

NPI EasyPrint

操作説明書

ver.1.1.2.0

2023/05/17

Nippon Primex Inc.

目次

1	NPI EasyPrint 概要.....	4
1. 1	概要.....	4
1. 2	対応 OS.....	4
1. 3	制限事項.....	5
1. 4	免責事項.....	5
2	NPI EasyPrint のセットアップ.....	6
2. 1	印刷イメージ作成ソフトを使用する場合.....	6
2. 2	サポートライブラリとして使用する場合.....	6
3	利用方法.....	7
3. 1	プリンター設定.....	7
3. 2	フォーマット設定.....	9
3. 2. 1	コマンド発行一覧.....	11
3. 2. 2	XML 出力やプリンタへのテンプレート登録を行う場合の CSV ファイルについて.....	15
3. 3	コンポーネント設定.....	17
3. 4	印刷.....	19
3. 4. 1	印刷枚数・複写枚数とデータベース使用時の印刷例.....	21
4	レイアウトデザイナーの操作.....	24
4. 1	コンポーネントの移動.....	24
4. 2	コンポーネントのサイズ変更（拡大・縮小）.....	25
4. 3	コンポーネントの回転.....	26
4. 4	メニュー操作.....	27
4. 4. 1	編集.....	27
4. 4. 2	表示.....	29
4. 4. 3	履歴.....	30
4. 4. 4	ツール.....	31
4. 4. 5	ヘルプ.....	34
4. 5	アイコンメニュー操作.....	35
4. 5. 1	拡大・縮小.....	35
4. 5. 2	コンポーネント操作アイコン.....	35
4. 5. 3	データベース操作.....	36
4. 6	コンポーネント情報表示枠.....	39
5	コンポーネントの説明.....	40
5. 1	テキストコンポーネント.....	40
5. 2	バーコードコンポーネント.....	42
5. 2. 1	バーコード一覧 / オプション詳細.....	44
5. 3	ラインコンポーネント.....	57
5. 4	ボックスコンポーネント.....	58

5. 5	イメージコンポーネント	60
5. 6	テーブルコンポーネント	61
6	データ種別・フォーマットについて	62
6. 1	Standard	63
6. 2	Serial Count	63
6. 3	Date Time	64
6. 4	Copy	64
6. 5	Substring	65
6. 6	Compute	66
6. 7	DataSearch / DataSearch2	68
6. 8	Join	69
6. 9	Attribute	69

1 NPI EasyPrint 概要

1. 1 概要

NPI EasyPrint は以下の用途にお使いいただけます。

1. 印刷イメージ作成ソフト

印刷イメージを GUI にて作成することができます(レイアウトデザイナー)

テキストやイメージ、バーコードの印刷はもちろん、ライン・ボックス・テーブルといったコンポーネントを揃えており、簡単に印刷券面を作成することができます。

作成した印刷券面はプリンター設定を行った後に、印刷画面から印刷することができます。

また様々な外部データベースとの連携も可能としています。

2. ラベル印刷ソフト

ラベルプリンターへの印字に対応しています。

(ラベルコマンドは ZPL II コマンドをサポートしております)

作成した印刷イメージは XML 形式ファイルで保存し、プリンターにテンプレート登録、印刷することが可能です。

3. ユーザーアプリケーションからの印刷サポートライブラリ

既に印刷イメージを作成し、XML 形式のファイルとして保存している場合にはお客様が作成されたアプリケーションから NPIGF.dll を用いて XML ファイルを読み込んで印刷を行うことが可能です。

その他の機能としてプリンター初期化、プリンターのステータス管理、テキスト、バーコード、セルコンポーネントのデータ取得・編集を提供しています。

1. 2 対応 OS

当アプリケーションは Windows10 (32/64bit) において動作確認を行っております。

他 OS で使用される際の不具合・損害等については責任を負いかねます。

1. 3 制限事項

1. XML、ZPL II（コマンド）出力は、EasyPrint 上での表示と実際に印字される結果が異なる場合があります。（ZPL II は Text コンポーネントが領域外にはみ出した分も印字される等）
使用する場合は予めテスト印字を行い、印字に差し支えの無いことを確認して利用ください。
2. ZPL II（コマンド）出力でバーコードを印字する際、EasyPrint 上で表示が出来ていても印字結果が INVALID - S 等のエラーになる恐れがあります。
EasyPrint 上では足りないデータ等を自動補完して表示を行っていますが、プリンタ側では補完が行われない為、ZPL コマンドに沿ったデータを入力して頂くようにお願いします。
3. ZPL II（コマンド）出力では一部プロパティが使用できません。使用できないプロパティや値が入力された際は自動でデータ置換を行います。
4. 印刷に関連する機能制限としては、共通ライブラリ及びドライバーの制限事項を参照してください。

1. 4 免責事項

本アプリケーションを使用することで何らかの問題が発生した場合において、本アプリケーション、プリンターの使用または使用不能から生ずるいかなる損害（間接的・偶発的・必然的・特殊・典型的損害、修理費用、訴訟費用、事業利益の損失、事業の中断、事業情報の損失、またはその他の金銭的損害を含むがこれらに限定されない）に関して弊社は一切責任を負いかねます。

弊社がかかる損害の可能性について知らされていた場合でも同様となります。

お客様各自の責任で使用をお願いします。

2 NPI EasyPrint のセットアップ

本製品の印字機能を使用する際は、NPI 統合ドライバをインストール、または共通ライブラリ Windows 版の DLL ファイル (NPrinterLib.dll) を System32 フォルダに配置する必要があります。

設定方法につきましては別途各設定ガイドをご参照ください。

NPI EasyPrint はフォルダ・ファイル構成を含め Zip 圧縮して配布します。

ユーザー任意の場所に展開してご使用ください。

・フォルダ・ファイル構成

NPIEasyPrint_[バージョン]_[YYYYMMDD]

Driver

NPI_IG3.2.11.0_20230216 (統合ドライバー Ver.3.2.11.0) ※2023/05/12 時点での最新ドライバー

NPIEasyPrint

Win32

NPIEasyPrint.exe (レイアウトデザイナー 32bit 版)

Win64

NPIEasyPrint.exe (レイアウトデザイナー 64bit 版)

NPIGF

NPIGFSample (NPIGF を使用したサンプルアプリケーション)

Win32

NPIGF.dll (プリンタ制御関数 DLL ファイル 32bit 版)

Win64

NPIGF.dll (プリンタ制御関数 DLL ファイル 64bit 版)

NPIGF ガイド.pdf (NPIGF.dll 使用説明書)

Distribution.txt (改訂履歴)

※当ガイドファイルは NPIEasyPrint.exe と同じフォルダに配置してあります。(32/64bit 共通)

2. 1 印刷イメージ作成ソフトを使用する場合

NPIEasyPrint.exe を起動して、イメージ作成・プリンターへの印字を行ってください。

手順については「[3 利用方法](#)」から説明いたします。

2. 2 サポートライブラリとして使用する場合

お客様が作成されたアプリケーションから NPIGF.dll を用いて印刷を行うことが可能です。

NPIGF.dll をアプリケーションと同じフォルダ、または System32 もしくは SysWOW64 フォルダに配置し、

DLL を読み込んでください。ご使用方法につきましては NPIGF ガイド.pdf とサンプルアプリケーションを

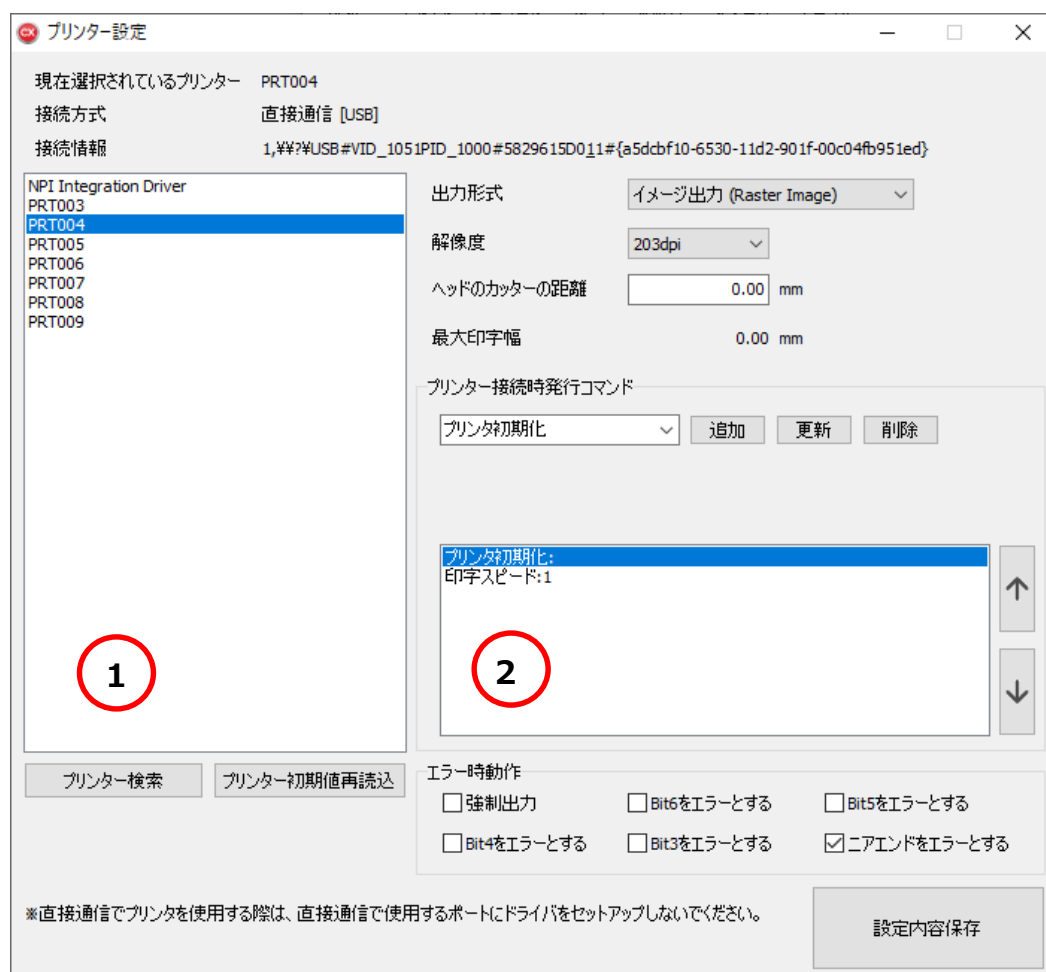
合わせてご確認ください。

3 利用方法

3.1 プリンター設定

最初にプリンターの設定を行います。

メニューの「ファイル」→「プリンター設定」の順にクリックしてください。



画面を開くと、使用できるプリンター一覧が左枠内(①)に表示されます。

使用したいプリンター名が見つからない場合は「プリンター検索」ボタンを押下してください。

プリンターを選択すると画面上部に接続方式が表示されます。

また、右にプリンター設定として出力形式・解像度・用紙タイプ・ヘッドのカッターの距離を選択・入力することができます。「プリンター初期値再読込」ボタンを押下するとプリンターから情報を取得し、初期値を設定することができます。

接続方式が直接通信のプリンタから情報を取得する際、直接通信で使用するポートにドライバがセットアップされているとプリンタ情報を正しく取得できません。

直接通信で使用したい場合はドライバを削除するか、Spooler を停止する必要があります。

出力形式は以下の 5 種類から選択します。

- ① イメージ出力 (GDI) : Windows API を使用してイメージを出力します。
- ② イメージ出力 (RasterImage) : イメージ形式で出力します (プリンターのイメージコマンド以外は使用しません)
- ③ XML : XML 形式で出力します。
- ④ ZPL II (イメージ) : イメージ形式で出力します (プリンターのイメージコマンド以外は使用しません)
- ⑤ ZPL II (コマンド) : 各コンポーネントに対応した ZPL II コマンドで出力します。

ドライバ(NEI)は①、②のみ、直接通信は①を除いたすべて、ドライバ(Zebra)は①、④、⑤の選択となります。

解像度は 203、300、400、600dpi の 4 種類から選択します。

使用されているプリンターに合った解像度を指定してください。

ヘッドのカッターの距離は印字後カット時のフィード量に使用されます。(Raster、XML 時)

最大印字幅は「プリンター初期値再読込」から情報取得した場合に設定されます。

ここで設定された印字幅はフォーマット作成時のページサイズ幅に値が反映されます。

プリンター接続時発行コマンドはコマンドとパラメータを選択し、「追加」ボタンを押下してコマンド一覧(②)に追加します。削除したい場合はコマンド一覧から選択し、「削除」ボタンを押下します。

出力形式がイメージ出力(GDI)、XML 選択時にはコマンドを選ぶことはできません。

指定できるコマンド一覧は「[3. 2. 1 コマンド発行一覧](#)」を参照してください。

コマンドの発行は上の項目から順に発行されます。順番の入れ替えは削除ボタン横の上下矢印で行えます。

エラー時動作はプリンターでの印刷時にステータスをチェックして印刷を行うかどうかを設定します。

各プリンターの仕様書にステータス (bit0～bit7) に関する説明があります。

印刷時に取得したステータスにおいて、エラー時動作のチェックを入れている項目で bit が ON になっている場合は印刷を行わず、エラーで処理を中断します。

初期状態はチェックなし (bit1、bit2 のみ ON になっていた場合はエラー対象) となります。

強制出力を選択するとステータスの状態に関わらず印刷処理を行います。

編集が終わりましたら「設定内容保存」ボタンを押下してください。

当アプリケーションと同じフォルダに設定ファイル(NPIEasyPrint.inf) が作成・更新されます。

※「プリンター接続時発行コマンド」は以下のタイミングで出力されます。

1. 当アプリケーション起動時
2. 設定内容保存ボタン押下時
3. プリンターステータスがオフラインからオンラインになった時

※当アプリケーション起動時に設定ファイル(NPIEasyPrint.inf)に前回 EasyPrint で選択したプリンターが指定されていない場合は出力されません。

3. 2 フォーマット設定

次にフォーマットの作成を行います。

メニューの「ファイル」→「新規作成」の順にクリックしてください。

- ・フォーマット名は半角英数字で指定してください。
- ・用紙種類は以下の4種類から選択してください。
 - 「連続紙」 … 一般的に使われるレシートロール紙
 - 「ラベル（ギャップ使用）」 … ギャップのあるラベル用紙
 - 「ラベル（BM 使用）」 … ブラックマークのあるラベル用紙
 - 「プリンタ設定使用」 … プリンタに設定されている用紙（ZPL のみ選択可能）
- ・ページサイズの幅・長さを mm 指定で入力してください。
- ・カットオプションを以下から選択してください。
 - ① プリンター設定で接続方式ドライバー（NEI）、直接通信を選択した場合
 - 「フルカット」
 - 「パーシャルカット」
 - 「カットしない」 （カット位置までの紙送りのみ行います）
 - 「連続」 （紙送り・カットは行いません）

