen

```

function Extract_C10($rawDataStr, $id, $aussen, $zu, $ab, $fort, $t3) {
    $rawDataStr = str_replace("\x07\x07", "\x07", $rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    if(count($dataArr) == 4) {
        $ID_T1 = CreateVariableByName($id, $aussen, 2);
        $ID_T2 = CreateVariableByName($id, $zu, 2);
        $ID_T3 = CreateVariableByName($id, $ab, 2);
        $ID_T4 = CreateVariableByName($id, $fort, 2);
        SetValueFloat($ID_T1, array_shift($dataArr) / 2 - 20);
        SetValueFloat($ID_T2, array_shift($dataArr) / 2 - 20);
        SetValueFloat($ID_T3, array_shift($dataArr) / 2 - 20);
        SetValueFloat($ID_T4, array_shift($dataArr) / 2 - 20);
    } else {

```

```

        LogMsg(__FUNCTION__,"Temperaturstatus :: more then 4 Data Bytes
received");
    }
    return $dataArr;
}

```

//Tastenstatus abrufen

```

function Extract_C12($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 1) {
        $tastenStatus = array_shift($dataArr);
        switch ($tastenStatus) {
            case 0:
                $Arr["Taste"] = "nichts gedrückt";
                break;
            case 255:
                $Arr["Taste"] = "Fehler";
                break;
            default:
                $Arr["Taste"] = $tastenStatus;
                break;
        }
    } else {
        $str = "Tastenstatus :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    // $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    SetValue($varID,$str);
    return $str;
}

```

//Analoge Eingänge abrufen

```

function Extract_C14($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 4) {
        $analog1 = array_shift($dataArr);
        $analog2 = array_shift($dataArr);
        $analog3 = array_shift($dataArr);
        $analog4 = array_shift($dataArr);
    }
}

```

```

    $Arr["Analog-1"] = $analog1 . " :: " . number_format($analog1*10/255,2) . "V";
    $Arr["Analog-2"] = $analog2 . " :: " . number_format($analog2*10/255,2) . "V";
    $Arr["Analog-3"] = $analog3 . " :: " . number_format($analog3*10/255,2) . "V";
    $Arr["Analog-4"] = $analog4 . " :: " . number_format($analog4*10/255,2) . "V";
} else {
    $str = "Analoge Eingaenge :: no valid Data received";
    $Arr["ERROR"] = $str;
    LogMsg(__FUNCTION__, $str);
}
//$Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
SetValue($varID, $str);
return $str;
}

```

//Sensordaten abrufen

```

function Extract_C98($rawDataStr, $id, $Sensordaten, $t1) {
    $rawDataStr = str_replace("\x07\x07", "\x07", $rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 17) {
        $Arr["EnthalpieSensor-Temperatur"] = array_shift($dataArr) / 2 - 20 . " °C";
        $Arr["EnthalpieSensor-Feuchtigkeit"] = array_shift($dataArr) . "%";
        $Arr["Analog-1"] = array_shift($dataArr) . "%";
        $Arr["Analog-2"] = array_shift($dataArr) . "%";
        $Arr["Enthalpie-Koeffizient"] = array_shift($dataArr) . "%";
        $Arr["Enthalpie-Time"] = array_shift($dataArr) * 12 . " min";
        $Arr["0x00"] = array_shift($dataArr);
        $Arr["Analog-1_ZuGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-1_AbGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-2_ZuGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-2_AbGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-3"] = array_shift($dataArr) . "%";
        $Arr["Analog-4"] = array_shift($dataArr) . "%";
        $Arr["Analog-3_ZuGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-3_AbGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-4_ZuGewuenscht"] = array_shift($dataArr) . "%";
        $Arr["Analog-4_AbGewuenscht"] = array_shift($dataArr) . "%";
    } else {
        $str = "Sensordaten :: no valid Data received";
        $Arr["ERROR"] = $str;
        //LogMsg(__FUNCTION__, $str);
    }
    // $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
}

```

```

        $varID=CreateVariableByName($id,$Sensordaten,$t1);
        SetValue($varID,$str);
        return $str;
    }

//RS232 Modus
function Extract_C9C($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 1) {
        $data = array_shift($dataArr);
        switch($data) {
            case 0:
                $Arr["RS232 Modus"] = "Ohne Verbindung";
                break;
            case 1:
                $Arr["RS232 Modus"] = "Nur PC";
                break;
            case 2:
                $Arr["RS232 Modus"] = "Nur CC-Ease";
                break;
            case 3:
                $Arr["RS232 Modus"] = "PC Master";
                break;
            case 4:
                $Arr["RS232 Modus"] = "PC Logmodus";
                break;
            default:
                $Arr["RS232"]["Modus"] = $data . " (unknown)";
                break;
        }
    } else {
        $str = "RS232 Modus :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    //$Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
        $varID=CreateVariableByName($id,$varID,$type);
    SetValue($varID,$str);
    return $str;
}

//Zeit Verzögerungen
function Extract_CCA($rawDataStr,$id,$Zeit,$t3) {

