Recommendations for the usage of heat meters in control applications

The following recommendations are based on the paper „AGFW-Empfehlungen für Wärmezähler mit M-Bus Interface für Regleranwendungen“ (Version 1 of Dec.30, 1996). It has been updated to reflect the actual work of WG4 of TC176 for a second edition of EN1434-3.

A) Heat meter

I) General

Heat meters for control applications should meet all standard requirements for normal heat meters including all requirements of EN1434. This is especially true for all requirements of measurement accuracy. In addition they should meet the requirements of M-Bus communication according to part 3 of this standard. All valid SND_UD telegrams should be acknowledged even if they are functionally not supported. SND_NKE telegrams should be acknowledged. The readout frequency should not be limited.
The answer delay of between 11 bit times and (330 bit times +50msek) = (4.6..187.5ms @2400 Baud) should be met.

II) Link Layer

The meters should (in addition to the default baudrate of 300 Baud) support also the baudrate of 2400 Baud which is used in all control communication. For switching from 300Baud to 2400 Baud either the baudrate switch command (CI=$BB) or an autospeed-detect technique should be supported. Only primary adressing is supported.

III) Application Layer: Data records

Each RSP_UD telegram of the heat meter should contain at least the following data records:

1.) Inlet temperature: Resolution better or equal to 0.1°C
2.) Outlet temperature: Resolution better or equal to 0.1°C
3.) Flow: Resolution better or equal to 0.2% of QNom
4.) Power: Resolution better or equal to 0.2% von PNom
5.) Status: At least general status bit of EN1434-3
6.) Additional values are allowed

The other data might vary. The sequence of the data records is arbitrary.

Heat meters with sequential multibyte telegrams should transmit these required data records in each telegram.

Heat meters which can fulfill all these requirement may ignore the function of an application reset (CI=$50) and the following subcode but must still acknowledge its reception.

Heat meters which cannot always automatically meet these requirements must support the function of „application reset with subcode“ at least for the combination of CI=$50 (application reset) with subcode $51 (following in the next byte) indicating that the heat meter is used in control application and that all its RSP_UD-telegrams must contain the control relevant data described above.

IV) Application Layer: Actuality of the data

1.) Temperature information: <=30s
2.) Flow information: Limiter applications: <=30s
3.) Flow information for regulation: For flow values between QMin and QMax: <=5s
4.) Status informationen: <=2min.

V) Application Layer: Acceptable data types (DIF´s):

1.) Binary: 8,16,24 or 32 Bit
2.) BCD: max. 2,4,6 or 8 digits
3.) ASCII, String-Data: not allowed
4.) Function type: always 0 (actual)
5.) The required data records shall use no DIF-Extension Other data records with DIF-Extensions are allowed.
6.) Thus the following DIF-types for the required data records should be supported: 1,2,3,4,$9,$A,$B,$C
7.) It is strongly recommended, that for new developments of control applications, the controller should also support the 32-bit floating point data type (Data type H, DIF=5).

VI) Application Layer: Acceptable units (VIF´s):

1.) Temperature: All acceptable units <=0.1°C
2.) Flow: All acceptable powers of ten of l/h, for which the resolution is better than or equal to 0.2% of QNenn.
3.) Power: All acceptable powers of ten of Watt, for which the resolution is better than or equal to 0.2% of PNenn.
4.) For the required data records VIF-Extensions are not allowed. Other data records may contain VIF-Extensions.

B) Controller

I) Start until first successful readout

After each power fail or other hard reset the controller should use the following sequence:

1.) Activate bus voltage to mark state.
2.) Wait >=5 seconds.
3.) Transmit at 300 Baud to each meter used:
SND_UD with CI=$BB to set bus baudrate to 2400 Baud.
If not acknowledged, repeat up to 2 times, then continue regardless of acknowledgement.
4.) Repeat step 3.) at 9600 Baud
5.) All following communication at 2400 Baud
6.) Send SND_NKE. If not acknowledged, repeat up to 2 times, then continue regardless of acknowledgement.
7.) Application reset with subcode: “Control“ via SND_UD with CI=$50 followed by $51. If not acknowledged, repeat up to 2 times, then continue regardless of acknowledgement.
8.) Periodic readout of each heat meter using REQ_UD2 to its (primary) adress. Up to two retries if heat meter does not answer with a correct telegramm.
9.) If still not successfull make up to three total restart attempts starting from step 3.
10.) If still no answer or no useful telegram received branch to system error handling.

II) Telegram processing

1.) Check Link Layer (Parity, Checksum etc.).
2.) Check frame, length, start/end bytes, adress, C-field, CI-field=$72.
3.) Check heat meter status bits.
4.) Segment data records: Take into account possible DIF- und VIF-Extensions.
5.) Consider all allowed datalengths of data records.
6.) Extract required records according their DIF and VIF.
7.) Convert data to internal data format
8.) Extract unit and power of ten from VIF and convert to internal controller units.