② プリンター設定で接続方式ドライバ（Zebra）を選択した場合

「手切り」

「剥離」

「巻取り」

「プリンタ設定使用」

「カット」

カット無し

…手切りと同様です。

ラベル印字毎にカット

…1枚毎にカットを行います。

ジョブ毎にカット

…指定枚数印字の最後にカットを行います。

複写毎にカット

…最後の複写ラベル印字時にカットを行います。

指定枚数毎にカット

…指定枚数毎にカットを行います。

※プリンタが対応していないカットオプションを指定した場合、印字動作が正しく行われない、またはプリンタが予期せぬ動作をする恐れがあります。

- ・ コマンド発行は「印字前コマンド」「印字後コマンド」の両方で設定します。

プリンター設定の出力形式により指定できるコマンドは異なります。

指定できるコマンド一覧は次ページで説明します。

- ・ コマンドの発行は上の項目から順に発行されます。順番の入れ替えは削除ボタン横の上下矢印で行えます。

3. 2. 1 コマンド発行一覧

コマンド名	説明	発行コマンド (Raster)	発行コマンド (ZPL)
プリンタ初期化	プリンタ初期化コマンドを実行します。	ESC @	–
印字スピード	印字速度の設定を行います。	GS S n	^PR
分割駆動	分割駆動の設定を行います。	GS %	–
印字濃度	印字濃度の設定を行います。 RasterImage 出力の際は%単位で手動入力します。	GS ~ n	~SD
LED ベゼル動作	ベゼル動作の設定を行います。	GS l n m	–
プレゼンタ動作	プレゼンタ動作の設定を行います。	ESC h n	–
印字通知設定	印字開始／終了設定コマンドを実行します。 0~3 までありますが、印字開始と同じ番号の印字終了コマンドを使用してください。 印字開始 1 と 3 はジョブ ID が 1 から自動で割り当てられます。 ※ラベル時は自動で印字開始 3 が行われるため使用できません。	GS G n	–
ページモードのクリア	ページバッファのクリアを行います。	CAN	–
頭出し設定	ラベルの頭出しを行います。 ※プリンタにより使用できる頭出し種別が異なる場合があります。	GS FF n	–
Y 位置補正 (dot)	ラベルの Y 軸の印字基点を dot 単位で調整します。	DC2 B xL xH y	^LT
メディアタイプ	メディアタイプを感熱紙と熱転写から選択します。		^MT
回転	印字を回転させます。	コマンドではなくイメージ自体を回転して出力します。	
印字後ポーズ処理	指定した印字後にポーズ状態になります。 ポーズの解除方法については使用するプリンタの仕様書を確認ください。	–	^PQ
リセット	プリンタリセットコマンドを実行します。 ※バッファのリセットが行われる恐れがあります。リセット前後に印字する際は、リセット動作が印字に影響しないことを確認して使用してください。(文字化け印字などの恐れ有)	DC1	~JR
ドキュメント継続	指定することで印字後のカットや紙送り動作が行われなくなり、次に印字されるフォーマットを繋げて印字できます。 ※ラベル出力時は使用できません。	コマンド発行無し	ラベル開始時の 頭出し動作の都合上未対応
バックフィード	印字後のバックフィード処理を設定します。	–	~JS
センサーLED	ラベルのセンサーLED を切り替えます。	GS R n	–
印字領域設定	フォーマットの幅に合わせた印字領域設定を行い、印字が中央に来るようにします。	レシート: GS W ラベル: DC2 W	–
その他コマンド	任意のコマンドを 16 進数文字列で入力できます。	任意	

発行コマンドを使用する際はプリンタがコマンドをサポートしていることを確認の上利用ください。

「データベース」を使用する場合は、赤枠タブをクリックして、データベース情報を入力します。

レシート / ラベルフォーマット

フォーマット名 OK

フォーマット情報 **データベース**

DB利用 ☒ 無 ☐ 有

データ形式 ☒ SQL ☐ CSV 検索実行

接続文字列 生成

SQL文 / CSVファイル名

フィールド関連付け情報

データベース項目名 → コンポーネント名 追加 削除

- ・ DB を利用する場合は「有」、利用しない場合は「無」を選択してください。
 - ・ 利用するデータベースの形式を選択してください。
 - SQL … データベース（※種類については[リンク先](#)参照）に接続し、SQL（SELECT 文のみ使用可）を実行して、データを取得
 - CSV … CSV ファイルを読み込み、データを取得。
 - フィールド関連付け情報の「データベース項目名」部分が「**CSV 番号**」の表記に変わります。
- ※「CSV 番号」とは CSV ファイル内の行をカンマで区切った時の順番です。
- CSV ファイルに、
- ABC,123,Tokyo
- と入っていた場合は「ABC」が CSV 番号 1、「123」が CSV 番号 2、「Tokyo」が CSV 番号 3 となります。
- ※データにカンマを含めたい場合は、レコードの先頭から末尾をダブルコーテーションで囲みます。
- CSV ファイルに、
- ABC,"123,456",Tokyo
- と入っていた場合は「ABC」が CSV 番号 1、「123,456」が CSV 番号 2、「Tokyo」が CSV 番号 3 となります。
- ※データにダブルコーテーションを含めたい場合は、レコードの先頭から末尾をダブルコーテーションで囲み、データとなるダブルコーテーションは 2 つ繋げて「""」表記します。
- CSV ファイルに、
- ABC,"123""456",Tokyo
- と入っていた場合は「ABC」が CSV 番号 1、「123"456」が CSV 番号 2、「Tokyo」が CSV 番号 3 となります。

- ・ 接続文字列は「SQL」使用時に入力します。「生成」ボタンを押下して、接続文字列生成ダイアログから文字列を生成してください。「CSV」選択時は入力する必要はありません。
- ・ SQL 文 / CSV ファイル名は「SQL」選択時は SELECT 文を入力します。
「CSV」選択時は CSV ファイル（フォルダパスを含む）を入力します。

※XML 出力とプリンタへのテンプレート登録を行う場合は[こちら](#)を参照してください。

上記までの項目を設定しましたら、右上の「検索実行」ボタンを押下してください。

SQL 選択時は「データベース項目名」部分に SELECT 文実行結果の項目名一覧

CSV 選択時は「CSV 番号」部分に CSV ファイルの 1 行をカンマで区切った項目数が選択できるようになります。

- ・ フィールド関連付け情報は以下のように指定します。

SQL 選択時：

「データベース項目名」と「コンポーネント名」を 1 セットとした形で「追加」ボタンを押下します。
追加されると、「追加」左下にある白枠内に登録されます。

（フォーマット新規作成時はコンポーネント名が選択できません。フォーマット作成後にコンポーネントを追加し、フォーマット編集でデータベース設定してください）

SQL 文で取得してきた結果に対して、まず先頭レコードの「データベース項目名」の値を
「コンポーネント名」で指定されたコンポーネントのデータに設定します。

例）SQL 文の取得結果が以下だった場合

	name	age	address
1	Tatsuya	25	Tokyo
2	Mai	34	Kanagawa
3	Miyuki	42	Chiba
4	Daisuke	40	Saitama
5	Makoto	52	Tokyo

フィールド関連付け情報を

name -> cmp1

age -> cmp2

address -> cmp3

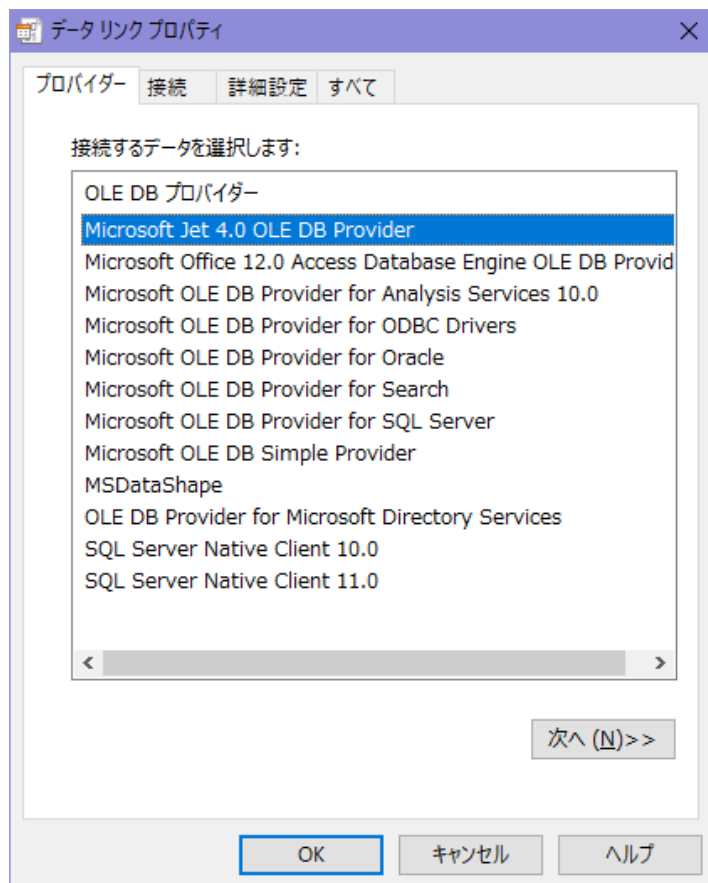
と指定したとします。

まずコンポーネント名 cmp1 のデータ部分に項目名 name で取得した「Tatsuya」が入ります。

同様にコンポーネント名 cmp2 のデータ部分に項目名 age で取得した「25」が入ります。

次のレコードにはメイン画面の操作パネルで進むことができます。

データベースに関しては、以下のダイアログから接続・設定ができるもののみ対応可能となっております。（接続文字列の「生成」ボタンを押下）



CSV 選択時：

「CSV 番号」と「コンポーネント名」を1セットとした形で「追加」ボタンを押下します。
追加されると、「追加」左下にある白枠内に登録されます。

CSV ファイルを読み込んだ結果に対して、まず先頭行の「CSV 番号」にある値を「コンポーネント名」で指定されたコンポーネントのデータに設定します。

例) CSV ファイルのデータが以下だった場合

Tatsuya,25,Tokyo
Mai,34,Kanagawa
Miyuki,42,Chiba
Daisuke,40,Saitama
Makoto,50,Tokyo

フィールド関連付け情報を

1 -> cmp1

2 -> cmp2

3 -> cmp3

と指定したとします。

まずコンポーネント名 cmp1 のデータ部分に csv 番号 1 で取得した「Tatsuya」が入ります。
同様にコンポーネント名 cmp2 のデータ部分に csv 番号 2 で取得した「25」が入ります。
次のレコード（行）にはメイン画面の操作パネルで進むことができます。

※CSV ファイルの各行の項目数（カンマで区切った項目）は全ての行で同じ数にしてください。

例) 不適切な CSV ファイル例

AAA, 10	← 項目数が 2
BBB, 20, xxx	← 項目数が 3
CCC, 30, 123, abc	← 項目数が 4

3. 2. 2 XML 出力やプリンタへのテンプレート登録を行う場合の CSV ファイルについて

レシート/ラベルフォーマット

フォーマット名: Format1

フォーマット情報: データベース

DB利用: ☐ 無 ☒ 有

データ形式: ☐ SQL ☒ CSV

接続文字列: 生成

SQL文 / CSVファイル名: 参照

E:\Data\File.csv

フィールド関連付け情報

CSV番号	コンポーネント名
2	Component2

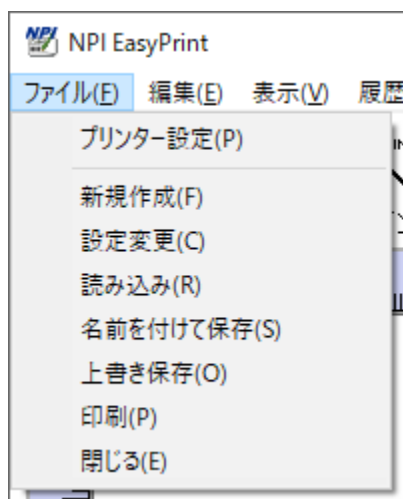
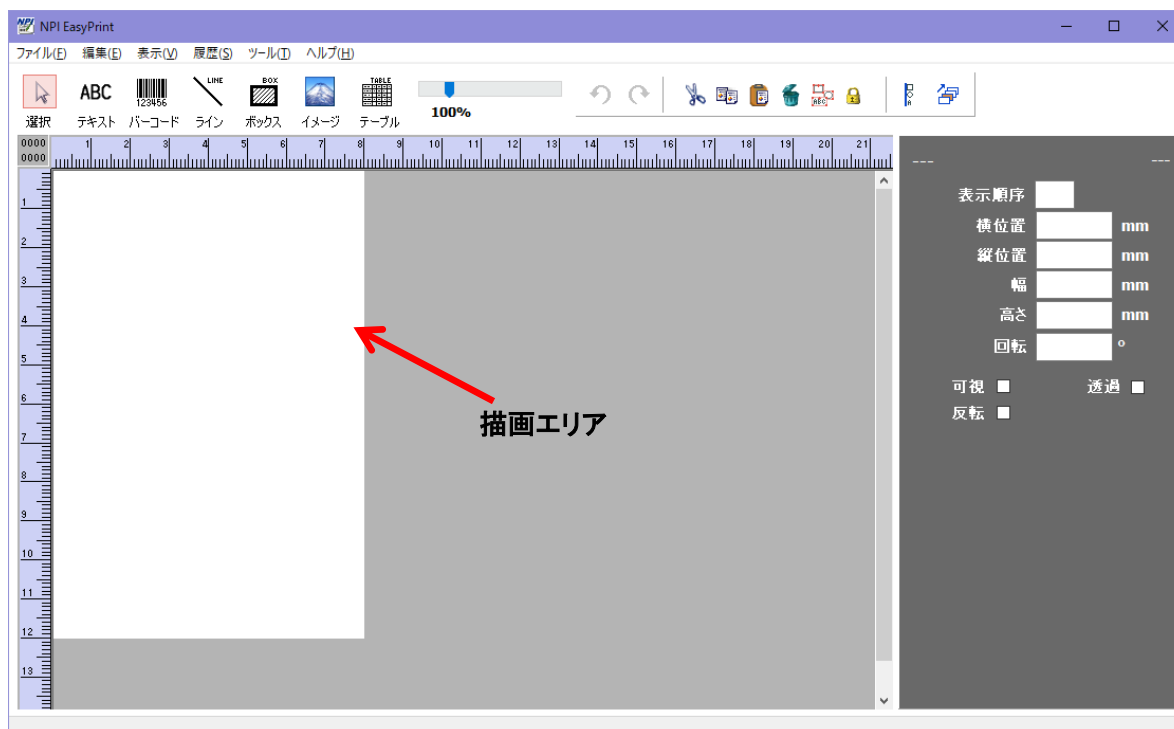
追加 削除

1 -> Component1
2 -> Component2

XML 出力とプリンタへのテンプレート登録を行う場合は、EasyPrint 上で PC 上の CSV ファイルを読み込んでフィールド関連付け情報を追加した後に、SQL 文/CSV ファイル名にプリンタの CSV ファイルパス(E:¥~又は U:¥~)を入力する必要があります。

その際 OK を押した時に CSV ファイルが読み込めないというエラーが表示されますが、印刷時にプリンタ側で CSV ファイル処理が行われる為問題ありません。

フォーマット設定が終わりましたら、「OK」ボタンを押下してください。
フォーマットを作成すると、画面上に白枠で描画エリアが生成されます。



既に作成したフォーマット内容を編集したい場合には「ファイル」→「設定変更」を押下してください。
フォーマット情報を保存したい場合には「ファイル」→「名前を付けて保存」を押下し、ファイルに保存します。（XML 形式で保存されます。ファイル名はお客様で自由に決めることができます。）
保存した XML ファイルは「ファイル」→「読み込み」にて再展開することができます。
また、「ファイル」→「上書き保存」にて読み込み済の XML ファイルに現在のフォーマット内容を上書きできます。

