

# OFSC ドライバーサンプルスタイルシート リファレンスマニュアル

---

## 概要

スタイルシートの種類やフォルダ構成について説明します。

## スタイルシートの編集

スタイルシートの変更方法を説明しています。

## ご注意

- 本書の内容の一部または全部を無断で転載、複写、複製、改ざんすることは固くお断りします。
- 本書の内容については、予告なしに変更することがあります。最新の情報はお問い合わせください。
- 本書の内容については、万全を期して作成いたしましたが、万一ご不審な点や誤り、記載もれなど、お気づきの点がありましたらご連絡ください。
- 運用した結果の影響については、上項にかかわらず責任を負いかねますのでご了承ください。
- 本製品がお客様により不適切に使用されたり、本書の内容に従わずに取り扱われたり、またはエプソンおよびエプソン指定の者以外の第三者により修理・変更されたことなどに起因して生じた損害などにつきましては、責任を負いかねますのでご了承ください。
- エプソン純正品およびエプソン品質認定品以外のオプションまたは消耗品を装着してトラブルが発生した場合には、責任を負いかねますのでご了承ください。

## 商標について

EPSON® および ESC/POS® はセイコーエプソン株式会社の登録商標です。

MS-DOS®, Microsoft®, Win32®, Windows®, Windows Server®, Windows Vista®, Visual Studio®, Visual Basic®, Visual C++®, Visual C#® は米国 Microsoft Corporation の米国およびその他の国における商標または登録商標です。  
その他、記載されている会社名、製品名は、各社の商標、または登録商標です。

© セイコーエプソン株式会社 2009-2011

---

# 安全のために

## 記号の意味

本書では以下の記号が使われています。それぞれの記号の意味をよく理解してから製品を取り扱ってください。

### 注意

ご使用上、必ずお守りいただきたいことを記載しています。この表示を無視して誤った取り扱いをすると、製品の故障や動作不良の原因になる可能性があります。

### 参考

補足説明や知っておいていただきたいことを記載しています。

## 使用制限

本製品を航空機・列車・船舶・自動車などの運行に直接関わる装置・防災防犯装置・各種安全装置など機能・精度などにおいて高い信頼性・安全性が必要とされる用途に使用される場合は、これらのシステム全体の信頼性および安全維持のためにフェールセーフ設計や冗長設計の措置を講じるなど、システム全体の安全設計にご配慮いただいた上で当社製品をご使用いただくようお願いいたします。

本製品は、航空宇宙機器、幹線通信機器、原子力制御機器、医療機器など、きわめて高い信頼性・安全性が必要とされる用途への使用を意図しておりませんので、これらの用途には本製品の適合性をお客様において十分ご確認のうえ、ご判断ください。

## 本書について

### 本書の目的

本書は、スタイルシートのサンプルプログラムを使って、アプリケーションを構築する人向けに、情報を提供することを目的としています。

### 本書の構成

本書は次のように構成されています。

第1章 [概要](#)

第2章 [スタイルシートの編集](#)

# 目次

■ 安全のために.....	3
記号の意味 .....	3
■ 使用制限 .....	3
■ 本書について .....	3
本書の目的 .....	3
本書の構成 .....	3
■ 目次 .....	4

---

## 概要.....5

■ はじめに .....	5
■ スタイルシート の種類.....	5
印刷結果 .....	6
■ フォルダ構成 .....	7
■ サンプルプログラムでスタイルシート を 使うには .....	9
■ Java 仮想マシン (JVM) の ヒープメモリーの拡張.....	9
Java Windows の場合 .....	9
Java Linux の場合 .....	9

---

## スタイルシート の編集..... 11

■ スタイルシート の編集方法.....	11
店舗ロゴの変更 .....	11
店舗情報の変更 .....	14
売り上げ商品情報のレイアウト の変更 .....	16
罫線の変更 .....	19
白伝票の削除 (TM-T90KP 専用) .....	24
ブザー音の変更 (TM-T90KP 専用) .....	25
バーコードの変更 .....	26
■ UPOS コマンド .....	28

# 概要

## はじめに

OFSC のサンプルスタイルシートは、OFSC サンプルプログラム上で動作することを前提に作成されています。

実際のアプリケーションでサンプルスタイルシートを使用する場合、アプリケーションが OFSC ドライバに渡す XML データの POSLog データによってスタイルシートを変更してください。

印刷レイアウトを変更する場合、スタイルシートを編集する必要があります。

サンプルプログラムの動作環境や、システム構築方法に関しては「OFSC ドライバー ユーザーズマニュアル」を参照してください。

## スタイルシートの種類

サンプルスタイルシートとして下記の物が用意されています。

### ■ 印刷スタイル用スタイルシート

印刷スタイル用スタイルシートはプリンターごと用意されています。  
(7 ページ「フォルダー構成」を参照してください。)

ファイル名	用途
Cooking.xml	調理指示単品伝票印刷
Serving.xml	調理指示一枚伝票印刷
Guest.xml	ゲスト伝票印刷
Receipt.xml	レシート印刷
Drawer.xml	ドロアーの開口
Re_Cooking.xml	再印刷用、調理指示単品伝票 (再印刷とは、1 度印刷が失敗した場合、OFSC ドライバーが迂回処理をして別のプリンターに印刷することです。詳細は” OFSC ドライバーユーザーズマニュアル” を参照してください。)
Re_Serving.xml	再印刷用、調理指示一枚伝票

### ■ Response スタイルシート

Response スタイルシートでは、エラーのハンドリングをしています。  
(エラーのハンドリングに関しては、「OFSC ドライバー ユーザーズマニュアル」を参照してください。)

ファイル名	用途
PrinterResponse.xml	PosPrinter レスポンス解析
DrawerResponse.xml	Cash Drawer レスポンス解析

### ■ CheckHealth 用スタイルシート

CheckHealth 用スタイルシートでは、各デバイスが正常に動作するか確認できます。

ファイル名	用途
Check_Kitchen.xml	KitchenPrinter のデバイスチェック
Check_Floor.xml	FloorPrinter のデバイスチェック
Check_Receipt.xml	ReceiptPrinter のデバイスチェック
Check_Drawer.xml	Cash Drawer のデバイスチェック

