Appendix F Data format of weather data

The following information consists of excerpts from the C++ source program, and aims to clarify the data structure of the weather data file.
The weather data file is a sequential file in which all received data are stored consecutively in a data record of constant size (WSPC_DATA).
At the beginning, there is a 4 byte integer containing flags: these indicate the availability of sensors.
Successive data records follow this as they are received from the interface.

Structure of the data record

```
struct WSPC_DATA
{
  time_t               zeit;
  WSPC_TEMP_FEUCHTE    temp[9];      // 8 Temp./humidity sensor
  WSPC_DRUCK           druck;        // Pressure sensor
  WSPC_WIND            wind;         // Wind sensor
  WSPC_REGEN           regen;        // Rain sensor
  WSPC_HELL_PYR        helligkeit;   // Brightness sensor
  WSPC_HELL_PYR        pyrano;       // Pyranometer
};

#pragma pack(1)                   // all the following data are byte aligned
struct WSPC_TEMP_HUMIDITY
{
  short           temp;
  unsigned char   feuchte;
  unsigned char   flag;
};                // structure for recording data from the temperature
                     sensors

struct WSPC_PRESSURE
{
  short           druck;
  unsigned char   flag;
};                // structure for recording pressure data
```

```
struct WSPC_WIND
{
  short              staerke;
  short              richtung;
  unsigned char      breite;
  unsigned char      flag;
};                   // structure for recording data from the
                     wind sensor

struct WSPC_RAIN
{
  short              zaehler;
  unsigned char      flag;
};                   // structure for recording data from the
                     rain sensor

struct WSPC_BRIGHTNESS_PYR
{
  int                wert;
  unsigned char      flag;
};                   // structure for recording data
                     from the brightness sensor
```

* Data transfer from the interface to the PC

Overview of commands

‘0’ : Activate communication
‘1’ : Request time
‘2’ : Select first saved data record
‘3’ : Select next data record
‘4’ : Request data record
‘5’ : Request status
‘D’ : Initialise interface (interval time, sensor addresses and sensor version)

Activation of data transfer

In its normal mode, the PC radio interface is inactive and does not react to V24 signals.

- For the interface to be supplied with power, DTR must be set (+12 V) and RTS deleted (-12 V).
- For activation, the activation command should be sent to the interface until a response is received from it.
- The PC radio interface activates its interface as soon as a sign appears. However, it takes up to approx. 30 ms until the oscillator is running smoothly and commands can be properly received.
- Once the interface has responded to the activation command, the data transfer can begin.
- The data exchange via the V24 interface has priority over reception from the sensors. Thus the reception of data should take place over longer intervals.
- Data records are therefore also only deleted after a special request from the PC. By comparing the transferred block numbers in the data record, the PC can determine whether the next data record has actually been selected.
- If no data traffic takes place over a period of 71 ms, the PC radio interface will again be deactivated.

Data formats

Data transfer: 19,200 Baud, Even Parity, 8 Bit, 2 Stop

Data frames for commands from the PC to the PC radio interface
<SOH> <Comm> [Para] <Test> <EOT>

<SOH>	is the start symbol of the data transfer
<Comm>	is the command for the interface
[Para]	are any parameters that may arise. For the parameters, Bit 7 is always set (so that they are not recognised as functional characters)
<Test>	is the negative test total, reached by entering the command and the following parameter. Bit 7 is always the value set here. (if, for example, the command has the value 31h, then Test must have the value CFh).
<EOT>	is the end symbol of the data transfer

Data frame for responses from the interface:
<STX> <length> [message] <test total> <ETX>

The start symbol transmitted will be <STX> while the end symbol will be <ETX> All symbols between the start and the end will be processed in such a way that no <STX> and no <ETX> occur.

<STX>	becomes the symbol series:	<ENQ> <DC2>
<ETX>	becomes the symbol series:	<ENQ> <DC3>
<ENQ>	becomes the symbol series:	<ENQ> <NAK>

- After a data package has been received, this editing must first be cancelled before the length and the test total can be checked.
- From the length you can derive the number of characters in the message.
- The test total is the negative 8-bit total of the number of bytes from <STX> up to the last character of the message. If all of the characters are added up (from STX to the test total), then the bottom 8 bits must yield the total 0.

In the case of incorrect data reception, the interface will send the message <NAK>.

	<STX>	<01h>	<NAK>	<0E8h>	<ETX>
Hex :	02	01	15	E8	03

Note: In the following description of the individual commands, only the actual message will be given as a response. The data frame is still to be added.