※データベース利用時の現在レコード位置は XML ファイルに保存できません。

XML ファイルで再読込した場合、先頭レコードが表示されます。

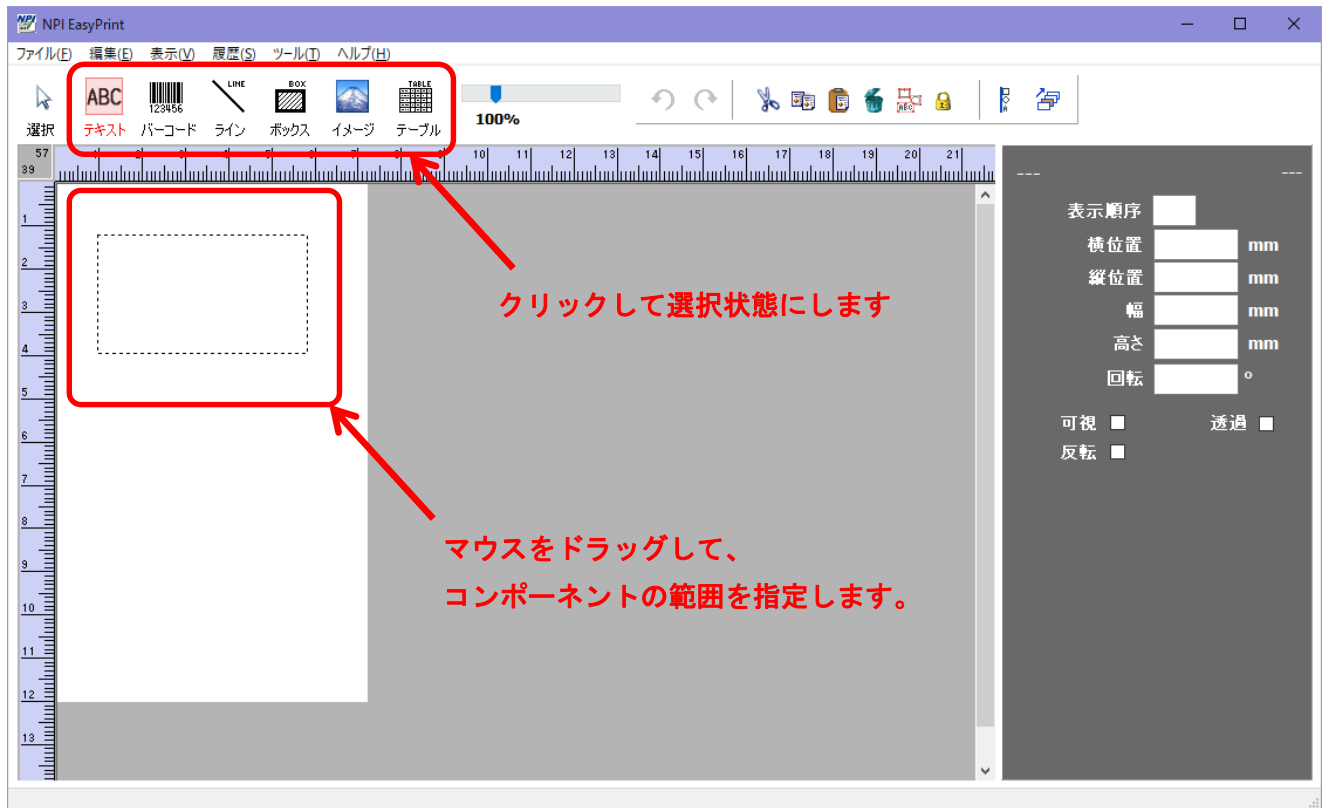
3. 3 コンポーネント設定

フォーマットが作成しましたら、次にコンポーネントを作成します。

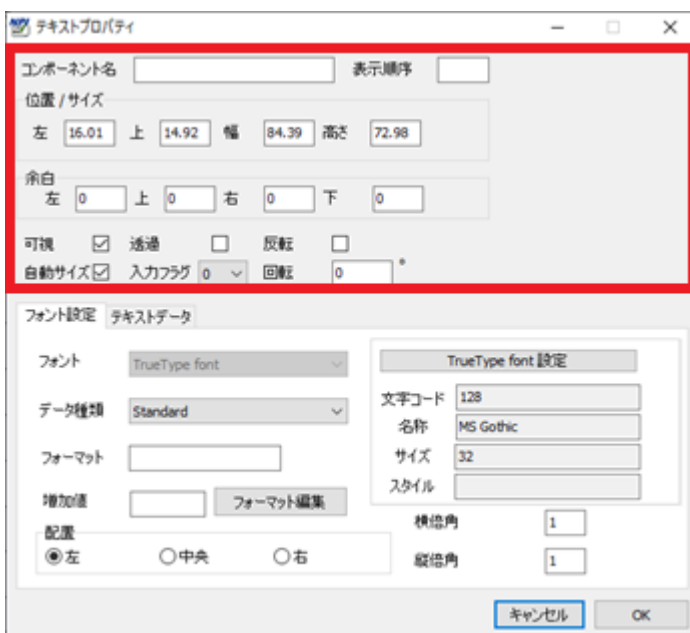
コンポーネントはテキスト・バーコード・ライン・ボックス・イメージ・テーブルの6種類となっています。

作成したいコンポーネントアイコンを押下してください（選択状態になります）

選択状態にした後、フォーマット上でマウスをドラッグして、コンポーネントの範囲を指定します。



コンポーネントの範囲を指定して、マウスを戻すとコンポーネント編集画面が表示されます。



コンポーネント共通項目（コンポーネントによって存在しない項目もあります）

- ・コンポーネント名を半角英数字にて入力してください。

何も入力しない場合は、「Component～」(～は任意の数字)で作成されます。

- ・ 表示順序を半角数字にて入力してください。

表示順序「1」が最背面に描画され、その前面に 2,3,...と続きます。

表示順序が一番大きいコンポーネントが最前面に描画されます。

- ・ 位置 / サイズ

フォーマット上の位置（コンポーネント左上座標）と幅・高さをそれぞれ mm で指定します。

小数点第 2 位まで指定することができます。

- ・ 余白

各データの余白部分を上下左右 mm で指定します。

- ・ 可視（初期状態は ON）

チェックを ON にすることで可視、OFF で不可視となります。

- ・ 透過（初期状態は OFF）

チェックを ON にすることで透過あり、OFF で透過なしとなります。

透過判定は領域全体に適用されるものとします。

透過なしの場合は領域全体が背景要素を隠す形となります。

- ・ 反転（初期状態は OFF）

チェックを ON にすることで白黒反転します。OFF で反転なしとなります。

領域全体に適用されるものとして余白部分も反転を行います。

- ・ 自動サイズ（初期状態は ON）

チェックを ON にすることで描画をコンポーネントの幅、高さに合わせます。

フォントサイズ指定やバーコード線幅指定などが無効になります。

- ・ 入力フラグ（初期状態は 0）

チェックを ON にすることで、XML テンプレート登録をした際にプリンタがデータ入力を要求します。

0：印刷時入力なし

1：印刷時に入力要求をする

2：フォーマット読込時に入力要求をする

- ・ 回転（初期状態は 0°）

0° ～ 359° の範囲でコンポーネントを回転させることができます。

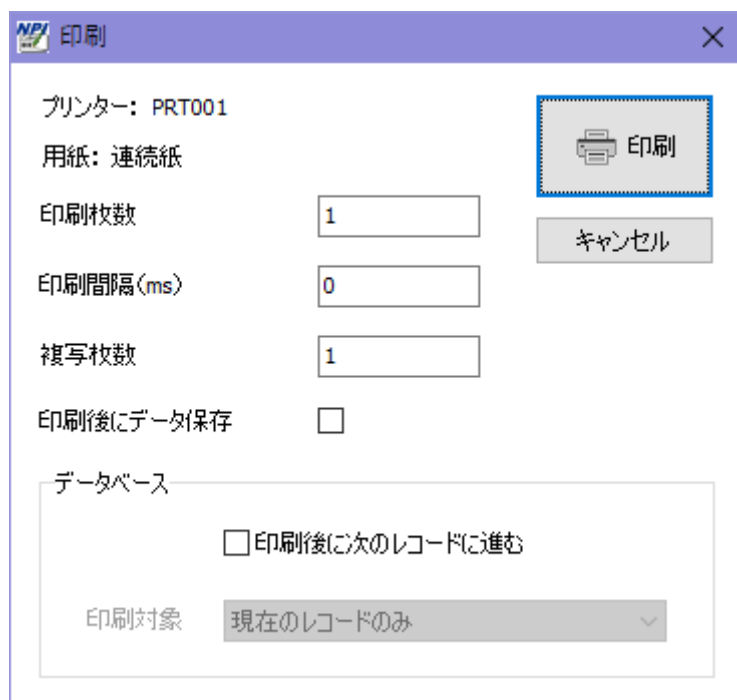
その他、各コンポーネントの詳細設定については「[5 コンポーネントの説明](#)」を参照してください。

3. 4 印刷

メニューの「ファイル」→「印刷」をクリックします。



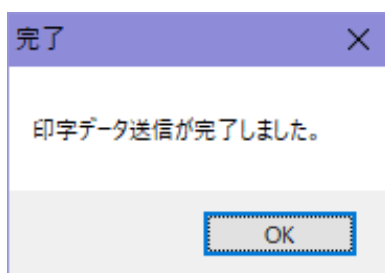
以下の印刷ダイアログが表示されます。



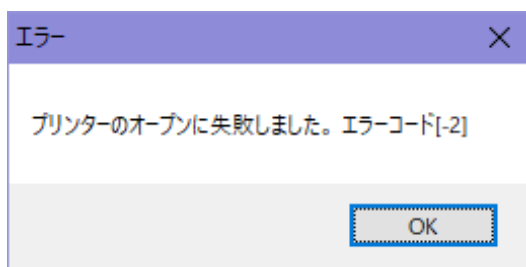
- ・プリンター部分には「プリンター設定」で選択したプリンター名が表示されます。
- ・用紙部分には「フォーマット設定」で選択した用紙種類が表示されます。
- ・印刷枚数に印刷したい枚数（1 以上）を指定します。

- ・印刷間隔（ms）に複数枚印刷する場合の印刷間隔をミリ秒で指定します。
（1 券面のデータを送信する間隔の為、実際に印刷される間隔とは若干異なることがあります）
- ・複写枚数（1 以上で指定）は 1 つのフォーマットを指定された複写枚数分印字します。
- ・印刷後にデータ保存にチェックを入れると、印刷後にフォーマットを保存するダイアログが開きます。既にファイル保存してある場合には上書き保存されます。
印刷毎に値が変化する場合（テキストコンポーネントで“Serial Count”を使用、データベース使用など）に最後に印刷した時の値を保存したい場合に使用します。
- ・データベース部分は“データベース”または“CSV ファイル”読み込み時に選択可能となります。
「印刷後に次のレコードに進む」にチェックを入れると、印刷後に次のレコードに移動します。
レコードが末尾で次のデータに進めないときはエラーとなります。
- ・“データベース”または“CSV ファイル”を使用した印刷時に「印刷後にデータ保存」にチェックを入れると印刷したデータ位置がファイル（読み込んでいるフォーマットファイルと同じフォルダに [フォーマット名]_DB.inf という名前）が保存されます。
フォーマットを再読み込みした場合に上記 inf ファイルがあれば、最後に印刷した位置からデータが表示されます。
- ・XML 出力の際はプリンタの CSV ファイルパスが指定されている為、印刷画面上でデータベースの設定を行えません。
正しくプリンタ上の CSV ファイルパスが指定されている場合は、設定に関わらず全件印刷が行われます
- ・印刷対象では「現在のレコードのみ」「全件印刷」を選択することができます。
「現在のレコードのみ」は表示しているデータのみを印刷します。
「全件印刷」はデータベースの全レコードを先頭から印刷します。
全件印刷を選択した場合には、「印刷後に次のレコードに進む」を有効にすることはできません。
（全件印刷なので自動的に最初から最後のレコードまで進みます）
また全件印刷実施後は、先頭レコードに移動されます。

印刷完了後は以下のダイアログが表示されます。



印刷に失敗した場合はエラーメッセージとエラーコードが表示されます。

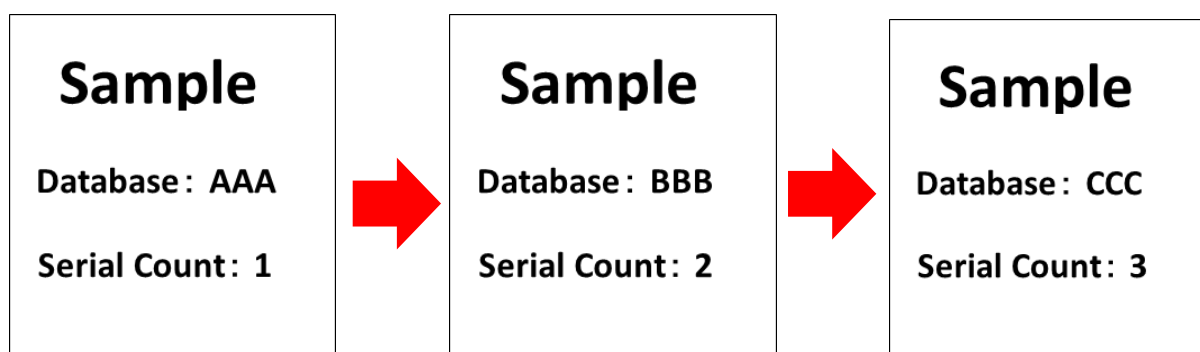


3. 4. 1 印刷枚数・複写枚数とデータベース使用時の印刷例

データベース印字と、Serial Count（例では+1 で数値カウントアップ）の印刷例を下記に説明します。

例 1)

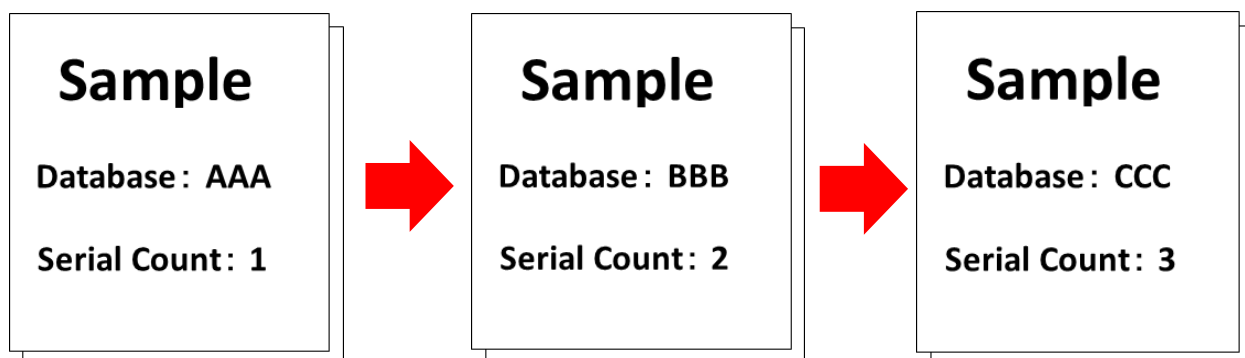
印刷枚数が 3、複写枚数が 1、データベース読み込み時（データは AAA、BBB、CCC…という形）で「印刷後に次のレコードに進む」のチェックを入れた場合の印刷



- ・「印刷枚数が 3」なのでデータベースレコードを 3 つ進める形になります。
- ・シリアルカウントを使用している場合は印刷枚数分カウントアップします。

例 2)

印刷枚数が 3、複写枚数が 2、データベース読み込み時（データは AAA、BBB、CCC…という形）で「印刷後に次のレコードに進む」のチェックを入れた場合の印刷



※同じものを 2 枚

※同じものを 2 枚

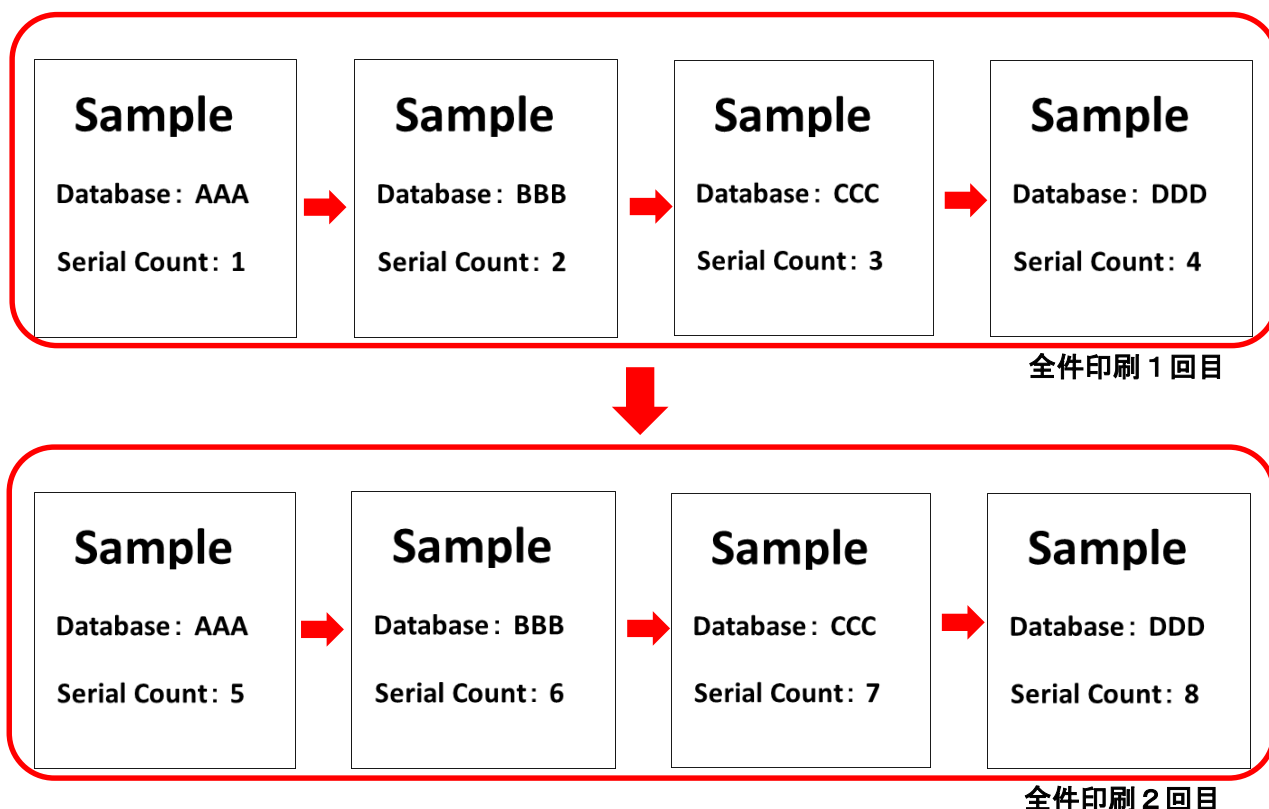
※同じものを 2 枚

- ・「印刷枚数が 3」なのでデータベースレコードを 3 つ進める形になります。
また「複写枚数が 2」なのでそれぞれ 2 枚ずつ印刷します。計 6 枚の印刷となります。