## 印刷結果

各レシートの印刷サンプルです。

調理指示単品伝票				調理指示一枚伝票									
<div style="display: flex; justify-content: space-between;"> <div> <b>1</b>  <b>調理単品</b>            0006         </div> <div> <b>アメリカンコーヒー</b>  <b>【新 規】</b>            1 名 山田         </div> <div> <b>A12卓</b>            11:30         </div> </div> <hr/> <div style="display: flex; justify-content: space-between;"> <div> <b>1</b>  <b>調理単品</b>            0006         </div> <div> <b>オレンジジュース</b>  <b>【新 規】</b>            1 名 山田         </div> <div> <b>A12卓</b>            11:30         </div> </div>				<div style="display: flex; justify-content: space-between;"> <div> <b>調理一枚</b>            人 数            時 刻         </div> <div> <b>【新 規】</b>            1            11:30         </div> <div> <b>テーブル</b>  <b>伝 票</b>  <b>担 当</b>            HT番号         </div> <div> <b>A12</b>  <b>0007</b>  <b>山田</b>  <b>HT10</b> </div> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="width: 5%;">数量</th> <th style="width: 95%;">品 名</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>アメリカンコーヒー</td> </tr> <tr> <td style="text-align: center;">1</td> <td>オレンジジュース</td> </tr> </tbody> </table> <div style="text-align: center; margin-top: 20px;"> </div>				数量	品 名	1	アメリカンコーヒー	1	オレンジジュース
数量	品 名												
1	アメリカンコーヒー												
1	オレンジジュース												

ゲスト 伝票				レシート																																	
<div style="display: flex; justify-content: space-between;"> <div> <b>【新 規】</b>            2009-02-25  <i>Sample Shop</i> </div> <div> <b>テーブル</b>  <b>伝 票</b>  <b>人 数</b>  <b>時 刻</b>  <b>担 当</b> </div> <div> <b>0008</b>  <b>A12</b>            1            11:30            山田         </div> </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <thead> <tr> <th style="width: 30%;">品 名</th> <th style="width: 10%;">数量</th> <th style="width: 60%;">金 額</th> </tr> </thead> <tbody> <tr> <td>01002 アメリカンコーヒー</td> <td style="text-align: center;">1</td> <td style="text-align: right;">200</td> </tr> <tr> <td>02004 オレンジジュース</td> <td style="text-align: center;">1</td> <td style="text-align: right;">300</td> </tr> <tr> <td>税 抜</td> <td style="text-align: center;">476</td> <td style="text-align: right;">消費税 24</td> </tr> <tr> <td>合計</td> <td></td> <td style="text-align: right;">500</td> </tr> </tbody> </table> <div style="text-align: center; margin-top: 10px;">           毎度ありがとうございます            またのご来店お待ちしております  </div>				品 名	数量	金 額	01002 アメリカンコーヒー	1	200	02004 オレンジジュース	1	300	税 抜	476	消費税 24	合計		500	<div style="text-align: center;"> <i>Sample Shop</i>            長野県松本店            □□□店            ○○県△△△市□□□ ×-××-××            TEL: xxxx-xx-xxxx         </div> <div style="text-align: center; margin-top: 20px;">           毎度ありがとうございます            またのご来店お待ちしております         </div> <div style="display: flex; justify-content: space-between; margin-top: 20px;"> <div>長野県松本店</div> <div>2009-02-25</div> <div>13:00</div> </div> <table style="width: 100%; margin-top: 10px;"> <tbody> <tr> <td style="width: 70%;">01002 アメリカンコーヒー</td> <td style="width: 10%; text-align: center;">1</td> <td style="width: 20%; text-align: right;">¥200</td> </tr> <tr> <td>02004 オレンジジュース</td> <td style="text-align: center;">1</td> <td style="text-align: right;">¥300</td> </tr> <tr> <td>税 抜</td> <td></td> <td style="text-align: right;">¥476</td> </tr> <tr> <td>消費税</td> <td></td> <td style="text-align: right;">24</td> </tr> <tr> <td>合 計</td> <td></td> <td style="text-align: right;">¥500</td> </tr> </tbody> </table> <div style="display: flex; justify-content: flex-end; margin-top: 10px;"> <div>2 点</div> <div>00009</div> <div>山田</div> </div>				01002 アメリカンコーヒー	1	¥200	02004 オレンジジュース	1	¥300	税 抜		¥476	消費税		24	合 計		¥500
品 名	数量	金 額																																			
01002 アメリカンコーヒー	1	200																																			
02004 オレンジジュース	1	300																																			
税 抜	476	消費税 24																																			
合計		500																																			
01002 アメリカンコーヒー	1	¥200																																			
02004 オレンジジュース	1	¥300																																			
税 抜		¥476																																			
消費税		24																																			
合 計		¥500																																			

## フォルダー構成

サンプルスタイルシートのフォルダー構成は以下のとおりです。

青字で書かれているスタイルシートは、サンプルプログラムの初期設定で実行する（[epsonsample]-[stylesheet] フォルダーに入っている）スタイルシートです。