Description of the commands and responses

Activate communication
<SOH> '0' (-total) <EOT>

This exists for the purpose of establishing communication with the interface.

Response from the interface: (1 byte)
<ACK>

Request time and date

<SOH> '1' (-total) <EOT>

Request for the time and date at the interface.

Response from the interface: (5 byte)
(hr) (min) (second) (day) (month/day of the week/flag)

hr.	1 byte	: Hour in BCD	(one: b0-3	ten: b4-7)
min.	1 byte	: minute in BCD	(one: b0-3	ten: b4-7)
sec.	1 byte	: second (binary !)		
Day	1 byte	: day in BCD	(one: b0-3	ten: b4-7)
month	1 byte	: b0-b3 → month (binary!)		
		: b4-b6 → day of the week		

Select next data record
<SOH> '2' (-total) <EOT>

This command serves to raise the ring pointer on the data records in the interface. After a data record has been requested, the pointer is not automatically raised, because it cannot recognise whether the data have been properly processed by the PC.
The PC must raise the pointer when this command, which is thus a confirmation for the interface, is entered.

Response from the interface: (1 byte)
1. next data record available: <ACK>
2. no data record available: <DLE>

Select first data record
<SOH> '3' (-total) <EOT>

This command serves to raise the ring pointer to the first data record saved in the interface.

Response from the interface: (1 byte)
<ACK>

Request data record
<SOH> ‘4’ (-total) <EOT>

Response from the interface: (36 byte)
1. no data available:
<DLE>

2. data available:
[block number Lo] [block number Hi] [time Lo] [time Hi] (32 byte data record)

Block number:	Number of the block in the memory (not connected to time. Monitors whether data records are transferred twice)
Time:	Current age of the data record in minutes.
Data:	Data record 32 bytes

Data record

The data record consists of 32 bytes, where every byte consists of two digits. Subsequently, bits 0-3 will be described as “L” and bits 4-7 as “H”.

Temperature/humidity sensor 1-8
The temperatures are transferred in BCD format as 3 digits, with the highest bit corresponding to the sign. This yields a range of values from –79.9 °C to +79.9 °C.
The humidity is transferred in BCD format – 20 %. This allows the third bit in the 10 value to be used as a new flag. → A transferred value of 75 represents a humidity of 95%. If the 1 value is greater than 9, then the humidity value is invalid (sensor has no humidity)

L1	Temp 1 in °C	b_3-b_0	→ 0.1	(0-9)
H1	Temp 1 in °C	b_7-b_4	→ 1	(0-9)
L2	Temp 1 in °C	b_2-b_0	→ 10	(0-7)
		b_3	→ Sign;	
H2	Humidity 1 in %	b_7-b_4	→ 1	(0-9)
			(if value > 9 → sensor has no humidity)	
L3	Humidity 1 in %	b_2-b_0	→ 10 - 2	(0-7)
		b_3	→ New flag;	
H3	Temp 2 in °C	b_7-b_4	→ 0.1	(0-9)
L4	Temp 2 in °C	b_3-b_0	→ 1	(0-9)
H4	Temp 2 in °C	b_6-b_4	→ 10	(0-7)
		b_7	→ Sign;	
L5	Humidity 2 in %	b_3-b_0	→ 1	(0-9)
			(if value > 9 → sensor has no humidity)	
H6	Humidity 2 in %	b_6-b_4	→ 10 - 2	(0-7)
		b_7	→ New flag;	
L6	Temp 3 in °C	b_3-b_0	→ 0.1	(0-9)
H6	Temp 3 in °C	b_7-b_4	→ 1	(0-9)
L7	Temp 3 in °C	b_2-b_0	→ 10	(0-7)
		b_3	→ Sign;	
H7	Humidity 3 in %	b_7-b_4	→ 1	(0-9)
			(if value > 9 → sensor has no humidity)	
L8	Humidity 3 in %	b_2-b_0	→ 10 - 2	(0-7)
		b_3	→ New flag;	
H8	Temp 4 in °C	b_7-b_4	→ 0.1	(0-9)
L9	Temp 4 in °C	b_3-b_0	→ 1	(0-9)
H9	Temp 4 in °C	b_6-b_4	→ 10	(0-7)
		b_7	→ Sign;	

L10 Humidity 4 % b_3-b_0 → 1 (0-9)
(if value > 9 → sensor has no humidity)
H10 Humidity 4 in % b_6-b_4 → 10 - 2 (0-7)
b_7 → New flag;