```

```

$rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
$rawDataArr = String2Arr($rawDataStr);
$dataArr = ExtractData($rawDataArr);
$Arr = array();
if(count($dataArr) == 8) {
    $Arr["Badezimmer Einschaltverzoeigerung"] = array_shift($dataArr) . " min";
    $Arr["Badezimmer Ausschaltverzoeigerung"] = array_shift($dataArr) . " min";
    $Arr["L1 Ausschaltverzoeigerung"] = array_shift($dataArr) . " min";
    $Arr["Stosslueftung"] = array_shift($dataArr) . " min";
    $Arr["Filter Zaehler"] = array_shift($dataArr) . " Wochen";
    $Arr["RF hoch Zeit kurz"] = array_shift($dataArr) . " min";
    $Arr["RF hoch Zeit lang"] = array_shift($dataArr) . " min";
    $Arr["Kuechenhaube Ausschaltverzoeigerung"] = array_shift($dataArr) . "
min";
} else {
    $str = "Zeit-Verz?gerungen :: no valid Data received";
    $Arr["ERROR"] = $str;
    LogMsg(__FUNCTION__,$str);
}
//$Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varID=CreateVariableByName($id,$Zeit,$t3);
SetValue($varID,$str);
return $str;
}

```

//Ventilationsstufe abrufen

```

function Extract_CCE($rawDataStr,$id,$varID,$aktuelleStufe,$type,$type2) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 14) {
        $AbluftAbwesend = array_shift($dataArr);
        $AbluftStufe1 = array_shift($dataArr);
        $AbluftStufe2 = array_shift($dataArr);
        $ZuluftAbwesend =array_shift($dataArr);
        $ZuluftStufe1 =array_shift($dataArr);
        $ZuluftStufe2 = array_shift($dataArr);
        $AbluftAktuell= array_shift($dataArr);
        $ZuluftAktuell= array_shift($dataArr);
        $AktuelleStufe= array_shift($dataArr);
        $ZuluftAktiv= array_shift($dataArr);
        $AbluftStufe3 =array_shift($dataArr);
        $ZuluftStufe3 = array_shift($dataArr);

        $Arr["Abluft-Abwesend"] = $AbluftAbwesend . " %";
    }
}

```

```

$Arr["Abluft-Stufe1"] = $AbluftStufe1 . " %";
$Arr["Abluft-Stufe2"] = $AbluftStufe2 . " %";
$Arr["Abluft-Stufe3"] = $AbluftStufe3 . " %";

$Arr["Zuluft-Abwesend"] = $ZuluftAbwesend . " %";
$Arr["Zuluft-Stufe1"] = $ZuluftStufe1 . " %";
$Arr["Zuluft-Stufe2"] = $ZuluftStufe2 . " %";
$Arr["Zuluft-Stufe3"] = $ZuluftStufe3 . " %";

$varID=CreateVariableByName($id,$aktuelleStufe,$type2);
                               SetValue($varID,$AktuelleStufe);
} else {
$str = "Ventilationsstufe :: no valid Data received";
$Arr["ERROR"] = $str;
  LogMsg(__FUNCTION__,$str);
}
//$Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varID=CreateVariableByName($id,$varID,$type);
SetValue($varID,$str);
return $str;
}

//Temperaturen abrufen
function Extract_CD2($rawDataStr,$id,$Temperaturen,$KomfortTemp,$t1,$t2) {
  $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
  $rawDataArr = String2Arr($rawDataStr);
  $dataArr = ExtractData($rawDataArr);
  $Arr = array();
  if(count($dataArr) == 9) {
    $Arr["KomfortTemperatur"] = number_format(array_shift($dataArr) / 2 - 20,1)
    ." °C";
    $ID_KomfortTemp=CreateVariableByName($id,$KomfortTemp,$t2);

    SetValue($ID_KomfortTemp,(int)$Arr["KomfortTemperatur"]);
    $Arr["T1-Aussenluft"] = number_format(array_shift($dataArr) /
    2 - 20,1) ." °C";
    $Arr["T2-Zuluft"] = number_format(array_shift($dataArr) / 2 - 20,1) ." °C";
    $Arr["T3-Abluft"] = number_format(array_shift($dataArr) / 2 - 20,1) ." °C";
    $Arr["T4-Fortluft"] = number_format(array_shift($dataArr) / 2 - 20,1) ." °C";

    $Fuehler = array_shift($dataArr);
    $FuehlerT1 = $Fuehler & hexdec(0x01);
    $FuehlerT2 = ($Fuehler & hexdec(0x02)) >> 1;
    $FuehlerT3 = ($Fuehler & hexdec(0x04)) >> 2;
    $FuehlerT4 = ($Fuehler & hexdec(0x08)) >> 3;