黒字のファイルは、サンプルプログラムで初期設定以外のスタイルシートを使用する場合に使用します。

赤字のファイルは、サンプルプログラムのファイルです。

（9 ページ「サンプルプログラムでスタイルシートを使うには」を参照してください。）

```

epsonsample..... : サンプルプログラム
├─ epsonsample.html
├─ driverControl.js (.NET 環境の場合、driverControl.vbs です。)
├─ formControl.js (.NET 環境の場合、formControl.vbs です。)
├─ epsonsample.css
├─ logo1.bmp ..... : ロゴデータ
├─ logo2.bmp ..... : ロゴデータ
├─ image
├─ stylesheet
└─ SampleStyleSheets..... : サンプルスタイルシート
    ├─ CheckHealth ..... : デバイスチェック
    │   └─ Check_Drawer.xml
    │   └─ Check_Floor.xml
    │   └─ Check_Kitchen.xml
    │   └─ Check_Receipt.xml
    └─ Cooking..... : 調理指示単品伝票
        ├─ TM-T70
        │   └─ Cooking.xml
        │   └─ Re_Cooking.xml
        ├─ TM-T88V
        │   └─ Cooking.xml
        │   └─ Re_Cooking.xml
        └─ TM-T90
            ├─ Horizontal
            │   └─ Cooking.xml
            │   └─ Re_Cooking.xml
            └─ Vertical
                └─ Cooking.xml
                └─ Re_Cooking.xml
    └─ Guest..... : ゲスト伝票
        ├─ TM-T70
        │   └─ Guest.xml
        ├─ TM-T88V
        │   └─ Guest.xml
        └─ TM-T90
            ├─ Horizontal
            │   └─ Guest.xml
            └─ Vertical
                └─ Guest.xml
        └─ TM-T90_RuledLine
            ├─ Horizontal
            │   └─ Guest.xml
            └─ Vertical
                └─ Guest.xml
    └─ Receipt..... : レシート
        ├─ TM-T70
        │   └─ Receipt.xml
        ├─ TM-T88V
        │   └─ Receipt.xml
        └─ TM-T90
            ├─ Horizontal
            │   └─ Receipt.xml
            └─ Vertical
                └─ Receipt.xml
    └─ Response..... : レスポンス解析
        └─ PrinterResponse.xml
    └─ Serving..... : 調理指示一枚伝票

```

epsonsample.....	: サンプルプログラム
SampleStyleSheets.....	: サンプルスタイルシート
CheckHealth.....	: デバイスチェック
Cooking.....	: 調理指示単品伝票
Guest.....	: ゲスト伝票
Receipt.....	: レシート
Response.....	: レスポンス解析
Serving.....	: 調理指示一枚伝票
TM-T70	
Serving.xml	
Re_Serving.xml	
TM-T88V	
Serving.xml	
Re_Serving.xml	
TM-T90	
Horizontal	
Serving.xml	
Re_Serving.xml	
Vertical	
Serving.xml	
Re_Serving.xml	
TM-T90_RuledLine	
Horizontal	
Serving.xml	
Re_Serving.xml	
Vertical	
Serving.xml	
Re_Serving.xml	



## サンプルプログラムでスタイルシートを使うには

サンプルプログラムには、初期設定で [epsonsample]-[stylesheet] に TM-T90KP 横置きプリンターのスタイルシートが入っています。(7 ページ「フォルダー構成」を参照してください。)

ただし、デバイスを変えて使用する場合にはスタイルシートを差し替える必要があります。

例：

初期設定では、レシートは TM-T88V 用のスタイルシート (Receipt.xml) で印刷します。

レシートを、TM-T90KP 横置きプリンターで印刷したい場合、TM-T90KP 横置きプリンター用のスタイルシート (Receipt.xml) を [epsonsample]-[stylesheet] に既にある、TM-T88V 用のスタイルシート (Receipt.xml) と差し替えてください。

## Java 仮想マシン (JVM) のヒープメモリーの拡張

送信する XML データが 2 MB を超える場合は、JVM のヒープメモリーを拡張する必要があります。

設定方法は以下のとおりです。(128MB のデータを送信できるようにする場合)

### Java Windows の場合

Tomcat モニターの設定で変更します。

- 1 Tomcat のタスクアイコンを右クリックし [Configure..] を選択します。
- 2 [Java] タブを選択します。
- 3 [Initial memory pool] に「128」を入力します。
- 4 [Maximum memory pool] に「256」を入力します。
- 5 Tomcat を再起動します。

### Java Linux の場合

CATARINA\_OPTS 環境変数の設定で変更します。

/etc/profile を以下のように編集します。編集後、コンピューターを再起動します。

```
>su-
JRE_HOME=/usr/java/jre1.6.0_11
CATALINA_HOME=/usr/local/tomcat/apache-tomcat-6.0.16
CATALINA_OPTS="-Xms128m -Xmx256m"
PATH=$JRE_HOME/bin:$CATALINA_HOME/bin:$PATH
export JRE_HOME CATALINA_HOME PATH CATALINA_OPTS
```



# スタイルシートの編集

スタイルシートを編集すると、各レシートの印刷レイアウトやブザーの音色などを変更することができます。

## スタイルシートの編集方法

- 店舗ロゴの変更（11 ページ「店舗ロゴの変更」）
- 店舗情報の変更（14 ページ「店舗情報の変更」）
- 売り上げ商品情報のレイアウトの変更（16 ページ「売り上げ商品情報のレイアウトの変更」）
- 罫線の変更（19 ページ「罫線の変更」）
- 白伝票の削除（24 ページ「白伝票の削除（TM-T90KP 専用）」）
- ブザー音の変更（25 ページ「ブザー音の変更（TM-T90KP 専用）」）
- バーコードの変更（26 ページ「バーコードの変更」）

### 参考

詳細なコメントがスタイルシートのソースに記載されています。  
本マニュアルと、合わせてご覧ください。

2

## 店舗ロゴの変更

レシート、およびゲスト伝票に印刷する店舗ロゴを変更します。

- 該当スタイルシート
  - Receipt.xml（レシート）
  - Guest.xml（ゲスト伝票）

Receipt.xml（レシート）



長野県松本店

□□□店

〇〇県△△市□□□ ×-××-××

TEL: XXXX-XX-XXXX

毎度ありがとうございます

またのご来店お待ちしております

長野県松本店    2009-02-25    13:00

01002 アメリカンコーヒー	1	¥200
02004 オレンジジュース	1	¥300
税 抜		¥476
消費税		24
合 計		¥500

2 点    00009    山田

Guest.xml（ゲスト伝票）

【新 規】

2009-02-25



テーブル  
伝 票  
人 数  
時 刻  
担 当

0008  
A12  
1  
11:30  
山田

品 名	数 量	金 額
01002 アメリカンコーヒー	1	200
02004 オレンジジュース	1	300
税 抜	476	消費税
合 計		24
		500

毎度ありがとうございます

またのご来店お待ちしております



青い部分を変更します。

## Receipt.xml

青字部分で印刷するイメージファイルと、印刷位置を指定します。

使用するイメージファイル名に書き換えることで、店舗ロゴを変更できます。このファイルは、プリンターに登録するのではなく、印刷の都度ファイルを読み込んで印刷します。

### Java 環境の場合

```
<!-- 店舗ロゴを印刷 -->
<PrintBitmap>
  <Station>PTR_S_RECEIPT</Station>
  <FileName>http://localhost:8080/epsonsample/logo1.bmp</FileName>
  <Width>PTR_BM_ASIS</Width>
  <Alignment>PTR_BM_CENTER</Alignment>
</PrintBitmap>
```