例 3)

印刷枚数が 2、複写枚数が 1、

データベース読み込み時（データは AAA、BBB、CCC、DDD という形で全 4 件）での
データベース「全件印刷」を選択した場合の印刷



「印刷枚数が 2」なので全件印刷を 2 回、また「複写枚数が 1」なので各 1 枚ずつ印刷します。

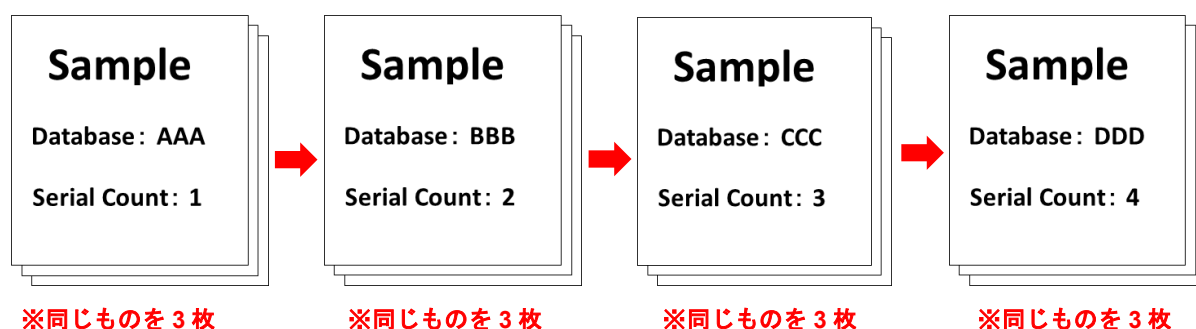
計 8 枚の印刷となります。

データベース印字内容は全件印刷毎に最初のデータに戻りますが、シリアルカウントは常に
カウントアップされて印字されます。

例 4)

印刷枚数が 1、複写枚数が 3、

データベース読み込み時（データは AAA、BBB、CCC、DDD という形で全 4 件）での
データベース「全件印刷」を選択した場合の印刷



・「印刷枚数が 1」なので全件印刷を 1 回、また「複写枚数が 3」なので各 3 枚ずつ印刷します。

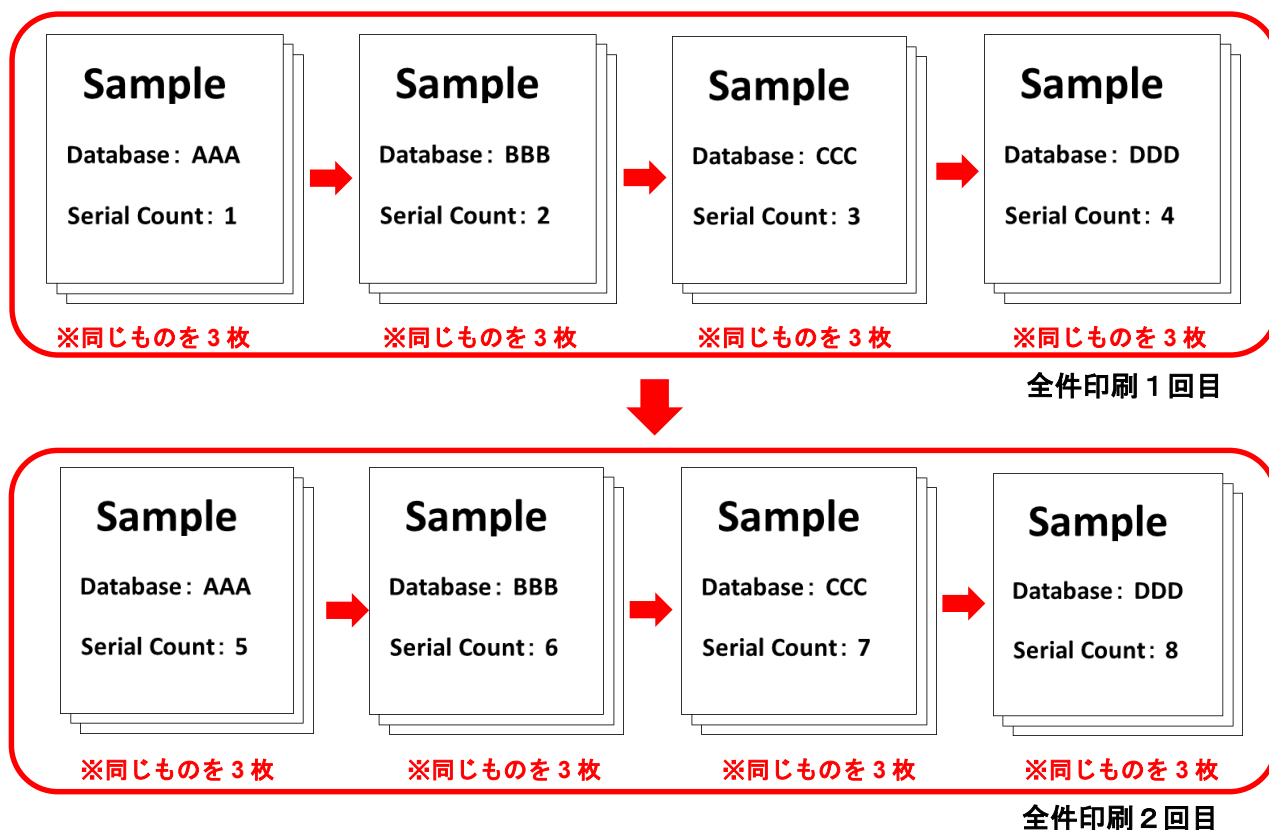
計 12 枚の印刷となります。

シリアルカウントは複写枚数 3 枚の印刷が終わった時点でカウントアップします。

例 5)

印刷枚数が 2、複写枚数が 3、

データベース読み込み時（データは AAA、BBB、CCC、DDD という形で全 4 件）でのデータベース「全件印刷」を選択した場合の印刷



「印刷枚数が 2」なので全件印刷を 2 回、また「複写枚数が 3」なので各 3 枚ずつ印刷します。

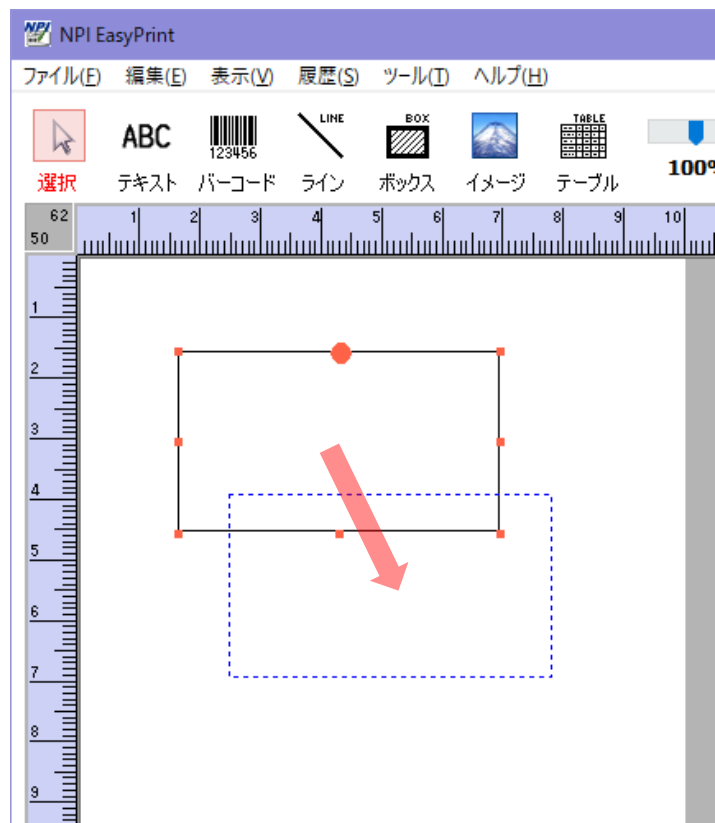
計 24 枚の印刷となります。

データベース印字内容は全件印刷毎に最初のデータに戻りますが、シリアルカウントは複写枚数 3 枚の印刷が終わった時点でカウントアップし、全件印刷 2 回目以降もデータは戻らずに加算されていきます。

4 レイアウトデザイナーの操作

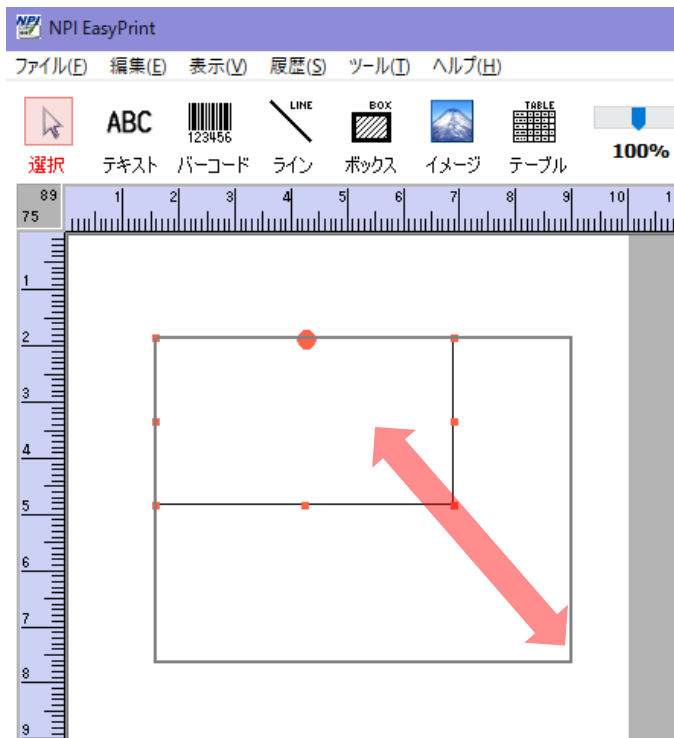
NPI EasyPrint ではレイアウトデザイナーと呼ばれるイメージ生成ツールを備えており、フォーマット作成後に各コンポーネントを追加・編集していくことで印刷券面を作成していきます。

4. 1 コンポーネントの移動



コンポーネント上にマウスを合わせ、クリックすると選択状態となります。
その後マウスをドラッグして移動させます。移動時には点線の青枠が表示されます。
マウスを離れた箇所にコンポーネントが移動します。

4. 2 コンポーネントのサイズ変更（拡大・縮小）



コンポーネント上にマウスを合わせ、クリックすると選択状態となります。

選択状態になるとコンポーネントの四隅、各辺の中央部分に■マークが表示されます。

（コンポーネント上辺のみ●マークの表示）

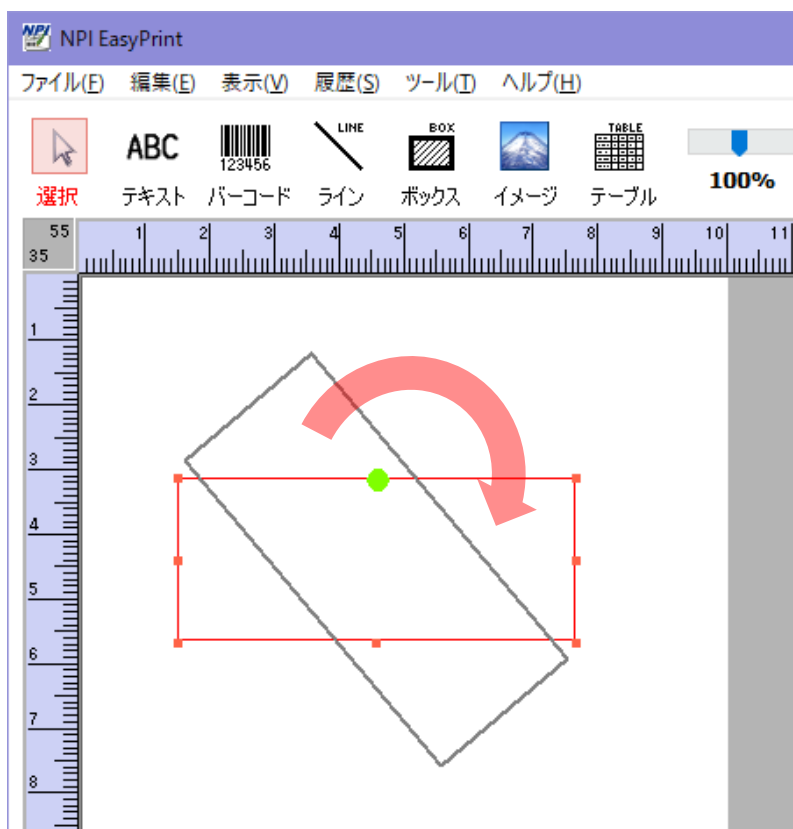
コンポーネントの四隅の■部分にマウスカーソルを合わせ、マウスドラッグして移動させることで拡大・縮小が行えます（縦・横サイズ変更可能）

コンポーネントの上辺●・下辺■は縦方向のみ拡大・縮小ができます。

右辺■・左辺■は横方向のみ拡大・縮小ができます。

拡大・縮小時には実線のグレー枠が表示されます。マウスを離れた箇所で拡大・縮小処理されます。

4. 3 コンポーネントの回転



コンポーネント上辺の●部分をダブルクリックすると赤丸から黄緑色に変わります。
ダブルクリックした後に、そのままマウスを押下したままでマウス移動させると回転させることができます。(ダブルクリックした後にマウスを離してしまうと回転できません)
回転操作時は実線のグレー枠が表示されます。マウスを離れた箇所で回転処理されます。

※移動・サイズ変更・回転については画面右側の情報表示枠、もしくはコンポーネントを右クリックして編集画面から行うこともできます。

4. 4 メニュー操作

4. 4. 1 編集

「編集」からコンポーネントに関する編集を行うことができます。



以下のメニューが表示されます。



① 元に戻す

コンポーネントの追加・編集・削除作業を前の状態に戻すことができます。

② やり直し

「元に戻す」で戻した作業をキャンセルします。

③ 切り取り

コンポーネントを選択した状態で実行すると、コンポーネントを削除し、クリップボード上にコピーします。

④ コピー

コンポーネントを選択した状態で実行すると、クリップボード上にコピーします。

⑤ 貼り付け

クリップボード上にコピーされているコンポーネント情報がある場合、フォーマット上に貼り付けます。

⑥ 削除

コンポーネントを選択した状態で実行すると、フォーマット上から削除できます。

⑦ すべて選択

フォーマット上にあるすべてのコンポーネントを選択状態にします。

⑧ 位置揃え

位置揃えをクリックすると、下記のサブメニューが展開します。

水平方向左に整列(A)
水平方向中央に整列(B)
水平方向右に整列(C)
垂直方向上に整列(D)
垂直方向中央に整列(E)
垂直方向下に整列(F)
水平方向等間隔に整列(G)
垂直方向等間隔に整列(H)

