EPSON®

EXCEED YOUR VISION

EpsonFpMate

Development Guide

REVISION HISTORY

Revision	Pages	Comments	Date	Author(s)
A	All	Preliminary	1st March 2005	C. Guietti
B	All	First edition	15th March 2005	C. Radaelli
C	All	Add some clarification sentences	27th May 2005	C. Radaelli
D	All	Management of display messages; Misspelling corrections; Review of the sample files	9th June 2005	C. Guietti
E	All	Added "Open Drawer" command Added modification of "Not allowed words" Added barcode printing	5th August 2005	C. Guietti
F	All	Added "Print from DGFE of last receipt" command	6th February 2006	C. Guietti
G	8	FP210 and FP285 have been replaced by FP81 and FP90II	7th February 2006	C. Guietti
H	All	Added: management of Fiscal closures log. management of graphic files direct EpsonFp2.ocx commands	21st June 2006	C. Guietti
I	All	Added: Option to avoid the Message Box: "Paper Near to end" Language options Command to read and save E/J Command to read and save Printer Status	6th December 2006	C. Guietti
J	All	Updated description of methods: printSetProperty printGetProperty printRecTotal (ticket payment mode)	11th January 2006	C. Guietti
K	All	Added notes related to fiscal receipt Errors corrections	23rd January 2006	C. Guietti
L	All	Added management of EFT-POS devices Added management of Fiscal Documents	16th May 2008	C. Guietti
M	All	Updated rec adjustment with uplift features Update sample files	21st May 2008	C. Guietti
N	All	Added info about error management (Printer off-line and paper near to end)	21st August 2008	C. Guietti
O	All	Path management for graphic files	22nd September 2008	C. Guietti
P	All	Added Greece functionalities	3rd December 2010	C. Guietti
Q	All	Improved writing style. Reformatted tables. Corrected minor errors.	1st April 2011	P. Barnett
R	10	Added "Restriction Option" to manage "Paper Near to End" events.	30th June 2011	C. Guietti

Revision	Pages	Comments	Date	Author(s)
S	All	Improvement of USB Printer management (more than one printer allowed).	17th April 2012	C. Guietti
T	All	Added management of external Plu.	15th May 2012	C. Guietti
U	All	Updated D-type management. Added management of Receipt Image Added management of Sound Buzzer	5th December 2012	C. Guietti
V	All	Added management of resetPrinter command	18th September 2014	C. Guietti
W	All	Document rewritten throughout. Added RT printer functions and related material. Expanded D-Type commands, fields and examples. Removed all Greek references.	14th March 2019	P. Barnett
X	All	Added printDuplicateReceipt examples. Removed EFT-POS Q-Pay chapter as no longer relevant. Added "Windows Scheduled Start on Reboot" chapter. beginTraining and endTraining enter and exit Demo RT mode in the case of RT printers (version 5.6.0.31 onwards). Added printRecItemVoid alias. Added annotations that PLU commands are not supported with RT models. Added aliases throughout. Added 115200 baud rate. Expanded queryPrinterStatus to included new RT status function (version 5.6.0.31 onwards). Updated printBarCode type names. Updated printRecMessage. Added D-Type void and refund documents (version 5.6.0.31 onwards). Expanded D-Type commands, fields and examples. Added output file keyword changes table.	18th September 2020	P. Barnett

Index

1   Overview

The aim of this manual is to provide information pertinent to software designers and software houses who wish to easily interface Epson fiscal printers using the EpsonFpMate fiscal driver. All communication protocol management is handled and controlled by EpsonFpMate.

Epson FP printers can be easily managed by means of simple TXT or XML files. It also has a control panel accessible via the Systray. EpsonFpMate can be programmed to create output log files that contain all sorts of useful information such as that relating to fiscal receipt print outs or from the issuing of commands.

EpsonFpMate is provided with two utilities:

- EpsonFpMateConfig: Used to set the environment.
- EpsonFp2Config: A simpler configurator (supports only one printer).

EpsonFpMate has been developed under Windows .NET technology. When using the Windows 98/2000/XP operating systems, the .NET framework (minimum version 2.0) must be loaded on your computer. Windows 7 and beyond should already include .NET Framework. Windows 10 either needs .Net Framework to be installed or .NET Framework 3.5 needs to be manually activated through the "activate/deactivate Windows functions" configurator.

2   List of Related Documents

FP xxx manuale operatore
EpsonFp2 Active X / OCX Control user manual

3   Abbreviations

EJ: Electronic Journal (Dispositivo Giornale di Fondo Elettronico – DGFE)

4   Installation

If the following errors occur, please make sure .NET Framework is installed.
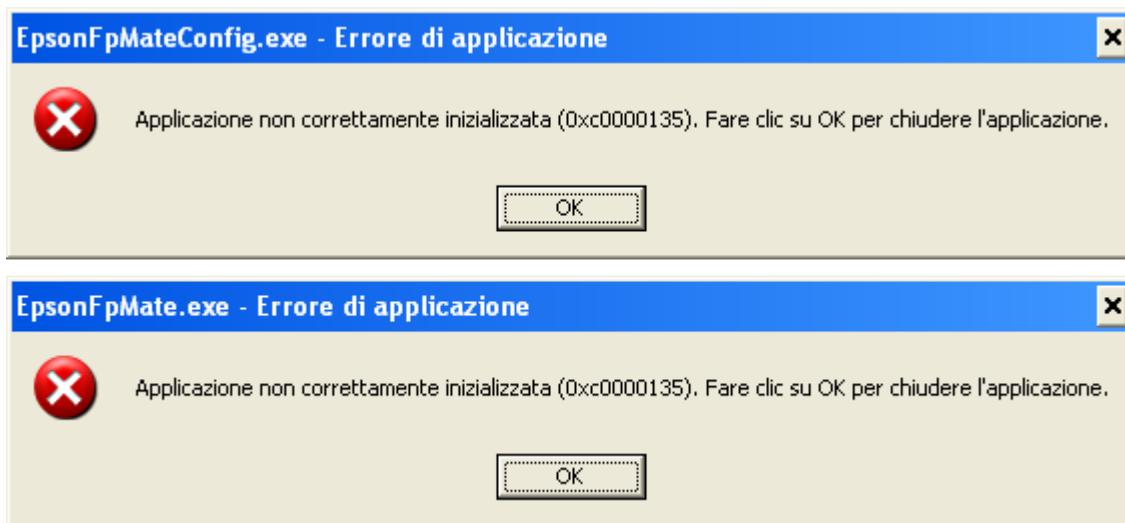
5   EpsonFpMateConfig

5.1   Overview

After having installed EpsonFpMate, the EpsonFpMateConfig tool must then be open. This utility creates and edits the "Settings.xml" file that is read by the EpsonFpMate application to configure its behaviour. The Settings.xml path depends on the version of EpsonFpMate, the variant of Windows and the type of active user. Examples are:

- C:\Program Files\EpsonFpMate
- C:\Documents and Settings\All Users\Dati applicazioni\EPSON\EpsonFpMate
- C:\ProgramData\EPSON\EpsonFpMate
- C:\Users\<username>\AppData\Local\VirtualStore\ProgramData\EPSON\EpsonFpMate

For two reasons, close EpsonFpMate before launching this utility. Firstly, EpsonFpMate only reads the new parameters when it is launched. Secondly, the communication test on the EpsonFpMateConfig tool could conflict with a running EpsonFpMate process. The following can be configured:

- Country Options
- Operating Setup
- Error and Paper Management
- Commands Enabled (Control Panel Buttons)
- Input File Options
- Output File Options
- Log Options
- Fiscal Printers
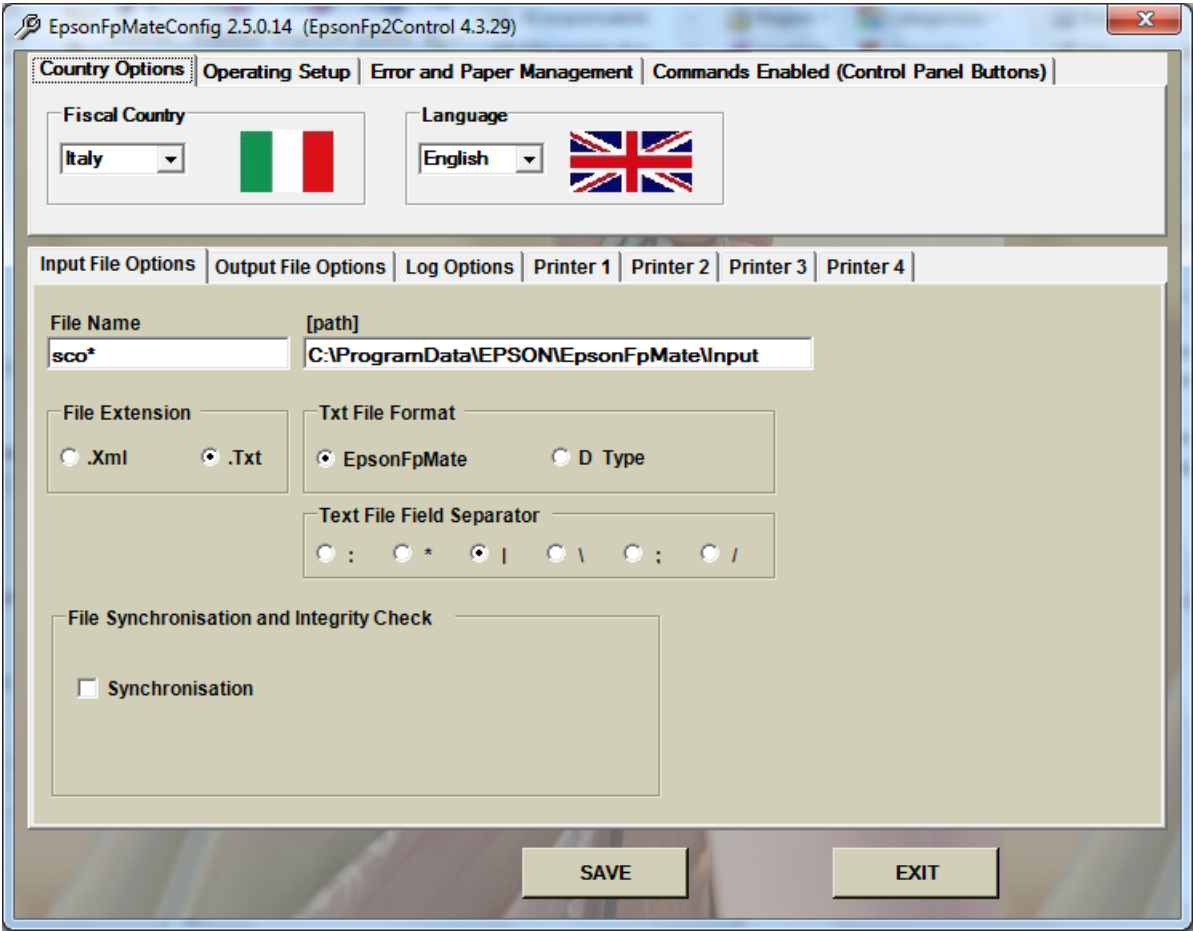
5.2   Country and Language Options

5.2.1   Fiscal Country

Selects the country where the printer is to be used. Choose Italy. The Greek option is no longer supported.
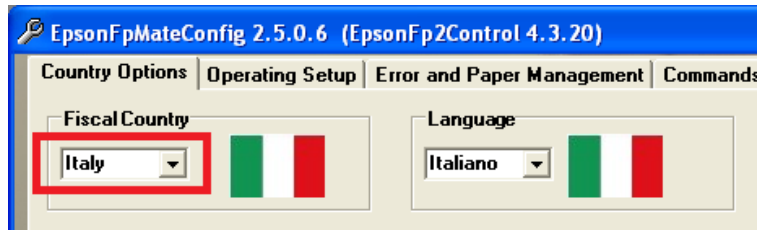
5.2.2   Language

Selects the language for menus, descriptions and messages generated by EpsonFpMate and in its Control Panel. N.B. This is not the same as the language used in the EpsonFpMateConfig forms. Choose Italian or English.
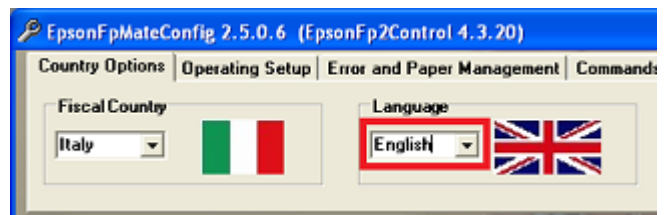
5.3 Operating Setup

5.3.1   Operating Mode

EpsonFpMate can operate in either "Free Running" or "Single Shot" mode.
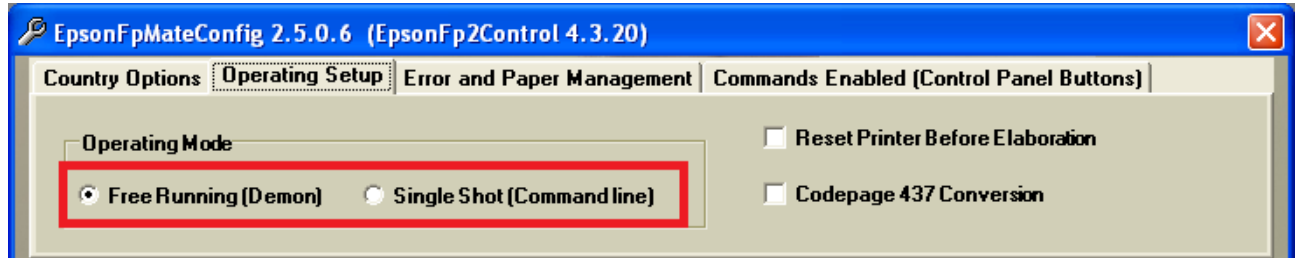
- In "Single Shot" mode, the application must be started by a command line string that has the file to be processed as an argument plus an optional check file argument separated by pipe ("|"). It then closes.

- In "Free Running" mode, once the application has been started, it is always running and scans (every 125 mS) for files to process in a chosen input folder. The following points should be taken into consideration:

    o The naming convention rule and the path where the input files will be located must be defined. Epson recommend using different filenames, adding a timestamp for example.

    o Input files are deleted immediately after processing whether successful or not.

- In "Free Running" mode, a watchdog process is also started (EpsonFpMateSpv). It monitors the EpsonFpMate process and will automatically restart it if it has closed unexpectedly.

5.3.2　Reset Printer Before Elaboration

EpsonFpMate can be configured to perform a reset or convert the character set:
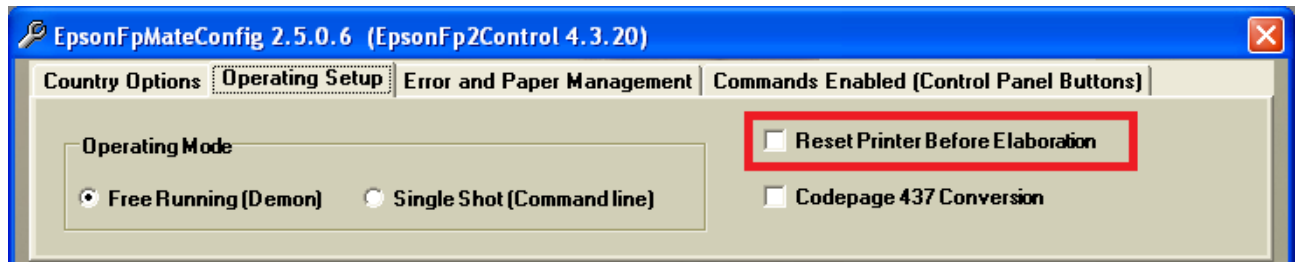
The "Reset Printer Before Elaboration" option applies to the following file types:

- printerFiscalReceipt
- printerNonFiscal
- printerFiscalReport
- printerFiscalDocument

When activated, a printer reset command is sent automatically to the printer prior to file elaboration. A reset carries out the following operations:

- Places the printer in the "STATO REGISTRAZIONE" condition irrespective of the current state.
- Closes any open non-fiscal receipt.
- Cancels any open fiscal receipt / commercial document.
- Cancels any open credit note / refund document or void document.
- Cancels any open direct invoice printing two copies.
- Cancels any open free invoice printing a single copy.
- Cancels any open box office ticket furthermore switching the printer into fiscal receipt mode ("SCONTRINO FISCALE").
- Unblocks the keyboard previously blocked via the directIO|opBloccoTastiera (alias directIO|DisableKeyboard) command.
- Erases the keyboard buffer.
- Terminates a possible SET 22 printer test condition.

5.3.3   Codepage 437 Conversion

Epson printers use codepage 437 to print extended characters such as accented vowels:
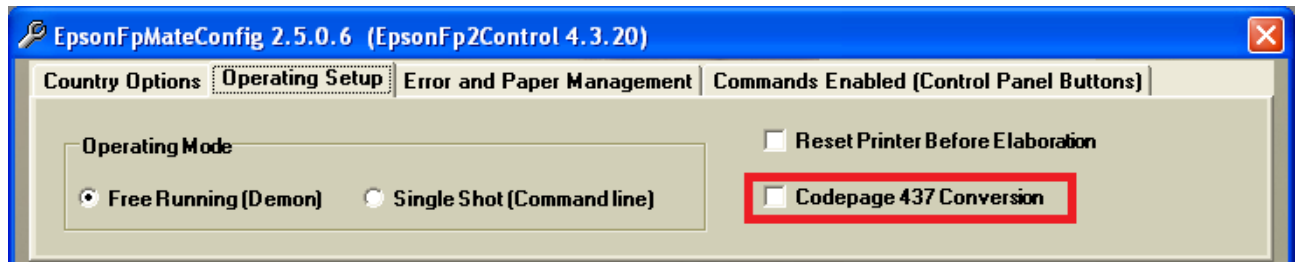
Windows PCs in Italy usually use codepage 1252. In order to seamlessly print and display extended characters such as those in item descriptions, EpsonFpMate can automatically perform a conversion to codepage 437. The setting also applies to lines read from the DGFE / MPD whereby the conversion is from codepage 437. The communication protocol document fully describes codepage 437. To correctly manage extended characters, Epson recommends input files with either UTF-8 or UNICODE encoding. Please note the following:

- Characters from 176 to 223 are not converted to or from as they are graphical.
- The Euro symbol does not exist in codepage 437 and is a special case. Euro symbol handling depends on the fiscal printer model:

  o **"MF" models**

    "MF" models cannot print the Euro symbol. They can however display it. By default, code 156 (£) is mapped to the Euro symbol via the SET 15/16 setting so it can be used in displayText messages. Code 156 received in a sale, refund or payment etc will both print and display the £ symbol rather than the Euro symbol.

  o **"RT" models**

    "RT" models can both print and display the Euro symbol. The SET 15/16 setting indicates the incoming code used for the mapping. The default is 156 (£). Code 156 received in a sale, refund or payment etc. will both print and display the Euro symbol. Code 156 received in a displayText message will display the Euro symbol. In addition, the Euro symbol is also hard mapped to code 96 (grave accent `) for printing only.

5.4   Error and Paper Management

If an error occurs, EpsonFpMate saves the error codes in the return file and furthermore stores them in the output log files. Furthermore, optional pop-up message boxes regarding errors and warnings can be activated. Certain options are exposed or hidden depending on previous selections.