### .NET 環境の場合

```
<!-- 店舗ロゴを印刷 -->
<PrintBitmap>
  <Station>PTR_S_RECEIPT</Station>
  <FileName>http://localhost/epsonsample/logo1.bmp</FileName>
  <Width>PTR_BM_ASIS</Width>
  <Alignment>PTR_BM_CENTER</Alignment>
</PrintBitmap>
```

#### ■ 使用できるイメージファイルのフォーマット

- bmp
- jpeg
- gif

#### 注意

PrintBitmap で画像ファイルを URL で指定する場合、ファイル名の拡張子は “.bmp”（小文字）にしてください。

#### ■ サンプルスタイルシートで印刷できる店舗ロゴのイメージサイズ

プリンター	横	縦
TM-T90KP	576 ピクセルまで	制限無し
TM-T88V	512 ピクセルまで	制限無し
TM-T70	576 ピクセルまで	制限無し

#### ■ Alignment は、ロゴを印刷する位置を指定します。

コマンド	意味
PTR_BM_LEFT	左寄せ
PTR_BM_CENTER	センタリング
PTR_BM_RIGHT	右寄せ

## Guest.xsl

青字部分で印刷するイメージファイルと、印刷位置を指定します。

使用するイメージファイル名に書き換えることで、店舗ロゴを変更できます。このファイルは、プリンターに登録するのではなく、印刷の都度ファイルを読み込んで印刷します。

### Java 環境の場合

```
<!-- 店舗ロゴを印刷 -->
<PrintBitmap>
  <Station>PTR_S_RECEIPT</Station>
  <FileName>http://localhost:8080/epsonsample/logo1.bmp</FileName>
  <Width>PTR_BM_ASIS</Width>
  <Alignment>PTR_BM_CENTER</Alignment>
</PrintBitmap>
```

### .NET 環境の場合

```
<!-- 店舗ロゴを印刷 -->
<PrintBitmap>
  <Station>PTR_S_RECEIPT</Station>
  <FileName>http://localhost/epsonsample/logo1.bmp</FileName>
  <Width>PTR_BM_ASIS</Width>
  <Alignment>PTR_BM_CENTER</Alignment>
</PrintBitmap>
```

#### ■ 使用できるイメージファイルのフォーマット

- bmp
- jpeg
- gif

#### 注意

PrintBitmap で画像ファイルを URL で指定する場合、ファイル名の拡張子は “.bmp”（小文字）にしてください。

#### ■ サンプルスタイルシートで印刷できる店舗ロゴのイメージサイズ

プリンター	横	縦
TM-T90KP	295 ピクセルまで	84 ピクセルまで
TM-T88V	262 ピクセルまで	74 ピクセルまで
TM-T70	295 ピクセルまで	84 ピクセルまで

#### ■ Alignment は、ロゴを印刷する位置を指定します。

コマンド	意味
PTR_BM_LEFT	左寄せ
PTR_BM_CENTER	センタリング
PTR_BM_RIGHT	右寄せ

## 店舗情報の変更

レシートに印刷する店舗情報を変更します。

### ■ 該当スタイルシート

- Receipt.xsl (レシート)

Receipt.xsl (レシート)

<i>Sample Shop</i>			
長野県松本店			
□□□店			
〇〇県△△△市□□□ ×-××-×××			
TEL: xxxx-xx-xxxx			
毎度ありがとうございます			
またのご来店お待ちしております			
長野県松本店	2009-02-25	13:00	
01002 アメリカンコーヒー	1	¥200	
02004 オレンジジュース	1	¥300	
税 抜		¥476	
消費税		24	
合 計		¥500	
2 点	00009	山田	

青い部分を変更します。

## Receipt.xml

青字部分が店舗情報です。印刷したい店舗情報に変更します。

```
<!-- 店舗情報を印刷 -->
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>¥x1b|cA □□□店 ¥n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>¥x1b|cA ○○県△△市□□□ X - X X - X X ¥n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>¥x1b|cATEL: xxxx-xx-xxxx ¥n¥n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>¥x1b|cA 毎度ありがとうございます ¥n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>¥n¥x1b|cA またのご来店をお待ちしております ¥n</Data>
</PrintNormal>
```

- “¥n” は改行の指定方法です。
- “¥x1b|cA” は UPOS のエスケープシーケンスコマンドで、中央寄せして印刷します。  
UPOS のエスケープシーケンスコマンドについては、[28 ページ「UPOS コマンド」](#)を参照してください。
- 1 行に印字できる文字数  
全角文字は 1 文字で 2 桁使用します。横倍角、4 倍角指定の場合、1 文字で 4 桁使用します。

プリンター	半角	漢字
TM-T90KP	48 桁	24 桁
TM-T88V	42 桁	21 桁
TM-T70	48 桁	24 桁

- <PrintNormal>・・・</PrintNormal> で指定する印刷データは 1 行です。  
印刷行数を増やす場合は、<PrintNormal>・・・</PrintNormal> をコピー＆ペーストし、<Data>・・・</Data> に追加する文字列を記述します。
- TM-T90KP の縦置き用のスタイルシートは 180 度回転が指定されています。そのため、印刷データは下から指定してください。

## 売り上げ商品情報のレイアウトの変更

各レシートに印刷する売り上げ商品情報のレイアウトを変更します。

売り上げ商品情報は全てのレシートに記述されています。使用機種、使用目的によって全て記述方法が異なります。そのため、主なものについて説明します。

### ■ 該当スタイルシート

- Cooking.xml（調理指示単品伝票）
- Serving.xml（調理指示一枚伝票）
- Guest.xml（ゲスト伝票）
- Receipt.xml（レシート）
- Re\_Cooking.xml（再印刷用、調理指示単品伝票）
- Re\_Serving.xml（再印刷用、調理指示一枚伝票）