コンポーネントを2つ以上（複数）選択したときに、整列を行うことができます。

⑨ 背面に移動

コンポーネントを選択した状態で実行すると、現在の表示順序を1つマイナスし、背面に移動します。

⑩ 最背面に移動

コンポーネントを選択した状態で実行すると、一番背面に移動します。
(表示順序を1にする)

⑪ 前面に移動

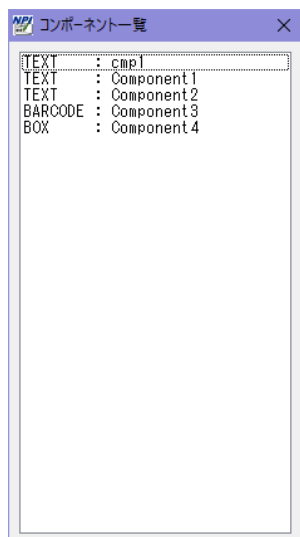
コンポーネントを選択した状態で実行すると、現在の表示順序を1つプラスし、前面に移動します。

⑫ 最前面に移動

コンポーネントを選択した状態で実行すると、一番前面に移動します。
(表示順序を一番高い値にする)

⑬ コンポーネント一覧表示

フォーマットに存在する全コンポーネントを「コンポーネント種別：コンポーネント名」の形で一覧表示します。ダブルクリックすることで詳細画面を開くこともできます。



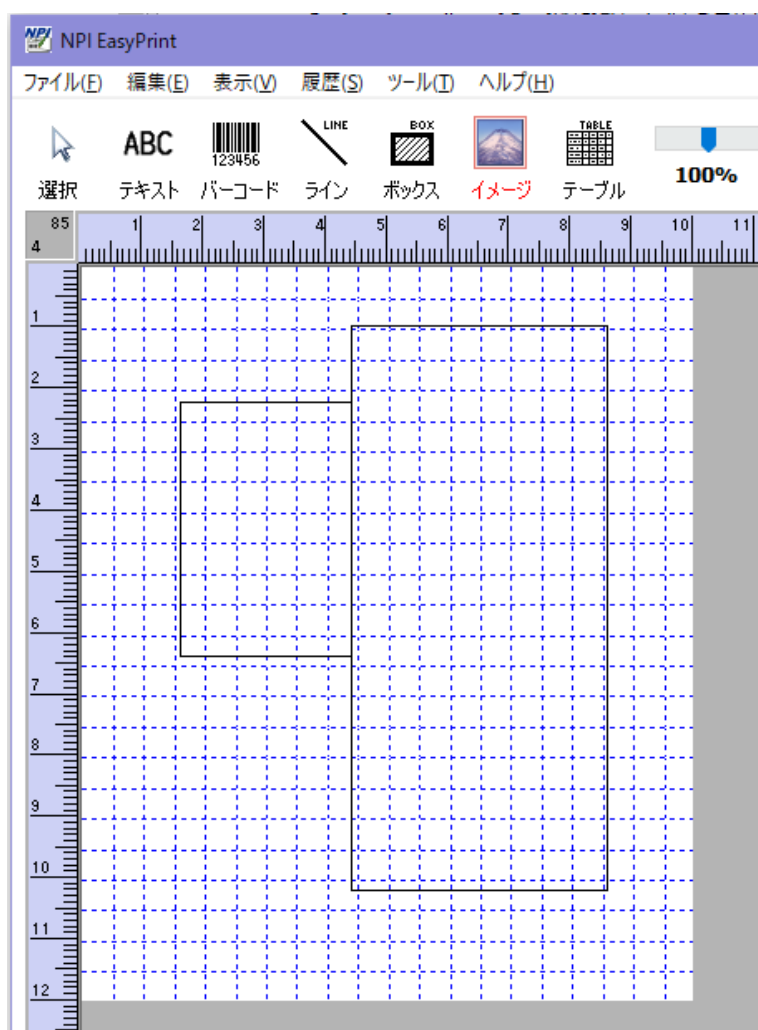
4. 4. 2 表示

「表示」をクリックすることで以下の3つの項目が開きます。



① グリッド表示

クリックすることで5mm 単位で青点線を表示します。再度クリックで非表示になります。



② 境界線表示

クリックすることですべてのコンポーネントの枠線を青で表示します。

表示は枠線の 1 ピクセル外側に表示されます。再度クリックで非表示になります。

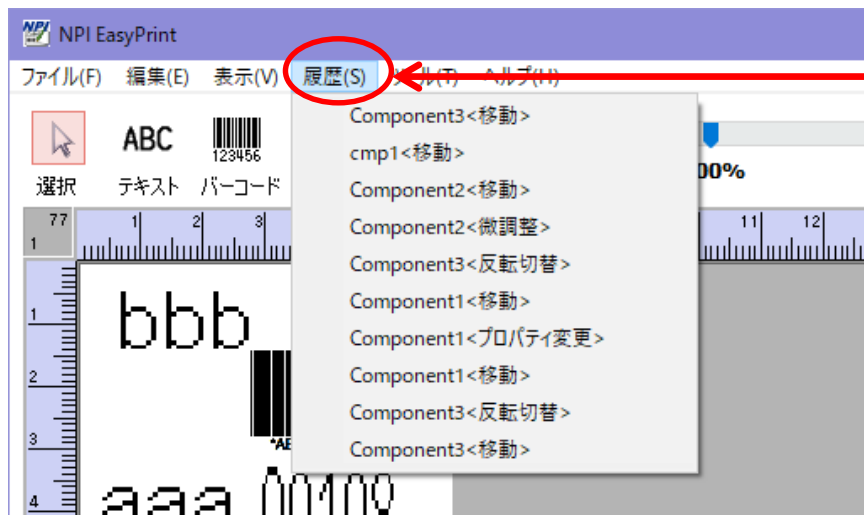
③ 言語設定

「日本語」と「英語」で表示切替ができます。

(一部ファイルオープンダイアログ等については使用されている Windows の言語設定に依存します)

4. 4. 3 履歴

「履歴」をクリックすることでコンポーネント操作における履歴一覧が表示されます。



クリックします。

履歴は 20 パターンまで保持できます。

コンポーネント追加・編集を行うごとに、「コンポーネント名<編集内容>」の組み合わせで履歴が下に追加されていきます。

(一番下が直近の履歴となり、上に行くほど過去の履歴となります)

履歴が 20 パターンを超えた時点で古いものから削除されます。

履歴をクリックすることで過去の編集状況に戻すことができます。

4. 4. 4 ツール

※テンプレート登録機能については今後のアップデートで対応予定です。

現在リリースしているバージョンでは使用できません。

「ツール」をクリックすることで「テンプレート登録」の項目が開きます。



テンプレート登録を選択することで、現在開いているフォーマットをテンプレート登録用に保存を行い、テンプレート登録用アプリケーションが起動します。

Template Register

表示(V)

プリンタ名: PRT001 [プリンタ検索] [OpenPrinter]

ファイル名: D:\Develop\EasyPrint2020\TemplateRegister\Win32\Release\TMP\RegTmp.xml [参照]
※ファイルデータにマルチバイト文字を含む際は文字コードをANSI(Shift-JIS)で保存してください。

XMLテンプレート登録 ☒ 処理終了時に紙送りを行う

保存先: 01:FlashROM

保存番号: 1

印字枚数入力: 0:入力を行わない

印字後保存/通知 ☐

対象コンポーネント: []

保存先: []

通知先: []

[XMLテンプレート登録]

プリンタ名：登録先のプリンタを選択し、OpenPrinter を実行して接続します。

※プリンタの言語は ESC/POS にする必要があります。

ファイル名：登録するテンプレートファイルや CSV ファイル等を選択します。

デフォルトでは EasyPrint で保存したテンプレートへのパスになっています。

処理終了時に紙送りを行う：登録コマンドの最後に紙送りコマンドを発行し、登録処理の完了を紙送り動作で判断します。

「XML テンプレート登録」、「CSV 登録」、「conf 登録」、「ファイル削除」を選択します。
次ページからそれぞれについて説明します。

1. XML テンプレート登録

保存先：FlashROM と USB が選択でき、それぞれ SerialCount のカウントアップ値を印字後に
保存するかを指定できます。

保存番号：保存するテンプレート番号を指定します。

印字枚数入力：印刷時に印字枚数の入力を行うかの選択が出来ます。

印字後保存/通知：印字後保存と通知の有無を指定します。有効にすることで対象コンポーネントに
指定したコンポーネントの値を保存先、通知先へ保存/通知を行います。

対象コンポーネント：保存/通知を行うコンポーネント名を指定します。

複数指定する場合はカンマ区切りで記入します。

保存先：印字後のデータ保存先を指定します。複数指定する場合はカンマ区切りで記入します。

内蔵 FlashROM = E:¥PrintInf¥xxx.csv

USB メモリ = U:¥PrintInf¥xxx.csv

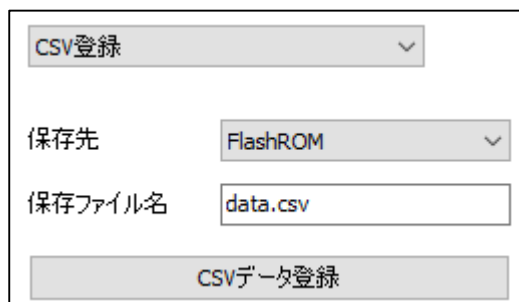
通知先：印字後の通知先を指定します。複数指定する場合はカンマ区切りで記入します。

UDP:192.168.1.10:49513

TCP:192.168.1.10:49513

XML テンプレート登録：登録処理を実行します。

2. CSV 登録



保存先：FlashROM と USB が選択できます。

保存ファイル名：保存後のファイル名を指定します。

CSV データ登録：登録処理を実行します。

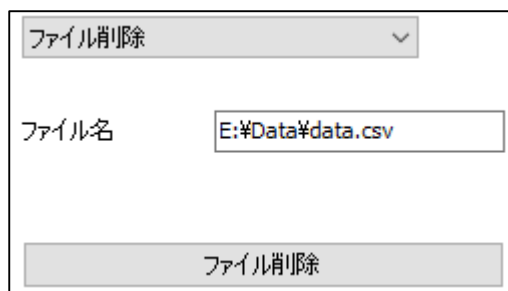
3. conf 登録



保存ファイル名：FlashROM と USB が選択できます。

conf データ登録：登録処理を実行します。

4. ファイル削除

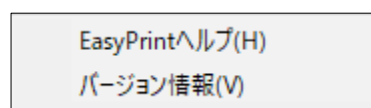


ファイル名：削除するプリンタのファイル名を指定します。。

ファイル削除：削除処理を実行します。

4. 4. 5 ヘルプ

「ヘルプ」をクリックすることで以下の2つの項目が開きます。



① EasyPrint ヘルプ

ヘルプファイル（当 PDF ファイル）を開きます。

② バージョン情報

NPIEasyPrint.exe（当アプリケーション）、使用している DLL（NPIGF.dll、NPrinterLib.dll）のバージョン情報が表示されます。

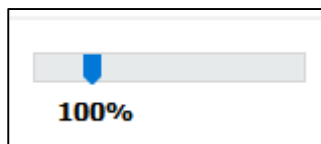


※上記に表示されているバージョンは表示例です。

実際のバージョンとは異なります。

4. 5 アイコンメニュー操作

4. 5. 1 拡大・縮小



フォーマット、およびルーラー（目盛り）表示を 50、75、100、125、150、200、300%で拡大・縮小することができます。表示の変更を行うだけで印刷時の拡大・縮小は行いません。

4. 5. 2 コンポーネント操作アイコン

下記アイコンはコンポーネント操作に使用します。

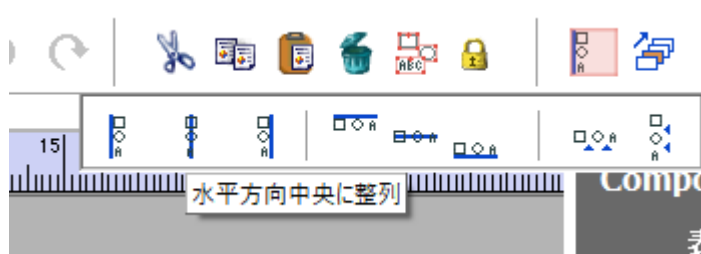


- ① 元に戻す
- ② やり直し
- ③ 切り取り
- ④ コピー
- ⑤ 貼り付け
- ⑥ 削除
- ⑦ すべて選択
- ⑧ ロック

クリックすると、すべてのコンポーネント編集を無効化します。
再度クリックすることで編集が有効化されます。

⑨ 整列

クリックすると各種位置揃えアイコンが開きます。
アイコンにマウスを合わせると、動作内容が表示されます。
コンポーネントを2つ以上選択した状態で実行してください。



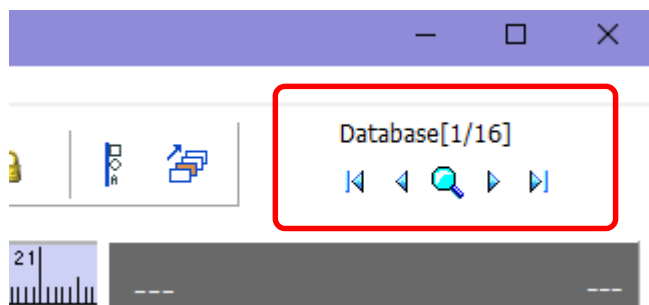
⑩ 順序

クリックするとコンポーネント順序移動アイコンが開きます。
⑨の整列同様、アイコンにマウスを合わせると、動作内容が表示されます。

※⑧のロック以外の動作は「[4. 4. 1 編集](#)」と同様の動作になります。

4. 5. 3 データベース操作

フォーマット作成時に DB 利用を「有」に設定し、データの読み込みに成功すると、画面右上にデータベースの操作アイコンが表示されます。



データベース読み込み時は **Database[x:n]** と表示されます。

CSV ファイル読み込み時は **CSV Data[x:n]** と表示されます。

(x は現在表示しているレコード行、n は総レコード数を表示します)



レコードの先頭行に移動します。



レコード行を 1 つ後ろに移動します。



レコード検索を行います (次ページ参照)