L11 Temp 5 in °C b_3-b_0 → 0.1 (0-9)
H11 Temp 5 in °C b_7-b_4 → 1 (0-9)
L12 Temp 5 in °C b_2-b_0 → 10 (0-7)
b_3 → Sign;
H12 Humidity 5 in % b_7-b_4 → 1 (0-9)
(if value > 9 → sensor has no humidity)
L13 Humidity 5 in % b_2-b_0 → 10 - 2 (0-7)
b_3 → New flag;

H13 Temp 6 in °C b_7-b_4 → 0.1 (0-9)
L14 Temp 6 in °C b_3-b_0 → 1 (0-9)
H14 Temp 6 in °C b_6-b_4 → 10 (0-7)
b_7 → Sign;
L15 Humidity 6 in % b_3-b_0 → 1 (0-9)
(if value > 9 → sensor has no humidity)
H15 Humidity 6 in % b_6-b_4 → 10 - 2 (0-7)
b_7 → New flag;

L16 Temp 7 in °C b_3-b_0 → 0.1 (0-9)
H16 Temp 7 in °C b_7-b_4 → 1 (0-9)
L17 Temp 7 in °C b_2-b_0 → 10 (0-7)
b_3 → Sign;
H17 Humidity 7 in % b_7-b_4 → 1 (0-9)
(if value > 9 → sensor has no humidity)
L18 Humidity 7 in % b_2-b_0 → 10 - 2 (0-7)
b_3 → New flag;

H18 Temp 8 in °C b_7-b_4 → 0.1 (0-9)
L19 Temp 8 in °C b_3-b_0 → 1 (0-9)
H19 Temp 8 in °C b_6-b_4 → 10 (0-7)
b_7 → Sign;
L20 Humidity 8 in % b_3-b_0 → 1 (0-9)
(if value > 9 → sensor has no humidity)
H20 Humidity 8 in % b_6-b_4 → 10 - 2 (0-7)
b_7 → New flag;

Rain sensor

For the rain sensor, rocker strokes are counted and the status of the current 7 bit binary counter is transmitted. In order to obtain a rainfall quantity, the current status should be subtracted from that of the previous counter and the difference should be multiplied by 300 ml. This will give the decrease in rainfall.

L21 Rain Lo b_3-b_0 → lowest 4-bit from 7-bit rocker counter
H21 Rain Hi b_6-b_4 → lowest 3-bit from 7-bit rocker counter
b_7 → New flag;

Wind sensor

The wind is transmitted in BCD format in km/h, where the 10 position represents an exception, as it can be a value > 9. A wind speed of 123.4 km/h would be transferred in the form <12> <3> <4>. This yields a range of values from 0 to 159.9 km/h.
The wind direction is transferred with a resolution of 5 °, where the bottom bit in the first nibble represents the 5 value. Both of the upper bits of the 100 nibble give the range of fluctuation of the wind direction ((±0 °; ±22.5 °; ±45 °: ±67.5 °)

L22 Wind in km/h b_3-b_0 → 0.1 (0-9)
H22 Wind in km/h b_7-b_4 → 1 (0-9)
L23 Wind in km/h b_3-b_0 → 10 (0-15) ‡ max.: 159.9 km/h
H23 Direction in ° b_7-b_4 → 10 (0-9)
L24 Direction in ° b_1-b_0 → 100 (0-3)
b_3-b_2 → Fluctuation range; 00→ ±0 °; 01→ ±22,5 °; 10→ ±45 °; 11→ ±67,5 °
H24 Direction in ° b_0 → 5° Flag (0 → xx0°; 1 → xx5°)
b_3 → New flag;

Indoor sensor

The air pressure is transmitted in BCD format in hPa, where the 100 position represents an exception, as it can be a value > 9. An air pressure of 1,023 hPa would be transferred in the form <10> <2> <3>. This yields a range of values from 0 to 1,299 hPa.
The temperature/humidity are transferred as described previously.

L25 Press. in hPa b_3-b_0 → 1 (0-9)
H25 Press. in hPa b_7-b_4 → 10 (0-9)
L26 Press. in hPa b_3-b_0 → 100 (0-12)
H26 Temp Int in °C b_7-b_4 → 0.1 (0-9)

L27 Temp Int in °C b_3-b_0 → 1 (0-9)
H27 Temp Int in °C b_6-b_4 → 10 (0-7)
b_7 → Sign;
L28 Humidity Int in % b_3-b_0 → 1 (0-9)
(if value > 9 → sensor has no humidity)
H28 Humidity Int in % b_6-b_4 → 10 - 2 (0-7)
b_7 → New flag;

Brightness sensor

The brightness is transferred as a 3-digit BCD value (0-999) and a 2-bit factor (*1, *10, *100, *1000) in lux. This yields a range of values from 0 to 200,000 lux.