#### Receipt.xml（レシート）

**Sample Shop**  
長野県松本店

□□□店  
〇〇県△△市□□□ ×-××-××  
TEL: XXXX-XX-XXXX

毎度ありがとうございます  
またのご来店お待ちしております

長野県松本店    2009-02-25    13:00

01002 アメリカンコーヒー	1	¥200
02004 オレンジジュース	1	¥300

税 抜	¥476
消費税	24
合 計	¥500

2 点    00009    山田

#### Guest.xml（ゲスト伝票）

【新 規】  
2009-02-25

**Sample Shop**

テーブル 0008  
伝 票 A12  
人 数 1  
時 刻 11:30  
担 当 山田

品 名	数 量	金 額
01002 アメリカンコーヒー	1	200
02004 オレンジジュース	1	300
税 抜	476	消費税
合 計		500

毎度ありがとうございます  
またのご来店お待ちしております



青い部分を変更します。



## Receipt.xml (Receipt¥TM-T88V¥Receipt.xml)

青字部分で印刷の指定をしています。

緑字部分にて商品番号、商品名、数量、金額の各文字列の長さをそろえています。

```
<xsl:template match="ixr:Sale">
  <!-- 商品 ID を取得 (5 桁に成形) -->
  <xsl:variable name="POSItemID" select="format-number(//ixr:POSItemID, '00000')"/>
  <!-- 商品名を取得 (全角 10 桁に成形) -->
  <xsl:variable name="Description" select="concat(ixr:Description, substring(' ',
1, 10 - string-length(ixr:Description)))"/>
  <!-- 商品数量を取得 (2 桁に成形) -->
  <xsl:variable name="Quantity" select="concat(substring(' ', 1, 2 - string-length(ixr:Quantity)),
ixr:Quantity)"/>
  <!-- 金額を取得 (8 桁に成形) -->
  <xsl:variable name="Temp" select="format-number(ixr:ExtendedAmount, '¥#,##0')"/>
  <xsl:variable name="ExtendedAmount" select="concat(substring(' ', 1, 8 - string-
length($Temp)), $Temp)"/>

  <!-- 商品番号、商品名、数量、金額を印刷 -->
  <PrintNormal>
    <Station>PTR_S_RECEIPT</Station>
    <Data><xsl:value-of select="$POSItemID"/> <xsl:value-of select="$Description"/> ¥t<xsl:value-of
select="$Quantity"/> ¥x1b|rA<xsl:value-of select="$ExtendedAmount"/> ¥n</Data>
  </PrintNormal>
</xsl:template>
```

- 金額の桁数は、桁区切り“,”と“¥”マークを含めた桁数で成形します。緑字の <!-- 金額を取得 (8 桁に成形) --> 部分にて設定します。
- 数量の印刷位置合わせはタブ (¥t) を使用しています。  
タブは8文字送ります。  
タブは、前の文字列の長さ（ここでは商品番号+商品名の長さ）が可変の場合、印刷位置が揃わないことがあります。  
実際に印字して問題が無いか確認してください。
- 金額は UPOS のエスケープシーケンスコマンドにて右寄せ (¥x1b|rA) で印刷しています。  
これにより、前の文字列の長さに関わらず位置をそろえることが出来ます。
- 売り上げ商品情報（商品番号、商品名、数量、金額）が 1 行に入るように設定してください。  
一行に入る桁数はプリンターによって異なります。  
全角文字は 1 文字で 2 桁使用します。横倍角、4 倍角指定の場合、1 文字で 4 桁使用します。

プリンター	文字数
TM-T90KP	48 桁
TM-T88V	42 桁
TM-T70	48 桁

## 印刷レイアウト

Receipt¥TM-T88V¥Receipt.xml の売り上げ商品情報は、以下のように印刷します。

商品 ID (5 桁)      商品名 (全角 10 文字 20 桁)      タブ      商品数量 (2 桁)      金額 (8 桁)

01002 アメリカンコーヒー      1      ¥300

## Guest.xml (Guest¥TM-T90\_RuledLine¥Vertical¥Guest.xml)

青字部分で印刷の指定をしています。

緑字部分にて商品番号、商品名、数量、金額の各文字列の長さをそろえています。

```
<xsl:template match="ixr:Sale">
  <!-- 商品 ID を取得 (5 桁に成形) -->
  <xsl:variable name="POSItemID" select="format-number(./ixr:POSItemID, '00000')"/>
  <!-- 商品名を取得 (全角 10 桁に成形) -->
  <xsl:variable name="Description" select="concat(ixr:Description, substring(' ', 1,
10 - string-length(ixr:Description)))/>
  <!-- 商品数量を取得 (2 桁に成形) -->
  <xsl:variable name="Quantity" select="concat(substring(' ', 1, 2 - string-length(ixr:Quantity)),
ixr:Quantity)"/>
  <!-- 金額を取得 (6 桁に成形) -->
  <xsl:variable name="Temp" select="format-number(ixr:ExtendedAmount, '#,###')"/>
  <xsl:variable name="ExtendedAmount" select="concat(substring(' ', 1, 6 - string-length($Temp)),
$Temp)"/>

  <!-- 商品番号、商品名、数量、金額を印刷 -->
  <PrintNormal>
    <Station>PTR_S_RECEIPT</Station>
    <Data>¥x1b|3C  ¥x1b|N <xsl:value-of select="$POSItemID"/>¥x1b|3C<xsl:value-of
select="$Description"/> ¥x1b|4C<xsl:value-of select="$Quantity"/>¥x1b|3C  ¥x1b|rA<xsl:value-
of select="$ExtendedAmount"/> ¥n</Data>
  </PrintNormal>
</xsl:template>
```