```

```

$FuehlerEWT = ($Fuehler & hexdec(0x10)) >> 4;
$FuehlerNachheizung = ($Fuehler & hexdec(0x20)) >> 5;
$FuehlerKuechenhaube = ($Fuehler & hexdec(0x40)) >> 6;

//$Arr["Fuehler-T1"] = $FuehlerT1;
//$Arr["Fuehler-T2"] = $FuehlerT2;
//$Arr["Fuehler-T3"] = $FuehlerT3;
//$Arr["Fuehler-T4"] = $FuehlerT4;
//$Arr["Fuehler-EWT"] = $FuehlerEWT;
//$Arr["Fuehler-Nachheizung"] = $FuehlerNachheizung;
//$Arr["Fuehler-Kuechenhaube"] = $FuehlerKuechenhaube;

//if($FuehlerEWT) { $Arr["EWT"] = number_format(array_shift($dataArr) / 2 -
20,1) ." °C"; } else { $Arr["EWT"] = "-";}
//if($FuehlerNachheizung) { $Arr["Nachheizung"] =
number_format(array_shift($dataArr) / 2 - 20,1) ." °C"; } else { $Arr["Nachheizung"] = "-";}
//if($FuehlerKuechenhaube) { $Arr["Kuechenhaube"] =
number_format(array_shift($dataArr) / 2 - 20,1) ." °C"; } else { $Arr["Kuechenhaube"] = "-
";}

} else {
$str = "Temperaturen :: no valid Data received";
$Arr["ERROR"] = $str;
LogMsg(__FUNCTION__, $str);
}
//$Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varID=CreateVariableByName($id,$Temperaturen,$t1);
SetValue($varID,$str);
return $str;
}

//Status abrufen
function Extract_CD5($rawDataStr,$id,$Status,$t1) {
$rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
$rawDataArr = String2Arr($rawDataStr);
$dataArr = ExtractData($rawDataArr);
$Arr = array();
if(count($dataArr) == 11) {
if(array_shift($dataArr)) { $Arr["Vorheizung"] = "vorhanden"; } else {
$Arr["Vorheizung"] = "nicht vorhanden"; };
if(array_shift($dataArr)) { $Arr["Bypass"] = "vorhanden"; } else { $Arr["Bypass"]
= "nicht vorhanden"; };
if(array_shift($dataArr)) { $Arr["Typ"] = "rechts"; } else { $Arr["Typ"] = "links"; };
if(array_shift($dataArr)) { $Arr["Groesse"] = "gross"; } else { $Arr["Typ"] =
"klein"; };

$Optionen = array_shift($dataArr);

```



```

        $Feuerstaette = $Optionen & hexdec(0x01);
    $Kuechenhaube = ($Optionen & hexdec(0x02)) >> 1;
    $Nachheizung = ($Optionen & hexdec(0x04)) >> 2;
    if($Feuerstaette) { $Arr["Option-Feuerstaette"] = "vorhanden"; } else {
$Arr["Option-Feuerstaette"] = "nicht vorhanden"; };
    if($Kuechenhaube) { $Arr["Option-Kuechenhaube"] = "vorhanden"; } else {
$Arr["Option-Kuechenhaube"] = "nicht vorhanden"; };
    if($Nachheizung) { $Arr["Option-Nachheizung"] = "vorhanden"; } else {
$Arr["Option-Nachheizung"] = "nicht vorhanden"; };

```

```

$xx = array_shift($dataArr);
$P10_17 = array_shift($dataArr);
$Arr["Status-P10"] = ($P10_17 & hexdec(0x01));
$Arr["Status-P11"] = ($P10_17 & hexdec(0x02)) >> 1;
$Arr["Status-P12"] = ($P10_17 & hexdec(0x04)) >> 2;
$Arr["Status-P13"] = ($P10_17 & hexdec(0x08)) >> 3;
$Arr["Status-P14"] = ($P10_17 & hexdec(0x10)) >> 4;
$Arr["Status-P15"] = ($P10_17 & hexdec(0x20)) >> 5;
$Arr["Status-P16"] = ($P10_17 & hexdec(0x40)) >> 6;
$Arr["Status-P17"] = ($P10_17 & hexdec(0x80)) >> 7;

```

```

$P18_19 = array_shift($dataArr);
$Arr["Status-P18"] = ($P18_19 & hexdec(0x01));
$Arr["Status-P19"] = ($P18_19 & hexdec(0x02)) >> 1;

```

```

$P90_96 = array_shift($dataArr);
$Arr["Status-P90"] = ($P90_96 & hexdec(0x01));
$Arr["Status-P91"] = ($P90_96 & hexdec(0x02)) >> 1;
$Arr["Status-P92"] = ($P90_96 & hexdec(0x04)) >> 2;
$Arr["Status-P93"] = ($P90_96 & hexdec(0x08)) >> 3;
$Arr["Status-P94"] = ($P90_96 & hexdec(0x10)) >> 4;
$Arr["Status-P95"] = ($P90_96 & hexdec(0x20)) >> 5;
$Arr["Status-P96"] = ($P90_96 & hexdec(0x40)) >> 6;

```

```

$Enthalpie = array_shift($dataArr);
switch($Enthalpie) {
    case 0:
        $Arr["Enthalpie"] = "abwesend";
        break;
    case 1:
        $Arr["Enthalpie"] = "anwesend";
        break;
    case 2:
        $Arr["Enthalpie"] = "ohne Fühler";
        break;
    default:
        $Arr["Enthalpie"] = "???";
        break;
}

```

```

    }
    $EWT = array_shift($dataArray);
    switch($EWT) {
        case 0:
            $Arr["EWT"] = "abwesend";
            break;
        case 1:
            $Arr["EWT"] = "geregelt";
            break;
        case 2:
            $Arr["EWT"] = "ungeregelt";
            break;
        default:
            $Arr["EWT"] = "???" ;
            break;
    }
} else {
    $str = "Status :: no valid Data received";
    $Arr["ERROR"] = $str;
    LogMsg(__FUNCTION__, $str);
}
//$Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varID=CreateVariableByName($id,$Status,$t1);
SetValue($varID,$str);
return $str;
}

