

RAMAR FLAT FILE LAYOUT

Overview – The RAMAR Flat interface is a 256 character fixed-width file format. Note that all fields make up one single line of text. Each line is a new record terminated with a <CR> <LF> sequence. An end of file terminator is not needed and the use of one will cause errors during the bit calculation which is performed before the file is loaded to ensure that the file contains a multiple of 256 characters. If the bit calculation does not pass then the file is not loaded.

The following table describes the RAMAR Flat file format in detail. FastTrackIT does not further specify the format of the information because, in general, the billing system will already define this format. However, the fields are limited to the number of characters specified under field width. In the case of numeric fields, the range of values is specified. Fields marked as required will cause a load failure or warning if left blank.

Filename and extension are fully customizable to suit the billing system. The default recommended extension is .DAT.

Table 1 Fields in RAMAR Flat File

Value	(Width) Range	Format	Required Value	Numeric Range	Load Error
Route Name	(32) 1 – 32	Alpha**	Yes		Failure
Customer Name	(32) 33 – 64	Alpha**	No		
Address	(80) 65 – 144	Alpha**	No		
Account	(32) 145 – 176	Alpha**	No		
Meter ID ¹	(32) 177 – 208	Alpha**	Yes		Warning
Meter Index ¹	(1) 209	Numeric	Yes	0 – 4	
Meter Dials	(1) 210	Numeric	Yes	0 – 6	Warning
Meter Decimals	(1) 211	Numeric	Yes	0 - (Meter Dials – 1)	Warning
Meter Flags	(4) 212 – 215	Alpha**	Ignored		
Meter Time	(14) 216 – 229	Alpha**	Ignored		
Meter Reading	(10) 230 – 239	Numeric*	Yes	0 – 16777215	Warning
Utility Code ²	(3) 240 – 242	Numeric*	Yes	0 – 255	Warning
XPDR ID ²	(8) 243 – 250	Numeric*	Yes	0 – 16777215	Warning
Tamper	(3) 251 – 253	Numeric*	Yes	0 – 255	
XPDR Flags	(2) 254 – 255	Alpha**	Ignored		
Meter Interface	(1) 256	Numeric	Yes	0 – 7	Warning

* Numeric fields should be left padded with zeros.

** Alpha fields may contain alpha-numeric or printable white space (i.e. not tab or control) characters.

¹ Meter ID + Meter Index must be unique

² Utility Code + XPDR ID must be unique

1. Route Name

Route Name can be used as an organizational tool because each unique route name will create a separate route folder in FastTrackIT. As many as 1024 routes are permitted in a single file.

2. Customer Name

This field stores the name of customer.

3. Address

This field is for customer address.

4. Account

This field can store the customer's account number.

5. Meter Index

This field determines the number of registers on a meter. 0 for a single register meter and the index number of the register for a multiple register (compound) meter. Non-zero values will be displayed as 'ID~n', where ID is the Meter ID and n is the Meter Index.

6. Meter Dials

This field allows the download file to specify the number of significant digits (left most) to represent in the reading. This can be used when it is only necessary to gather the reading from the white dials on the meter. If Meter Dials is 0, then FastTrackIT returns the complete reading obtained from the TransPondIT with no adjustment. Even though our maximum reading is 16,777,215 (eight digits) we can only read up to 6 dials since we would run out of digits half-way through the seventh dial.

NOTE: To pass the unaltered reading from the meter to billing put a zero in the Meter Dials & Decimals fields.

7. Meter Decimals

This field allows the download file to specify the number of digits after the decimal (black dials on the meter) to represent in the reading.

Caution: It is up to the billing system to ensure that the combination of the Meter, Register and TransPondIT and the Meter Dials and Meter Decimals fields are selected such that the reading returned from FastTrackIT is in a format compatible with the billing system's expected data.

Example:

Reading on meter 1234.56 (units)

TransPondIT returns 123456

Meter Dials = 4; Meter Decimals = 0 – FastTrackIT returns 1234 as the billable reading.

Meter Dials = 5; Meter Decimals = 1 – FastTrackIT returns 12345

8. Meter Flags Field

This field is ignored from the download file and may therefore contain any characters.

Four characters (in order)

M or space - indicating that a meter reading was manual i.e. not radio
O or space - indicating that a meter reading overflow has (not) been detected
R or space - indicating that a meter reverse reading has (not) been detected
Unused

9. Meter Time Field

This field is ignored from the download file and may therefore contain any characters.

Fourteen numeric digits (or spaces) indicating read time as YYYYMMDDHHMMSS. This field will be left as 14 spaces if no reading has been achieved.

10. Meter Reading Field

Ten numeric digits (left padded with 0's) used as previous reading for download and new reading for upload. If a new reading is not collected the previous reading is returned.

11. Utility Code

Code provided to the customer by the business partners. The code should be the same for all TransPondITs.

12. TransPondIT ID

Unique ID stored in each TransPondIT

13. Tamper

The Tamper field is ignored from the download file and therefore may contain any alphanumeric data.

The upload file will however contain tamper codes. Tamper flag indicates that the TransPondIT wiring has been tampered with for a pulse type meter or for an encoded meter the data being provided by the meter is invalid.