L29 Brightn. in lux b_3-b_0 → 1 (0-9)
H29 Brightn. in lux b_7-b_4 → 10 (0-9)
L30 Brightn. in lux b_3-b_0 → 100 (0-9)
H30 Factor b_5-b_4 → Factor (0→*1; 1→*10; 2→*100; 3→*1000)
b_6 → Sunshine flag;
b_7 → New flag;

Pyranometer

The irradiation effect is transferred as a 3-digit BCD value (0-999) and a 2-bit factor (*1, *10, *100, *1000) in $^1/_{10}$ W/m+. This yields a range of values from 0 to 99,900.0 W/m+.

L31 Effect in W/m+ b_3-b_0 → 1 (0-9)
H31 Effect in W/m+ b_7-b_4 → 10 (0-9)
L32 Effect in W/m+ b_3-b_0 → 100 (0-9)
H32 Factor b_5-b_4 → Factor (0→*1; 1→*10; 2→*100; 3→*1000)
b_7 → New flag;

The new flag reveals whether the sensor data has been received again in the time period between the reception of this data record and of the last data record If this flag is not in place, this means that the value is equal to twice that of the previous data record.

Request status
<SOH> ‘5’ (-total) <EOT>

Response from the interface: (16 byte – WS2500 ; 18 byte – WS2500PC)

B1	Temp sensor status 1
B2	Temp sensor status 2
B3	Temp sensor status 3
B4	Temp sensor status 4
B5	Temp sensor status 5
B6	Temp sensor status 6
B7	Temp sensor status 7
B8	Temp sensor status 8

B9	Rain sensor status		
B10	Wind sensor status		
B11	Brightness sensor status		
B12	Pyranometer status		
B13	Indoor sensor status		
B14	Interval time in min.		
B15	B_0	: 0 → WS2500 German/1 →	WS2500 English
		(0→ without HF	1→ with HF)
	B_1	: 0 → not DCF synchronised	1→ DCF synchronised
	B_2	: 0 → without DCF	1 → with DCF
	B_3	: 0 → Protocol V1.2	1 → Protocol V1.1
	B_4	: 0 → WS2500	1 → WS2500PC
	$B_5 – B_7$	: Indoor sensor address (0-7)	
B16	Version number		
B17	$B_0 – B_2$	: Rain sensor address (0-7)	
	B_3	: —	
	-		
	$B_4 – B_6$	: Wind sensor address (0-7)	
	B_8	: —-	
B18	$B_0 – B_2$	: Pyranometer sensor address (0-7)	
	B_3	: —	
	-		
	$B_4 – B_6$	: Brightness sensor address (0-7)	
	B_8	: —-	

Status of sensors:

- < 16→ unavailable
- = 16 → OK
- 17 .. 255 → Reception interference quantity +16

Interface init

<SOH> ‘D’ [interval] [V wind/rain] [V brightn./pyrano] [V indoor/version] (-total) <EOT>

The interface is re-initialised. In this way the interval time, the sensor addresses and the protocol version are re-set. Then the interface runs an HF initialisation.

[interval] → Time interval in which the data records are written to the buffer.

The value range extends over a period of 2 min. .. 63 min.
The recording time is calculated from: Time interval * 1024 [min]
Bit 7 is always set for this value.

[V wind/rain]	→		Wind and rain sensor addresses.
		$B_0 – B_2$	: Rain sensor address (0-7)
		B_3	: free
		$B_4 – B_6$	: Wind sensor address (0-7)
		B_8	: always 1
[V brightn./pyrano]	→		Brightness sensor and pyranometer address
		$B_0 – B_2$	: Pyranometer address (0-7)
		B_3	: free
		$B_4 – B_6$	: Brightness sensor address (0-7)
		B_8	: always 1
[V indoor/version]	→		Indoor sensor and protocol version addresses
		B_0	: 0 → Protocol V1.2 1 → Protocol V1.1
		$B_1 – B_3$	: free
		$B_4 – B_6$	: Indoor sensor address (0-7)
		B_8	: always 1

Response from the interface: (1 byte)
<ACK>

[#] format_wdat
[#] format_interf