```

//Störungen und Filter abrufen

```

function Extract_CDA($rawDataStr,$id,$Fehler,$letzterfehler,$Filter,$t1) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArray = ExtractData($rawDataArr);
        $Arr = array();
            $Arr["Filter"] = "";
            $Arr["ERROR"]="";
    $Arr["AktuellerFehler-A"] = "";
    $Arr["AktuellerFehler-E"] = "";
    $Arr["LetzterFehler-A"] = "";
    $Arr["LetzterFehler-E"] = "";
    $Arr["VorletzterFehler-A"] = "";
    $Arr["VorletzterFehler-E"] = "";
    $Arr["VorvorletzterFehler-A"] = "";
    $Arr["VorvorletzterFehler-E"] = "";
    $Arr["AktuellerFehler-EA"] = "";
    $Arr["LetzterFehler-EA"] = "";
    $Arr["VorletzterFehler-EA"] = "";
}

```

```

$Arr["VorvorletzterFehler-EA"] = "";
$Arr["AktuellerFehler-A_high"] = "";
$Arr["LetzterFehler-A_high"] = "";
$Arr["VorletzterFehler-A_high"] = "";
$Arr["VorvorletzterFehler-A_high"] = "";
if(count($dataArr) == 17) {
    $Arr["ERROR"]="";
    $Arr["AktuellerFehler-A"] = array_shift($dataArr);
    $Arr["AktuellerFehler-E"] = array_shift($dataArr);
    $Arr["LetzterFehler-A"] = array_shift($dataArr);
    $Arr["LetzterFehler-E"] = array_shift($dataArr);
    $Arr["VorletzterFehler-A"] = array_shift($dataArr);
    $Arr["VorletzterFehler-E"] = array_shift($dataArr);
    $Arr["VorvorletzterFehler-A"] = array_shift($dataArr);
    $Arr["VorvorletzterFehler-E"] = array_shift($dataArr);
    if(array_shift($dataArr)) { $Arr["Filter"] = "voll"; } else { $Arr["Filter"] = "OK"; };
    $Arr["AktuellerFehler-EA"] = array_shift($dataArr);
    $Arr["LetzterFehler-EA"] = array_shift($dataArr);
    $Arr["VorletzterFehler-EA"] = array_shift($dataArr);
    $Arr["VorvorletzterFehler-EA"] = array_shift($dataArr);

    $Arr["AktuellerFehler-A_high"] = array_shift($dataArr);
    $Arr["LetzterFehler-A_high"] = array_shift($dataArr);
    $Arr["VorletzterFehler-A_high"] = array_shift($dataArr);
    $Arr["VorvorletzterFehler-A_high"] = array_shift($dataArr);
} else {
    $str = "Stoerungen :: no valid Data received";
    $Arr["ERROR"] = $str;
    LogMsg(__FUNCTION__,$str);
}
// $Arr["RawData"] = String2Hex($rawDataStr);
$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$varIDFilter=CreateVariableByName($id,$Filter,$t1);
$varIDError=CreateVariableByName($id,$letzterfehler,$t1);
$varID=CreateVariableByName($id,$Fehler,$t1);
    setValue($varIDFilter,$Arr["Filter"]);
    setValue($varIDError,"");
    if ($Arr["ERROR"]=="Stoerungen :: no valid Data received")
$Arr["ERROR"]="";
        if (($Arr["ERROR"]<>"") or ($Arr["AktuellerFehler-A"]<>"") or
($Arr["AktuellerFehler-E"]<>"") or ($Arr["AktuellerFehler-A_high"]<>""))
        {
            setValue($varIDError,"ComfoBoxFEHLER: ".$Arr["ERROR"]."
FehlerA: ".$Arr["AktuellerFehler-A"]." / FehlerE: ".$Arr["AktuellerFehler-E"]." /
FehlerA_high: ".$Arr["AktuellerFehler-A_high"]);}
        else
            {setValue($varIDError,"");}

```

```

        SetValue($varID,$str);
        return $str;
    }

//Betriebsstunden
function
Extract_CDE($rawDataStr,$id,$Betriebsstunden,$HST1,$HST2,$HST3,$HSTgesamt,$FilterCountdown,$t3) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 20) {
        $Abwesend = array_shift($dataArr)*256*256 + array_shift($dataArr) * 256 +
array_shift($dataArr);
        $Stufe1 = array_shift($dataArr)*256*256 + array_shift($dataArr) * 256 +
array_shift($dataArr);
        $Stufe2 = array_shift($dataArr)*256*256 + array_shift($dataArr) * 256 +
array_shift($dataArr);

        $Frostschutz = array_shift($dataArr) * 256 + array_shift($dataArr);
        $Vorheizung = array_shift($dataArr) * 256 + array_shift($dataArr);
        $Bypass = array_shift($dataArr) * 256 + array_shift($dataArr);
        $Filter = array_shift($dataArr) * 256 + array_shift($dataArr);
        $Stufe3 = array_shift($dataArr)*256*256 + array_shift($dataArr) * 256 +
array_shift($dataArr);
        $varID=CreateVariableByName($id,$HST1,1);
        SetValue($varID,$Stufe1);
        $varID=CreateVariableByName($id,$HST2,1);
        SetValue($varID,$Stufe2);
        $varID=CreateVariableByName($id,$HST3,1);
        SetValue($varID,$Stufe3);
        $varID=CreateVariableByName($id,$HSTgesamt,1);
        SetValue($varID,$Stufe1+$Stufe2+$Stufe3);

        $Arr["Abwesend"] = $Abwesend . " h -> " .
number_format((float)$Abwesend/24/360,2) . " Jahre";
        $Arr["Stufe-1"] = $Stufe1 . " h -> " . number_format((float)$Stufe1/24/360,2) . "
Jahre";
        $Arr["Stufe-2"] = $Stufe2 . " h -> " . number_format((float)$Stufe2/24/360,2) . "
Jahre";
        $Arr["Stufe-3"] = $Stufe3 . " h -> " . number_format((float)$Stufe3/24/360,2) . "
Jahre";

        $Arr["Gesamtlaufzeit"] =
($Abwesend+$Stufe1+$Stufe2+$Stufe3) . " h -> " .
number_format((float)($Abwesend+$Stufe1+$Stufe2+$Stufe3)/24/360,2) . " Jahre";
        $Arr["Bypass"] = $Bypass . " h -> " . number_format((float)$Bypass/24/360,2) .
" Jahre";
    }
}