レコード行を 1 つ前に移動します。

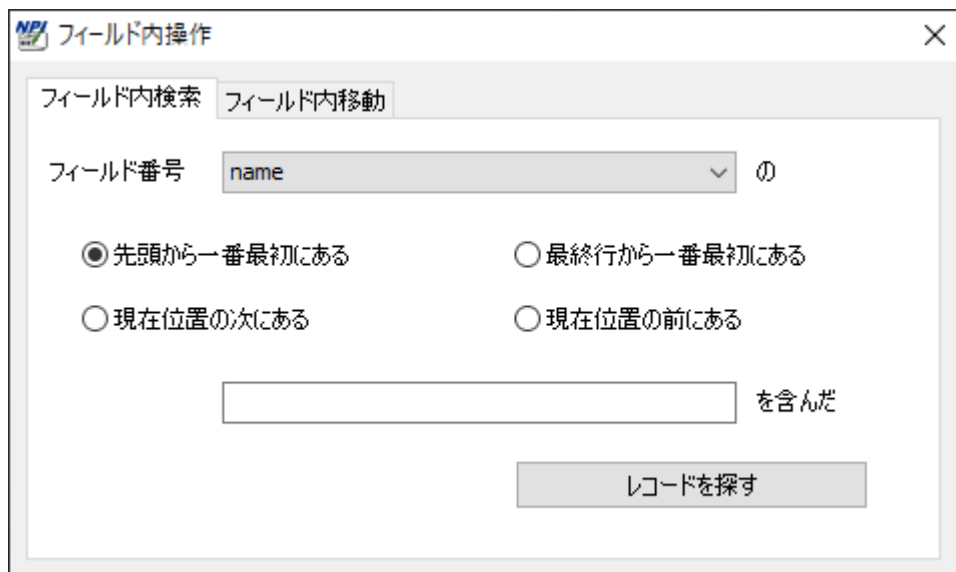


レコードの最終行に移動します。

次ページではレコード検索について説明します。

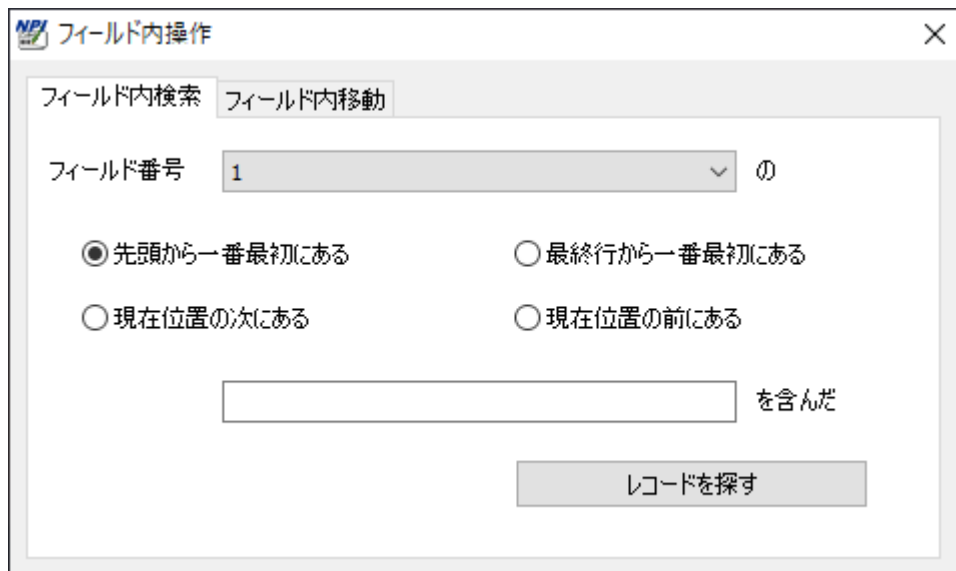
レコード検索アイコンをクリックすると、以下のダイアログが開きます。
フィールド内操作は「フィールド内検索」と「フィールド内移動」の2機能があり、
タブで切り替えます。

・データベース読み込み時（フィールド内検索）



フィールド名にはデータベースから読み込んだ項目名が選択できます。
項目を選択後に、「レコードをどこから検索するか」を4種類から選択し、
最後に項目を検索する文字列を入力し、「レコードを探す」ボタンを押下します。
レコードが検索できると、取得できたレコード行の結果に表示が更新されます。

・CSV ファイル読み込み時（フィールド内検索）



フィールド名は CSV ファイル番号を選択できます。

CSV ファイルが、

ABC,12345,xyz

だった場合は「ABC」が1、「12345」が2、「xyz」が3となります。

検索方法はデータベース読み込み時と同様となります。

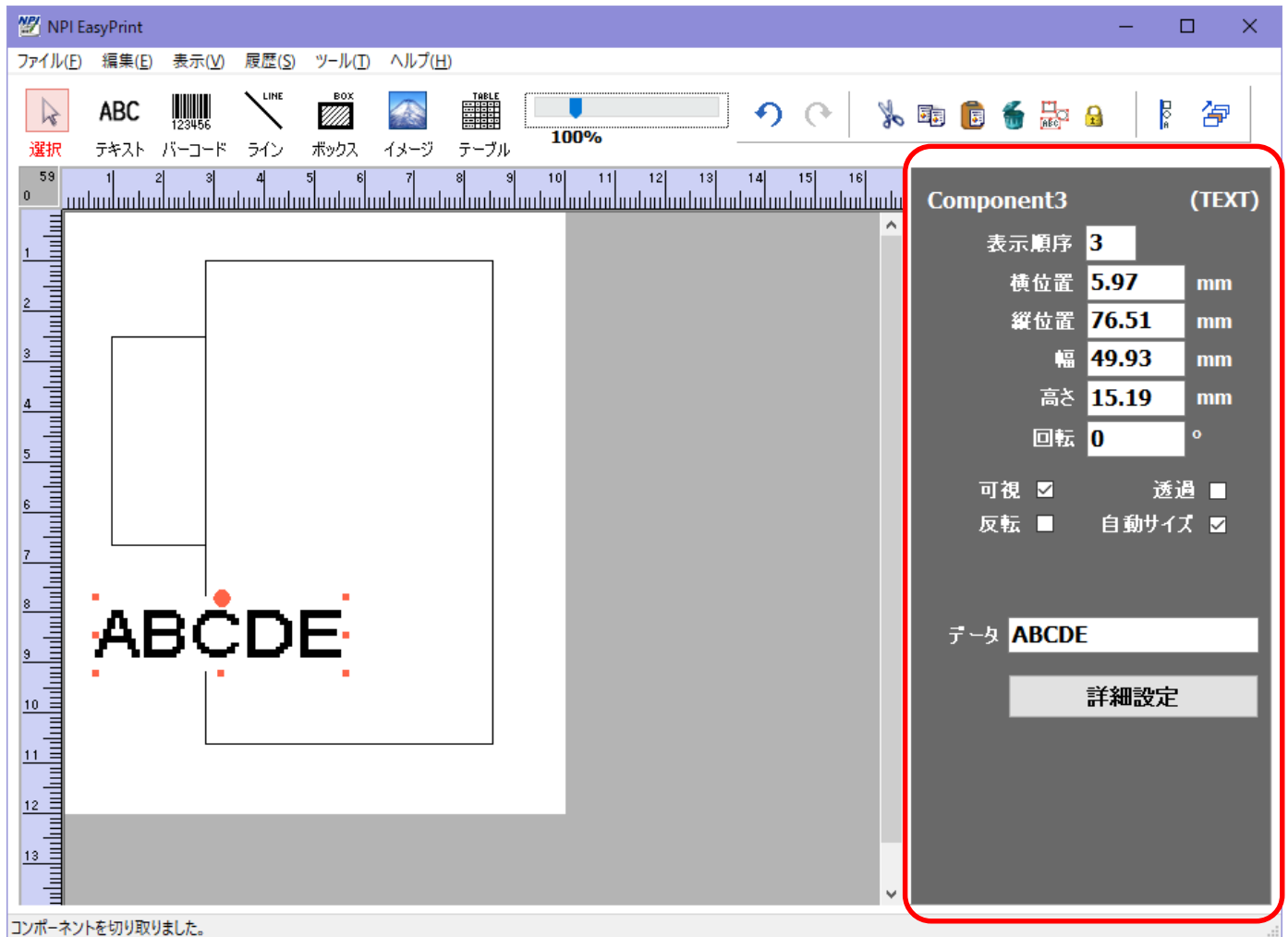
- ・ データベース、CSV ファイル読み込み時共通（フィールド内移動）

The screenshot shows a dialog box titled 'フィールド内操作' (Field Operation) with a close button (X) in the top right corner. Inside the dialog, there are two tabs: 'フィールド内検索' (Field Search) and 'フィールド内移動' (Field Move). The 'フィールド内移動' tab is currently selected. Below the tabs, there is a label 'レコード行' (Record Line) followed by a text input field containing the number '1'. To the right of the input field is a button labeled '移動する' (Move).

レコード行の移動ができます。範囲は 1 からレコード上限数で指定してください。

4. 6 コンポーネント情報表示枠

コンポーネントを選択すると、画面右側の赤枠部分に各種情報が表示されます。



コンポーネントを右クリック、またはダブルクリックすることで編集ダイアログが表示されますが、情報表示枠で直接編集することもできます。

(ただし基本的な項目のみとなります。詳細データの変更はダイアログから編集してください)

5 コンポーネントの説明

レイアウトデザイナーではテキスト、バーコード、ライン、ボックス、イメージ、テーブルの6種類のコンポーネントを用意しています。下記に説明します。

(共通設定項目については「[3. 3 コンポーネント設定](#)」を参照してください)

5. 1 テキストコンポーネント

指定したエリア内に文字を描画します。

- ・ フォントは「TrueType font」が初期設定となっており、プリンター設定で ZPL II (コマンド)・XML を選択された場合のみ Device フォントが利用できます。
- ・ データ種類、フォーマット、増加値については「[6 データ種別・フォーマットについて](#)」を参照してください。
- ・ 配置は「左」「中央」「右」から選択します。コンポーネント枠内の配置箇所として使用します。
- ・ TrueType font 設定ボタンを押下することでフォント設定のダイアログが表示されます。
- ・ 横倍角、縦倍角を指定することでテキスト描画を拡大することができます。
- ・ テキストデータ タブをクリックし、テキストデータを登録します。

テキストプロパティ

コンポーネント名 表示順序

位置 / サイズ

左 上 幅 高さ

余白

左 上 右 下

可視 ☒ 透過 ☐ 反転 ☐

自動サイズ ☒ 入力フラグ 回転 °

フォント設定 | テキストデータ

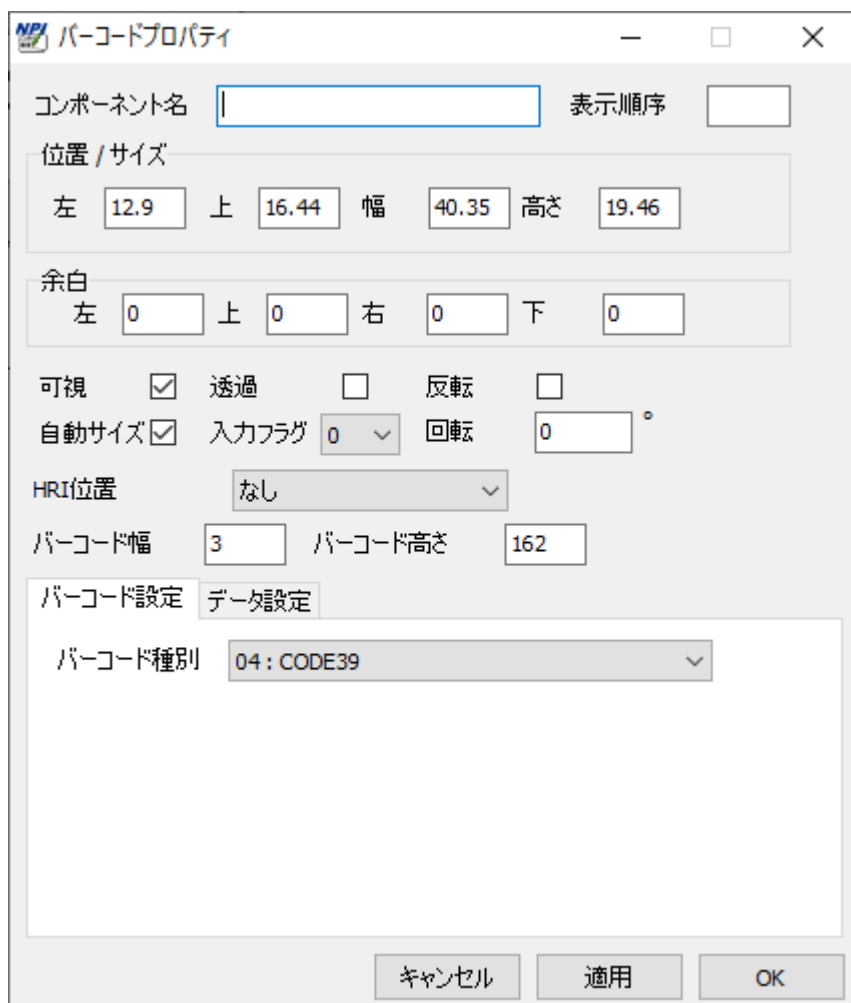
ABCDE|

キャンセル OK

枠内にテキストデータを登録します

5. 2 バーコードコンポーネント

指定したエリア内にバーコードを描画します。



The image shows a Windows-style dialog box titled "バーコードプロパティ" (Barcode Properties). It contains several sections for configuring a barcode component. At the top, there are fields for "コンポーネント名" (Component Name) and "表示順序" (Display Order). Below this is a "位置 / サイズ" (Position / Size) section with input boxes for Left (12.9), Top (16.44), Width (40.35), and Height (19.46). The next section is "余白" (Margin) with input boxes for Left (0), Top (0), Right (0), and Bottom (0). There are checkboxes for "可視" (Visible), "透過" (Transparent), and "反転" (Invert). Below these are checkboxes for "自動サイズ" (Automatic Size) and "入力フラグ" (Input Flag), and a "回転" (Rotation) field set to 0 degrees. The "HRI位置" (HRI Position) is set to "なし" (None). The "バーコード幅" (Barcode Width) is 3 and "バーコード高さ" (Barcode Height) is 162. At the bottom, there are tabs for "バーコード設定" (Barcode Settings) and "データ設定" (Data Settings). Under "バーコード設定", the "バーコード種別" (Barcode Type) is set to "04: CODE39". At the very bottom are "キャンセル" (Cancel), "適用" (Apply), and "OK" buttons.

コンポーネント名 表示順序

位置 / サイズ

左 上 幅 高さ

余白

左 上 右 下

可視 ☒ 透過 ☐ 反転 ☐

自動サイズ ☒ 入力フラグ 回転 °

HRI位置

バーコード幅 バーコード高さ

バーコード設定 データ設定

バーコード種別

キャンセル 適用 OK

- ・ HRI 位置は「なし」と Font A・B・D についてそれぞれ「上」「下」「上下」から選択することができます。
- ・ バーコード幅は 1～20 までの間で指定できます。
- ・ バーコード高さは 1～999 までの間で指定できます。
- ・ バーコード種別、オプションについては「[バーコード一覧/オプション詳細](#)」を参照してください。
オプションのあるバーコードのみ、バーコード種別の下部にオプションが表示されます。

バーコードプロパティ

コンポーネント名 表示順序

位置 / サイズ

左 上 幅 高さ

余白

左 上 右 下

可視 ☒ 透過 ☐ 反転 ☐

自動サイズ ☒ 入力フラグ 回転 °

HRI位置

バーコード幅 バーコード高さ

バーコード設定 データ設定

データ種別

フォーマット

増加値

バーコードデータ

☐ 16進数で入力する

- ・データ種類、フォーマット、増加値については巻末の[データ種別・フォーマットについて](#)を参照してください。
- ・バーコードデータは、バーコードにしたいデータを入力します。
MaxiCode でモードが 2 または 3 だった場合、先頭 15 文字を 1 次メッセージとして処理します。
合成シンボル(Composite)は' | ' を区切りとして扱い、' | ' が見つかるまでを各バーコードの文字列データ、それ以降の文字を PDF417 部分の文字列データとして取り扱います。' | ' 自体はどちらの文字列データにも含みません。
- ・16 進数で入力するを有効にすると、バーコードデータに制御文字等を 16 進数文字列で入力できるようになります。入力中のバーコードデータは自動で変換されます。

5. 2. 1 バーコード一覧 / オプション詳細
対応バーコード一覧

ID	名称	文字数	文字の種類	備考
0	UPC-A	12 15 (EAN-2) 18 (EAN-5)	数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ 10 桁以下の場合、11 桁になるまで上位ビットを 0 で埋めて 12 桁目にチェックデジットを結合します。 ・ チェックデジットの計算値と 12 桁目の数値が異なる場合は計算値を優先させます。 ・ + 以降の文字をアドオンコードとして 9 本分の余白を空けて生成します。 ・ + 以降が 2 桁以下なら EAN-2、3 桁以上 5 桁以下なら EAN-5 のアドオンコードを生成します。
1	UPC-E	8 11 (EAN-2) 14 (EAN-5) 12 (UPC-A)	数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ 6 桁以下の場合、7 桁になるまで上位ビットを 0 で埋めて 8 桁目にチェックデジットを結合します。 ・ チェックデジットの計算値と 8 桁目の数値が異なる場合は計算値を優先させます。 ・ + 以降の文字をアドオンコードとして 9 本分の余白を空けて生成します。 ・ + 以降が 2 桁以下なら EAN-2、3 桁以上 5 桁以下なら EAN-5 のアドオンコードを生成します。 ・ + を含まない 11 又は 12 桁の入力であった場合はデータを 8 桁に圧縮して生成します。 ・ アドオンコードは 8 桁以下で UPC-E データを入力した時のみ有効です。
2	JAN-13 (EAN-13)	13 16 (EAN-2) 19 (EAN-5)	数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ 11 桁以下の場合、12 桁になるまで上位ビットを 0 で埋めて 13 桁目にチェックデジットを結合します。 ・ チェックデジットの計算値と 13 桁目の数値が異なる場合は計算値を優先させます。 ・ + 以降の文字をアドオンコードとして 9 本分の余白を空けて生成します。 ・ + 以降が 2 桁以下なら EAN-2、3 桁以上 5 桁以下なら EAN-5 のアドオンコードを生成します。
3	JAN-8 (EAN-8)	8 11 (EAN-2) 14 (EAN-5)	数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ 6 桁以下の場合、7 桁になるまで上位ビットを 0 で埋めて 8 桁目にチェックデジットを結合します。 ・ チェックデジットの計算値と 8 桁目の数値が異なる場合は計算値を優先させます。 ・ + 以降の文字をアドオンコードとして 9 本分の余白を空けて生成します。 ・ + 以降が 2 桁以下なら EAN-2、3 桁以上 5 桁以下なら EAN-5 のアドオンコードを生成します。
4	CODE39	可変長	数字 (0~9) 英字 (A~Z) 記号 (\$%*+-. /スペース)	<ul style="list-style-type: none"> ・ Start/Stop Code の “*” を必ず入力してください。 ・ 先頭と末尾以外に “*” を入力すると生成されません。 ・ チェックディジットは自動付与されません