5.4.1   Message Boxes

When enabled, EpsonFpMate requires operator intervention in order to decide how to manage the error:
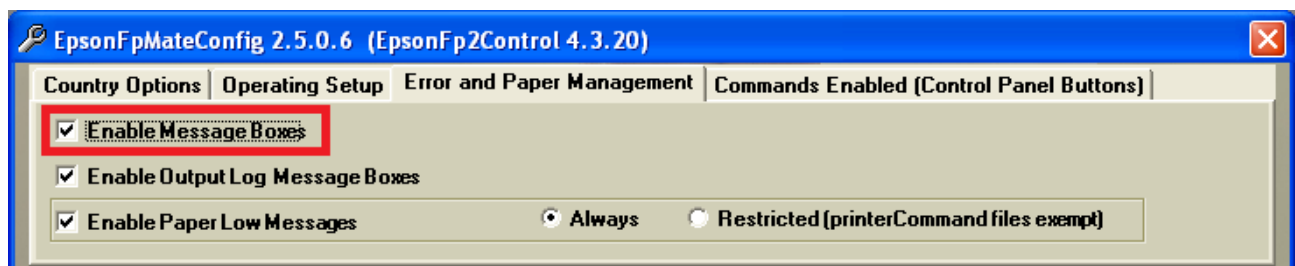
When disabled, EpsonFpMate does not require operator intervention. Furthermore, as soon as the printer goes offline (typically caused by a paper out condition), EpsonFpMate waits the configurable "Paper Replace Time" period (default 80 seconds):
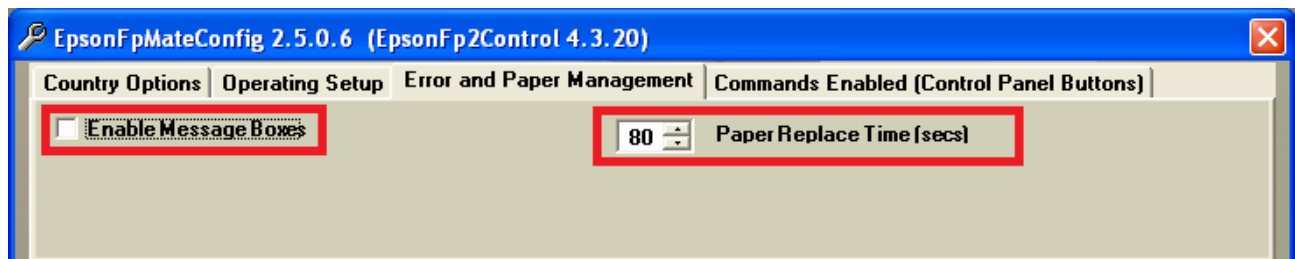
5.4.2   Output Log Message Boxes

Output Log messages relate to errors during file elaboration and are saved in the Output Log (if enabled). The details of the error can also be presented as a message box:
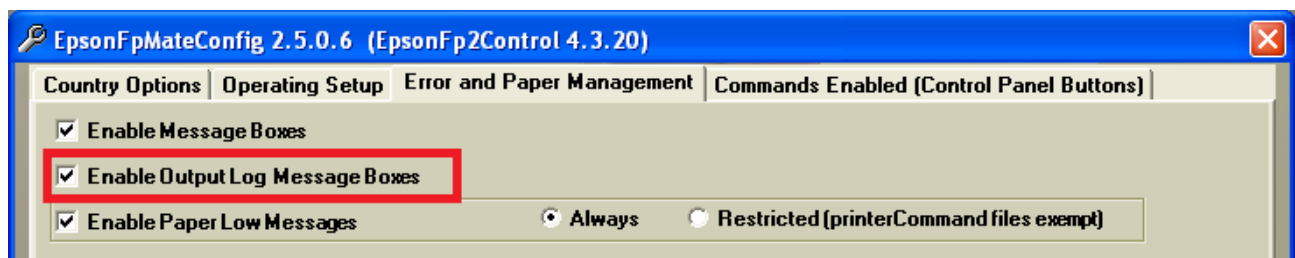
5.4.3   Paper Low Messages

The diagnostic "Paper Low" warning can also appear as a message box:
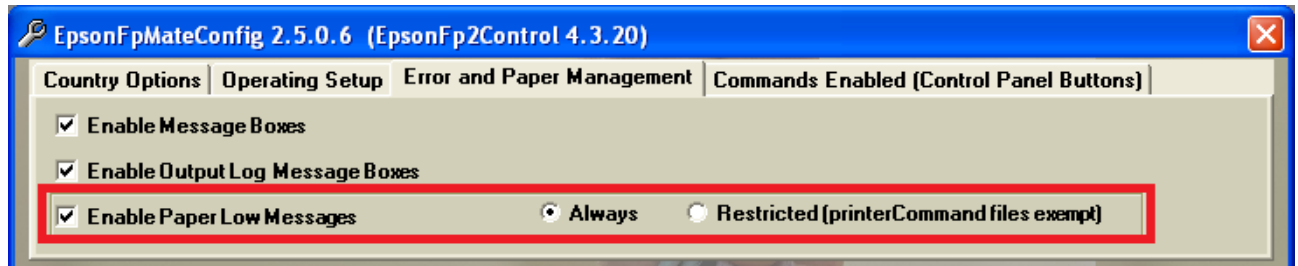
If the "Restricted" option is selected, paper low warnings do not appear when a printerCommand file is processed.

5.4.4   Message Box Setting Combinations

Message Box	Paper Low Message	Paper Low Condition	Out of Paper Event (Printer Offline)
Enabled	Enabled – sub option "Always"	All receipts are printed in their entirety. Afterwards, a message box appears on the screen indicating that the paper is low. To proceed, the operator simply clicks the OK button.	The behaviour depends at which point the paper out occurred. Paper Out after having processed and sent all file commands: A paper low alert appears on the screen. The operator can simply acknowledge it. After replacing the paper roll and closing the printer cover, the fiscal printer firmware may or may not reprint the fiscal receipt or commercial document depending on the exact point the paper ran out.
Enabled	Enabled – sub option "Restricted"	All receipts are printed in their entirety. The message box indicating that the paper is low does not appear when "printerCommand" input files are processed.	Paper Out before having processed and sent all file commands:
Enabled	Disabled	Paper low condition has no effect. It is ignored by EpsonFpMate. It continues working normally.	A windows form appears requesting either to retry the current file or cancel the current operation. In the latter case, the output file indicates that the task was not completed. A second message box appears indicating Error 3 (offline). After replacing the paper roll and closing the printer cover, if a fiscal receipt or commercial document was in progress, the fiscal firmware reprints the lines up to point that the paper ran out. If Retry is selected, the remaining lines are then printed out. Non-fiscal receipts are never reprinted automatically by the fiscal firmware.
Disabled	Selection not shown		If the printer goes back online within the "Paper Replace Time" period, processing resumes. If the timeout expires, EpsonFpMate cancels the current file and writes appropriate information in the output files. If it was printing a (fiscal/non-fiscal) receipt or commercial document, it remains in the open condition.

5.4.5   Printer Power Supply Interruptions

The error management described in the previous chapter also applies whenever a printer power supply interruption occurs while EpsonFpMate is active.

Power Supply Interruption whilst there are no EpsonFpMate Requests for the Printer	Power Supply Interruption whilst Printing a Document
Minimal effect. After power is restored to the printer the system is ready as before. In the case of TCP/IP, the connection will be lost and there will be a small delay when the next file is processed or when the control panel is opened while EpsonFpMate reopens the connection.	As with printer offline or out of paper events, the behaviour is related to the Message Box settings. 1) Message Box Enabled A windows form alert appears on the screen regarding the event. Once the printer is ready, the operator can either choose to either cancel or retry. Retry means that EpsonFpMate proceeds from the point at which communication was lost. Cancel abandons the current file processing. It does not cancel the open document. 2) Message Box Disabled EpsonFpMate abandons the current file processing. The return file indicates that the current working state has been interrupted because the printer is offline (the task has not been completed). In both cases, if it was printing a (fiscal/non-fiscal) receipt or commercial document, it remains in the (fiscal/non-fiscal) receipt open condition. Upon power restoration, the printer prints a power interruption message, cuts the paper and then prints it again up to the same point. This is also true with reports.

5.4.6   Control Panel Buttons – Enabling and Disabling

The EpsonFpMate control panel allows some common requests to be sent to the printer by an operator. There are twelve request buttons on the EpsonFpMate control panel. Some of them can be disabled with EpsonFpMateConfig. The buttons that can be enabled/disabled are:

- Z-report printout (Fiscal daily closure).
- Void receipt.
- Forced closure of open fiscal receipt or commercial document.
- Forced closure of open non-fiscal receipt or management document.
- EpsonFpMate program shutdown. When disabled, it is not possible to close the program even using the "Alt F4" sequence.
- Date and time setup.

Disabled buttons are not visible on the EpsonFpMate control panel:
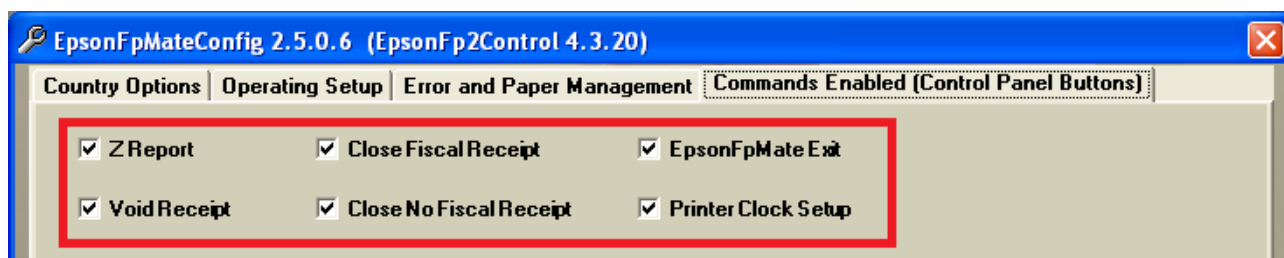
5.5   Input File Options

The following parameters can be configured:

- File Name and [path] – Mandatory. The input file naming convention rule and the path where EpsonFpMate searches for files when operating in "Free Running" mode. Paths can also be on shared network drives. EpsonFpMate is not case-sensitive to filenames.
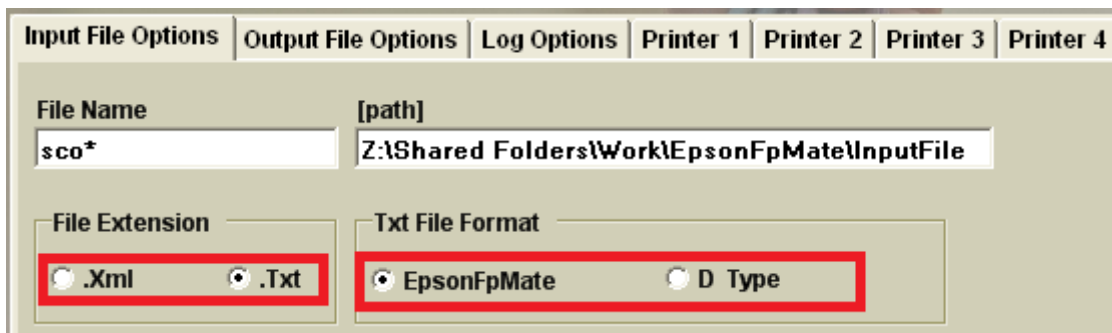
The filename does not include the extension since EpsonFpMate uses either .txt or .xml fixed extensions. The wildcard * is permitted; Epson recommends using it in conjunction with fixed strings rather than using it alone.

Furthermore, Epson does not recommend using the same filename repeatedly but rather to use sequence numbers and/or timestamps in the filename. This aids maintenance and troubleshooting.

- File Extension and Txt File Format – EpsonFpMate supports three input file formats. Initially choose between two file extensions:

  - .Xml

  - .Txt (of which there are two possible formats):

    - EpsonFpMate.
    - D Type. EpsonFpMate can parse D Type formatted files.

In both cases EpsonFpMate is not case-sensitive to file extensions.

- Text File Field Separator – The choices only appear if the File Extension is ".Txt" and the format is EpsonFpMate. In this mode, the separator (delimiter) used to differentiate the different data fields inside the records must be selected. The default is "|" (pipe). Commas and dots are not valid as they could be confused with prices.

- File Synchronisation and Integrity Check – Provides extra file handling capabilities:

  o When "Synchronisation" is selected, EpsonFpMate expects an additional check file associated with each input file. Firstly, the input file is saved in the input directory. Secondly, the check file (having the same filename but with a different extension) is saved perhaps after a short delay. This sequence is controlled by your software application. The check file indicates to EpsonFpMate that the file transfer of the input file has been completed thus preventing truncated files from being read. The check file can be empty. The default extension is "on". EpsonFpMate supports extensions up to three characters in length. The dot character "." must not be used.

  o When "Integrity Check" is selected, EpsonFpMate also verifies the integrity of the input file. Your software application must provide the MD5 string in the check file. EpsonFpMate applies an MD5 algorithm to the associated input file data and compares it to the string in the check file. The input file is only processed if it's MD5 result and the check file's string match, otherwise no processing is performed. MD5 is the only algorithm type currently supported.

5.6 Output File Options

The following parameters can be configured:

- Output File – If enabled, it creates an output file in the directory specified under [path] for each input file received. Epson do not recommend using the same input filename time after time but rather to use sequence numbers and/or timestamps in the filename. This is because the output file of the same name would be constantly overwritten without any warnings.
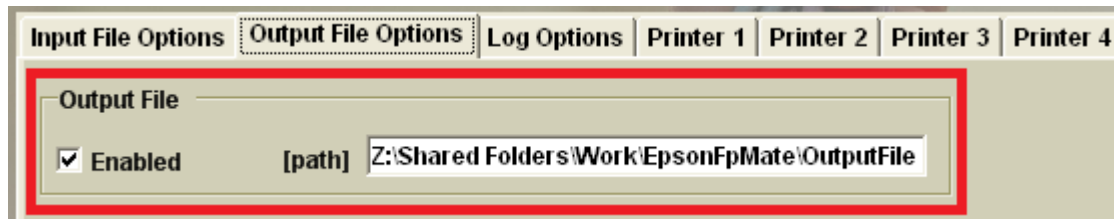
- Output Filename Timestamp Insertion – If selected, EpsonFpMate automatically modifies the output filename appending the timestamp in the yyyymmddhhmmss format.

  - Output Filename Prefix / Suffix – Permits one of four possible modes:

    1. Prefix – The user configurable string is inserted as a prefix in the output filename.

    2. Suffix – The user configurable string is inserted as a suffix in the output filename. If Output Filename Timestamp Insertion is activated, the suffix is placed after the timestamp.

    3. None – No prefix or suffix is inserted in the filename.

    4. Retrospective – EpsonFpMate behaves in the same manner as prior to the introduction of the new prefix / suffix options. The fixed "Out" string is inserted as prefix or suffix according to the type of file:

       a. TXT file – The txt extension is replaced with Out.
          e.g. sco1234.Out / sco1234-20200904121430.Out

       b. XML file – "Out" is inserted as a prefix and the xml extension is unaltered.
          e.g. Outsco1234.xml / OutSco1234-20200904121430.xml

    In both cases Timestamp Insertion is still possible.
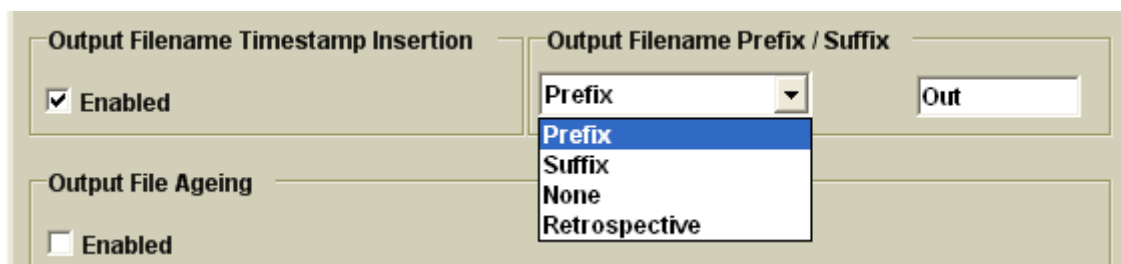
- Output File Ageing – Sets automatic deletion of old output files. There are two thresholds that can be configured:

    1. Max Days – The maximum permissible age of the file after which it will be deemed ready for deletion.
    2. Max Old Output Files – The maximum number of output files allowable.

In both cases the value zero or blank deactivates the setting. Either one of the thresholds or both can be configured.
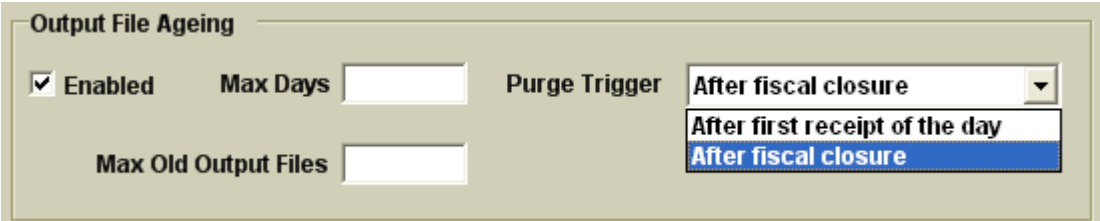
The deletion routine is activated in one of two ways:

    1. After first receipt of the day.
    2. After fiscal closure.

The routine begins by deleting files deemed old by virtue of the "Max Days" setting. During this sequence, the number of output files is recalculated. After the sequence has finished, EpsonFpMate compares the number of remaining files with the "Max Old Output Files" setting. If the number of files exceeds the threshold, it begins deleting them (in age order starting from the oldest file).

The "After first receipt of the day" option tells EpsonFpMate to perform ageing after having emitted fiscal receipt / commercial document number 1 of the day.

The "After fiscal closure" option tells EpsonFpMate to perform ageing after processing the printZReport, printZTotReport or printXZReport commands in a printerFiscalReport input file or by clicking the "Daily Closure" or "Daily Closure (Financial Report)" buttons on the EpsonFpMate control panel.

5.7 Log Options

Up to four types of log can be enabled:

- Input Log – Keeps a copy of the processed input files in a single file. The default name is "InputLogFpMate".

- Output Log – Keeps track of error conditions in a single file. The file is only written to if errors have been encountered. The default name is "OutputLogFpMate".

  Z-Report Log – Logs printerFiscalReport file processing and "Daily Closure" or "Daily Closure (Financial Report) " button click events on the EpsonFpMate control panel. The default name is "RepZLog".

- EpsonFpMate Log – Detailed log that describes the principal EpsonFpMate functions such as file elaboration, printer connections and errors. The default name is "EpsonFpMateLog".

The file storage locations are individually configured under [path]. Network drive shares can also be specified.