```

```

        $Arr["Filter"] = $Filter . " h -> ". number_format((float)$Filter/24/360,2) . "
Jahre";
        $FilterCountdown=CreateVariableByName($id,$FilterCountdown,$t3);
        SetValue($FilterCountdown,182-(int)number_format((float)$Filter/24,2));
        $Arr["Frostschutz"] = $Frostschutz . " h -> ".
number_format((float)$Frostschutz/24/360,2) . " Jahre";
        $Arr["Vorheizung"] = $Vorheizung . " h -> ".
number_format((float)$Vorheizung/24/360,2) . " Jahre";
    } else {
        $str = "Betriebsstunden :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    //$Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    $varID=CreateVariableByName($id,$Betriebsstunden,$t3);
    SetValue($varID,$str);
    return $str;
}

//Bypassregelung abrufen
function Extract_CE0($rawDataStr,$id,$Modus,$Stufe,$Bypass,$t1,$t2,$t3) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 7) {
        array_shift($dataArr);
        array_shift($dataArr);
        $Arr["Bypass-Faktor"] = array_shift($dataArr);
        $Arr["Bypass-Stufe"] = array_shift($dataArr);
        $Arr["Bypass-Korrektur"] = array_shift($dataArr);
        array_shift($dataArr);
        $Arr["Modus"] = array_shift($dataArr);
        if($Arr["Modus"]==1) { $Arr["Modus"] = "Sommermodus"; } else { $Arr["Modus"]
= "Wintermodus"; };
        $BypassModus=CreateVariableByName($id,$Modus,$t1);
        $BypassStufe=CreateVariableByName($id,$Stufe,$t2);
        SetValue($BypassStufe,(int)$Arr["Bypass-Stufe"]);
        SetValue($BypassModus,$Arr["Modus"]);
    } else {
        $str = "Bypassregelung :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    //$Arr["RawData"] = String2Hex($rawDataStr);

```

```

$Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
$str = Array2HtmlTable($Arr);
$VisID=CreateVariableByName($id,$Bypass,$t3);
SetValue($VisID,$str);
return $str;
}

// Vorheizung abrufen => in meinem Fall uninteressant, da kein Frostschutz eingebaut !!
function Extract_CE2($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 6) {
        switch(array_shift($dataArr)) {
            case 0:
                $Arr["StatusKlappe"] = "geschlossen";
                break;
            case 1:
                $Arr["StatusKlappe"] = "offen";
                break;
            case 2:
                $Arr["StatusKlappe"] = "unbekannt";
                break;
            default:
                $Arr["StatusKlappe"] = "???" ;
                break;
        }
        if(array_shift($dataArr) {$Arr["Frostschutz"] = "aktiv";} else
        {$Arr["Frostschutz"] = "inaktiv";}
        if(array_shift($dataArr) {$Arr["Vorheizung"] = "aktiv";} else
        {$Arr["Vorheizung"] = "inaktiv";}
        $Arr["Frostminuten"] = array_shift($dataArr) * 256 + array_shift($dataArr) . "
min";
        if(array_shift($dataArr) {$Arr["Frostsicherheit"] = "extra sicher";} else
        {$Arr["Frostsicherheit"] = "sicher";}
        } else {
            $str = "Vorheizung :: no valid Data received";
            $Arr["ERROR"] = $str;
            LogMsg(__FUNCTION__,$str);
        }
        // $Arr["RawData"] = String2Hex($rawDataStr);
        $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
        $str = Array2HtmlTable($Arr);
        SetValue($varID,$str);
        return $str;
    }
}

```

```

// RF Status abrufen, uninteressant, da nicht verbaut
function Extract_CE6($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 7) {

        $RF_Adr4 = array_shift($dataArr);
        $RF_Adr3 = array_shift($dataArr);
        $RF_Adr2 = array_shift($dataArr);
        $RF_Adr1 = array_shift($dataArr);
        $RF_ID = array_shift($dataArr);

        $Arr["RF Adresse"] = dechex($RF_Adr1) . " ". dechex($RF_Adr2) . " ".
dechex($RF_Adr3) . " ". dechex($RF_Adr4);
        $Arr["RF ID"] = dechex($RF_ID);
    } else {
        $str = "RF Status :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    SetValue($varID,$str);
    return $str;
}

```

```

// Letzte 8 mal Vorheizung => in meinem Fall uninteressant
function Extract_CEA($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 8) {
        $Arr["Wert 8 (Neuster Wert)"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 7"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 6"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 5"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 4"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 3"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 2"] = array_pop($dataArr) . " ?C";
        $Arr["Wert 1 (Aeltester Wert)"] = array_pop($dataArr) . " ?C";
    } else {
        $str = "Letzte 8 mal Vorheizung :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
}