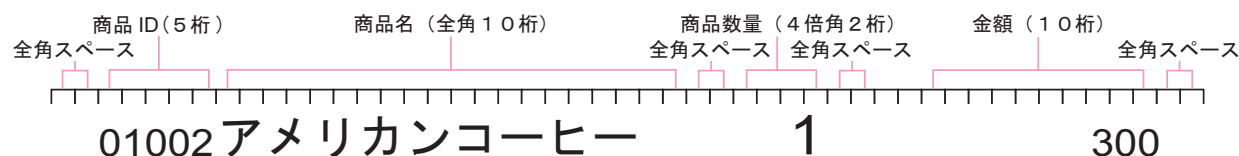
- 金額の桁数は、桁区切り“,”と“¥”マークを含めた桁数で成形します。緑字の<!-- 金額を取得 (6桁に成形) -->部分にて設定します。
- 数量の位置合わせは、商品番号と商品名の文字列を固定にすることで行っています。
- 金額はUPOSのエスケープシーケンスコマンドにて右寄せ(¥x1b|rA)で印刷しています。  
これにより、前の文字列の長さに関わらず位置をそろえることが出来ます。
- 売り上げ商品情報(商品番号、商品名、数量、金額)が1行に入るように設定してください。  
一行に入る桁数はプリンターによって異なります。  
全角文字は1文字で2桁使用します。横倍角、4倍角指定の場合、1文字で4桁使用します。

プリンター	文字数
TM-T90KP	48 桁
TM-T88V	42 桁
TM-T70	48 桁

### 印刷レイアウト

Guest¥TM-T90\_RuledLine¥Vertical¥Guest.xmlの売り上げ商品情報は、以下のように印刷します。

(ゲスト用伝票には罫線が引かれるため、罫線と文字が重ならないように罫線部分に全角スペースを入れています。罫線については、19ページ「罫線の変更」を参照してください。)



## 罫線の変更

罫線の印刷レイアウトを変更します。

### ■ 罫線の印刷方法

罫線の印刷方法には、以下の方法があります。どちらを使用しても、印刷結果に大きな差はありません。

- 罫線のフォントを使用
- プリンターの罫線描画機能を使用

### ■ 該当スタイルシート

- Serving.xml（調理指示一枚伝票）
- Guest.xml（ゲスト伝票）
- Re\_Serving.xml（調理指示一枚伝票）

Serving.xml（調理指示一枚伝票）

調理一枚	【新 規】	テーブル	A12
人 数	1	伝 票	0007
時 刻	11:30	担 当	山田
		HT番号	HT10

	数量	品 名
<input type="checkbox"/>	1	アメリカンコーヒー
<input type="checkbox"/>	1	オレンジジュース



青い部分を変更します。

## 罫線のフォントを使用(Serving¥TM-T90¥Horizontal¥Serving.xsl)

罫線のフォント（「┌」、「─」、「│」、「＋」、「└」など）を使用して、罫線を印刷します。

青字部分を変更し、罫線を編集します。

```
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> ┌───────────────────────────────────┐ \n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> │ │ 数量 │                品                名                │ \n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> ┌───┬───┬──────────────────────────┐ \n</Data>
</PrintNormal>

<!-- 売り上げ商品情報を印刷 -->
<xsl:apply-templates select="//ixr:Sale"/>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> \x1b|3C │ │ │ │ │ \x1b|N\n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> \x1b|3C │ │ │ │ │ \x1b|N\n</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data> ┌───────────────────────────────────┐ \n</Data>
</PrintNormal>
```

- TM-T90KP 用サンプルスタイルシートの場合、1 行が 48 桁になるようフォントを並べます。  
プリンターの機種ごと、1 行あたりに並べるフォントの数は異なります。  
全角文字は 1 文字で 2 桁使用します。横倍角、4 倍角指定の場合、1 文字で 4 桁使用します。

プリンター	文字数
TM-T90KP	48 桁
TM-T88V	42 桁
TM-T70	48 桁

## プリンターの罫線描画機能を使用 (Serving¥TM-T90\_RuledLine¥Horizontal¥Serving.xml)

プリンターの罫線描画機能を使用して、罫線を印刷します。

罫線は、横方向と縦方向のみです。斜め罫線は描画できません。

青字部分で線の種類と、罫線を指定をしています。

```
<!-- 罫線の座標指定を行いやすくするため、単位を 1/100mm に設定 -->
<SetProperty>
  <MapMode>PTR_MM_METRIC</MapMode>
</SetProperty>
<!-- 横罫線の描画
  Data の 2 は太さ 2、Object の 0,7200 は、左端を基点に座標 0 から 72mm の長さで罫線を描画 -->
<DirectIO>
  <Command>1000</Command>
  <Data>2</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO">0,7200</String>
  </Object>
</DirectIO>
<!-- 縦罫線の描画
  Data の 2 は太さ 2、Object の 0,7194 は、左端を基点に 0mm(0 ドット目) と 71.94mm(575 ドット目)
  の位置で縦罫線を描画 -->
<DirectIO>
  <Command>1010</Command>
  <Data>2</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO">0,7194</String>
  </Object>
</DirectIO>
<!-- 縦罫線の描画
  Data の 1 は太さ 1、Object の 700,1600 は、左端を基点に 7mm と 16mm の位置で縦罫線を描画 -->
<DirectIO>
  <Command>1010</Command>
  <Data>1</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO">700,1600</String>
  </Object>
</DirectIO>
<!-- 横罫線と文字がつかないように 1.5mm の改行 -->
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>\x1b|150uF</Data>
</PrintNormal>
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>      数量          品          名          \n</Data>
</PrintNormal>
      :
      :
<!-- 縦罫線の描画の中止。Data 0 は全ての罫線を指定する意味 -->
<DirectIO>
  <Command>1010</Command>
  <Data>0</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO"></String>
  </Object>
</DirectIO>
```