Two further settings are possible:

- Max Log File Size – Defines the maximum log file size allowable. Once the maximum size is reached, the file is renamed by inserting "-" followed by the timestamp and ".old" before the file extension. For example:
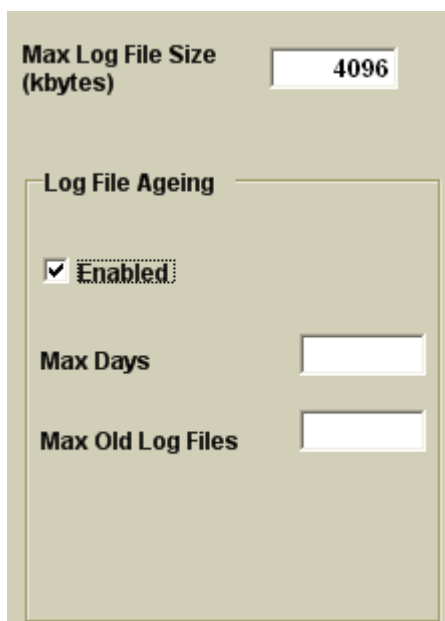
  OutputLogFpMate.txt becomes OutputLogFpMate-20200904160301.old.txt

  A new log file is then created.

- Log File Ageing – There are two thresholds that can be configured:

  1. Max Days – The maximum permissible age of the file after which it will be deemed ready for deletion.
  2. Max Old Log Files – The maximum number of log files allowable.

In both cases the value zero or blank deactivates the setting. Either one of the thresholds or both can be configured. Each of the four log file types are treated separately.

Whenever a log file exceeds the "Max Log File Size" threshold, EpsonFpMate initiates ageing prior to renaming and creating a new file. The routine begins by deleting files deemed old by virtue of the "Max Days" setting. During this sequence, the number of log files is recalculated. After the sequence has finished, EpsonFpMate compares the number of remaining files with the "Max Old Log Files" setting. If the number of files exceeds the threshold, it begins deleting them (in age order starting from the oldest file).

```
Max Log File Size
(kbytes)                    4096

┌─Log File Ageing──────────────┐
│                              │
│  ☑ Enabled                   │
│                              │
│                              │
│  Max Days         [        ] │
│                              │
│  Max Old Log Files [        ]│
│                              │
│                              │
└──────────────────────────────┘
```

5.8 Other Logs

EpsonFpMate also writes the following log files irrespective of the EpsonFpMateConfig settings:

1. EpsonFpMateCrashLog.txt
2. EpsonFpMateSpvLog.txt (watchdog log file)

The path depends on the version of EpsonFpMate, the variant of Windows and the type of active user. Examples are:

C:\Program Files\EpsonFpMate
C:\Documents and Settings\All Users\Dati applicazioni\EPSON\EpsonFpMate
C:\ProgramData\EPSON\EpsonFpMate
C:\Users\<username>\AppData\Local\VirtualStore\ProgramData\EPSON\EpsonFpMate

5.9 Printers

Printer communication options need only be set if more than one fiscal printer is connected to the PC and managed by EpsonFpMate otherwise the EpsonFp2Config utility can be used. EpsonFpMate can manage up to four printers. For each installed printer, the following must be configured:

- Printer Installed (checkbox).
- Communication Channel (Serial RS-232, LAN TCP/IP or USB) and the related parameters.
- Cash Register Number (CashDesk ID / Numero Cassa): range 0 - 999.
- JavaPOS Mode (checkbox).

EpsonFpMate auto-detects the printer model.

- Printer Installed
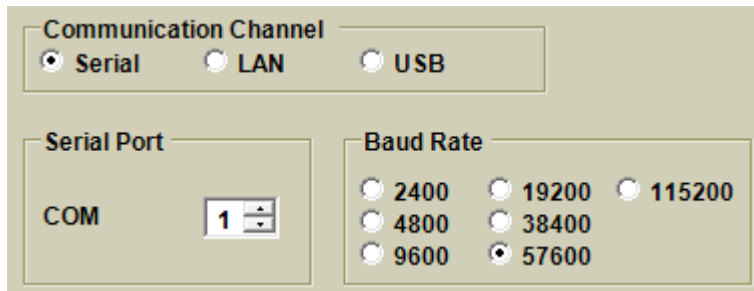
  Up to four printers can be managed. Warning: If only one printer is connected to the PC and is managed by EpsonFpMate, the communication setup may be performed using the alternative EpsonFp2Config tool. In this case, the communication parameters cannot be modified with EpsonFpMateConfig and are greyed out. In the example below, LAN communication settings were configured using the EpsonFp2Config tool:

- Communication Channel

    Choose one of the three communication channels. If Serial is selected, the COM port number and baud rate parameters must be configured:

    If LAN is selected, the TCP/IP address and port number (default 9100) must be configured. EpsonFpMate can also be used over wireless connections or RNDIS (TCP/IP over USB) still selecting the LAN radio button.

    In USB mode, the fiscal serial number must be configured as it is the only way to distinguish two or more USB connected fiscal printers:

    If an MF style serial number is configured and a similar RT serial number is found or vice-versa during the communication test, EpsonFpMateConfig will ask whether to accept it. The EpsonFpMate process handles both serial number styles. Therefore, if an MF printer switches to RT mode, EpsonFpMate will continue to operate. if an RT printer switches to MF mode, EpsonFpMate will continue to operate and will ignore the RT device type identifier.

- Cash Register Number

  The cash register number corresponds with the NUMERO CASSA line programmed with SET 15/11. A zero value deactivates printing of the NUMERO CASSA line. Accepted values range from 1 to 999.

- JavaPOS Mode

  If selected, EpsonFpMate activates JavaPOS mode on the printer (SET 14/29 flag). In this condition, the printer requires the separate "CloseFiscalReceipt" closure command after payment has been completed according to the UPOS international standard. Furthermore, EpsonFpMate will automatically send the closure command when necessary such as when voiding or closing receipts from the control panel. If the D Type input file format is activated, this setting has no effect as the JavaPOS mode is forced off.

- Communication Test Button

  EpsonFpMate will open a connection to the printer and first check if it is an RT printer or not. Then it reads the fiscal serial number which is then displayed to allow for confirmation. If the connection is via the USB port and EpsonFp2Config was not used to configure the communication parameters, the retrieved serial number will be compared with the one programmed in this tool. If an MF style serial number is configured and a similar RT serial number is found or vice-versa, EpsonFpMateConfig requests a confirmation from the operator. If none match, a list of retrieved fiscal serial numbers is displayed.

6   EpsonFpMate in Free Running Mode

6.1   System Tray Icon and Application Window

Once the system has been configured (using EpsonFpMateConfig) the EpsonFpMate application can be started. Its icon is a bridge and it remains visible in the system tray.

It then scans for input files. If any are found, it processes them sequentially (in no specific order). Depending on the settings, it can generate output and/or log files. By left or right clicking on the system tray icon, the application window becomes visible:

If the Exit button has been disabled, the bottom part of the form will change accordingly:

In this case Alt-F4 can be used to close the program.

The application window can be hidden (minimised) or the program itself can be closed using the appropriate buttons (Hide and Program Exit respectively).

6.2   EpsonFpMate Control Panel

The control panel can be used to perform local management functions such as a fiscal daily closure. It appears after clicking the "Enable Commands" button. If one or more commands have been disabled in EpsonFpMateConfig, the related buttons will not be visible.

If all buttons are enabled, the form will appear as follows:
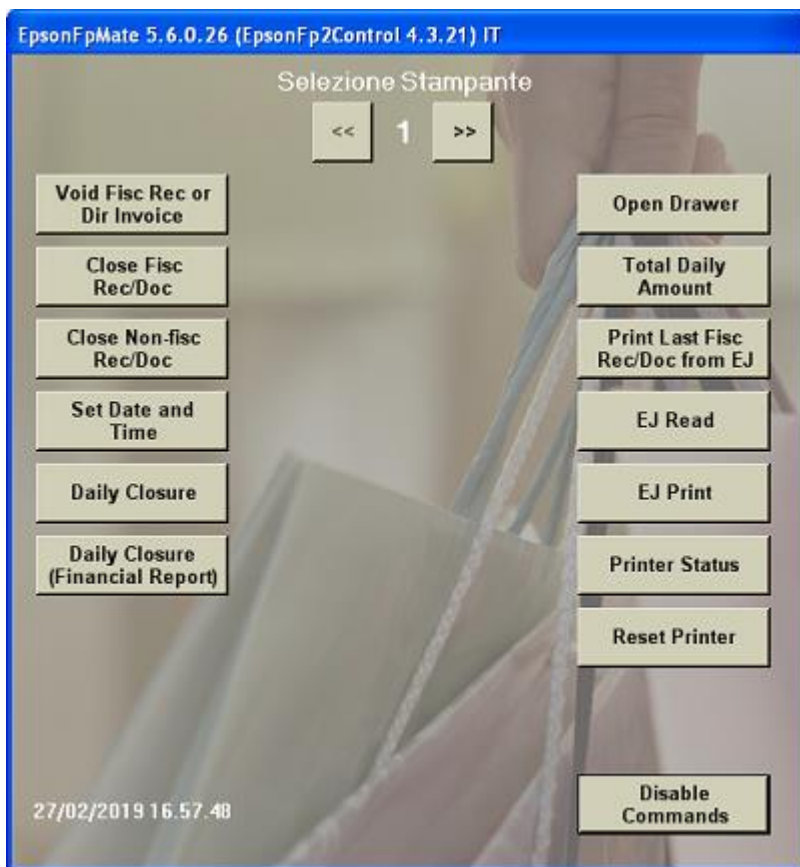
Selezione Stampante (select printer) is only displayed if more than one printer is configured in EpsonFpMateConfig. Commands can be sent to the printer in order to:

- Read the following:
    - Printer status (including RT status)
    - Electronic Journal content (not invoices)
    - Current fiscal daily total amount
- Print the following:
    - Z reports (daily fiscal closures)
    - A financial X-01 report followed by a Z report
    - Electronic Journal content (not invoices)
    - Reprint the last emitted receipt by using content from the Electronic Journal (since duplicates are prohibited by law, a non-fiscal receipt is issued that incorporates all the details of the last fiscal receipt).

- Program the printer date and time (using the computer's system time)
- Close an open fiscal receipt / commercial document
- Close an open non-fiscal receipt / management document
- Void an open fiscal receipt / commercial document
- Open the cash drawer
- Reset the printer

The Electronic Journal content can be read or printed either by date or by number. Invoices are not included.

Read content can be saved as a text file:

6.3   Windows Scheduled Start on Reboot

EpsonFpMate can be automatically launched whenever the PC is rebooted. EpsonFpMate does not need to be configured any differently. When setting this up in your operating system, please note the following considerations:

- The working directory (path) must be set in the windows parameters otherwise EpsonFpMate will not find the watchdog executable file (EpsonFpMateSpv.exe) and will consequently abort. A typical path is C:\Programmi (x86)\EpsonFpMate.

- Whenever the scheduler launches multiple processes, EpsonFpMate should be the last one in the sequence.

7   EpsonFpMate in Single Shot Mode

In "Single Shot" mode, EpsonFpMate is executed by a command line string that has the file to be processed as an argument plus an optional check file argument separated by pipe ("|"). In addition, an alternative Settings.xml file can be indicated. After having processed the line, EpsonFpMate closes. If file arguments are omitted, EpsonFpMate asks the operator if to open the control panel and switch to Free Running mode.

Command line examples:

C:\Programmi (x86)\EpsonFpMate\EpsonFpMate.exe C:\EpsonFpMate\Input\ScoDocComm.txt

C:\Programmi (x86)\EpsonFpMate\EpsonFpMate.exe C:\EpsonFpMate\Input\ScoDocComm.txt|
C:\EpsonFpMate\Input\ScoDocComm.on

C:\Programmi (x86)\EpsonFpMate\EpsonFpMate.exe
C:\EpsonFpMate\Settings\Settings.xml|C:\EpsonFpMate\Input\ScoDocComm.txt

C:\Programmi (x86)\EpsonFpMate\EpsonFpMate.exe
C:\EpsonFpMate\Settings\Settings.xml|C:\EpsonFpMate\Input\ScoDocComm.txt|
C:\EpsonFpMate\Input\ScoDocComm.on

8   Connectivity

Whenever EpsonFpMate is launched in free running mode it automatically configures the following parameters for each printer:

- Fixed:
  - ○ ACK SET 14/27 – Set to 0 (no)
  - ○ SET 14/11 – Set to 1 (paper low on request)

- Per printer according to EpsonFpMateConfig:
  - ○ Cash Register Number – SET 15/11
  - ○ JavaPOS Mode – SET 14/29

It also checks the following:

- Printer status
- Fiscal printer serial number
- DGFE / MPD status
- Printer clock
- Printer model

When the program closes it simply closes the communication port without sending any commands.

9   Input File Setup (EpsonFpMate Format)

9.1   Input File Types

EpsonFpMate can handle two different file formats:

- XML
- TXT

When TXT EpsonFpMate format files are used, a user defined separator (delimiter) character is used to differentiate the fields inside each record.

The input file structure must contain three sections:

- Input file type identifier
- Printer number (only necessary if more than one printer is being managed by EpsonFpMate)
- File body with commands and fields

The # and _ prefixes can be used to comment out a TXT line. The _ prefix can be used to comment out an XML tag.

9.2   Input File Type Identifiers

The first part of the file contains the header and it identifies the file type.

Type Identifier Keyword	Input File Type
printerFiscalReceipt	Fiscal receipt / commercial document (and direct invoice)
printerFiscalReport	Financial report and daily closure (X and Z report)
printerNonFiscal	Non-fiscal receipt / management document
printerCommand	Command (for example a printer reset)
printerFiscalDocument	Fiscal document (not commercial document) (free invoice rather than direct invoice)
printerVoidReceipt	Void receipt / document command *
applicationExit	EpsonFpMate program close command *

* These types of file have no body.

In XML mode the root node is set to the desired keyword.

Input File Specification				
Fiscal Receipt	**Non-fiscal Receipt**	**Daily Closure**	**Command**	**Fiscal Document**
printerFiscalReceipt	printerNonFiscal	printerFiscalReport	printerCommand	printerFiscalDocument

Message to customer display (displayText)
Direct activation of EpsonFp2.ocx method (directIO)
Printer beep request (soundBuzzer)
Setup of EpsonFp2.ocx properties (printSetProperty)
Reading of EpsonFp2.ocx properties (printReadProperty)

Fiscal Receipt	Non-fiscal Receipt	Daily Closure	Command	Fiscal Document
beginFiscalReceipt	beginNonFiscal	printXReport	queryEjContent	beginFiscalDocument
printRecItem	printNormal	printZTotReport	queryPrinterStatus	printFiscalDocumentLine
printRecRefund	printRecMessage	printZReport	openDrawer	printFiscalDocAmount
printRecVoidItem / printRecItemVoid	printBarCode		getDate	endFiscalDocument
printRecItemAdjustment	printGraphicCoupon		queryEjContent	printFiscalDocument
printRecSubtotalAdjustment	endNonFiscal		queryEjContentByNumbers	openDrawer
printRecSubtotal	printDuplicateReceipt		pluTableDownload *	getDate
printRecMessage	beginTraining		pluTableUpload *	printNormal
printBarCode	endTraining		pluPriceUpdate *	printRecMessage
printQRCode	openDrawer		pluDelete *	
printGraphicCoupon	printQRCode		pluRead *	
printRecTotal	printGraphicCoupon		pluReadReset *	
printRecVoid			sendReceiptImage	
openDrawer			resetPrinter	
closeFiscalReceipt			* Not RT models	

Input File Specification (files without body)

Void Receipt Command	**EpsonFpMate close command**
printerVoidReceipt	applicationExit

9.3   Destination Printer Selection

The second record in the input file indicates the specific printer to be addressed. It is optional but must be used if more than one printer is configured.

- Txt Mode:

  Printer[Field Separator][Number]   e.g.: Printer|3

  This example denotes printer number 3 and assumes the field separator character (delimiter) is pipe.

- Xml Mode
  <Printer Num="3" />

9.4   File Bodies

The content of the body depends on the file type to be processed – fiscal receipt / commercial document, non-fiscal receipt etc. The following text examples all assume that the separator used is pipe.

9.4.1   Records Managed Regardless of the File Type

Command	Description
displayText	Customer display messages
directIO	Direct Activation of EpsonFp2 OCX method
printSetProperty	Setup of EpsonFp2 OCX properties
printReadProperty	Reading of EpsonFp2 OCX properties
soundBuzzer	Sound internal beeper

9.4.1.1   Customer Display Messages

The displayText command sends text messages to the customer display. You cannot insert carriage returns or line feeds so use spaces to pad out line 1 and begin line 2. Both lines are 20 characters long.

- Txt Format
  displayText|nOpe|Display Text Message

- Xml Format
  <displayText Ope="nOpe" Text="Display Text Message" />

Text has alias text.

Where:

- **displayText** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **DisplayText Message** indicates the text to display (maximum 40 characters).

9.4.1.2   Activation of EpsonFp2 OCX Method

OCX method calls enrich the functionality of EpsonFpMate. The first entry after directIO is the OCX method name. Subsequent method arguments are then included in the command line.

- Txt Format
  directIO|Method|Data0|Data1|Data2| . . . |DataN

- Xml Format
  <directIO   OcxMethod="Method"   Arg0="Data0"   Arg1="Data1"   Arg2="Data2"   ...
  ArgN="DataN" />

OcxMethod has alias ocxMethod.

Where:

- **directIO** is the command.
- **Method** is name of the EpsonFp2 OCX method to be used (please refer to the list of methods in the EpsonFp2 development guide).
- **DataX** represents the Xth parameter / argument to be passed to the method. The parameter sequence order must be the same as the OCX method's argument order as indicated in the related EpsonFp2 development guide. The number, the type, the format and the meaning of the arguments depend upon the method called.

Note: Square brackets [ ] in methods in the EpsonFp2.ocx development guide signifies optional parameters. They can be all present, partially present or missing in the list of parameters.

9.4.1.3   Setup of EpsonFp2 OCX Properties

printSetProperty sets one EpsonFp2.ocx property. A common usage is to set timeouts just before a command.

- Txt Format
  printSetProperty|Property|Value

- Xml Format
  <printSetProperty OcxProperty="Property" SetUpValue="Value" />

OcxProperty has alias ocxProperty.
SetUpValue has aliases setupValue and setUpValue.

Where:

- **printSetProperty** is the command.
- **Property** is the name of the desired EpsonFp2 OCX property to be set.
- **Value** is the value to be assigned to the selected property.

9.4.1.4   Reading of EpsonFp2 OCX Properties

printReadProperty reads the current value of the selected EpsonFp2 OCX property.

- Txt Format
  printReadProperty|Property

- Xml Format
  <printReadProperty OcxProperty="Property" />

OcxProperty has alias ocxProperty.

Where:

- **printReadProperty** is the command.
- **Property** is the name of the EpsonFp2 OCX property to be read.

9.4.1.5   Sound Buzzer (FP-81 II and FP-90 III models only)

Sends the beep command to sound the internal printer buzzer.

- Txt Format
  SoundBuzzer|nOpe|nCyles|bLength|bFrequency

- Xml Format
  <SoundBuzzer Ope="nOpe" BeepCycles="nCycles" BeepLength="bLength"
  BeepFrequency="bFrequency" />

Where:

- **SoundBuzzer** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **nCycles** is the number of desired "beeps" (range 0 - 99. 0=continuous).
- **bLength** is the pulse length in tens of milliseconds (range 1 - 999).
- **bFrequency** is the beep frequency in Hz (range 0 - 9999).