```

```

    }
    $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    SetValue($varID,$str);
    return $str;
}

// EWT Nachheizung => in meinem Fall uninteressant, da nicht eingebaut !!
function Extract_CEC($rawDataStr,$varID) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $rawDataArr = String2Arr($rawDataStr);
    $dataArr = ExtractData($rawDataArr);
    $Arr = array();
    if(count($dataArr) == 8) {
        $Arr["EWT-niedrig"] = array_shift($dataArr) . " °C";
        $Arr["EWT-hoch"] = array_shift($dataArr) . " °C";
        $Arr["EWT-speedUp"] = array_shift($dataArr) . " %";
        $Arr["Kuechenhaube-speedUp"] = array_shift($dataArr) . " %";
        $Arr["NachheizungLeistung"] = array_shift($dataArr);
        $Arr["NachheizungLeistung_l-Parameter"] = array_shift($dataArr) . " " .
array_shift($dataArr);;
        $Arr["NachheizungTempGewuenscht"] = array_shift($dataArr) . " °C";
    } else {
        $str = "EWT Nachheizung :: no valid Data received";
        $Arr["ERROR"] = $str;
        LogMsg(__FUNCTION__,$str);
    }
    $Arr["RawData"] = String2Hex($rawDataStr);
    $Arr["Update-TimeStamp"] = date('d-m-Y H:i:s',time());
    $str = Array2HtmlTable($Arr);
    SetValue($varID,$str);
    return $str;
}

function Extract_Unknown($rawDataStr) {
    $rawDataStr = str_replace("\x07\x07","\x07",$rawDataStr);
    $str = "WRL WARNING :: unknown Command received -> " . String2Hex($RawData)
. "(" . $RawData;
    LogMsg(__FUNCTION__,$str);
    return $str;
}

//
~~~~~
~~~~~

function ExtractString($rawDataStr) {

```



```

        $rawDataArr = String2Arr($rawDataStr);
        $dataArr = ExtractData($rawDataArr);
        return Arr2String($dataArr);
    }

function ExtractData($byteArr) {
    $crc = array_pop($byteArr);
    //echo "\r\nrcrc: " . $crc;
    //print_r($byteArr);
    $crcOK = CheckCRC($byteArr,$crc);
    if($crcOK) {
        //print_r($byteArr);
        //echo "\r\nlen: " . dechex($byteArr[2]);
        return array_slice($byteArr,3,ord($byteArr[2]));
    } else {
        return array ("\x00");
    }
}

function CheckCRC($byteArr,$crc) {
    $crcCalculated = CalcCRC($byteArr);
    //echo "\r\nrcrc: " . $crc;
    //echo "\r\nrcrcCalculated: " . $crcCalculated;
    if ($crc == $crcCalculated) {
        //echo "\r\nCRC OK";
        return true;
    } else {
        //echo "\r\nCRC !not! OK";
        LogMsg(__FUNCTION__,"CRC NOT OK >: " . String2Hex(Arr2String($byteArr)));
        return false;
    }
}

function CalcCRC($byteArr) {
    $crc = 0;
    $cnt07 = 0;
    foreach($byteArr as $byte) {
        //echo "\r\n Byte: " . $byte . " :: " . dechex($byte) . " :: " . chr($byte);
        /*
        if(($byte == 07) AND ($cnt07==0)) {
            $cnt07++;
        } else {
            $cnt07 = 0;
            $crc = $crc + $byte;
        }
        */
        $crc = $crc + $byte;
    }
}

```

```

    }
    $crc = $crc + 173;
    $crc = $crc & 255;
    return $crc;
}

```

```

function String2Hex($string){
    $hex="";
    for ($i=0; $i < strlen($string); $i++){
        //$hex .= dechex(ord($string[$i]));
        $hex .= "0x" . sprintf("%02X", ord($string[$i])) . " ";
    }
    return $hex;
}

```

```

function String2Arr($string){
    $arr = array();
    for ($i=0; $i < strlen($string); $i++){
        $arr[] = ord($string[$i]);
    }
    return $arr;
}

```

```

function Arr2String($arr){
    $str = "";
    foreach ($arr as $arrElem) {
        $str .= chr($arrElem);
    }
    return $str;
}

```

```

function DebugArray($arr) {
    foreach ($arr as $arrElem) {
        echo "\r\n Byte: " . $arrElem . " :: " . dechex($arrElem) . " :: " . chr($arrElem);
    }
}

```

```

function startsWith($haystack, $needle)
{
    return !strncmp($haystack, $needle, strlen($needle));
}

```

```

function LogMsg($functionName,$logMessage) {

    $logEntry = date('Y-m-d H:i:s', time()) . " : " . $functionName . " : " . $logMessage;
}