			Start/Stop Code (*)	・桁数上限は Start/Stop Code を含めて 76 文字です。
5	ITF	偶 数	数字 : 0~9	・文字数が奇数の場合は先頭に 0 を追加します。 ・桁数上限は 89 文字です。
6	CODABAR (NW-7)	可変長	数字 (0~9) 記号 (\$+-. /:) Start/Stop Code (ABCD) (abcd)	・Start/Stop Code が入力されていない場合は A を Start/Stop Code としてバーコードを生成します。 ・桁数上限は 60 文字です。
7	CODE128	CODE128 仕様詳細 をご参照願います		
8	CODE93	可変長	ASCII 文字	・HRI 文字の前後に “□” を自動付加します。 ・制御文字の HRI 文字は “■” と英字 1 文字を組み合わせで印字します。*1 ・桁数上限は 107 文字です。
9	ISBN コード	13	数字 (0~9) 記号 (Xx)	・9 桁以下の場合は末尾にチェックデジットを結合し、接頭記号は 978 を使用します。 ・10 桁又は 13 桁で計算値とチェックデジットの値が異なる場合は計算値を優先させます。 ・13 桁の場合、接頭記号は 979 と 978 を使えます。
10	Industrial 2 of 5	可変長	数字 (0~9)	・桁数上限は 40 文字です。
11	Matrix 2 of 5	可変長	数字 (0~9)	・桁数上限は 80 文字です。
12	NEC Matrix	可変長	数字 (0~9)	・桁数上限は 80 文字です。
13	カスタマ バーコード	20	数字 (0~9) 英字 (A~Z) 英字 (a~z) 記号 (-)	・20 桁未満の場合も生成可能です。
14	MSI Code	可変長	数字 (0~9)	・オプションによって算出するチェックデジットの種類が変わります。 ・チェックデジットが無しの場合と mod11 の場合は 55 文字まで入力できます。それ以外は 18 文字までとなります。
15	GS1-128 (EAN-128)	可変長	DEL 以外の制御文字を除いた ASCII コード文字	・GS1 規格に沿ったデータでなかった場合バーコードを生成できません。 ・AI 文字は” []” で囲んで渡します。 ・HRI 文字の AI 文字は” ()” で囲まれます。
16	GS1 DataBar Omnidirectional	14	数字 (0~9)	・AI 文字は 01 で固定のため AI 文字を除いた 14 桁までのデータを渡します。 ・14 桁目のチェックデジットは自動で計算されます。14 桁の入力を行った場合 14 桁目は処理されません。 ・12 桁以下だった場合は上位ビットを 0 で埋めて処理を行います。
17	GS1 DataBar Truncated	14	数字 (0~9)	・バーの高さが低くなった GS1 DataBar Omnidirectional です。処理内容は GS1 DataBar Omnidirectional と同様です。
18	GS1 DataBar Limited	14	数字 (0~9)	・AI 文字は 01 で固定のため AI 文字を除いた 14 桁までのデータを渡します。

				<ul style="list-style-type: none"> ・ 14 桁目のチェックデジットは自動で計算されます。14 桁の入力を行った場合 14 桁目は処理されません。 ・ 12 桁以下だった場合は上位ビットを 0 で埋めて処理を行います。 ・ 13 桁以上の場合、1 桁目は 0 か 1 である必要があります。どちらでもない場合生成されません。
19	GS1 DataBar Expanded	可変長	<p>1 つ目の AI 文字が[01]だった場合：その AI 文字のデータのみ数字（0～9）</p> <p>1 つ目の AI 文字が[01]じゃなかった場合：数字（0～9）</p> <p>英字（A～Z）</p> <p>英字（a～z）</p> <p>記 号</p> <p>(! "% & ' () * + , - . / : ; < > = ? _</p> <p>スペース [])</p>	<ul style="list-style-type: none"> ・ GS1 規格に沿ったデータでなかった場合バーコードを生成できません。 ・ AI 文字は” [] ” で囲んで渡します。 ・ HRI 文字の AI 文字は” () ” で囲まれます。
20	CODE39 (Option specified)	可変長	<p>数字（0～9）</p> <p>英字（A～Z）</p> <p>記号（\$%*+-. / スペース）</p> <p>Start/Stop Code（*）</p>	<ul style="list-style-type: none"> ・ Start/Stop Code の自動入力オプションを無効にしている場合は “*” を必ず入力してください。 ・ 先頭と末尾以外に “*” を入力すると生成されません。 ・ 自動チェックデジットオプションを有効にするとストップコードの手前に自動付与されます ・ 桁数上限は Start/Stop Code を含めて 76 文字です。
21	CODE128 (Option specified)	CODE128 仕様詳細 をご参照願います		
22	CODE93 (Option specified)	可変長	ASCII 文字	<ul style="list-style-type: none"> ・ オプションで HRI 文字の前後に “□” を付加するか選択できます。 ・ 制御文字の HRI 文字は “■” と英字 1 文字を組み合わせで印字します。*1 ・ 桁数上限は 107 文字です。
30	QR コード	可変長	シフト JIS コード文字	<ul style="list-style-type: none"> ・ オプションで誤り訂正レベルとバージョンとマスクパターンの指定ができます。
31	マイクロ QR コード	可変長	シフト JIS コード文字	<ul style="list-style-type: none"> ・ オプションで誤り訂正レベルとバージョンの指定ができます。 ・ 桁数上限はバージョン 4 の誤り訂正レベル L の時に 35 桁の数字、21 文字の英数字、15 バイトの文字となります。
32	Aztec Code	可変長	ASCII 文字	<ul style="list-style-type: none"> ・ オプションで誤り訂正レベルとサイズ指定とリーダー初期化データの付与とサイズの自動選択時に compact Aztec Code を生成するかを指定できます。 ・ 誤り訂正レベルとサイズ指定は同時に指定できません。その場合サイズ指定が優先されます。
33	PDF417	可変長	シフト JIS コード文字	<ul style="list-style-type: none"> ・ オプションでチェックデジットの数とデータ部の列数指定とリーダー初期化データの有無と高さ比を指定できます。

				<ul style="list-style-type: none"> ・バーコードリーダーによっては読み取り結果の先頭に文字コード情報が表示される場合があります。 <p>リーダーの設定で先頭 2 文字を削除する等で対応ください。</p>
34	MicroPDF417	可変長	シフト JIS コード文字	<ul style="list-style-type: none"> ・オプションでデータ部の列数指定とリーダー初期化データの有無を選択できます。 ・桁数上限は数字のみの入力で最大 245 文字です。入力文字により桁数上限は減少します。 ・バーコードリーダーによっては読み取り結果の先頭に文字コード情報が表示される場合があります。 <p>リーダーの設定で先頭 2 文字を削除する等で対応ください。</p>
35	MaxiCode	縦 33 横 30	<p>1 次メッセージ：数字（0～9）スペース</p> <p>2 次メッセージ：ASCII 文字 （ASCII 文字以外の文字があってもバーコード化しますが、読み取りを保証できません）</p>	<ul style="list-style-type: none"> ・mode2、mode3 の場合は先頭 15 桁を 1 次メッセージとして扱います。 ・1 次メッセージが 15 桁に満たない場合は先頭をスペースで埋めて渡す必要があります。 <p>例” 123456”</p>
36	DataMatrix	可変長	シフト JIS コード文字	<ul style="list-style-type: none"> ・オプションでリーダー初期化データの付与、生成サイズの指定が行えます。
37	GS1 DataBar Stacked	14	数字（0～9）	<ul style="list-style-type: none"> ・AI 文字は 01 で固定のため AI 文字を除いた 14 桁までのデータを渡します。 ・14 桁目のチェックデジットは自動で計算されます。14 桁の入力を行った場合 14 桁目は処理されません。 ・12 桁以下だった場合は上位ビットを 0 で埋めて処理を行います。
38	GS1 DataBar Stacked Omnidirectional	14	数字（0～9）	<ul style="list-style-type: none"> ・AI 文字は 01 で固定のため AI 文字を除いた 14 桁までのデータを渡します。 ・14 桁目のチェックデジットは自動で計算されます。14 桁の入力を行った場合 14 桁目は処理されません。 ・12 桁以下だった場合は上位ビットを 0 で埋めて処理を行います。
39	GS1 DataBar Expanded Stacked	可変長	<p>1 つ目の AI 文字が[01]だった場合：その AI 文字のデータのみ数字（0～9）記号（[]）</p> <p>1 つ目の AI 文字が[01]じゃなかった場合：数字（0～9） 英字（A～Z） 英字（a～z） 記号</p>	<ul style="list-style-type: none"> ・GS1 規格に沿ったデータでなかった場合バーコードを生成できません。 ・AI 文字は” []” で囲んで渡します。 ・オプションで列数を 1～10 の間で指定できます。指定がなかった場合は 2 列で生成します。 ・AI 文字が[01]から始まる場合、[01]を含めて 2 つ以上の AI 文字が含まれていないと読み取りを保証できません。

			(!'"%&' ()*+, -./:;<>=?_ スペース[])	
40	EAN-8 Composite	可変長	MicroPDF417 部分: 数字 (0 ~9) 英字 (A~Z) 英字 (a~z) 記 号 (!'"%&' ()*+, -./:;<>=?_ スペース[]) EAN8 部分: 数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ EAN-8 部分の処理内容は上記 EAN-8 と同じです。
41	EAN-13 Composite	可変長	MicroPDF417 部分: 数字 (0 ~9) 英字 (A~Z) 英字 (a~z) 記 号 (!'"%&' ()*+, -./:;<>=?_ スペース[]) EAN13 部分: 数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ EAN-13 部分の処理内容は上記 EAN-13 と同じです。
42	UPC-A Composite	可変長	MicroPDF417 部分: 数字 (0 ~9) 英字 (A~Z) 英字 (a~z) 記 号 (!'"%&' ()*+, -./:;<>=?_ スペース[]) UPC-A 部分: 数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ UPC-E 部分の処理内容は上記 UPC-A と同じです。
43	UPC-E Composite	可変長	MicroPDF417 部分: 数字 (0 ~9) 英字 (A~Z) 英字 (a~z) 記 号 (!'"%&' ()*+, -./:;<>=?_ スペース[]) UPC-E 部分: 数字 (0~9) 記号 (+)	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ UPC-E 部分の処理内容は上記 UPC-E と同じです。
44	GS1-128 Composite	可変長	PDF417 部分: 数字 (0~9) 英字 (A~Z)	<ul style="list-style-type: none"> ・ MicroPDF417 又は PDF417 (CC-C) を付加させた合成シンボルを生成します。

			<p>英字 (a~z)</p> <p>記 号</p> <p>(!'"%&' () *+, -. /:;<>=?_ スペース[])</p> <p>GS1-128 部分:制御文字を除いた ASCII 文字</p>	<p>・PDF417 部分のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。</p> <p>・オプションで合成シンボルのタイプを CC-A, CC-B, CC-C から選択できます。指定がない場合 CC-A が指定されます。</p> <p>・GS1-128 部分の処理内容は上記 GS1-128 と同じです。</p>
45	GS1 DataBar Omnidirectional Composite	可変長	<p>MicroPDF417 部分:数字(0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!'"%&' () *+, -. /:;<>=?_ スペース[])</p> <p>DataBar 部分:数字(0~9)</p>	<p>・MicroPDF417 を付加させた合成シンボルを生成します。</p> <p>・MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。</p> <p>・オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。</p> <p>・GS1 DataBar 部分の処理内容は上記 GS1 DataBar Omnidirectional と同じです。</p>
46	GS1 DataBar Truncated Composite	可変長	<p>MicroPDF417 部分:数字(0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!'"%&' () *+, -. /:;<>=?_ スペース[])</p> <p>DataBar 部分:数字(0~9)</p>	<p>・MicroPDF417 を付加させた合成シンボルを生成します。</p> <p>・MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。</p> <p>・オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。</p> <p>・GS1 DataBar 部分の処理内容は上記 GS1 DataBar Truncated と同じです。</p>
47	GS1 DataBar Limited Composite	可変長	<p>MicroPDF417 部分:数字(0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!'"%&' () *+, -. /:;<>=?_ スペース[])</p> <p>DataBar 部分:数字(0~9)</p>	<p>・MicroPDF417 を付加させた合成シンボルを生成します。</p> <p>・MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。</p> <p>・オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。</p> <p>・GS1 DataBar 部分の処理内容は上記 GS1 DataBar Limited と同じです。</p>
48	GS1 DataBar Expanded Composite	可変長	<p>MicroPDF417 部分:数字(0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!'"%&' () *+, -. /:;<>=?_ スペース[])</p>	<p>・MicroPDF417 を付加させた合成シンボルを生成します。</p> <p>・MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。</p> <p>・オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。</p> <p>・GS1 DataBar 部分の処理内容は上記 GS1 DataBar Expanded と同じです。</p>

			<p>DataBar 部分:1 つ目の AI 文字が[01]だった場合その AI 文字のデータのみ数字 (0~9)</p> <p>1 つ目の AI 文字が[01]じゃなかった場合: 数字 (0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!"%&'()*+,-./:;<>=?_スペース)</p>	
49	GS1 DataBar Stacked Composite	可変長	<p>MicroPDF417 部分: 数字 (0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!"%&'()*+,-./:;<>=?_スペース[])</p> <p>DataBar 部分: 数字 (0~9)</p>	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ GS1 DataBar 部分の処理内容は上記 GS1 DataBar Stacked と同じです。
50	GS1 DataBar Stacked Omnidirectional Composite	可変長	<p>MicroPDF417 部分: 数字 (0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!"%&'()*+,-./:;<>=?_スペース[])</p> <p>DataBar 部分: 数字 (0~9)</p>	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A, CC-B から選択できます。指定がない場合 CC-A が指定されます。 ・ GS1 DataBar 部分の処理内容は上記 GS1 DataBar Stacked Omnidirectional と同じです。
51	GS1 DataBar Expanded Stacked Composite	可変長	<p>MicroPDF417 部分: 数字 (0~9)</p> <p>英字 (A~Z)</p> <p>英字 (a~z)</p> <p>記 号</p> <p>(!"%&'()*+,-./:;<>=?_スペース[])</p> <p>DataBar 部分:</p> <p>1 つ目の AI 文字が[01]だった場合: その AI 文字のデータのみ数字 (0~9)</p>	<ul style="list-style-type: none"> ・ MicroPDF417 を付加させた合成シンボルを生成します。 ・ MicroPDF417 のデータは GS1 規格に沿ったデータで AI 文字を” []” で囲んで渡します。 ・ オプションで合成シンボルのタイプを CC-A と CC-B から選択できます。指定がない場合 CC-A になります。 ・ オプションで列数を 1~10 の間で指定できます。指定がなかった場合は 2 列で生成します。 ・ GS1 DataBar 部分の処理内容は上記 GS1 DataBar Expanded Stacked と同じです。