9.4.2   Fiscal Receipt / Commercial Document File Body

In this case, the body consists of one or more records of different types. The structure is also relevant for direct type invoices:

Command	Description
printRecItem	Sale
printRecRefund	Goods return
printRecVoidItem printRecItemVoid	Cancel a previous sale (storno)
printRecItemAdjustment	Department discount or surcharge
printRecSubtotalAdjustment	Subtotal discount or surcharge
printRecSubtotal	Prints subtotal
printRecMessage	Additional free text row or description
printBarCode	Print barcode
printQRCode	Print QR code
printGraphicCoupon	Print graphic coupon
printRecTotal	Payment
openDrawer	Open the cash drawer
printRecVoid	Void the open fiscal receipt / commercial document
beginFiscalReceipt	Open a fiscal receipt / commercial document
closeFiscalReceipt	Close a fiscal receipt / commercial document (JavaPOS mode only)

Note: Apart from when issuing a credit note receipt, each fiscal receipt file must comply with the following conditions:

- It must include at least one sale row
- The total amount of the receipt cannot be negative
- The payment must be completed

A fiscal receipt / commercial document is automatically opened if the printer receives a sale command whilst in the STATO REGISTRATION condition meaning that beginFiscalReceipt can be omitted. If JavaPOS mode is deactivated, receiving enough payment(s) to cover the total will close it. In this case, printRecTotal must be the last line. Multiple PrintRecTotal lines can be used if for example a customer pays partly by card and partly by cash (assuming the total due hasn't already been covered). In JavaPOS mode closeFiscalReceipt must be used after payment. Even though footer texts and barcodes appear below the total line for example, the footer and barcode command lines must be placed before "printRecTotal". The only exception is the displayText command. This can be placed after printRecTotal or closeFiscalReceipt when a message is to be sent to the customer display.

9.4.2.1   Sale Items

printRecItem prints a sale item. If it is the first item and beginFiscalReceipt is not used, a new fiscal receipt / commercial document is automatically opened.

- Txt Format
  printRecItem|nOpe|SaleDescription|xx,xxx|yy,yy|nDep|Just

- Xml Format
  <printRecItem   Ope="nOpe"   Text="SaleDescription"   Qty="xx,xxx"   UnitCost="yyy,yy" Dep="nDep" Just="Just" />

Text has alias text.

Where:

- **printRecItem** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **SaleDescription** is the description composed of a maximum of 32 or 38 characters depending on the printer model.
- **xx,xxx** represents the quantity (range 0,001 - 9999,999). This cannot be zero.
- **yy,yy** represents the unit price (range 0,00 - 9999999,99).
- **nDep** represents the department number (range 1 - 99).
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

9.4.2.2 Customer Returns (Resi)

Used when goods are returned by a customer and a refund (reso) transaction must be printed. In the case of an RT printer in service, if a refund is received in an open commercial document, it is automatically converted to a cancel sale (storno) transaction. This is because sales and refunds can no longer exist together in the same commercial document.

- Txt Format
  printRecRefund|nOpe|RefundDescription|xx,xxx|yy,yy|nDep|Just

- Xml Format
  <printRecRefund Ope="nOpe" Text="RefundDescription" Qty="xx,xxx" UnitCost="yyy,yy" Dep="nDep" Just="Just" />

Text has alias text.

Where:

- **printRecRefund** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **RefundDescription** is a description composed of a maximum of 32 or 38 characters depending on the printer model.
- **xx,xxx** represents the quantity (range 0,001 - 9999,999). This cannot be zero.
- **yy,yy** represents the unit price (range 0,00 - 9999999,99).
- **nDep** represents the department number (range 1 - 99).
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

9.4.2.3   Cancel Sale (Storno)

printRecVoidItem or the alias printRecItemVoid is used to void one of the previous sale operations.

- Txt Format
printRecVoidItem|nOpe|VoidItemDescription|xx,xxx|yy,yy|nDep|Just

- Xml Format
<printRecVoidItem   Ope="nOpe"   Text="VoidItemDescription"   Qty="xx,xxx"
UnitCost="yyy,yy" Dep="nDep" Just="Just" />

Text has alias text.

Where:

- **printRecVoidItem or printRecItemVoid** is the command.
- **nOpe** is the operator ID (range 1 - 12)
- **VoidItemDescription** is a description composed of a maximum of 32 or 38 characters depending on the printer model.
- **xx,xxx** represents the quantity (range 0,001 - 9999,999). This cannot be zero.
- **yy,yy** represents the unit price (range 0,00 - 9999999,99).
- **nDep** represents the department number (range 1 - 99).
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

9.4.2.4   Item or Department Specific Discount/Surcharge

A discount or surcharge is applied to either the previous sale or refund operation or to a specific department. If based on the last transaction and the latter was a correction (storno), the printer returns Error 11.

- Txt Format
printRecItemAdjustment|nOpe|DiscountDescription|DiscUp|yyy,yy|nDep|Just

- Xml Format
<printRecItemAdjustment   Ope="nOpe"   Text="DiscountDescription"   Type="DiscUp" Amount="yyy,yy" Dep="nDep" Just="Just" />

Type has alias type.
Text has alias text.

Where:

- **printRecItemAdjustment** is the command.
- **nOpe** is the operator ID (range 1 - 12)
- **DiscountDescription** is a description of up to 32 or 38 characters depending on the printer model.
- **DiscUp** selects one of four different operation types:
    - 0 = Discount based on the last transaction.
    - 3 = Department specific discount.
    - 5 = Surcharge based on the last transaction.
    - 8 = Department specific surcharge.
- **yyy,yy** represents the discount amount (range 0,01 - 9999999,99). This cannot be zero.
- **nDep** represents the department number (range 1 - 99).
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

Notes:

1. A zero discount will throw Error 16 from the printer and the print-out will not complete.
2. In the case of surcharges on refunds or discounts on sales, the minus sign must never be used – the printer automatically prints it.
3. Whenever DiscUp = 0 – If the last transaction was a sale, the discount is negative. If the last transaction was a refund, the discount is positive.
4. Whenever DiscUp = 5 – If the last transaction was a sale, the surcharge is positive. If the last transaction was a refund, the surcharge is negative.
5. If the last transaction was a correction (storno), the printer throws Error 11 and the print-out will not complete.
6. nDep only has significance when DiscUp equals 3 or 8. Furthermore, these two types can be used as the first operation to automatically open a fiscal receipt / commercial document.

9.4.2.5   Discount/Surcharge on Subtotal

A discount or surcharge is applied to the subtotal.

- Txt Format
  printRecSubtotalAdjustment|nOpe|DiscountDescription|DiscUp|yyy,yy|NU|Just

- Xml Format
  <printRecSubtotalAdjustment   Ope="nOpe"   Text="DiscountDescription"   Type="DiscUp" Amount="yyy,yy" Dep="NU" Just="Just" />

Type has alias type.
Text has alias text.

Where:

- **printRecSubtotalAdjustment** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **DiscountDescription** is a description composed of a maximum of 32 or 38 characters depending on the printer model.
- **DiscUp** selects between four different types of operation:
  - o 1 = Discount on subtotal with subtotal printed out.
  - o 2 = Discount on subtotal without subtotal printed out.
  - o 6 = Surcharge on subtotal with subtotal printed out.
  - o 7 = Surcharge on subtotal without subtotal printed out.
- **yyy,yy** represents the discount amount (range 0,01 - 9999999,99). This cannot be zero.
- **NU** represents an unused numeric field not handled by the printer.
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

Notes:

1. A zero discount will throw Error 16 from the printer and the print-out will not complete.
2. In the case of discounts, the minus sign must never be used – the printer automatically prints it.
3. The eventual subtotal line is printed before the discount or surcharge.
4. If DiscUp is either 1 or 2 and the subtotal is zero or negative, the printer throws Error 17 and the print-out will not complete.

9.4.2.6   Print Subtotal

Prints or displays the subtotal or both.

- Txt Format
  printRecSubtotal|nOpe|prnOption

- Xml Format
  <printRecSubtotal Ope="**nOpe**" Type="**prnOption**" />

Type has alias type.

Where:

- **printRecSubtotal** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **prnOption** selects between three different types of operation:
  - 0 = Print and display
  - 1 = Print only
  - 2 = Display only

9.4.2.7   Additional Line / Customer Loyalty Programme Number / EFT-POS

Additional lines can be printed in four positions. Additional header commands must be placed before the open command or before the first transaction as they must be preloaded in the printer memory. Trailer and additional trailer commands must be placed before the payment command.

The Customer Loyalty Programme Number appears in the rec.json file.

- Txt Format
printRecMessage|nOpe|Type|Num|Font|RowText

- Xml Format
<printRecMessage Ope="Nope" Type="Type" Index="Num" Font="Font" Text="RowText" />

Type has alias type.
Text has alias text.

Where:

- **printRecMessage** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **Type** defines the row type:
    o 1 = Additional fiscal receipt / commercial document or credit note header.
    o 2 = Trailer line (before MF / RT logotype) in fiscal receipt / commercial document or credit note.
    o 3 = Additional trailer / promo line (after MF / RT logotype) in fiscal receipt / commercial document or credit note.
    o 4 = Additional description (in the body of a fiscal receipt / commercial document).
    o 7 = Customer loyalty programme number (Only FP-81 II and FP-90 III models).
    o 8 = Instructs the printer to print out EFT-POS lines that would have been buffered after an electronic payment. They are printed after trailer type 2 lines.
- **Num** indicates the row number index:
    o Range 1 - 9 for additional header (type 1).
    o Range 1 - 99 for trailer and additional trailer (types 2 and 3).
    o Not applicable with type 4.
    o Not applicable with type 7.
- **Font** indicates the font to be used for the entire row:
    o 1 = Normal.
    o 2 = Bold.
    o 3 = Double height.
    o 4 = Bold double height.
    o Not applicable with type 4 (font is always 1 – normal due to fiscal law layout requirements).
    o Not applicable with type 7.
- **RowText** represents the text to be printed or loaded. The maximum length is 40 or 46 characters depending on the printer model. For type 4, up to 32 or 38 characters are accepted depending on the printer model (fiscal law layout requirements). Further characters are ignored.

N.B: Additional header command lines (type 1) must be placed before opening the receipt (opened automatically by the first fiscal operation or by beginFiscalReceipt).

9.4.2.8   Bar Code Printing

Bar codes are printed at the end of the fiscal receipt or commercial document. The command must be pre-loaded being positioned before printRecTotal. Only one barcode can be printed. Either a barcode or a QR code can be printed but not both.

- Txt Format
  printBarCode|nOpe|Position|BcWidth|BcHeight|HriEnabling|HriFont|CodeType|BCText

- Xml Format
  <printBarCode Ope="nOpe" Pos="Position" Width="BcWidth" Height="BcHeight" Hri="HriEnabling" Font="HriFont" Tipo="CodeType" Text="BcText" />

Tipo has aliases codeType and tipo.
Text has aliases text, code and Code.

Where:

- **printBarCode** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **Position** defines the starting position from the left margin (range 0 - 511). The special 900, 901 and 902 values can be used to indicate left, centred and right alignment respectively.
- **BcWidth** indicates the dot width of the individual bars (range 1 - 8)
- **BcHeight** indicates the height of the bar code measured in print dots (range 1 - 255)
- **HriEnabling** selects one of three ways to print the alphanumeric representation of the barcode or to disable it altogether. You can choose between the number or the name:
    - o 0 / DISABLED
    - o 1 / BELOW
    - o 2 / ABOVE
    - o 3 / TWICE (prints it above and below)
- **HriFont** indicates the font to be used for the HRI string. You can choose between the number or the name:
    - o 0 / FontA
    - o 1 / FontB
- **CodeType** indicates the bar code standard. You can choose between the code number or the name:
    - o 65 / UPC-A or UPCA
    - o 66 / UPC-E or UPCE
    - o 67 / EAN-13 or EAN13
    - o 68 / EAN-8 or EAN8
    - o 69 / CODE-39 or CODE39
    - o 70 / ITF
    - o 71 / CODABAR
    - o 72 / CODE-93 or CODE93
    - o 73 / CODE-128 or CODE128

- The following can be used with FP-81 II and FP-90 III models. You can choose between the code number or the name:

    - o 74 / GS1-128 or GS1128
    - o 75 / GS1-DATABAR-OMNIDIRECTIONAL or GS1DATABAROMNIDIRECTIONAL
    - o 76 / GS1-DATABAR-TRUNCATED or GS1DATABARTRUNCATED
    - o 77 / GS1-DATABAR-LIMITED or GS1DATABARLIMITED
    - o 78 / GS1-DATABAR-EXPANDED or GS1DATABAREXPANDED

- **BcText** represents the bar code itself.

Notes:

- The special 900, 901 and 902 values indicate left, centred or right horizontal justification.

- The character set and the string length are defined by the standards related to each bar code system.

- The starting position, the string length and the bar code width must be defined according to the paper roll size. If there is not enough room for the row, the barcode will not be printed (whitespace will be visible instead).

- According to the CODE128 standard, one of the following character sets must be specified:

    o To select CODE A, the first two characters of the RowText must be "{A"
    o To select CODE B, the first two characters of the RowText must be "{B"
    o To select CODE C, the first two characters of the RowText must be "{C"

    When employing CODE128, note that these first two characters are not part of the bar code itself and are therefore not printed. They are present only for selecting the character set.

9.4.2.9   QR Code Printing

QR codes are only supported with FP-81 II and FP-90 III models. They are printed at the end of the fiscal receipt or commercial document in the same manner as for barcodes (the command must be pre-loaded being positioned before printRecTotal). Only one QR code can be printed. Either a QR code or barcode can be printed but not both.

- Txt Format
printBarCode|nOpe|Position|0|QRCodeSize|QRCodeDataType|ErrorCorrectionLevel|CodeType|QRCodeData

- Xml Format
<printQRCode Ope="nOpe" Pos="Position" Size="QRCodeSize" DataType="QRCodeDataType" ErrorCorrection="ErrorCorrectionLevel", Tipo="CodeType" Text="QRCodeData" />

- Alternative Xml Format using printBarCode
<printBarCode Ope="nOpe" Pos="Position" Height ="QRCodeSize" Hri="QRCodeDataType" Font="ErrorCorrectionLevel", Tipo="CodeType" Text="QRCodeData" />

Where:

- **printQRCode** is the command. **printBarCode** is also accepted for backwards compatibility.
- **nOpe** is the operator ID (range 1 - 12).
- **Position** defines the horizontal alignment 0, 1 and 2 values can be used to indicate left, centred and right respectively.
- **QRCodeSize** indicates the QR code size (range 1 - 16)
- **QRCodeDataType** selects one of two representations:
  - BINARY
  - ALPHANUMERIC
- **ErrorCorrectionLevel** selects one of four levels:
  - LOW
  - MEDIUMLOW
  - MEDIUMHIGH
  - HIGH
- **CodeType** indicates the QR code standard. You can choose between the code number or the name:
  - 91 / QRCODE1
  - 92 / QRCODE2 (has better Smartphone compatibility)
- **QRCodeData** represents the QR code itself. If binary is chosen, a hexadecimal sequence must be used. For example, HELLO = 48454C4C4F. The letters A to F can also be lower case.

9.4.2.10   Graphic coupon printing

One graphic coupon can be printed at the end of the receipt or commercial document after the additional row, description or bar code / QR code. JavaPOS Mode must be off. The graphic file must be in the Epson format.

- Txt Format
printGraphicCoupon|nOpe|FileName

- Xml Format
<printGraphicCoupon Ope="nOpe" GraphicFile="FileName" />

Where:

- **printGraphicCoupon** is the command
- **nOpe** is the operator ID (range 1 - 12).
- **FileName** is either:
  - The name of the graphic file to be printed (if the file resides in the pre-configured "input" directory).
  - The complete path (including the disk drive and filename) if the file is located elsewhere.

9.4.2.11   Payment section

One or more records can be defined but usually it is composed of one record; more than one means that the payment is composed of several partial payments. In this case care must be taken to correctly order the payments as the printer will not accept further payment commands if the total due has already been paid.

- Txt Format
  printRecTotal|nOpe|PaymentDescription|yyy,yy|Mode|Index|Just

- Xml Format
  <printRecTotal   Ope="nOpe" Text="PaymentDescription" Amount="yyy,yy" Type="Mode" Index="NCard" Just="Just" />

Amount has alias Payment.
Tipo has aliases codeType and tipo.
Text has aliases text, code and Code.

Where:

- **printRecTotal** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **PaymentDescription** is a description composed of a maximum of 32 or 38 characters depending on the printer model.
- **yyy,yy** represents the amount (range 0,00 - 9999999,999). A value of zero is interpreted as the outstanding amount to pay.
- **Mode** indicates the payment method:
  - o   0 = Cash
  - o   1 = Cheque
  - o   2 = Credit/Credit Card
  - o   3 = Ticket
- **NCard** is used with either cash, credit cards or tickets to select for example the type of credit card or ticket that has been used.
- **Just** is used to set the description justification. The maximum number of characters that can appear on the display is 20. If Just = 1, the first 20 characters appear; whereas if Just = 2, the last 20 characters.

9.4.2.12   Open Cash Drawer

openDrawer opens the cash drawer. There are no parameters.

- Txt Format
  openDrawer|nOpe

- Xml Format
  <openDrawer Ope="nOpe" />

Where:

- **openDrawer** is the command.
- **nOpe** is the operator ID (range 1 - 12).

9.4.2.13 Void Fiscal Receipt

printRecVoid voids the current open fiscal receipt. There are no parameters.

- Txt Format
  printRecVoid|nOpe

- Xml Format
  <printRecVoid Ope="nOpe" />

Where:

- **printRecVoid** is the command.
- **nOpe** is the operator ID (range 1 - 12).

9.4.2.14 Begin Fiscal Receipt

beginFiscalReceipt should be used for correctness but can sometimes be omitted if for example the first transaction is a sale. It opens a new fiscal receipt / commercial document. The retail header lines are automatically printed along with any header logo and/or additional header lines. There are no parameters.

- Txt Format
  beginFiscalReceipt|nOpe

- Xml Format
  <beginFiscalReceipt Ope="nOpe" />

Where:

- **beginFiscalReceipt** is the command.
- **nOpe** is the operator ID (range 1 - 12).

9.4.2.15 Close Fiscal Receipt

This command is only required if the JavaPOS Mode has been activated. This is because the fiscal receipt / commercial document remains open after the payment has been completed (as per the UPOS international standard). This command is therefore required to close it.

- Txt Format
  closeFiscalReceipt|nOpe or endFiscalReceipt|nOpe

- Xml Format
  <closeFiscalReceipt Ope="nOpe" />

Where:

- **closeFiscalReceipt** is the command. The **endFiscalReceipt** alias is also accepted.
- **nOpe** is the operator ID (range 1 - 12).

9.4.3   Prohibited Wordings in Free Text Lines

Certain wordings are prohibited on fiscal receipts and commercial documents as additional text or descriptions (printRecMessage command). They are as follows:

- Totale
- Sconto
- Importo
- Contante

In addition, Totale is not permitted as a payment description (printRecTotal).