Tamper Count Error Codes for Encoded Meter Types

Meter Type

Error Code	Description of Problem
ABB Scancoder	
00	No errors
01	Open circuit wheel contacts
02	Short circuit wheel contacts
64	No communication with meter
Sensus ECR, Schlumberger ARB5 & 6	
00	No errors
64	No data or bad data
Pulse >0	Incremented each time circuit is made/broken

The numbers are in decimal. For the ABB they may be added together to indicate more than one error.

For encoded meter, if disconnected and then reconnected the tamper count will return to zero and the meter reading will be correct.

14.XPDR Flags Field

This field is ignored from the download file and may therefore contain any characters.

Two characters (in order)

X, T, S, M, A or space

- X - Strange readings detected from transponder.
- T - Tamper detected (pulse interface only)
- S - Failed to read meter 'some' (0-50% of messages), (encoder interface only)
- M - Failed to read meter 'most' (50-100% of messages), (encoder interface only)
- A - Failed to read meter 'all' (100% of messages), (encoder interface only)
- Space - Failed to read meter 'none' (0% of messages), (encoder interface only)

W, S, M, A or space

- W - 'Wrap' detected (pulse interface only)
- S - Read bad data 'some' (0-50% of messages), (encoder interface only)
- M - Read bad data 'most' (50-100% of messages), (encoder interface only)
- A - Read bad data 'all' (100% of messages), (encoder interface only)
- Space - Read bad data 'none' (0% of messages), (encoder interface only)

15.Meter Interface Field

Table 2 – RAMAR Flat Meter Interface Codes

Meter Interface	Meter Type	TransPondIT
X	Unknown	N/A
0	Standard Pulse Generator	TT-915/F-X1 TT-915/F-X7
1	Short Pulse	TT-915/F-X1
2	Dual Pulse	TT-915/F-X2
3	ABB Scancoder	TT-915/F-X3
4	Standard ECR Meter	TT-915/F-X4
5	Schlumberger ARB5	TT-915/F-X5
6	Schlumberger ARB6	TT-915/F-X6
7	TBA	T B A

TMC – TRANSPONDER/METER CONNECTIONS

Certain Billing Interfaces and therefore the relevant FastTrackIT Data Interfaces do not support the inclusion of data in the Billing System to link a Meter Register to a Transponder. To allow the FastTrackIT system to be used successfully with such systems, a method for providing the logical connection (linkage) information is provided by FastTrackIT. This method is called TMC.

1. Invoking TMC

TMC is invoked automatically where the FastTrackIT Data Interface indicates that the Billing Interface does not support linkage information.

TMC information is held in the FastTrackIT database until the billing interface file is saved or unloaded. At this point the data is saved to a file (or files). These files are created on a route-by-route basis and have the name 'route_name.tmc'. These files are saved into the same directory from which the billing interface file was loaded.

Please note that it is the responsibility of the end user to ensure that these files are adequately backed up and are available should the same billing file be reloaded for subsequent meter reading sessions.

2. TMC file format

A TMC file is essentially a simple comma separated variables file. As such it can be maintained using Microsoft Excel (or other spreadsheets that can load and save in CSV format).

The file consists of a number of lines where each line contains at least 4 (four) comma-separated fields but may contain 5 (five). The first blank line terminates the file.

2.1. Fields

Utility Code, Transponder ID, “Meter ID”, Register, MIF

Utility Code: Numeric range 0...255 (standard utility code). This is the utility code of the transponder as configured during installation.

Transponder ID: Numeric in range 0...16777215 (standard transponder id). This is the id of the transponder as configured during installation. The combination of utility code and transponder id must be unique.

Meter ID: This is the meter serial number optionally quoted text up to 32 characters. Note that neither or both quotes must be present.

Register: Numeric in range 0...9. An index of 0 indicates that there is only one register in the meter. Multiple register meters must have sequentially allocated indices. The combination of meter id and meter register index must be unique.

MIF: Optional Meter Interface type. See the RAMAR Flat Meter Interface Codes Table in Appendix B for a complete listing of meter types.

Example:

```
11, 2130760, 2130760, 0, 5  
11, 4226421, 422642, 1,  
11, 4226422, 422642, 2  
11, 601, "ZA 1985562", 0
```

In the above example, there are 4 transponder/meter connections. All transponders have been configured in utility code 11.

In the first connection, the transponder has been configured with the same id as the Meter ID (This will only be possible if the meter id is solely numeric and in the range 0...16777215). The meter is a single register meter and therefore the meter index is 0. The transponder MIF has been specified as 5.

In the second and third connection, the meter is a dual register meter (as indicated by the indices 1 and 2). The transponders attached to the meter have been configured with the same id as the meter BUT with the register index appended (by multiplying the meter id by 10 and adding the register index). This will only be possible if the meter id is solely numeric, in the range 0...1677721 and the register index is in the range 1...5. Note that this simple encoding scheme potentially creates duplication of ids between multiple register and single register meters and must therefore be used consistently and with the utmost caution.

In the final connection, the meter is again a single register but in this case, due to the presence of one or more non-numeric characters in the meter id it is not possible to configure the transponder with the meter id. Therefore a unique (within utility code) id has been manually allocated.