- 罫線の始点と長さの値の単位を、MapMode プロパティ（ドライバが使用する単位を指定するプロパティ）で設定します。

サンプルスタイルシートでは、MapMode プロパティを PTR\_MM\_METRIC に設定しています。

PTR\_MM\_METRIC に設定すると、罫線の始点と長さの値の単位は 1/100mm になります。

（MapMode プロパティを PTR\_MM\_DOTS にすると、横罫線の始点と長さの値の単位はドットになります。）

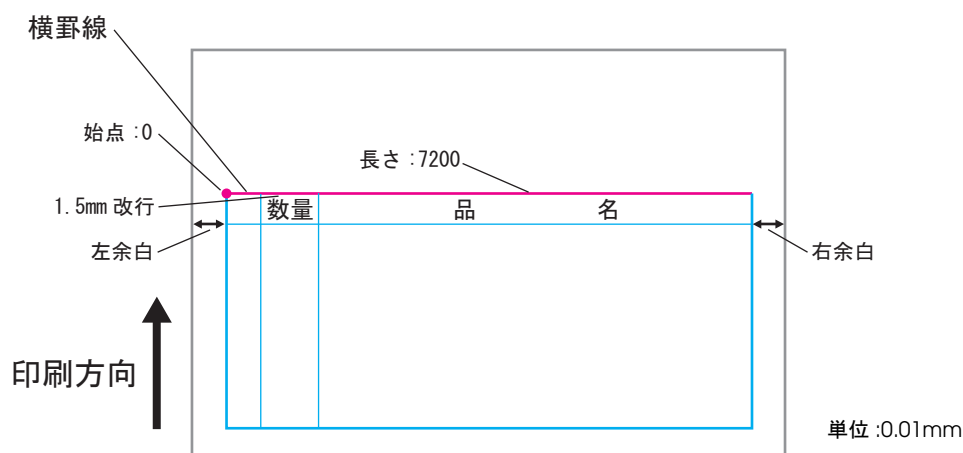
- 罫線は DirectIO を使用します。
- 横罫線の描画は、Command で 1000 を、縦罫線の描画は Command で 1010 を指定します。
- Data は罫線の種類を指定します。

Data の値	意味
1	1 ドットの実線
2	2 ドットの実線
3	4 ドットの実線
4	1 ドットの二重線（線の間隔は 1 ドット）
5	2 ドットの二重線（線の間隔は 1 ドット）
6	4 ドットの二重線（線の間隔は 1 ドット）

- 罫線描画のデータ指定方法が、横罫線と縦罫線では異なります。

- 横罫線

Object で横罫線の始点と長さを指定します。フォーマットは“始点, 長さ”で、始点と長さをカンマ (,) で区切ります。同一のラインに複数罫線を印刷する場合は、コロン (:) で区切ります（例：“始点 1、長さ 1: 始点 2、長さ 2”）。



始点は印刷方向に対し左側です。（180 度反転印刷の場合でも基準は変更されません。）

始点 + 長さの値が印刷領域を超えた場合、エラーになります。設定可能な最大値は 7200 です。

TM-T90KP で用紙幅 80mm の場合、左余白 2.65mm、右余白 4.85mm、印字領域は 72mm になります。

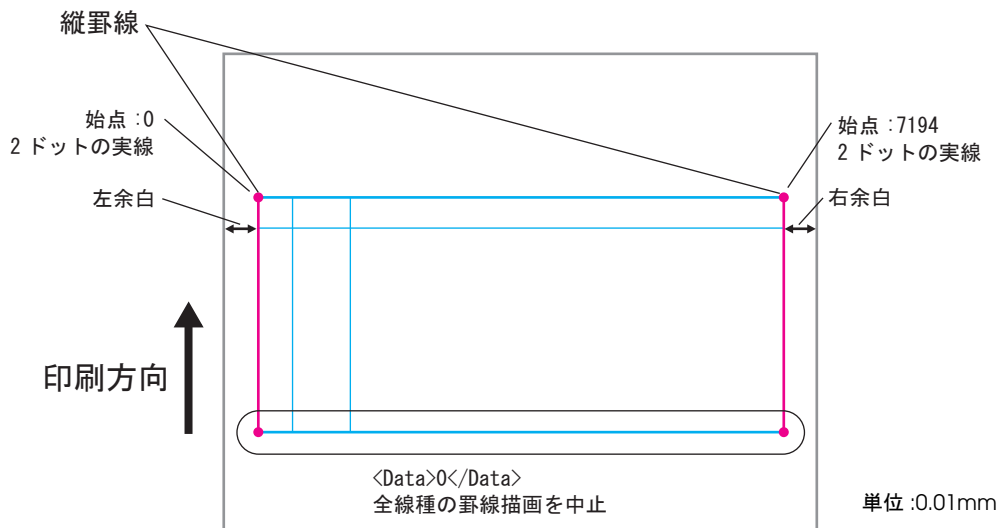
横罫線を描画した場合、次の行で印刷する文字と重ならないようにするため、1.5mm 改行します。文字を印刷した次の行に横罫線を引く場合、文字の下には余白が入るので、改行する必要はありません。

- 縦罫線

縦罫線は指定した行から罫線を引き始め、縦罫線をキャンセルした行まで罫線を引きます。

Object は縦罫線の始点を指定します。複数の罫線を印刷する場合は、始点をカンマ (,) で区切ります。

また線種の異なる縦罫線を描画する場合、再度 DirectIO を記述します。



始点は印刷方向に対し左側です。(180度反転印刷の場合でも基準は変更されません。) 始点の値が印刷領域を超えた場合エラーになります。設定可能な最大値は 7194 です。

縦罫線を指定後、次の行を印刷したときに指定された位置に縦罫線が印刷されます。

Data に 0 を指定した場合、全線種の罫線の印刷を停止します。

Object が空の場合 (始点の設定がない)、Data で指定されている線種の罫線の印刷を停止します。

```
<DirectIO>
  <Command>1010</Command>
  <Data>1</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO"></String>
  </Object>
</DirectIO>
```

- ドライバ内ではドット単位で計算されます。単位を PTR\_MM\_METRIC で 1/100mm に設定している場合、始点/長さの値は下記の計算式でドットに変換されます。

ドット値 = 設定値 \* 203 / 2540 (小数点以下は全て切り上げ)

## 白伝票の削除（TM-T90KP 専用）

調理指示単品伝票の印刷時に、オーダーの区切りを分かりやすくするために、白伝票を挿入するようにしています。  
白伝票を作成したくない場合、削除してください。

### ■ 該当スタイルシート

- Cooking.xml（調理指示単品伝票印刷）

Cooking.xml（調理指示単品伝票印刷）

白伝票

1      アメリカンコーヒー			
<b>調理単品</b>	【新 規】	A12卓	
0006	1 名    山田	11:30	

## Cooking.xml

白伝票を作成しない場合、下記の青色の部分削除してください。

```
<!-- 白伝用 カットコマンド -->
<PrintNormal>
  <Station>PTR_S_RECEIPT</Station>
  <Data>\x1b|100rP</Data>
</PrintNormal>
<!-- 印刷データ作成 -->
<xsl:apply-templates select="//ixr:Sale" />
```