These words cannot be printed even if the characters are separated by non-alphabetic characters (e.g.: Cont-ant#e; s?c'o"n%t o,). They should not be present in any description text. EpsonFpMate does not perform any pre-processing operation on prohibited wordings. The fiscal printer will alter any message that attempts the printing of any of the above words on a fiscal receipt or commercial document. Neither the printer nor EpsonFpMate will generate any error. The way in which these words are modified is "hard coded" in the fiscal printer firmware. Examples:

- Totale becomes Total*
- Importo becomes Import*
- Sconto becomes Scont*
- Contante becomes Contant*

9.4.4   Credit Note Fiscal Receipt File Body (MF models only)

Since by law fiscal receipts cannot be negative, credit notes are issued whenever a client returns goods and does not purchase replacement items that would otherwise bring the subtotal to a zero or positive amount. The operations allowed in a Credit Note (Nota di Credito) are almost the same as those in a normal fiscal receipt and are listed in the following table:

Command	Description
printRecRefund	Goods return
printRecItemAdjustment	Sale discount / surcharge
printRecSubtotalAdjustment	Subtotal discount / surcharge
printRecSubtotal	Print subtotal
printRecMessage	Additional row or description
printBarCode	Print barcode
printQRCode	Print QR code
printGraphicCoupon	Print graphic coupon
printRecTotal	Payment

Note: Each credit note fiscal receipt file requires:

- The open command
- At least one Goods Return row
- The payment row (closure). The operator field is the only one that has relevance.

9.4.4.1   Opening a Credit Note

Credit notes are opened in a special way. The first command must be "printRecMessage" with type=4 or type=1 and the description must begin with the special 16-character phrase "PRATICA DI RESO ". It is important to use capitals and separate the words with blanks plus add a trailing blank. There are then a further 16 characters that must be used to uniquely identify the credit note. The retail outlet software application must take care of any sequencing or referencing.

9.4.4.2   Closing a Credit Note

A Credit Note is closed by a single payment command and if JavaPOS mode is active a subsequent closure command. Partial payments are not permitted. The amount indicated in the command line is ignored as are all the other parameters apart from the operator. The payment description is therefore not printed. The amount reported on the credit note is the one computed by the fiscal printer.

9.4.5   Commercial Refund Document (Documento di Reso) (RT models only)

Epson RT fiscal printers can print out commercial refund documents related to goods refund operations. It means for example that customers can be refunded the same amount that was paid when the item was originally bought without having to purchase new items to make the commercial document positive. The commands allowed depend on the SET 14/58 flag. In Ndc mode (1 / SI), the same commands as for credit notes are allowed and the same rules are followed. In non-Ndc mode (0 / NO), the same commands as for commercial documents are allowed and the same rules are followed. The file type remains printerFiscalReceipt. If the original document exists on the EJ / MPD, individual totals for each of the 20 tax groups present in the refund document cannot exceed the same totals in the original document.

9.4.5.1   Opening a Commercial Refund Document

To open a commercial refund document, a special type 4 "printRecMessage" command must be used that has to conform to the following format:

- REFUND zzzz nnnn ddmmyyyy sssssssssss

Where:

- zzzz equals the four-digit Z report number of the original document.
- nnnn equals the four-digit document number of the original document.
- ddmmyyyy equals the date of the original document.
- sssssssssss equals the eleven-character fiscal serial number of the printer that emitted the original document. If an MF model was used or the printer was in MF mode, use 99MEY999999.

It is important to use capitals and separate the fields with blanks. beginFiscalReceipt is not required with commercial refund documents but can be still used for conformity if desired (placing it after printRecMessage). If the fiscal serial number is not the same as the printer that receives the request, an EJ search is not performed. Currently there is no fiscal printer intercommunication function. The special printRecMessage command must be the first in the commercial refund document sequence (but after any type 1 "printRecMessage" optional additional header lines).

9.4.5.2   Closing a Commercial Refund Document

In Ndc mode (flag SET 14/58 = 1 / SI), the same rules as for credit notes are followed except that partial payments are accepted. In non-Ndc mode (0 / NO), the same rules as for commercial documents are followed.

9.4.6   Commercial Void Document (Documento di Annullo) (RT models only)

Epson RT fiscal printers can print out commercial void documents. It means for example that an erroneously printed commercial document can be completely voided. With automatic printing, only printRecMessage is used otherwise the commands allowed depend on the SET 14/59 flag. In Ndc mode (1 / SI), the same commands as for credit notes are allowed and the same rules are followed. In non-Ndc mode (0 / NO), the same commands as for commercial documents are allowed and the same rules are followed. The file type is printerFiscalReceipt with printRecMessage followed by the original transactions in case the document is not found. If the original document exists on the EJ / MPD, the opening command will automatically print and close the new void document. In this case, the subsequent lines in the file are ignored.

9.4.6.1   Opening a Commercial Void Document

To open a commercial void document, a special type 4 "printRecMessage" sub-element must be used that has to conform to the following format:

- VOID zzzz nnnn ddmmyyyy sssssssssss

Where:

- zzzz equals the four-digit Z report number of the original document.
- nnnn equals the four-digit document number of the original document.
- ddmmyyyy equals the date of the original document.
- sssssssssss equals the eleven-character fiscal serial number of the printer that emitted the original document. If an MF model was used or the printer was in MF mode, use 99MEY999999.

It is important to use capitals and separate the fields with blanks. beginFiscalReceipt is not required with commercial void documents but can be still used for conformity if desired (placing it after printRecMessage). If the fiscal serial number is not the same as the printer that receives the request, an EJ/MPD search is not performed. Currently there is no fiscal printer intercommunication function.

The special printRecMessage command must be the first in the commercial void document sequence (but after any type 1 "printRecMessage" optional additional header lines).

9.4.6.2   Closing a Commercial Void Document

Only valid if automatic printing does not take place. In Ndc mode (flag SET 14/59 = 1 / SI), the same rules as for credit notes are followed except that partial payments are accepted. In non-Ndc mode (0 / NO), the same rules as for commercial documents are followed.

9.4.7   Non-fiscal Receipt / Management Document File Body

EpsonFpMate can print non-fiscal receipts or management documents (documenti gestionali). The keyword "printerNonFiscal". The following commands are used in a non-fiscal file:

Command	Description
beginNonFiscal	Open a non-fiscal receipt or management document
printNormal	Print text line
printRecMessage	Print additional header or EFT-POS lines
printBarCode	Print barcode
printQRCode	Print QR code
printGraphicCoupon	Print graphic coupon
endNonFiscal	Close non-fiscal receipt or management document
openDrawer	Open the cash drawer
printDuplicateReceipt	Print duplicate of last Fiscal Receipt or Commercial Document from EJ / MPD
beginTraining	Enter Demo RT or training mode
endTraining	Exit Demo RT or training mode

9.4.7.1   Begin a Non-fiscal Receipt

beginNonFiscal is used to open a non-fiscal receipt / management document:

- Txt Format
  beginNonFiscal|nOpe

- Xml Format
  <beginNonFiscal Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.7.2   Print Bar Codes and QR Codes

The printing of bar codes and QR codes uses the same commands and rules as described in the Fiscal Receipt / Commercial Document section. Differently from them there is no limit to the number of barcodes and QR codes that can be printed.

9.4.7.3   Print Graphic Coupon

The printing of graphic coupons uses the same commands and rules as described in the Fiscal Receipt / Commercial Document section. The only difference is that instead of only being possible to be printed once at the end, in this case it can be printed as many times as you like making the printing of several different graphic images possible.

9.4.7.4   Print Text Line

printNormal is used. Up to four fonts are supported. The format is as follows:

- Txt Format
  printNormal|nOpe|Font|RowText

- Xml Format
  <printNormal Ope="nOpe" Font="Font" Text="RowText" />

Text has alias text.

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "Font" indicates one of four possible fonts:
    - 1 = Normal
    - 2 = Bold
    - 3 = Double
    - 4 = Double and bold
- "RowText" contains the text.

9.4.7.5   Additional Header or EFT-POS lines

Up to nine additional header lines can be printed. Additional header commands must be placed before the open command as they must be preloaded in the printer memory. A request to print EFT-POS lines should be positioned whilst the document is open.

- Txt Format
printRecMessage|nOpe|Type|Num|Font|RowText

- Xml Format
<printRecMessage Ope="Nope" Type="Type" Index="Num" Font="Font" Text="RowText" />

Type has alias type.
Text has alias text.

Where:

- **printRecMessage** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **Type** defines the row type:
    - o 1 = Additional non-fiscal receipt / management document header.
    - o 8 = Instructs the printer to print out EFT-POS lines that would have been buffered after an electronic payment.
- **Num** indicates the row number index:
    - o Range 1 - 9 for additional header (type 1).
    - o Not applicable with type 8 EFT-POS.
- **Font** indicates the font to be used for the entire row:
    - o 1 = Normal.
    - o 2 = Bold.
    - o 3 = Double height.
    - o 4 = Bold double height.
    - o Not applicable with type 8 EFT-POS.

- **RowText** represents the text to be printed or loaded. The maximum length is 40 or 46 characters depending on the printer model.

N.B: Additional header command lines (type 1) must be placed before opening the receipt by beginNonFiscal).

9.4.7.6   Close Non-fiscal Receipt / Management Document

endNonFiscal closes a non-fiscal receipt / management document. The format is as follows:

- Txt Format
endNonFiscal|nOpe

- Xml Format
<endNonFiscal Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.7.7   Open Cash Drawer

The open cash drawer command uses the same commands and rules as described in the Fiscal Receipt section.

9.4.7.8   Reprint Last Fiscal Receipt / Commercial Document (from EJ / MPD)

printDuplicateReceipt is used to print a copy of the last emitted fiscal receipt or commercial document. Since exact duplicates are prohibited by law, the reprinted content is in fact embedded in a non-fiscal receipt / management document (automatically handled by the fiscal printer).

- Txt Format
  printDuplicateReceipt|nOpe

- Xml Format
  <printDuplicateReceipt Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.7.9   Enter Demo RT / Training Mode

To enter demo RT / training mode the fiscal printer must be in the logical day opened = false condition. A fiscal daily closure may therefore be necessary. Once in this mode, non-fiscal print outs are emitted. All spaces are replaced with question marks (?) to render print outs unusable in a retail outlet. Fiscal daily closures are inhibited whilst in training mode but in demo RT mode they are permitted

- Txt Format
  beginTraining|nOpe

- Xml Format
  <beginTraining Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.7.10　Exit Demo RT / Training Mode

endTraining exits from demo RT / training mode.

- Txt Format
  endTraining|nOpe

- Xml Format
  <endTraining Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.8　Report File Body

Three possible report print outs are possible. The printer can perform a Z daily closure report, an X-01 report or both. In the case of an RT printer, tax authority XML transmission is automatically handled by the printer.

Command	Description
printZReport	Daily closure (Z Report)
printZTotReport	Daily closure (Z Report) with additional X-01 daily financial report
printXReport	Daily X-01 financial report only

A Z report also sets the printer to the logical day opened = false condition.

9.4.8.1　Z Daily Closure Report

- Txt Format
  printZReport|nOpe

- Xml Format
  <printZReport Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.8.2   Z Daily Closure with X-01 Financial Report

- Txt Format
  printZTotReport|nOpe

- Xml Format
  <printZTotReport Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.8.3   X-01 Financial Report Only

- Txt Format
  printXReport|nOpe

- Xml Format
  <printXReport Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.9   Command File Body

EpsonFpMate can extract administrative/diagnostic information from the printer. The "printerCommand" identifier is used.

The following list of commands can be used in printerCommand files:

Command	Description
queryEjContent	Collect EJ / MPD content by range of dates and save in a file
queryEjContentByNumbers	Collect EJ / MPD content by single date and range of receipt / document numbers and save in a file
queryPrinterStatus	Collect basic or RT status information from the printer and save in a file
openDrawer	Open cash drawer
getDate	Read date and time from the printer's real time clock
sendReceiptImage	Send BMP receipt image file to "Scontrino Sicuro" service
pluTableDownload *	Download PLU data to printer SD/MMC memory card
pluTableUpload *	Read PLU data from printer SD/MMC memory card
pluPriceUpdate *	Update PLU price list
pluDelete *	Delete SD/MMC memory card PLU entries
pluRead *	Read PLU sales data
pluReadReset *	Read PLU sales data and reset counters
resetPrinter	Soft reset printer

* Not RT models.

9.4.9.1   Collect EJ / MPD Content by Range of Dates and Save in a File

The file will appear in the output directory. If the file already exists it will be overwritten without any warning. Invoices are not included (please see example chapter for DirectIO alternative).

- Txt Format
  queryEjContent|nOpe|dataType|stD|stM|stY|endD|endM|endY|file

- Xml Format
  <queryEjContent Ope="nOpe" Type="dataType" stDay="stD" stMonth="stM" stYear="stY" endDay="endD" endMonth="endM" endYear="endY" fileName="file" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "dataType" indicates the data type filter (not currently supported).
- "stD" indicates the start day (1 - 31).
- "stM" indicates the start month (1 - 12).
- "stY" indicates the start year (YYYY).
- "endD" indicates the end day (1 - 31).
- "endM" indicates the end month (1 - 12).
- "endY" indicates the end year (YYYY).
- "file" indicates the desired filename. Paths are not supported.

9.4.9.2   Collect EJ / MPD Content by Single Date and Range of Receipt / Document Numbers and Save in a File

The file will appear in the output directory. If the file already exists it will be overwritten without any warning. Invoices are not included (please see example chapter for DirectIO alternative). Setting both start and end numbers to 9999 together the current date has a special significance equivalent to printDuplicateReceipt.

- Txt Format
  queryEjContentByNumbers|nOpe|dataType|day|month|year|stN|endN|file

- Xml Format
  <queryEjContentByNumbers Ope="nOpe" Type="dataType" stDay="day" stMonth="month" stYear="year" stNumber="stN" endNumber="endN" fileName="file" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "dataType" indicates the data type filter (not currently supported).
- "day" indicates the day (1 - 31).
- "month" indicates the month (1 - 12).
- "year" indicates the year (YYYY).
- "stN" indicates the start number (1 - 9999).
- "endN" indicates the end number (1 - 9999).
- "file" indicates the desired filename. Paths are not supported.

9.4.9.3   Collect Basic or RT Status Information from the Printer and Save in a File

The file will be written in the output directory. If the file already exists it will be overwritten without any warning. From version 5.6.0.31, the command has been expanded. An optional statusType field can be used to indicate whether to return the basic printer status or the RT status. If the new field is omitted, EpsonFpMate retains the original behaviour returning the basic printer status.

- Txt Format
  queryPrinterStatus|nOpe|file or
  queryPrinterStatus|nOpe|nStatusType|file or

- Xml Format
  <queryPrinterStatus Ope="nOpe" fileName="file" /> or
  <queryPrinterStatus Ope="nOpe" statusType="nStatusType" fileName="file" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "nStatusType" indicates one of two modes:
  0 = Basic printer status
  1 = RT printer status
- "file" indicates the desired filename. Paths are not supported.

Example basic status file

Time=05/09/2020 17.27.28
EnvironmentOpSystem=Microsoft Windows NT 5.1.2600 Service Pack 3
EpsonFpMateVersion=5.6.0.33
EpsonFp2ControlVersion=4.3.29
PrinterModel=EPSON FP90III
PrinterSN=EY99000015
FB_SwVer=10.02
FM_SwVer=4.3
FM_State=0
PrinterDiag=00110
DgfeNum=1
DgfePerc=0
DgfeSize=4GB

Example RT status file

```
Time=11/09/2020 19:37:50
EnvironmentOpSystem=Microsoft Windows NT 6.1.7601 Service Pack 1
EpsonFpMateVersion=5.6.0.33
EpsonFp2ControlVersion=4.3.29
PrinterModel=EPSON FP90III RT MODIFICATO
PrinterSN=99MEY000203
RTDeviceType=M
RTMainState=2
RTSubState=8
RTDayOpened=False
RTNonWorkingPeriod=True
RTNumberOfCurrentFiles=0
RTNumberOfOldFiles=0
RTNumberOfRejectedFiles=27
RTDateDeviceCertificateExpiration=09/12/2022
RTDateCAAECertificateExpiration=11/01/2038
RTFirmwareBuildNumber=213
RTSDFileSystemVersionNumber=2
RTTrainingDemoSimulationMode=0
RTFirmwareUpdateResult=0
RTNumberOfArchivedRejectedFiles=0
Spare=000
```

9.4.9.4   Open Cash Drawer

The cash drawer open command used in printerCommand files uses the same commands and rules as described in the Fiscal Receipt section.

9.4.9.5   Read Date and Time from the Printer's Real Time Clock

The file will be written in the output directory.

- Txt Format
  getDate

- Xml Format

The date and time read from the printer is written in the EpsonFpMate output file. Only hours and minutes can be read – seconds are always zero.

9.4.9.6   Send BMP Receipt Image File to "Scontrino Sicuro" Service (FP-81 II and FP-90 III Models only)

The last emitted receipt / document is forwarded to the "Scontrino Sicuro" service as a BMP image file. The image file is generated in fiscal firmware. EpsonFpMate first sends a telephone number (customer code) verification request to the printer and if successful sends a transfer request.

- Txt Format
  sendReceiptImage|nOpe|customerCode

- Xml Format
  <sendReceiptImage Ope="nOpe" UserId="customerCode" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "customerCode" indicates the unique telephone number ("Scontrino Sicuro" customer registration ID code).

9.4.9.7   Download PLU Data to Printer SD/MMC Memory Card

MF models only. pluTableDownload is used to load entries to the external PLU database that resides on the printer's SD/MMC card. The PLU table source data is an XML file with a specific structure. The root element must be named "PLUTable". Each PLU entry is defined as a "PLUItem" element with the following attributes:

- Code: PLU entry number from 1 to 20000. Zero can be used but is not recommended.
- Barcode: Unique EAN13 barcode linked to the PLU entry.
- Description: Personalised product description text that will be printed on the receipt.
- Price: Unit price.
- Category: Not used.
- Department: Printer department from 0 to 99. The value 0 deactivates the PLU entry.

- Txt Format
  pluTableDownload|nOpe|xmlPluFile.xml

- Xml Format
  <pluTableDownload Ope="nOpe" pluDataBaseXmlFile="pluFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "pluFile.xml" indicates the PLU data source XML filename. The file must reside in the input directory. Paths are not supported.

9.4.9.8   Read PLU Data from Printer SD/MMC Memory Card

MF models only. pluTableUpload is used to read all or partial entries from the external PLU database that resides on the printer's SD/MMC card. An optional filter file can be specified which contains a list of the desired PLU entries. The PLU list file is an XML file with a specific structure. The root element must be named "PLUList". Each PLU entry is defined as a "PLUItem" element with the following attribute:

- Code: PLU entry number from 1 to 20000.

Data is saved in an XML file to the configured Output File path.

- Txt Format
  pluTableUpload|nOpe|pluFile.xml
  or with optional filter
  pluTableUpload|nOpe|pluFile.xml|pluListFile.xml

- Xml Format
  <pluTableUpload Ope="nOpe" pluDataBaseXmlFile="pluFile.xml" />
  or with optional filter
  <pluTableUpload Ope="nOpe" pluDataBaseXmlFile="pluFile.xml pluListXmlFile="pluListFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "pluFile.xml" indicates the destination XML filename.
- "pluListFile.xml" optionally indicates the name of a filter file. If it is not found, EpsonFpMate reads all 20000 PLU entries from the printer.