```

```

$Path1 = IPS_GetKernelDir();
$Path2 = "logs\";
$File = "ComfoAir550.log"; //date("Y-m-d His",time())
$Filename = $Path1 . $Path2 . $File;

$datei = fopen($Filename , "a+");
fwrite($datei, $logEntry . "\r\n");
fclose($datei);
}

function Array2HtmlTable($Arr) {
    //print_r($Arr);
    $html = "<table width='100%' cellpadding='0'>";
    $keys = array_keys($Arr);
    for($i=0;$i<count($keys);$i++)
    {
        $html.= "<tr>";
        switch ($keys[$i]) { // spezzie farbdarstellungen im htmlbox
            case "Gesamtlaufzeit":
                $html.= "<td><font color='#7E7979'>" . $keys[$i] . "</font></td><td
align='right'><font color='#7E7979'>" . $Arr[$keys[$i]] . "</font></td>";
                break;
                case "Update-TimeStamp":
                    $html.= "<td><font size=2 color='#7E7979'>" . $keys[$i] .
"</font></td><td align='right'><font size=2 color='#7E7979'>" . $Arr[$keys[$i]]
."</font></td>";
                    break;
                    case "RawData":
                        $html.= "<td><font size=2 color='#7E7979'>" . $keys[$i] . "</td><td
align='right'><font size=2 color='#7E7979'>" . $Arr[$keys[$i]] . "</font></td>";
                        break;
                        default:
                            $html.= "<td>" . $keys[$i] . "</td><td align='right'>" . $Arr[$keys[$i]]
."</td>";
                            break;
                            }
                            $html.= "</tr>";
                            }
                            $html.= "</table>";
                            return $html;
                            }

function CreateVariableByName($id, $name, $type) { //Variablen auslesen bzw.
erstellen
    global $scriptid;
    $scriptid = $_IPS['SELF'];
    $vid = @IPS_GetVariableIDByName($name, $id);
    if($vid===false) {

```

```

$vid = IPS_CreateVariable($type);
IPS_SetParent($vid, $id);
IPS_SetName($vid, $name);
}
return $vid;
}

```

?>

...`

**Denkt daran, dass Ihr die ID's in den Scripten entsprechend Eurer ID's anpassen müsst,** damit Ihr ein besseres Verständnis bekommt hier die Gesamtsicht meines Objektbaumes (grün gekennzeichnet -> manuell erstellen, gelb gekennzeichnet -> diese Variablen werden automatisch erstellt)

ID	Name	Typ	Wert	Aktualisi
37023	Luftungsanlage			
33743	Automatiksteuerung			
15168	Luftung	Register Variable		
30142	5453	String	<table width="100%" cellspa...	23:32:46
16433	Abluft U/Min	Integer	1197	23:32:46
29463	Abluft's	Integer	35	23:32:46
11521	Ablufttemperatur	Float	16,5	23:32:45
49593	Aussenkühltemperatur	Float	-0,5	23:32:45
54317	Badezimmerschalter	Integer	0	23:32:46
45748	Betriebsstunden	String	<table width="100%" cellspa...	23:32:45
17138	Bootloader Version	String	<table width="100%" cellspa...	23:32:46
16812	Bypass	String	<table width="100%" cellspa...	23:32:45
57405	Bypassmodus	String	Wintermodus	23:32:45
59881	Bypassstufe	Integer	0	23:32:45
30047	Eingangs	String	<table width="100%" cellspa...	23:32:46
52139	Fehler	String	<table width="100%" cellspa...	23:32:45
12785	Filter	String	OK	23:32:45
15893	Filter Countdown Tage	String	154	23:32:45
32824	Firmware Version	String	<table width="100%" cellspa...	23:32:46
46040	Fortlufttemperatur	Float	2,5	23:32:45
51050	Klappenstatus	String	<table width="100%" cellspa...	23:32:45
54809	Komfort-Temperatur	Integer	21	23:32:45
18307	Letzter Fehler	String	ComfoBoxFEHLER: FehlerA...	23:32:45
51679	Status	String	<table width="100%" cellspa...	23:32:46
57338	Stunden gesamt	Integer	108511	23:32:45
31701	Stunden Stufe 1	Integer	39057	23:32:45
29517	Stunden Stufe 2	Integer	67547	23:32:45
33075	Stunden Stufe 3	Integer	1907	23:32:45
44443	Temperaturen	String	<table width="100%" cellspa...	23:32:45
54916	Ventilationsstufen	String	<table width="100%" cellspa...	22:24:01
31924	Ventilatorstatus	String	<table width="100%" cellspa...	23:32:46
54653	Ventilatorstufe	Integer	2	23:32:46
54190	Zeitverzögerung	String	<table width="100%" cellspa...	23:32:46
41415	Zuluft U/Min	Integer	1179	23:32:46
57521	Zuluft's	Integer	35	23:32:46
56516	Zulufttemperatur	Float	8,5	23:32:45
44712	RK-Luftung	PHP-Skript	44712.ips.php	23:32:46
46451	Luftung auslesen	PHP-Skript	46451.ips.php	23:32:46
18545	Ereignis: Alle 1 Tag(e) alle 15 Sekunde(n)	Ereignis		23:32:45
38023	Komforttemperatur setzen	PHP-Skript	38023.ips.php	23:22:45
31358	Komforttemperatur Soll	Integer	21	23:22:45
12229	Ereignis: Bei Änderung von 'Komforttemperatur Soll (Luftungsanlage/Komforttemperatur setzen)'	Ereignis		23:22:45
45367	Luefterstufe setzen	PHP-Skript	45367.ips.php	22:18:50
40012	Luefterstufe Soll	Integer	2	22:18:50
30553	Ereignis: Bei Änderung von 'Luefterstufe Soll (Luftungsanlage/Lufterstufe setzen)'	Ereignis		22:18:50
18321	Reset Gerätefilter	PHP-Skript	18321.ips.php	22:22:48
47785	Reset Filter	Boolean	false	21:29:48
33929	Ereignis: Bei bestimmtem Wert (true) von 'Reset Filter (Luftungsanlage/Reset Gerätefilter)'	Ereignis		21:29:48
41860	PV-Anlage			
25687	Shelly			
42363	Sonos			
28645	Surveillance			
14187	Wetter			
27505	BildArchiv	BildArchiv		
36155	SmartHome View	Medien	media/36155.ipsView	23:22:07
24226	Client Socket (ModBus Gateway #59298) - Huawei	Client Socket	192.168.1.110:502	
56521	Multicast Socket (KNX Discovery #59293)	Multicast Socket		
51407	NetatmoWeatherID (NetatmoWeatherConfig #56673)	NetatmoWeatherID		
50765	Serial Port - ComfoAir 350	Serial Port	COM3 (9600,8,1,N)	
58033	Server Socket (MQTT Server #23169) - Shelly	Server Socket	0.0.0.0:1024	
32877	SSE Client (MieleAtHomeSplitter #30008)	SSE Client		
27572	UDP Socket (KNX Gateway #23519)	UDP Socket	192.168.1.10:3671	
53565	Cutter ComfoAir 350	Zerschneider		
23519	KNX Gateway	KNX Gateway		
30008	MieleAtHomeSplitter	MieleAtHomeSplitter		
59298	ModBus Gateway - Dongle - Huawei	ModBus Gateway		
11331	ModBus Gateway - Wechseleiferter - Huawei	ModBus Gateway		
23169	MQTT Server - Shelly	MQTT Server		
24289	Sonos Splitter	Sonos Splitter		
	Kern Instanzen			