## ブザー音の変更 (TM-T90KP 専用)

調理指示単品伝票の印刷後に鳴らすブザー音を変更します。

ブザー音をスタイルシートで変更にするには、TM-T90KP のディップスイッチを変更する必要があります。  
(詳細は、「TM-T90KP ユーザーズマニュアル」を参照してください。)

### ■ 該当スタイルシート

- Cooking.xml (調理指示単品伝票)

## Cooking.xml

青字部分で音色や間隔を指定しています。

```
<!-- ブザー鳴呼 Data 1 は音色、Object " 3,200" は 200ms 間隔で 3 回鳴呼の意味 -->
<DirectIO>
  <Command>1100</Command>
  <Data>1</Data>
  <Object>
    <String xmlns="http://www.epson.co.jp/UnifiedPOS/POSPrinter/DirectIO">
      3,200
    </String>
  </Object>
</DirectIO>
```

■ Data は音色を指定します。音色は 5 種類あり、1 ～ 5 の数字で指定します。

■ ブザーを繰り返し鳴らす場合、Object で繰り返し回数、および繰り返す時の間隔時間を指定します。  
フォーマットは回数と間隔時間をカンマ (,) で区切ります (“回数, 間隔時間”)。間隔時間の単位はミリ秒です。

## バーコードの変更

調理指示一枚伝票、およびゲスト伝票に印刷しているバーコードを変更します。  
ここでは、ゲスト伝票を例に説明します。

### ■ 該当スタイルシート

- Serving.xml（調理指示一枚伝票）
- Guest.xml（ゲスト伝票）
- Re\_Serving.xml（調理指示一枚伝票）


### Guest.xml（ゲスト伝票）

【新 規】 2009-02-25	テーブル 伝 票 人 数 時 刻 担 当	0008 A12 1 11:30 山田
---------------------	----------------------------------	---------------------------------

*Sample Shop*

品 名	数量	金 額
01002 アメリカンコーヒー	1	200
02004 オレンジジュース	1	300
税 抜	476	消費税
合計		500

毎度ありがとうございます  
またのご来店お待ちしております



青い部分を変更します。

## Guest.xsl

青字部分で印刷するイメージファイルを指定しています。使用するイメージファイル名に書き換えることで、店舗ロゴを変更できます。

```
<!-- シーケンス番号をバーコードで印刷 -->
<PrintBarCode>
  <Station>PTR_S_RECEIPT</Station>
  <Data><xsl:value-of select="$SequenceNumber"/></Data>
  <Symbology>PTR_BCS_Code39</Symbology>
  <Height>64</Height>
  <Width>256</Width>
  <Alignment>PTR_BC_CENTER</Alignment>
  <TextPosition>PTR_BC_TEXT_NONE</TextPosition>
</PrintBarCode>
```

■ Data はバーコードのデータを指定します。サンプルスタイルシートでは、POSLog データのシーケンス番号を指定し、POSLog データのシーケンス番号がバーコード印刷されます。

■ Symbology は、バーコードの種類を指定します。印刷できる主なバーコードは以下のとおりです。

バーコードの種類	Symbology に指定する文字列
JAN8(EAN8)	PTR_BCS_JAN8
JAN13(EAN13)	PTR_BCS_EAN13
ITF	PTR_BCS_ITF
CODABAR	PTR_BCS_Codabar
CODE39	PTR_BCS_Code39
CODE93	PTR_BCS_Code93

■ Height は、バーコードの高さを指定します。高さの指定はドット単位で行い、指定可能な値は 1 ～ 255 です。MapMode プロパティで PTR\_MM\_METRIC を指定した場合、バーコードの高さの単位は 1/100mm になります。

■ Width は、バーコードの幅を指定します。幅の指定はドット単位で行います。MapMode プロパティで PTR\_MM\_METRIC を指定した場合、バーコードの幅の単位は 1/100mm になります。バーコードの幅は、指定した値を元にドライバが調整します。基本的に指定した値以内に入る幅を選択し印刷します。

プリンター	指定可能な値
TM-T90KP	1 ～ 576
TM-T88V	1 ～ 512
TM-T70	1 ～ 576

■ Alignment は、バーコードを印刷する位置を指定します。

コマンド	意味
PTR_BC_LEFT	左寄せ
PTR_BC_CENTER	センタリング
PTR_BC_RIGHT	右寄せ

■ TextPosition は HRI 文字の印刷の有無を指定します。

コマンド	意味
PTR_BC_TEXT_NONE	HRI 文字印刷無し
PTR_BC_TEXT_ABOVE	バーコードの上に HRI 文字を印刷
PTR_BC_TEXT_BELOW	バーコードの下に HRI 文字を印刷

# UPOS コマンド

文字列を印刷する場合に使用する、UPOS のエスケープシーケンスコマンドで主だったものを説明します。  
エスケープシーケンスは PrintNormal で使用します。

## ■ 文字属性に関する設定

エスケープシーケンスコマンド	意味
¥x1b bC	太字
¥x1b rvC	反転
¥x1b lC	通常の文字の大きさ
¥x1b 2C	横倍角
¥x1b 3C	縦倍角
¥x1b 4C	4 倍角
¥x1b cA	センタリング
¥x1b rA	右寄せ
¥x1b N	文字属性をリセット

## ■ 用紙カットに関する設定

エスケープシーケンスコマンド	意味
¥x1b 90P	用紙カット（1 点残し）
¥x1b 90rP (TM-T90KP 専用)	用紙カットの予約 (エプソンオリジナル : Cooking.xml で使用)

## ■ 改行に関する設定

エスケープシーケンスコマンド	意味
¥x1b #uF	# は数字をいれ、# で指定された分だけ改行（ペーパーフィード）します。 数字の単位は MapMode プロパティに従います。（サンプルスタイルシートの初期設定ではミリに設定されています。）
¥x1b #lF	# は数字をいれ、# で指定された行数だけ改行。