9.4.9.9   Update PLU Price List

MF models only. One or more prices can be updated from an XML PLU price list file with a specific structure. The root element must be named "PLUPriceList". Each PLU entry is defined as a "PLUItem" element with the following attributes:

- Code: PLU entry number from 0 to 20000.
- Barcode: EAN13 barcode.
- Price: Updated unit price

Code can be omitted provided Barcode is present. Barcode can be omitted provided Code is not 0. Both can be present. If Code is 0, Barcode is used.

- Txt Format
pluPriceUpdate|1|PluPriceFile.xml

- Xml Format
<pluPriceUpdate Ope="nOpe" pluListXmlFile="PluPriceFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "pluPriceFile.xml" indicates the XML PLU price list filename. The file must reside in the input directory. Paths are not supported.

9.4.9.10   Delete SD/MMC Memory Card PLU Entries

MF models only. Deletes one or more entries from the external PLU database that resides on the printer's SD/MMC card. An XML file specifies the entries to delete and has a specific structure. The root element must be named "PLUList". Each PLU entry is defined as a "PLUItem" element with one or both of the following attributes:

- Code: PLU entry number from 0 to 20000.
- Barcode: EAN13 barcode.

Code can be omitted provided Barcode is present. Barcode can be omitted provided Code is not 0. Both can be present. If Code is 0, Barcode is used.

- Txt Format
pluDelete|nOpe|pluListFile.xml

- Xml Format
<pluDelete Ope="1" pluListXmlFile="pluListFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "xmlPluListFile.xml" indicates the XML filename (lists the PLUs to delete). The file must reside in the input directory. Paths are not supported.

9.4.9.11   Read PLU Sales Data

MF models only. pluRead is used to read all or partial sales amounts per PLU entry from the external PLU database that resides on the printer's SD/MMC card. An optional filter file can be specified which contains a list of the desired PLU entries. The PLU list file is an XML file with a specific structure. The root element must be named "PLUList". Each PLU entry is defined as a "PLUItem" element with one or both of the following attributes:

- Code: PLU entry number from 0 to 20000.
- Barcode: PLU entry EAN13 barcode.

Code can be omitted provided Barcode is present. Barcode can be omitted provided Code is not 0. Both can be present. If Code is 0, Barcode is used.

Data is saved in an XML file to the configured Output File path. The root element in the file is called "PLUSalesData". Each PLU entry is then defined as a "PLUItem" element with the following attributes:

- Code: PLU entry number from 1 to 20000.
- Barcode: Only present if value non-zero.
- Date: Date that counter was read according to the printer's real time clock
- Time: Time that counter was read according to the printer's real time clock
- Items: Number of sold items
- Amount: Total sales amount

- Txt Format
  pluRead|nOpe|PluSalesData.xml
  or with optional filter
  pluRead|nOpe|PluSalesData.xml|pluListFile.xml

- Xml Format
  <pluRead Ope="1" pluSaledXmlFile="PluSalesData.xml" />
  or with optional filter
  <pluRead Ope="1" pluSaledXmlFile="PluSalesData.xml" pluListXmlFile="pluListFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "xmlPluListFile.xml" indicates the destination XML filename.
- "pluListFile.xml" optionally indicates the name of a filter file. If it is not found, EpsonFpMate reads the sales amounts for each of the 20000 PLU entries from the printer.

9.4.9.12   Read PLU Sales Data and Reset Counters

MF models only. pluReadReset is used to read all or partial sales amounts per PLU entry from the external PLU database that resides on the printer's SD/MMC card. After having read each entry, the sold items and amount counters are reset to zero by the printer. An optional filter file can be specified which contains a list of the desired PLU entries. Data is saved in an XML file to the configured Output File path. Both input and output XML files follow the same rules as pluRead.

- Txt Format
  pluReadReset|nOpe|PluSalesData.xm
  or with optional filter
  pluReadReset|nOpe|PluSalesData.xml|pluListFile.xml

- Xml Format
  <pluReadReset Ope="1" pluSaledXmlFile="PluSalesData.xml" />
  or with optional filter
  <pluReadReset   Ope="1"   pluSaledXmlFile="PluSalesData.xml"
  pluListXmlFile="pluListFile.xml" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).
- "xmlPluListFile.xml" indicates the destination XML filename.
- "pluListFile.xml" optionally indicates the name of a filter file. If it is not found, EpsonFpMate reads the sales amounts for each of the 20000 PLU entries from the printer.

9.4.9.13   Printer Reset

Soft resets the printer carrying out the following operations:

- Places the printer in the "STATO REGISTRAZIONE" condition irrespective of the current state.
- Closes any open non-fiscal receipt / management document.
- Cancels any open fiscal receipt / commercial document.
- Cancels any open credit note.
- Cancels any open direct invoice printing two copies.
- Cancels any open free invoice printing a single copy.
- Cancels any open box office ticket furthermore switching the printer into fiscal receipt mode ("SCONTRINO FISCALE").
- Unblocks the keyboard previously blocked via the 1-055 command.
- Erases the keyboard buffer.
- Terminates an eventual SET 22 printer test condition.

- Txt Format
  resetPrinter|nOpe

- Xml Format
  <resetPrinter Ope="nOpe" />

Where:

- "nOpe" specifies the operator ID (range 1 - 12).

9.4.10 Fiscal Document File Body (MF models only)

Not to be confused with commercial RT documents, this type of file is used to print invoices and open personalised fiscal receipts (ricevute fiscali). Since the latter is only opened, it is used in conjunction with a fiscal receipt formatted file. The following list of commands can be used in printerFiscalDocument files:

Command	Description
beginFiscalDocument	Open free invoice
printFiscalDocumentLine	Print a free invoice text row
printFiscalDocAmount	Print free invoice total amount line
endFiscalDocument	Close free invoice
printFiscalDocument	Print invoice based on last fiscal receipt or open personalised fiscal receipt

9.4.10.1 Free Invoice Prerequisites

Each file must contain:

- The beginFiscalDocument open free invoice command along with the total invoice amount.
- The invoice body including the retail header, client and VAT breakdown. printFiscalDocumentLine is used and an unlimited number of text rows can be included in the file. The only requirement is that they must be placed after the open and before the close commands.
- The endFiscalDocument close free invoice command.

The print amount command (printFiscalDocAmount) is optional.

9.4.10.2 Open a Free Invoice

beginFiscalDocument is used to open a free invoice. The invoice total amount including VAT must be specified. To open a free invoice based on the fiscal receipt, a Direct IO call must be used.

- Txt Format
  beginFiscalDocument|nOpe|invAmount

- Xml Format
  <beginFiscalDocument Ope="nOpe" Amount="invAmount" />

Where:

- **beginFiscalDocument** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **invAmount** is the total invoice amount.

9.4.10.3   Print Free Invoice Text Row

Used to construct the invoice line by line. Retail header and client lines must also be printed in this way. Up to four fonts can be used.

- Txt Format
  printFiscalDocumentLine|nOpe|Font|RowText

- Xml Format
  <printFiscalDocumentLine Ope="nOpe" Font="Font" Text= "RowText" />

Where:

- **printFiscalDocumentLine** is the command.
- **nOpe** specifies the operator ID (range 1 - 12).
- **Font** indicates one of the following fonts:
    - 1 = Normal
    - 2 = Bold
    - 3 = Double
    - 4 = Double and bold
- **RowText** specifies the row text.

9.4.10.4   Print Free Invoice Amount

Request that the printer print the total line. The amount is the one specified with beginFiscalDocument. This is an optional command and can be used in case extra free text lines are required after the total line.

- Txt Format
  printFiscalDocAmount|nOpe

- Xml Format
  <printFiscalDocAmount Ope="nOpe" />

Where:

- **printFiscalDocAmount** is the command.
- **nOpe** is the operator ID (range 1 - 12).

9.4.10.5 Close Free Invoice

The free invoice is closed. A second copy is printed. If printFiscalDocAmount was not used, the fiscal printer automatically prints the total line. EpsonFpMate returns a file that reports the result of the operation. If the result is OK, it also reports the date, time and invoice amount.

- Txt Format
  endFiscalDocument|nOpe

- Xml Format
  <endFiscalDocument Ope="nOpe" />

Where:

- **endFiscalDocument** is the command.
- **nOpe** is the operator ID (range 1 - 12).

9.4.10.6   Invoice based on Last Fiscal Receipt or Personalised Fiscal Receipt

The following operations are allowed:

- Print Invoice based on the last fiscal receipt which can be printed by the fiscal printer itself or by a secondary printer connected to the fiscal printer serial port.

- Open Personalised fiscal receipt (ricevuta fiscale). It is only supported if a secondary printer is connected to the fiscal printer serial port or the fiscal printer is an FP-H6000 with integrated slip printer.

- Txt Format
  printFiscalDocument|nOpe|docType|docNumber

- Xml Format
  <printFiscalDocument Ope="nOpe" Document="docType" Number="docNumber" />

Where:

- **printFiscalDocument** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **docType** is one of two document types (Invoice or Receipt).
- **docNumber** is the invoice or personalised fiscal receipt number (range 1 - 99999). With invoices it is not the receipt number which it is based on.

Additional header / invoice client lines

The printRecMessage command can be used to instruct the printer to print additional header lines and/or invoice client lines. Because they must be preloaded in the fiscal printer's memory, they must be placed before the printFiscalDocument command.

- Txt Format
  printRecMessage|nOpe|Type|Num|1|RowText

- Xml Format
  <printRecMessage Ope="Nope" Type="Type" Index="Num" Font="1" Text="RowText" />

Type has alias type.
Text has alias text.

Where:

- **printRecMessage** is the command.
- **nOpe** is the operator ID (range 1 - 12).
- **Type** defines the row type:
    - 5 = Additional header line
    - 6 = Invoice client line
- **Num** indicates the row number index:
    - Range 1 - 20 for additional header (type 5).
    - Range 1 - 5 for invoice client line (types 6).
- **Font** is always normal so can be 1.
- **RowText** represents the text to be loaded. The maximum length is 40 or 46 characters depending on the printer model.

9.4.10.7 Important Notes

- Previous set-up of the fiscal printer is required (SET 25 and SET 26).

- Personalised fiscal receipts (ricevute fiscali) can only be printed by a secondary printer connected to the fiscal printer serial port or by the FP-H6000 integrated slip printer.

- Invoices based on a fiscal receipt can only be printed after having first emitted a fiscal receipt ("printerFiscalReceipt" file). The invoice contains the data present in the previous receipt with some exceptions.

- To print out a personalised fiscal receipt, printFiscalDocument command is used with the "Receipt" indication. A "printerFiscalReceipt" file must then be processed immediately.

- The invoice number must be zero or **greater** than the last invoice number printed (it doesn't have to be previous + 1, it could be previous + 5 for example).

- The personalised fiscal receipt number must be zero or **greater** than the last one (it doesn't have to be previous + 1, it could be previous + 5 for example).

- If a file contains the "printFiscalDocument" command, it cannot include free invoice commands.

- The EpsonFpMate output file does not include the amount. The value is set to zero. Totaliser readings can be performed before and after to calculate the amount.

10   Input File Setup (WinEcrCom File D-Type Format)

10.1   Description

EpsonFpMate can also process input files in the WinEcrCom D-type format. The JavaPOS mode must be deactivated. Fiscal receipts, direct invoices, non-fiscal receipts and daily closures can only be emitted. The set of supported commands is a subset of the complete WinEcrCom command set. Some WinEcrCom commands are accepted by EpsonFpMate but they have no effect in the fiscal printer. It is also possible to use some alias names for WinEcrCom commands.

The following table lists all the WinEcrCom commands with the alias and notes related to them:

Command	Description	Alias	Notes
CANC	Clear button emulation	CLEAR	No effect at printer level.
CHIAVE	Key button emulation	KEY	No effect at printer level.
VEND	Sale, void and refund operations	SALE	Only department. PLU operations are not allowed.
SCONTO	Absolute value adjustment operations	DISC	Negative amounts are interpreted as a surcharge.
PERCA / PERCB	Percentage adjustment	None	Negative amounts are interpreted as a surcharge. The PERCB command is identical in all respects to the PERCA command. EpsonFpMate processes both commands (in precisely the same manner).
SUBT	Subtotal print and display	None	
PRMSG	Text to print	None	It is usually treated as an additional description in the body of the fiscal receipt. If sent just before the payment, it is treated as a trailer line. In the case of invoices, up to two lines can be printed as a footer. Cannot be the first line in the file.
CHIU	Payment	CHIUS CLOSE	One of four possible payment types are accepted.
CORT	Additional trailer lines	None	Up to ten lines can be printed.
FATTURA	Fattura Diretta	None	First line of file. Up to five client lines can be optionally printed.
NOFIS	Non-fiscal receipt	None	Free text lines and barcodes are supported.
report	Fiscal daily closure / Z report	AZZGIO	Two modes are possible (with or without X-01 financial report). The output filename is specified. AZZGIO can be used without arguments. In this case, the X-01 financial report is omitted.
VIS	Message for customer display	None	
ALLEG	Printing QR codes	None	The following parameters are hard-coded: • Medium-low error correction • Alphanumeric • Type 92
INP	Fiscal tax code (codice fiscale) Commercial refund documents Commercial void documents	None	The INP command has many variations. Please see the INP Fields section and examples.

For details relating to the record structures, please refer to WinEcrCom documents.

10.2   Fields (if Expected)

10.2.1   VEND (Alias SALE) Fields

Department – REP, DEP or R
Unit price – PREZZO, PRICE or P
Quantity – QTY, QUANTITY or Q. If missing, quantity = 1
Description – DES
Refund – RESO or REFUND
Cancel sale – STORNO or VOID

10.2.2   SCONTO (Alias DISC) Fields

Amount – VALORE, VALUE or VAL. Negative amounts are interpreted as a surcharge.
Description – DESCR.
Mode – SUBTOT or absence of SUBTOT – If SUBTOT is present, the discount or surcharge is
applied to the subtotal. If not present, it is applied with reference to the last sale.

10.2.3   PERCA / PERCB Fields

Percentage – ALIQ or RATE. Negative amounts are interpreted as a surcharge.
Description – DESCR.
Mode – SUBTOT or absence of SUBTOT – If SUBTOT is present, the discount or surcharge is
applied to the subtotal. If not present, it is applied with reference to the last sale.

10.2.4   PRMSG Fields

Text – LINE, RIGA or DES

The specific printer number can be indicated using the following special wording placed at the
beginning of the sequence:

- LINE='PRINTER|<printer number>'

10.2.5   CHIU (Aliases CHIUS and CLOSE) Fields

Type – TEND or T. The following values are accepted:

    0 or >5 – Cash
    1 – Cheque
    2 or 4 – Ticket
    3 or 5 – Credit card

Amount – IMPORTO, AMOUNT or IMP
Description – DES

10.2.6   CORT Fields

Line 1 – R1
...
...
Line 10 – R10

All "R" identifiers must be in the same single "CORT" line. Lines are printed after the fiscal serial number (additional trailer).

10.2.7   FATTURA Fields

NOME1 = Client row 1
NOME2 = Client row 2
NOME3 = Client row 3
NOME4 = Client row 4
NOME5 = Client row 5

10.2.8   NOFIS Fields

Open – APRI or OPEN
Close – CHIUDI or CLOSE
Text – LINE, RIGA or DES
Barcode – BARCODE. Must be preceded by a text field of the code itself. See example below. The following hard-coded parameters are used:

- Left margin – 120 dots.
- Barcode width – 1 dot.
- HRI – Below barcode.
- HRI Font – Font B.
- Type – CODE 39.

10.2.9   report (Alias AZZGIO) Fields

num – Ignored
modo – Regards X-01 Financial report: 1=Do not include and 2=Include
file – Output filename

AZZGIO can be used without fields. It can also be used without the file argument.

10.2.10   VIS Fields

CLI1 – Text for display

10.2.11   ALLEG Fields (QR Code)

ON – Empty tag positioned before CHIU
RIGA, BARCODE, SCALA and ALLINEAMENTO – Positioned after CHIU where
- RIGA – QR code data
- BARCODE – Value ignored but keyword must be present
- SCALA – Size from 1 (small) to 16 (large)
- ALLINEAMENTO – Horizontal alignment (0=left, 1=centred and 2=right)

FINE – Empty tag positioned after RIGA

10.2.12   INP Fields (Fiscal Tax Code (Codice Fiscale) and Refund and Void Documents)

TERM – The following values indicate the relative function:

- 49 – Fiscal tax code (codice fiscale)
- 61 – Ignored
- 145 – Depends on lengths of ALFA and NUM fields
- 180 – Commercial refund document
- 188 – Commercial void document

ALFA – Depends on TERM value and/or ALFA length as follows:

- TERM=49 – ALFA=Fiscal tax code (codice fiscale)
- TERM=145 and ALFA length=8 – Reference document consisting of Z report number and document number without spaces or hyphens plus padded with zeroes if necessary
- TERM=145 and ALFA length=11 – Fiscal serial number

NUM – TERM must be 145 and NUM length must be six. Indicates reference document date

See examples for complete sequences.

10.3   Post Description Line with SCONTO, PERCA and PERCB Commands ("0D")

The above three commands can be appended with a post description line using "0D" as a delimiter and PRMSG as the second command. There must be exactly 65 characters before the "0D". For example:

PERCA   ALIQ= 10.00 , SUBTOT   **0D**PRMSG LINE='Test'

10.4   Commercial Refund Document

Since the VEND sale command is used, flag SET 14/58 must be set to non-NdC mode (0 / NO).

10.5   Commercial Void Documents

Since the VEND sale command is used, flag SET 14/59 must be set to non-NdC mode (0 / NO). EpsonFpMate automatically determines if the void is automatic or manual.