**Sofern Ihr das alles erledigt habt, könnt Ihr das Script "Lueftung auslesen" einmal starten**

57521	~	Zuluft%	Integer	35	23:53:01
56516		Zulufttemperatur	Float	8,5	23:53:00
44712		RX-Lueftung	PHP-Skript	44712.ips.php	23:53:01
46451		Lueftung auslesen	PHP-Skript	46451.ips.php	23:53:01
18545		Ereignis: Alle 1 Tag(e) alle 15 Sekunde(n)	Ereignis		23:53:00

Wenn alles richtig konfiguriert ist, sollten sich dann die **ganzen Variablen automatisch unterhalb der "Lueftung" Register Variablen erzeugen.**

Folgende Scripte hat mir @smudo noch freundlicherweise zur Verfügung gestellt für die Einstellung der **\*\*Komforttemperatur\*\***

...

**// Script Komforttemperatur**

```
<?php
$ID=50765; //ID der Seriellen Schnittstelle
Include('ComfoAir.ips.php');
$temp=GetValueInteger(31358); //Hier die ID der Variablen für die Temperatur.
    if (($temp>=15)and($temp<=25));
    {
        SendCommand(array(0x00,0xD3),array(($temp+20)*2),$ID);//Komforttemperatur
    setzen
    }
... 
```

das **setzen der Lüfterstufe**

...

**// Script Luefterstufe setzen**

```
<?php

if($_IPS["SENDER"] == "Variable")
{
    $ID=50765; //ID der Seriellen Schnittstelle
        Include('ComfoAir.ips.php');
        $Stufe=GetValueInteger(40012); //Hier die ID der von dir angelegten Variablen für die
    Stufe angeben.
        if (($Stufe>=1)and($Stufe<=4)){
            SendCommand(array(0x00,0x99),array(0x0.$Stufe),$ID);
        }
... 
```

## und den **Reset des Filters**

```\n

```
// Script Reset des Filters
```

```
<?php
```

```
$ID=50765; //ID der Seriellen Schnittstelle
```

```
Include('ComfoAir.ips.php');
```

```
SendCommand(array(0x00,0xDB),array(0,0,0,1),$ID); // Reset des Filters
```

```\n

Hier noch die Details aus dem Objektbaum zu den 3 zuvor genannten Scripten

|       |   |            |               |
|-------|---|------------|---------------|
| 44712 | ■ RKL-Lueftung  | PHP-Skript | 44712.ips.php |
| 46451 | ▼ ■ Lueftung auslesen   | PHP-Skript | 46451.ips.php |
| 18545 | ● Ereignis: Alle 1 Tag(e) alle 15 Sekunde(n)  | Ereignis   |               |
| 38023 | ▼ ■ Komforttemperatur setzen  | PHP-Skript | 38023.ips.php |
| 31358 | □ Komforttemperatur Soll  | Integer    | 21            |
| 12229 | ● Ereignis: Bei Änderung von "Komforttemperatur Soll (Lüftungsanlage/Komforttemperatur setzen)" | Ereignis   |               |
| 45367 | ▼ ■ Luefterstufe setzen   | PHP-Skript | 45367.ips.php |
| 40012 | □ Luefterstufe Soll   | Integer    | 2             |
| 30593 | ● Ereignis: Bei Änderung von "Luefterstufe Soll (Lüftungsanlage/Luefterstufe setzen)"           | Ereignis   |               |
| 18321 | ▼ ■ Reset Gerätefilter  | PHP-Skript | 18321.ips.php |
| 47785 | □ Reset Filter  | Boolean    | false         |
| 33929 | ● Ereignis: Bei bestimmtem Wert (true) von "Reset Filter (Lüftungsanlage/Reset Gerätefilter)"   | Ereignis   |               |