11   Output Files

When the **Output File** option is selected in EpsonFpMateConfig, EpsonFpMate generates an output file. The following details are reported:

- Date and time
- Printer number
- Cash Id number
- Operation result (ok or ko)
- If Result = ok
  - Receipt number
  - Total receipt amount
  - From version 5.6.0.29 (Txt) and 5.6.0.31 (XML), the current Z report number and document number combination is included in a new ZDocNum attribute.
- If Result = ko
  - Error code
  - Short error description

The following table lists some keyword changes that were adopted:

Input file type	Keyword prior to version 5.6.0.31	Keyword from version 5.6.0.31 onwards	Comments
printerFiscalReceipt	ReceiptReport	FiscalReceiptReport	Missing Fiscal keyword portion
printerVoidReceipt			
printRecVoid			
printerCloseReceipt			
printerFiscalDocument	FiscalDocumentReport	FiscalDocumentReport	Unchanged
printerNonFiscal	XML file type: NonFiscalReceiptReport TXT file type: Not Managed (error)	NonFiscalReceiptReport	Corrected error with TXT file type
printerCommand	Command	CommandReport	Missing Report keyword portion
printerFiscalReport	ReportReport	FiscalDailyReport	Missing Fiscal keyword portion. Added Daily keyword portion for clarity

12 Sample Files

12.1   Text Delimited File Examples

12.1.1   Fiscal Receipt (in MF mode)

```
printerFiscalReceipt
Printer|1
printRecMessage|1|1|1|1|First additional header type 1 line
printRecMessage|1|1|2|1|Second additional header type 1 line
printRecMessage|1|4|1|1|First additional body type 4 line
printRecItem|1|Sale department 1 22% VAT|1|100,0|1|1
printRecMessage|1|4|1|1|Second additional body type 4 line
printRecItem|1|Sale department 2 10% VAT|1,234|100,0|2|1
printRecVoidItem|1|Sale correction dept 1 22% VAT|1|100,0|1|1
printRecItem|1|Sale department 3 4% VAT|2,5|100,17|3|1
printRecMessage|1|4|1|1|Third additional body type 4 line
printRecRefund|1|Refund department 2 10% VAT|10|5,0|2|1
printRecItem|1|Sale department 13 VAT exempt|12,13|216,17|13|1
printRecItemAdjustment|1|Discount on last sale|0|123,45|13|1
printRecSubtotalAdjustment|1|Discount on subtotal|1|300,12|0|1
printRecSubtotal|1|1
printRecMessage|1|2|1|1|First trailer type 2 line
printRecMessage|1|2|2|1|Second trailer type 2 line
printRecMessage|1|3|1|1|First additional trailer type 3 line
printRecMessage|1|3|2|1|Second additional trailer type 3 line
printBarCode|1|901|2|66|TWICE|FontA|CODE39|01234567ABCDEF
printRecTotal|1|Credit card index 3|6000,0|2|3|1
displayText|1|Customer Display    Printed Fisc Receipt
```

12.1.2   Commercial Document (in RT mode)

printerFiscalReceipt
Printer|2
printRecMessage|1|1|1|1|First additional header type 1 line
printRecMessage|1|1|2|1|Second additional header type 1 line
printRecMessage|1|4|1|1|First additional body type 4 line
printRecItem|1|Sale department 1 22% VAT|1|100,0|1|1
printRecMessage|1|4|1|1|Second additional body type 4 line
printRecItem|1|Sale department 2 10% VAT|1,234|100,0|2|1
printRecVoidItem|1|Sale correction dept 1 22% VAT|1|100,0|1|1
printRecItem|1|Sale department 3 4% VAT|2,5|100,17|3|1
printRecMessage|1|4|1|1|Third additional body type 4 line
printRecItem|1|Sale department 13 VAT exempt|12,13|216,17|13|1
printRecItemAdjustment|1|Discount on last sale|0|123,45|13|1
printRecItem|1|Sale department 10 VAT excluded|12,13|216,17|10|1
printRecSubtotalAdjustment|1|Discount on subtotal|1|300,12|0|1
printRecSubtotal|1|1
printRecMessage|1|2|1|1|First trailer type 2 line
printRecMessage|1|2|2|1|Second trailer type 2 line
printRecMessage|1|3|1|1|First additional trailer type 3 line
printRecMessage|1|3|2|1|Second additional trailer type 3 line
printBarCode|1|901|2|66|TWICE|FontA|CODE39|01234567ABCDEF
printRecTotal|1|Credit card index 3|6000,0|2|3|1
displayText|1|Customer Display   Printed Fisc Receipt

12.1.3   Credit Note Fiscal Receipt (MF mode only)

printerFiscalReceipt
Printer|1
printRecMessage|1|1|1|1|First additional header type 1 line
printRecMessage|1|1|2|1|Second additional header type 1 line
printRecMessage|1|4|1|1|PRATICA DI RESO 1/2019
printRecRefund|1|Refund 1|10|5,0|1|1
printRecRefund|1|Refund 2|13|6,0|1|1
printRecRefund|1|Refund 3|1,5|7,0|1|1
printRecItemAdjustment|1|Discount on last transaction|0|1,23|1|1
printRecMessage|1|2|1|1|First trailer type 2 line
printRecMessage|1|2|2|1|Second trailer type 2 line
printRecMessage|1|3|1|1|First additional trailer type 3 line
printRecMessage|1|3|2|1|Second additional trailer type 3 line
printRecSubtotal|1|1
printBarCode|1|901|2|100|BELOW|FontA|CODABAR|A0123456789ABCD
printRecTotal|1| |0|0|1|1
displayText|1|Arrivederci e grazie

12.1.4   Commercial Refund Document

printerFiscalReceipt
Printer|2
printRecMessage|1|1|1|4|Additional Header
printRecMessage|1|4|1|4|REFUND 0166 0051 06092020 99MEX000004
printRecRefund|1|T-SHIRT|1|10|1|1
printRecTotal|1|CONTANTI|10,00|0|0|1

12.1.5   Commercial Void Document

printerFiscalReceipt
Printer|2
printRecMessage|1|1|1|4|Additional Header
printRecMessage|1|4|1|4|VOID 0221 0006 06092020 99MEY000005
printRecRefund|1|T-SHirt|1|10|1|1
printRecTotal|1|CONTANTI|10|0|1|1

12.1.6   Z Report

printerFiscalReport
Printer|1
printZReport|1
displayText|1|Message on   customer display

12.1.7   Non-Fiscal Receipt / Management document with Text Bar Codes and QR Codes

printerNonFiscal
Printer|1
printRecMessage|1|1|1|1|First additional header type 1 line
printRecMessage|1|1|2|1|Second additional header type 1 line
beginNonFiscal|1
printNormal|1|1|Line font 1
printNormal|1|2|
printNormal|1|2|CODE39 barcode standard
printBarCode|1|901|2|66|TWICE|FontA|CODE39|01234567ABCD
printNormal|1|2|
printNormal|1|2|Line font 2
printNormal|1|3|Line font 3
printNormal|1|2|
printNormal|1|2|CODE93 barcode standard
printBarCode|1|901|2|66|TWICE|FontA|CODE93|01234567ABCDEF
printNormal|1|2|
printNormal|1|4|Line font 4
printNormal|1|2|
printNormal|1|2|CODE 128 subtype A barcode standard
printBarCode|1|901|2|100|TWICE|FontA|CODE128|{A01234567ABCDEF
printNormal|1|2|
printNormal|1|2|CODE 128 subtype B barcode standard
printBarCode|1|901|2|100|TWICE|FontA|CODE128|{B01234567ABCDEF
printNormal|1|2|
printNormal|1|2|CODE 128 subtype C barcode standard
printBarCode|1|901|2|66|TWICE|FontA|CODE128|{C0123456789
printNormal|1|2|
printNormal|1|2|QR code type 2 standard
printQRCode|1|1||8|ALPHANUMERIC|LOW|QRCODE2|123456789
displayText|4|Customer Display   Printed Non Fisc Doc
endNonFiscal|1

12.1.8   Print Duplicate Fiscal Receipt / Commercial Document

printerNonFiscal
Printer|1
printDuplicateReceipt|1

12.1.9   Direct IO

printerCommand
printer|1
directIO|opRepartoLight|1|1|Mobile phone|100,0|1|1
directIO|opAddRowDesc|1|4|1|1|IMEI:0001213540
directIO|opRepartoLight|1|2|Product description|12,34|1|1
printReadProperty|LastOPCode
printReadProperty|LastOP
directIO|opSubTot|1
directIO|opPaymentLight|1|Cash payment|300,00|0|2|1

12.1.10   Double Direct IO (Native Protocol)

printerCommand
printer|1
directIO|DirectIO|321701
printReadProperty|LastRxFrameInDirectIoMode

12.1.11   Fiscal Receipt / Commercial Document with Graphic Coupon

printerFiscalReceipt
printer|1
beginFiscalReceipt|1
printRecItem|1|Mobile phone|1|100,0|1|1
printRecMessage|1|4|1|1|IMEI:0001213540
printGraphicCoupon|1|coupon.bin
printRecSubtotal|1|1
printRecTotal|1|Cash payment|300,00|0|1|1

12.1.12   Non-fiscal Receipt / Management Document with Graphic Coupon

printerNonFiscal
printer|1
beginNonFiscal|1
printGraphicCoupon|1|coupon.bin
endNonFiscal|1

12.1.13   Non-fiscal Receipt / Management Document with EFT-POS Lines

printerNonFiscal
printer|1
beginNonFiscal|1
printRecMessage|1|8|1|1|
endNonFiscal|1

12.1.14   Free Invoice Document (MF mode only)

printerFiscalDocument
Printer|1
beginFiscalDocument|1|1234,56
printFiscalDocumentLine|1|1|Font 1
printFiscalDocumentLine|1|2|Font 2
printFiscalDocumentLine|1|3|Font 3
printFiscalDocumentLine|1|4|Font 4
printFiscalDocAmount|1
printFiscalDocumentLine|1|1|Line after total
endFiscalDocument|1

12.1.15   Invoice Based on the Last Fiscal Receipt

printerFiscalDocument
Printer|1
printRecMessage|1|5|1|1|First additional header type 5 line
printRecMessage|1|5|2|1|Second additional header type 5 line
printRecMessage|1|6|1|1|First client line
printRecMessage|1|6|2|1|Second client line
printFiscalDocument|1|Invoice|0

12.1.16   Personalised Fiscal Receipt (Ricevuta Fiscale)

printerFiscalDocument
Printer|1
printFiscalDocument|1|Receipt|0

12.1.17   EJ /MPD Query

printerCommand
Printer|1
queryEjContent|1|0|7|3|2020|8|3|2020|7-8_March.txt

12.1.18   EJ /MPD Invoice Query

printerCommand
Printer|1
directIO|QueryContent|1|2|7|3|2020|8|3|2020|C:\Epson\Invoices_7-8_March.txt

12.1.19   Printer Status Query

printerCommand
Printer|1
queryPrinterStatus|1|Printer1_Status.txt

12.1.20   Printer RT Status Query

printerCommand
Printer|1
queryPrinterStatus|1|1|Printer1_RTStatus.txt

12.1.21   Customer Display Message

printerCommand
Printer|1
displayText|1|12345678901234567890123456789012345678901234567890

12.1.22   Download PLU Table to Printer – MF Models Only

printerCommand
Printer|1
pluTableDownload|1|Default_20000_Plu.xml

12.1.23   Upload Complete PLU Table from Printer – MF Models Only

printerCommand
Printer|1
pluTableUpload|1|CompletePluTable.xml

12.1.24   Upload Partial PLU Data from Printer Filtered by a Defined List – MF Models Only

printerCommand
Printer|1
pluTableUpload|1|PartialPluTable.xml|PluList_01000.xml

12.1.25   Delete One or More PLU Entries – MF Models Only

printerCommand
Printer|1
pluDelete|1|PartialPluTable.xml|PluList_01000.xml

12.1.26   Update One or More PLU Prices – MF Models Only

printerCommand
Printer|1
pluPriceUpdate|1|PluPriceList_01000.xml

12.1.27   Collect All PLU Sales Data Counters – MF Models only

printerCommand
Printer|1
pluRead|1|PluSalesData.xml

12.1.28   Collect Partial PLU Sales Data Counters Filtered by a Defined List – MF Models Only

printerCommand
Printer|1
pluRead|1|PluSalesData.xml|PluList_01000.xml

12.1.29   Collect and Reset all PLU Sales Data Counters – MF Models Only

printerCommand
Printer|1
pluReadReset|1|PluSalesData.xml

12.1.30   Collect and Reset Partial PLU Sales Data Counters Filtered by a Defined List – MF Models Only

printerCommand
Printer|1
pluReadReset|1|PluSalesData.xml|PluList_01000.xml

12.1.31   Exit Application (Close EpsonFpMate)

applicationExit

12.2 XML File Examples

12.2.1   Fiscal Receipt (in MF mode)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
 <Printer Num="1" />
 <printRecMessage Ope="1" Text="First additional header type 1 line" Type="1" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second additional header type 1 line" Type="1" Index="2" Font="1" />
 <beginFiscalReceipt Ope="1" />
 <printRecMessage Ope="1" Text="First additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecItem Ope="1" Text="Sale department 1 22% VAT" Qty="1" UnitCost="100" Dep="1" Just="1" />
 <printRecMessage Ope="1" Text="Second additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecItem Ope="1" Text="Sale department 2 10% VAT" Qty="1,234" UnitCost="100" Dep="2" Just="1" />
 <printRecVoidItem Ope="1" Text="Sale correction dept 1 22% VAT" Qty="1" UnitCost="100" Dep="1" Just="1" />
 <printRecItem Ope="1" Text="Sale department 3 4% VAT" Qty="2,5" UnitCost="100,17" Dep="3" Just="1" />
 <printRecMessage Ope="1" Text="Third additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecRefund Ope="1" Text="Refund department 2 10% VAT" Qty="10" UnitCost="5" Dep="2" Just="1" />
 <printRecItem Ope="1" Text="Sale department 13 VAT exempt" Qty="12,13" UnitCost="216,17" Dep="13" Just="1" />
 <printRecItemAdjustment Ope="1" Text="Discount on last sale" Type="0" Amount="123,45" Dep="13" Just="1" />
 <printRecSubtotalAdjustment Ope="3" Text="Discount on subtotal" Type="1" Amount="300,12" Dep="0" Just="2" />
 <printRecSubtotal Ope="1" Type="1" />
 <printRecMessage Ope="1" Text="First trailer type 2 line" Type="2" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second trailer type 2 line" Type="2" Index="2" Font="1" />
 <printRecMessage Ope="1" Text="First additional trailer type 3 line" Type="3" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second additional trailer type 3 line" Type="3" Index="2" Font="1" />
 <printBarCode Ope="1" Pos="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE39" Text="0123456789" />
 <printRecTotal Ope="1" Text="Credit card index 3" Amount="6000,0" Type="2" Index="3" Just="1" />
 <displayText Ope="1" Text="Message on        customer display" />
</printerFiscalReceipt>
```

12.2.2   Commercial Document (in RT mode)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
 <Printer Num="2" />
 <printRecMessage Ope="1" Text="First additional header type 1 line" Type="1" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second additional header type 1 line" Type="1" Index="2" Font="1" />
 <beginFiscalReceipt Ope="1" />
 <printRecMessage Ope="1" Text="First additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecItem Ope="1" Text="Sale department 1 22% VAT" Qty="1" UnitCost="100" Dep="1" Just="1" />
 <printRecMessage Ope="1" Text="Second additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecItem Ope="1" Text="Sale department 2 10% VAT" Qty="1,234" UnitCost="100" Dep="2" Just="1" />
 <printRecVoidItem Ope="1" Text="Sale correction dept 1 22% VAT" Qty="1" UnitCost="100" Dep="1" Just="1" />
 <printRecItem Ope="1" Text="Sale department 3 4% VAT" Qty="2,5" UnitCost="100,17" Dep="3" Just="1" />
 <printRecMessage Ope="1" Text="Third additional body type 4 line" Type="4" Index="1" Font="1" />
 <printRecItem Ope="1" Text="Sale department 13 VAT exempt" Qty="12,13" UnitCost="216,17" Dep="13" Just="1" />
 <printRecItemAdjustment Ope="1" Text="Discount on last sale" Type="0" Amount="123,45" Dep="13" Just="1" />
 <printRecItem Ope="1" Text="Sale department 10 VAT excluded" Qty="12,13" UnitCost="216,17" Dep="10" Just="1" />
 <printRecSubtotalAdjustment Ope="3" Text="Discount on subtotal" Type="1" Amount="300,12" Dep="0" Just="2" />
 <printRecSubtotal Ope="1" Type="1" />
 <printRecMessage Ope="1" Text="First trailer type 2 line" Type="2" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second trailer type 2 line" Type="2" Index="2" Font="1" />
 <printRecMessage Ope="1" Text="First additional trailer type 3 line" Type="3" Index="1" Font="1" />
 <printRecMessage Ope="1" Text="Second additional trailer type 3 line" Type="3" Index="2" Font="1" />
 <printBarCode Ope="1" Pos="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE39" Text="0123456789" />
 <printRecTotal Ope="1" Text="Credit card index 3" Amount="6000,0" Type="2" Index="3" Just="1" />
 <displayText Ope="1" Text="Message on        customer display" />
</printerFiscalReceipt>
```

12.2.3   Credit Note Fiscal Receipt (MF mode only)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
  <Printer Num="1" />
  <printRecMessage Ope="1" Text="First additional header type 1 line" Type="1" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second additional header type 1 line" Type="1" Index="2" Font="1" />
  <printRecMessage Ope="1" Text="PRATICA DI RESO 1/2019" Type="4" Index="1" Font="1" />
  <printRecRefund Ope="1" Text="Refund 1" Qty="10" UnitCost="5" Dep="1" Just="1" />
  <printRecRefund Ope="1" Text="Refund 2" Qty="13" UnitCost="6" Dep="1" Just="1" />
  <printRecRefund Ope="1" Text="Refund 3" Qty="1,5" UnitCost="7" Dep="1" Just="1" />
  <printRecItemAdjustment Ope="2" Text="Discount on last transaction" Type="0" Amount="1,23" Dep="1" Just="1"
      />
  <printRecSubtotal Ope="1" Type="1" />
  <printRecMessage Ope="1" Text="First trailer type 2 line" Type="2" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second trailer type 2 line" Type="2" Index="2" Font="1" />
  <printRecMessage Ope="1" Text="First additional trailer type 3 line" Type="3" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second additional trailer type 3 line" Type="3" Index="2" Font="1" />
  <printBarCode Ope="1" Pos="901" Width="2" Height="66" Hri="BELOW" Font="FontA" Tipo="CODABAR"
      Text="A0123456789ABCD" />
  <printRecTotal Ope="1" Text="" Amount="0" Type="0" Index="0" Just="1" />
  <displayText Ope="1" Text="Arrivederci e grazie" />
</printerFiscalReceipt>
```

12.2.4   Commercial Refund Document

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
  <Printer Num="2" />
  <printRecMessage Ope="1" Text="Additional Header" Type="1" Index="1" Font="4" />
  <printRecMessage Ope="1" Text="REFUND 0166 0053 06092020 99MEX000004" Type="4" Index="1" Font="4" />
  <printRecRefund Ope="1" Text="T-SHIRT" Qty="1" UnitCost="10" Dep="1" Just="1" />
  <printRecTotal Ope="1" Text="CONTANTI" Amount="10" Type="0" Index="1" Just="1" />
</printerFiscalReceipt>
```

12.2.5   Commercial Void Document

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
  <Printer Num="2" />
  <printRecMessage Ope="1" Text="Additional Header" Type="1" Index="1" Font="4" />
  <printRecMessage Ope="1" Text="VOID 0166 0053 06092020 99MEY000005" Type="4" Index="1" Font="4" />
  <printRecRefund Ope="1" Text="T-SHIRT" Qty="1" UnitCost="10" Dep="1" Just="1" />
  <printRecTotal Ope="1" Text="CONTANTI" Amount="10" Type="0" Index="1" Just="1" />
</printerFiscalReceipt>
```

12.2.6   Z Report

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReport>
  <Printer Num="1" />
  <printZReport Ope="1" />
  <displayText Ope="1" Text="Message on        customer display" />
</printerFiscalReport>
```

12.2.7   Non-Fiscal Receipt / Management document with Text Bar Codes and QR Codes

```
<?xml version="1.0" encoding="utf-16" ?>
<printerNonFiscal>
  <Printer Num="1" />
  <printRecMessage Ope="1" Text="First additional header type 1 line" Type="1" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second additional header type 1 line" Type="1" Index="2" Font="1" />
  <beginNonFiscal Ope="1" />
  <printNormal Ope="1" Font="1" Text="Line font 1" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="CODE39 barcode standard" />
  <printBarCode Ope="1" Position="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE39"
      Text="01234567ABCD" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="Line font 2" />
  <printNormal Ope="1" Font="3" Text="Line font 3" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="CODE93 barcode standard" />
  <printBarCode Ope="1" Position="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE93"
      Text="01234567ABCDEF" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="4" Text="Line font 4" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="CODE 128 subtype A barcode standard" />
  <printBarCode Ope="1" Position="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE128"
      Text="{A01234567ABCDEF" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="CODE 128 subtype B barcode standard" />
  <printBarCode Ope="1" Position="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE128"
      Text="{B01234567ABCDEF" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="CODE 128 subtype C barcode standard" />
  <printBarCode Ope="1" Position="901" Width="2" Height="66" Hri="TWICE" Font="FontA" Tipo="CODE128"
      Text="{C0123456789" />
  <printNormal Ope="1" Font="2" Text="" />
  <printNormal Ope="1" Font="2" Text="QR code type 2 standard" />
  <printQRCode Ope="1" Position="1" Size="8" DataType="ALPHANUMERIC" ErrorCorrectioni="LOW" Tipo="QRCODE2"
      Text="123456789" />
  <displayText Ope="1" Text="Message on        customer display" />
  <endNonFiscal Ope="1" />
</printerNonFiscal>
```

12.2.8   Print Duplicate Fiscal Receipt / Commercial Document

```
<?xml version="1.0" encoding="utf-16" ?>
<printerNonFiscal>
  <Printer Num="1" />
  <printDuplicateReceipt Ope="1" />
</printerNonFiscal>
```

12.2.9   Direct IO

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <directIO OcxMethod="opRepartoLight" Arg0="1" Arg1="1" Arg2="Mobile phone" Arg3="100,0" Arg4="1" Arg5="1"
      />
  <directIO OcxMethod="opAddRowDesc" Arg0="1" Arg1="4" Arg2="1" Arg3="1" Arg4="IMEI:0001213540" />
  <directIO OcxMethod="opRepartoLight" Arg0="1" Arg1="1" Arg2="Product description" Arg3="12,34" Arg4="1"
      Arg5="1" />
  <printReadProperty OcxProperty="LastOPCode" />
  <printReadProperty OcxProperty="LastOP" />
  <directIO OcxMethod="opSubTot" Arg0="0" />
  <directIO OcxMethod="opPaymentLight" Arg0="1" Arg1="Cash payment" Arg2="300,00" Arg3="0" Arg4="2"
      Arg5="1" />
  </printerCommand>
```

12.2.10   Double Direct IO (Native Protocol)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <directIO OcxMethod="DirectIO" Arg0="321701" />
  <printReadProperty OcxProperty="LastRxFrameInDirectIoMode" />
</printerCommand>
```

12.2.11   Fiscal Receipt / Commercial Document with Graphic Coupon

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalReceipt>
  <Printer Num="1" />
  <beginFiscalReceipt Ope="1" />
  <printRecItem Ope="1" Text="Mobile phone" Qty="1" UnitCost="100" Dep="1" Just="1" />
  <printRecMessage Ope="1" Text="IMEI:0001213540" Type="4" Index="1" Font="1" />
  <printGraphicCoupon Ope="1" GraphicFile="coupon.bin" />
  <printRecTotal Ope="1" Text="Cash payment" Amount="300" Type="0" Index="1" Just="1" />
</printerFiscalReceipt>
```

12.2.12   Non-fiscal Receipt / Management Document with Graphic Coupon

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerNonFiscal>
  <Printer Num="1" />
  <beginNonFiscal Ope="1" />
  <printGraphicCoupon Ope="1" GraphicFile="coupon.bin" />
  <endNonFiscal Ope="1" />
</printerNonFiscal>
```

12.2.13   Non-fiscal Receipt / Management Document with EFT-POS Lines

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerNonFiscal>
  <Printer Num="1" />
  <beginNonFiscal Ope="1" />
  <printRecMessage Ope="1" Text="" Type="8" Index="1" Font="1" />
  <endNonFiscal Ope="1" />
</printerNonFiscal>
```

12.2.14   Free Invoice Document (MF mode only)

```xml
<?xml version="1.0" encoding="utf-16" ?>
  <printerFiscalDocument>
  <Printer Num="1" />
  <beginFiscalDocument Ope="1" Amount="1234,56" />
  <printFiscalDocumentLine Ope="1" Font="1" Text="Font 1" />
  <printFiscalDocumentLine Ope="1" Font="2" Text="Font 2" />
  <printFiscalDocumentLine Ope="1" Font="3" Text="Font 3" />
  <printFiscalDocumentLine Ope="1" Font="4" Text="Font 4" />
  <printFiscalDocAmount Ope="1" />
  <printFiscalDocumentLine Ope="1" Font="1" Text="Line after total" />
  <endFiscalDocument Ope="1" />
</printerFiscalDocument>
```

12.2.15   Invoice Based on the Last Fiscal Receipt

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalDocument>
  <Printer Num="1" />
  <printRecMessage Ope="1" Text="First additional header type 5 line" Type="5" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second additional header type 5 line" Type="5" Index="2" Font="1" />
  <printRecMessage Ope="1" Text="First client line" Type="6" Index="1" Font="1" />
  <printRecMessage Ope="1" Text="Second client line" Type="6" Index="2" Font="1" />
  <printFiscalDocument Ope="1" Document="Invoice" Number="0" />
</printerFiscalDocument>
```

12.2.16   Personalised Fiscal Receipt (Ricevuta Fiscale)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerFiscalDocument>
  <Printer Num="1" />
  <printFiscalDocument Ope="1" Document="Receipt" Number="0" />
</printerFiscalDocument>
```

12.2.17   EJ / MPD Query

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <queryEjContent Ope="1" dataType="0" stDay="7" stMonth="3" stYear="2020" endDay="8" endMonth="3"
      endYear="2020" fileName="7-8_March.txt" />
</printerCommand>
```

12.2.18   EJ / MPD Invoice Query

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <directIO OcxMethod="QueryContent" Arg0="1" Arg1="2" Arg2="7" Arg3="3" Arg4="2020" Arg5="8" Arg6="3"
      Arg7="2020" Arg8="C:\Epson\Invoices_7-8_March.txt" />
</printerCommand>
```

12.2.19   Printer Status Query

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <queryPrinterStatus Ope="1" fileName="Printer1_Status.txt" />
</printerCommand>
```

12.2.20   Printer RT Status Query

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <queryPrinterStatus Ope="1" statusType="1" fileName="Printer1_RTStatus.txt" />
</printerCommand>
```

12.2.21   Customer Display Message

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
    <Printer Num="1" />
    <displayText Ope="1" Text="12345678901234567890123456789012345678901234567890" />
</printerCommand>
```

12.2.22   Download PLU Table to Printer – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluTableDownload Ope="1" pluDataBaseXmlFile="Default_20000_Plu.xml" />
</printerCommand>
```

12.2.23   Upload Complete PLU Table from Printer – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluTableUpload Ope="1" pluDataBaseXmlFile="CompletePluTable.xml" />
</printerCommand>
```

12.2.24   Upload Partial PLU Data from Printer Filtered by a Defined List – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
  <printerCommand>
  <Printer Num="1" />
  <pluTableUpload Ope="1" pluDataBaseXmlFile="PartialPluTable.xml" pluListXmlFile="PluList_01000.xml" />
</printerCommand>
```

12.2.25   Delete One or More PLU Entries – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluDelete Ope="1" pluListXmlFile="PluList_01000.xml" />
</printerCommand>
```

12.2.26   Update One or More PLU Prices – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluPriceUpdate Ope="1" pluListXmlFile="PluPriceList_01000.xml" />
</printerCommand>
```

12.2.27   Collect All PLU Sales Data Counters – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluRead Ope="1" pluSaledXmlFile="PluSalesData.xml" />
</printerCommand>
```

12.2.28   Collect Partial PLU Sales Data Counters Filtered by a Defined List – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluRead Ope="1" pluSaledXmlFile="PluSalesData.xml" pluListXmlFile="PluList_01000.xml" />
</printerCommand>
```

12.2.29   Collect and Reset All PLU Sales Data Counters – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
  <Printer Num="1" />
  <pluReadReset Ope="1" pluSaledXmlFile="PluSalesData.xml" />
</printerCommand>
```

12.2.30   Collect and Reset Partial PLU Sales Data Counters Filtered by a Defined List – MF Models Only

```xml
<?xml version="1.0" encoding="utf-16" ?>
<printerCommand>
 <Printer Num="1" />
 <pluReadReset Ope="1" pluSaledXmlFile="PluSalesData.xml" pluListXmlFile="PluList_01000.xml" />
</printerCommand>
```

12.2.31   Exit Application (Close EpsonFpMate)

```xml
<?xml version="1.0" encoding="utf-16" ?>
<applicationExit />
```

12.3 PLU XML List File Examples – MF Models Only

12.3.1   PLU Table

```xml
<PLUTable>
  <PLUItem Code="1" Description="Descr. Plu N. 1" Barcode="8000000000001" Price="500,01" Category="2"
      Department="2" />
  <PLUItem Code="2" Description="Descr. Plu N. 2" Barcode="8000000000002" Price="500,02" Category="3"
      Department="3" />
  <PLUItem Code="3" Description="Descr. Plu N. 3" Barcode="8000000000003" Price="500,03" Category="4"
      Department="4" />
  <PLUItem Code="4" Description="Descr. Plu N. 4" Barcode="8000000000004" Price="500,04" Category="5"
      Department="5" />
  <PLUItem Code="5" Description="Descr. Plu N. 5" Barcode="8000000000005" Price="500,05" Category="6"
      Department="6" />

  …

  <PLUItem Code="19996" Description="Descr. Plu N. 19996" Barcode="8000000019996" Price="699,96"
      Category="7" Department="96" />
  <PLUItem Code="19997" Description="Descr. Plu N. 19997" Barcode="8000000019997" Price="699,97"
      Category="8" Department="97" />
  <PLUItem Code="19998" Description="Descr. Plu N. 19998" Barcode="8000000019998" Price="699,98"
      Category="9" Department="98" />
  <PLUItem Code="19999" Description="Descr. Plu N. 19999" Barcode="8000000019999" Price="699,99"
      Category="10" Department="99" />
  <PLUItem Code="20000" Description="Descr. Plu N. 20000" Barcode="8000000020000" Price="700" Category="1"
      Department="1" />
</PLUTable>
```

12.3.2   PLU List

```xml
<PLUList>
  <PLUItem Code="1" />
  <PLUItem Code="3" />
  <PLUItem Code="4" />
  <PLUItem Code="8" />
  <PLUItem Code="9" />
  <PLUItem Code="10" />
  <PLUItem Code="12" />

  …

  <PLUItem Code="19990" />
  <PLUItem Code="20000" />
</PLUList>
```

12.3.3   PLU Price List

```xml
<PLUPriceList>
  <PLUItem Code="1" Price="800,01" />
  <PLUItem Code="2" Price="800,02" />
  <PLUItem Code="3" Price="800,03" />
  <PLUItem Code="4" Price="800,04" />
  <PLUItem Code="5" Price="800,05" />
  <PLUItem Code="6" Price="800,06" />
  <PLUItem Code="14" Price="800,14" />
  <PLUItem Code="16" Price="800,16" />
  <PLUItem Code="18" Price="800,18" />
  <PLUItem Code="19" Price="800,19" />
  <PLUItem Code="20" Price="800,2" />

  ...

  <PLUItem Code="19996" Price="999,96" />
  <PLUItem Code="19997" Price="999,97" />
  <PLUItem Code="19998" Price="999,98" />
  <PLUItem Code="19999" Price="999,99" />
  <PLUItem Code="20000" Price="1000" />
</PLUPriceList>
```

12.3.4   PLU Sales Data

```xml
<PLUSalesData>
  <PLUItem Code="1" Date="14/05/12" Time="17:42" Items="1" Amount="800,01" />
  <PLUItem Code="2" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="3" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="4" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="5" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="6" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="7" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="8" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="9" Date="14/05/12" Time="17:42" Items="1" Amount="800,09" />
  <PLUItem Code="10" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="11" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="12" Date="14/05/12" Time="17:42" Items="0" Amount="0" />
  <PLUItem Code="13" Date="14/05/12" Time="17:42" Items="0" Amount="0" />

  ...

  <PLUItem Code="997" Date="14/05/12" Time="17:43" Items="0" Amount="0" />
  <PLUItem Code="998" Date="14/05/12" Time="17:43" Items="0" Amount="0" />
  <PLUItem Code="999" Date="14/05/12" Time="17:43" Items="0" Amount="0" />
  <PLUItem Code="1000" Date="14/05/12" Time="17:43" Items="0" Amount="0" />
</PLUSalesData>
```

12.4 WinEcrCom File D-Type Format Examples

12.4.1   Fiscal Receipt 1

```
VEND    REP=1 , PREZZO=      5.50, QTY=   1.00, DES='san marzano '
VEND    REP=1 , PREZZO=      8.80, QTY=   1.00, DES='verace      '
VEND    REP=1 , PREZZO=      8.50, QTY=   1.00, DES='sant agata  '
PERCA   ALIQ=  10.00 , SUBTOT
PRMSG LINE='Prestazione Gratuita                                   '
SCONTO VALORE=  2.80 , SUBTOT
PRMSG LINE='La tua card oggi ti ha                                 '
PRMSG LINE='dato diritto ad uno                                    '
PRMSG LINE='sconto di E.     5,08                                  '
PRMSG LINE='Promo per E.                                           '
PRMSG LINE='per infocard visita il                                 '
PRMSG LINE='sito www.epson.it                                      '
PRMSG LINE='                                                       '
PRMSG LINE='Punti     0 Al 24/09/10                                '
PRMSG LINE='                                                       '
PRMSG LINE='Tavolo N.  36                                          '
CORT R1='--- CORTESIA 1 ----------', R2='--- CORTESIA 2 ----------'
CHIU    t=1, imp=   17.72, des='cash'
```

12.4.2   Fiscal Receipt 2

```
; Esempio di scontrino di vendita inviato a ECR mediante ECRCOM.EXE
; ----------------------------------------------------------------
;
CLEAR         ;preme il tasto C
CHIAVE REG    ;conferma che la cassa si trovi in assetto Registrazione

VEND REP=1,PREZZO=25.00            ;vendita semplice a reparto 1
SCONTO VAL=500                     ;sconto assoluto diretto su reparto

VEND REP=2,PREZZO=25.00            ;vendita semplice a reparto 2
VEND REP=3,PREZZO=25.00            ;vendita semplice a reparto 3
VEND REP=4,PREZZO=25.00            ;vendita semplice a reparto 4

VEND REP=2,PREZZO=25.00,STORNO     ;storno della vendita semplice a reparto 2

CORT R1='PROVA MESSAGGIO 1',R2='PROVA MESSAGGIO 2' ;messaggio di cortesia

SUBT                              ;subtotale
CHIUS T=1,IMP=50.00               ;chiusura mista : 50.00 in contanti
CHIUS T=2                         ;                 e il resto a credito
```

12.4.3   Fiscal Receipt with Fiscal Tax Code (Codice Fiscale) on Printer 2

```
PRMSG LINE='PRINTER|2'
VEND REP=1, DES='HAMBURGER INCROCIO C', PREZZO=   2.08
VEND REP=1, DES='HAMBURGER INCROCIO C', PREZZO=   2.02
VEND REP=1, DES='HAMBURGER INCROCIO C', PREZZO=   2.02
VEND REP=1, DES='HAMBURGER INCROCIO C', PREZZO=   2.03
INP TERM=61
INP ALFA='ABCDEF12G34H567S',TERM=49
PRMSG LINE='-------------'
PERCA ALIQ=   5.00, SUBTOT
CHIUS t=0, imp=  10.05
```

12.4.4   Fiscal Receipt with QR Code

```
VEND    REP=1,  PREZZO=      0.00, QTY=  2.000, DES='Persone    '
VEND    REP=1,  PREZZO=      2.00, QTY=  2.000, DES='Servizio   '
VEND    REP=1,  PREZZO=      3.30, QTY=   1.00, DES='bruschetta '
VEND    REP=1,  PREZZO=      5.00, QTY=   1.00, DES='arancini e f'
PERCA   ALIQ=  90.00 , SUBTOT
NOFIS RIGA=''
PRMSG LINE='Operatore : MICHELE - MICHELE #17              '
PRMSG LINE='Tavolo : 1                                     '
PRMSG LINE='Conto : 48356                                  '
ALLEG ON
CHIU    t=1,  imp=      1.23
ALLEG RIGA='http://www.mysite.com/myApp/', BARCODE=158, SCALA=4, ALLINEAMENTO=1
ALLEG FINE
```

12.4.5   Commercial Refund Document

```
INP TERM=180
INP TERM=145
INP ALFA='99MEY000203', TERM=145
INP ALFA=00210042, TERM=145
INP NUM=060220, TERM=145
CLEAR
VEND REP=1,      PREZZO=10.00, DES='TEST RESO'
CHIUS
```

12.4.6   Commercial Void Document

```
INP TERM=188
INP TERM=145
INP ALFA='99MEY000203', TERM=145
INP ALFA=00210042, TERM=145
INP NUM=060220, TERM=145
CLEAR
VEND REP=1, PRE=69.99, DES='TEST VOID'
CHIUS
```

12.4.7   Non-fiscal Receipt with Barcode

```
NOFIS APRI
NOFIS DES='TEST'
NOFIS DES='123456789', BARCODE
NOFIS CHIUDI
```

12.4.8   Invoice

```
FATTURA NOME1=NOME_ONE, NOME2=NOME_TWO, NOME3=NOME_THREE, NOME4=NOME_FOUR,
NOME5=NOME_FIVE
VEND REP=1, DES='SALE 1', PREZZO=   2.08
PRMSG LINE='MESSAGGIO CORTESIA 1'
PRMSG LINE='MESSAGGIO CORTESIA 2'
PERCA ALIQ=   5.00, SUBTOT
CHIUS t=0, imp=  10.05
```

12.4.9   Display Message

```
VIS CLI1='1234567890          1234567890           '
```

12.4.10   Z Report without X-01 Financial Report

```
report num=1, modo=1, file='Z_Report.txt'
```

or

```
AZZGIO num=1, modo=1, file='Z_Report.txt'
```

12.4.11   Z Report with X-01 Financial Report

```
report num=1, modo=2, file='X and Z Reports.txt'
```

or

```
AZZGIO num=1, modo=2, file='X and Z Reports.txt'
```

12.4.12   Concatenation

EpsonFpMate can process multiple printouts in a single file. For example:

```
VEND    REP=1 , PREZZO=     8.80, QTY=   1.00, DES='PANINO'
CHIU   t=1, imp=    10.00
NOFIS APRI
NOFIS DES='TEST'
NOFIS DES='123456789', BARCODE
NOFIS CHIUDI
```