			1 つ目の AI 文字が[01]じゃなかった場合：数字（0～9） 英字（A～Z） 英字（a～z） 記号 (!"#\$%&'()*+,-./:;<=>?_スペース)	
52	GS1 QRコード	可変長	DEL 以外の制御文字を除いた ASCII 文字	・ AI 文字は” []” で囲んで渡します。 ・ GS1 規格に沿ったデータでなかった場合バーコードを生成できません。
53	GS1 Aztec Code	可変長	DEL 以外の制御文字を除いた ASCII 文字	・ AI 文字は” []” で囲んで渡します。 ・ GS1 規格に沿ったデータでなかった場合バーコードを生成できません。
54	GS1 DataMatrix	可変長	DEL 以外の制御文字を除いた ASCII 文字	・ AI 文字は” []” で囲んで渡します。 ・ GS1 規格に沿ったデータでなかった場合バーコードを生成できません。

*1：CODE93 制御文字の HRI 文字表記

ASCII	16 進	HRI 文字	ASCII	16 進	HRI 文字
NUL	00	■U	DLE	10	■P
SOH	01	■A	DC1	11	■Q
STX	02	■B	DC2	12	■R
ETX	03	■C	DC3	13	■S
EOT	04	■D	DC4	14	■T
ENQ	05	■E	NAK	15	■U
ACK	06	■F	SYN	16	■V
BEL	07	■G	ETB	17	■W
BS	08	■H	CAN	18	■X
HT	09	■I	EM	19	■Y
LF	0A	■J	SUB	1A	■Z
VT	0B	■K	ESC	1B	■A
FF	0C	■L	FS	1C	■B
CR	0D	■M	GS	1D	■C
SO	0E	■N	RS	1E	■D
SI	0F	■O	US	1F	■E
			DEL	7F	■T

オプション一覧

記載されていない値を設定した場合はオプションが無視されます。

1次元バーコードオプション

ID	バーコード種別	オプション名	動作
14	MSI Code	チェックディジットの有無切替	mod 無し, mod10, mod1010, mod11, mod1110
20	Code39	チェックディジットの自動入力	モジュラス 43 のチェックディジットの自動付与切替
		Start/Stop Code の自動入力	データ前後への*自動付与切替
21	Code128	特殊キャラクターの手動入力	CODE128 の仕様詳細参照
22	Code93	HRI 文字の前後に□を付与する	HRI 文字前後の□付与切替

2次元バーコードオプション

ID	バーコード種別	オプション名	動作
30	QR コード	誤り訂正レベル	誤り訂正レベルの指定
		バージョン	バージョンの指定 21×21～177×177 のモジュールを指定します。
		マスクパターン	マスクパターンの指定
31	マイクロ QR コード	誤り訂正レベル	誤り訂正レベルの指定
		バージョン	バージョンの指定 11×11～17×17 のモジュールを指定します。
32	Aztec Code	誤り訂正レベル	誤り訂正レベルの指定
		バージョン	バージョンの指定 1～4 は 15×15～27×27 モジュールの Compact Aztec Code を指定します。 5～36 は 19×19～151×151 モジュールの Aztec Code を指定します。
		サイズ自動時に Compact Aztec を生成しない	バージョンが自動の際に Compact Aztec を生成しないようになります。
		リーダー初期化データの付与	リーダー初期化データを付与します。
33	PDF417	誤り訂正レベル	チェックデジット数の指定
		列数	データ部の列数指定
		高さ比指定	1 モジュールの幅に対する高さ比を指定
		リーダー初期化データの付与	リーダー初期化データを付与します。

34	Micro PDF417	列数	データ部の列数指定
		リーダー初期化データの付与	リーダー初期化データを付与します。
35	MaxiCode	モード	<p>モードの選択</p> <p>2 = 1 次メッセージを 9 桁の郵便番号として扱う</p> <p>3 = 1 次メッセージを 6 桁の郵便番号として扱う</p> <p>4 = 誤り訂正レベル (1 次メッセージ EEC, 2 次メッセージ SEC)</p> <p>5 = 誤り訂正レベル (全文 EEC)</p> <p>6 = 誤り訂正レベル (全文 SEC)</p> <p>※モード 2 と 3 はデータの先頭 15 文字を 1 次メッセージとして扱います。</p>
36	DataMatrix	サイズ	サイズ指定
		サイズ自動時に必ず正方形で生成する	長方形の DataMatrix が生成されなくなります。
		リーダー初期化データの付与	リーダー初期化データを付与します。
39	GS1 DataBar Expanded Stacked	列数	データ部の列数の指定
40	EAN-8 Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
41	EAN13 Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
42	UPC-A Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
43	UPC-E Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
44	GS1-128 Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
45	GS1 DataBar Omnidirectional Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
46	GS1 DataBar Truncated Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
47	GS1 DataBar Limited Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
48	GS1 DataBar Expanded Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
49	GS1 DataBar Stacked Composite	コンポーネントタイプ	コンポジットコンポーネントの種類

50	GS1 DataBar Stacked Omnidirectional Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
51	GS1 DataBar Expanded Stacked Composite	コンポーネントタイプ	コンポジットコンポーネントの種類
52	GS1 QR コード	誤り訂正レベル	誤り訂正レベルの指定
		バージョン	バージョンの指定 21×21～177×177 のモジュールを指定します。
		マスクパターン	マスクパターンの指定
53	GS1 Aztec Code	誤り訂正レベル	誤り訂正レベルの指定
		バージョン	バージョンの指定 1～4 は 15×15～27×27 モジュールの Compact Aztec Code を指定します。 5～36 は 19×19～151×151 モジュールの Aztec Code を指定します。
		サイズ自動時に Compact Aztec を生成しない	バージョンが自動の際に Compact Aztec を生成しないようになります。
54	GS1 DataMatrix	サイズ	サイズ指定
		サイズ自動時に必ず正方形で生成する	長方形の DataMatrix が生成されなくなります。

[CODE128 の仕様詳細]

1. 概要

ASCII 128 文字（数字、アルファベット大文字／小文字、記号、制御文字）と 1 つのバーパターンで 2 桁の数字を表すことができます。文字数は可変長でストップコード、チェックデジットは自動付加されます。

2. 文字の種類

- ・コードセット A：〔00〕 h～〔5F〕 h の ASCII 文字を表すことができます。
- ・コードセット B：〔20〕 h～〔7F〕 h の ASCII 文字を表すことができます。
- ・コードセット C：数字 2 桁（00～99）を 1 つのバーパターンで表すことができます。

・特殊キャラクター

オプションで特殊キャラクターの手動入力を有効にしなかった場合はバーコードデータを元に自動で下記特殊キャラクターを追加します。

- ①スタートコード/コード選択キャラクター（CODE A、CODE B、CODE C）
バーコードデータの先頭は、必ずこのキャラクターとしてください。
コードセットをデータの途中で切り替えることができます。
- ②シフトキャラクター（SHIFT）
コードセット A：SHIFT 直後の 1 文字をコードセット B として扱います。
コードセット B：SHIFT 直後の 1 文字をコードセット A として扱います。
コードセット C：使用不可
- ③ファンクションキャラクター（FNC1、FNC2、FNC3、FNC4）
用途はアプリケーションによります。
コードセット C は、“FNC1” のみ使用可能です。

3. 詳細仕様

- ・制御文字/特殊キャラクターの HRI 文字仕様を以下に示します。
 - ①制御文字（〔00〕 h～〔1F〕 h、〔7F〕 h）
スペースを表示します。
 - ②スタートコード/コード選択キャラクター（CODE A、CODE B、CODE C）
印字しません。
 - ③シフトキャラクター（SHIFT）
印字しません。
 - ④ファンクションキャラクター（FNC1、FNC2、FNC3、FNC4）
スペースを表示します。

以下特殊キャラクターの手動入力時の仕様

- ・バーコードデータの先頭がスタートコードでなかった場合、処理を中止します。
- ・“{” とその直後の組合せが特殊キャラクターに該当しないときは、処理を中止します。
- ・“{” を文字として使用するときは、“{” と入力してください。
- ・選択されているコードセットで使用できない文字があるときは、印字はしますがスキャナー等での読取りはできません。
- ・特殊キャラクター手動入力時のコードセット C の数値入力は、2 桁を 1 つの 16 進数として入力します。
（詳細はコードセット表参照）
自動入力の場合は文字列（0x30 ～ 0x39）として 1 桁ずつ渡します。

4. キャラクターセット表

特殊キャラクター

	ASCII	16 進
CODE A	{A	〔7B〕 h+ 〔41〕 h
CODE B	{B	〔7B〕 h+ 〔42〕 h
CODE C	{C	〔7B〕 h+ 〔43〕 h
SHIFT	{S	〔7B〕 h+ 〔53〕 h
FNC 1	{1	〔7B〕 h+ 〔31〕 h
FNC 2	{2	〔7B〕 h+ 〔32〕 h
FNC 3	{3	〔7B〕 h+ 〔33〕 h
FNC 4	{4	〔7B〕 h+ 〔34〕 h

[各コードセット使用可能文字一覧]

コードセットA

HEX	0	1	2	3	4	5
0	NUL	DLE	SP	0	@	P
1	SOH	DC1	!	1	A	Q
2	STX	DC2	“	2	B	R
3	ETX	DC3	#	3	C	S
4	EOT	DC4	\$	4	D	T
5	ENQ	NAK	%	5	E	U
6	ACK	SYN	&	6	F	V
7	BEL	ETB	‘	7	G	W
8	BS	CAN	(8	H	X
9	HT	EM)	9	I	Y
A	LF	SUB	*	:	J	Z
B	VT	ESC	+	;	K	[
C	FF	FS	,	<	L	¥
D	CR	GS	-	=	M]
E	SO	RS	.	>	N	^
F	SI	US	/	?	O	_

コードセットB

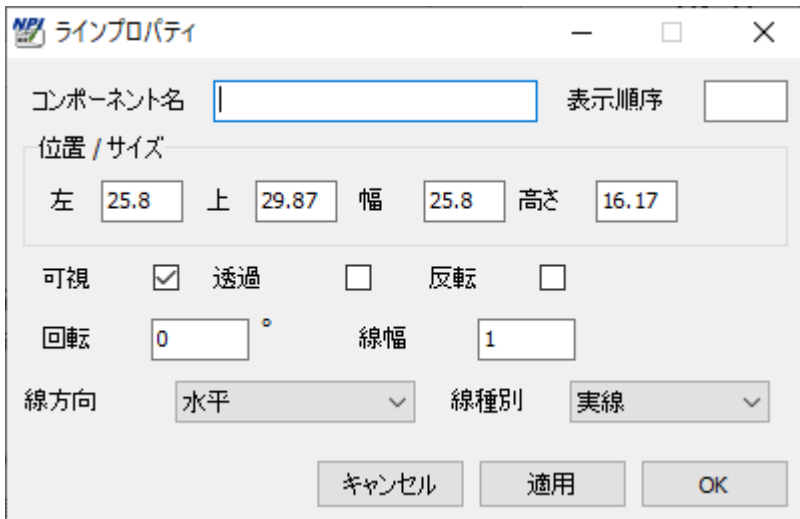
HEX	2	3	4	5	6	7
0	SP	0	@	P	`	p
1	!	1	A	Q	a	q
2	“	2	B	R	b	r
3	#	3	C	S	c	s
4	\$	4	D	T	d	t
5	%	5	E	U	e	u
6	&	6	F	V	f	v
7	‘	7	G	W	g	w
8	(8	H	X	h	x
9)	9	I	Y	i	y
A	*	:	J	Z	j	z
B	+	;	K	[k	{
C	,	<	L	¥	l	
D	-	=	M]	m	}
E	.	>	N	^	n	~
F	/	?	O	_	o	DEL

コードセットC

HEX	0	1	2	3	4	5	6
0	00	16	32	48	64	80	96
1	01	17	33	49	65	81	97
2	02	18	34	50	66	82	98
3	03	19	35	51	67	83	99
4	04	20	36	52	68	84	
5	05	21	37	53	69	85	
6	06	22	38	54	70	86	
7	07	23	39	55	71	87	
8	08	24	40	56	72	88	
9	09	25	41	57	73	89	
A	10	26	42	58	74	90	
B	11	27	43	59	75	91	
C	12	28	44	60	76	92	
D	13	29	45	61	77	93	
E	14	30	46	62	78	94	
F	15	31	47	63	79	95	

5. 3 ラインコンポーネント

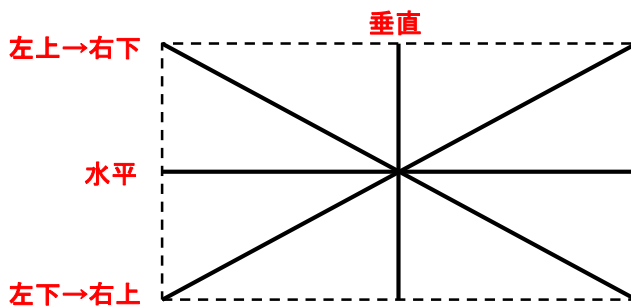
指定したエリア内に線を描画します。



The dialog box titled 'ラインプロパティ' (Line Properties) contains the following fields and controls:

- コンポーネント名: A text input field.
- 表示順序: A text input field.
- 位置 / サイズ: A group box containing four input fields: 左 (25.8), 上 (29.87), 幅 (25.8), and 高さ (16.17).
- 可視: A checked checkbox.
- 透過: An unchecked checkbox.
- 反転: An unchecked checkbox.
- 回転: A text input field with the value 0 and a degree symbol.
- 線幅: A text input field with the value 1.
- 線方向: A dropdown menu with '水平' (Horizontal) selected.
- 線種別: A dropdown menu with '実線' (Solid line) selected.
- Buttons: キャンセル (Cancel), 適用 (Apply), and OK.

- ・ 線幅は 1～255 までの間で指定できます。
- ・ 線方向は「水平」「垂直」「左上→右下」「左下→右上」から選択します。



- ・ 線種別は「実線」、「破線」、「点線」、「一点鎖線」、「二点鎖線」から選択します。

<破線>



<点線>



<一点鎖線>



<二点鎖線>



5. 4 ボックスコンポーネント

指定したエリア内に四角形・円形を描画します。

ボックスプロパティ

コンポーネント名 表示順序

位置 / サイズ

左 上 幅 高さ

可視 ☒ 透過 ☐ 反転 ☐

回転 ° 線幅

線種別 ペイント種別

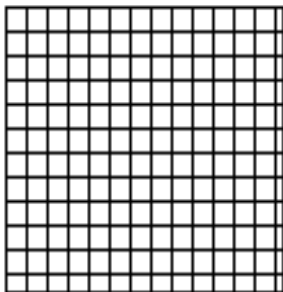
コーナー設定

☒ 四角 ☐ 円形 ☐ 自動

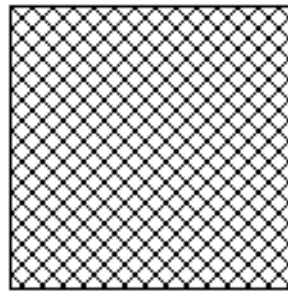
幅 /99999

高さ /0

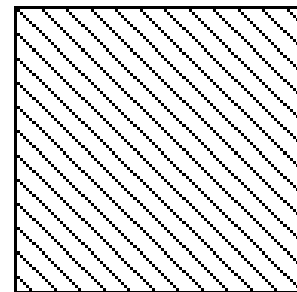
- ・ 線幅は 1～255 までの間で指定できます。
- ・ 線種別は「実線」、「破線」、「点線」、「一点鎖線」、「二点鎖線」から選択します。
(ラインコンポーネントと同様です)
- ・ ペイント種別は「塗りつぶしなし」、「縦横網掛け」、「左右対角線網掛け」、「右下がり対角線」、「右上がり対角線」、「垂直線」、「水平線」、「塗りつぶし」から選択します。



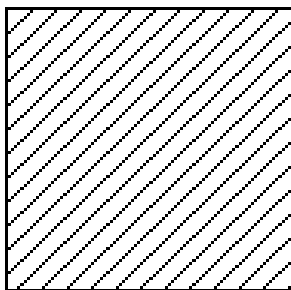
縦横網掛け



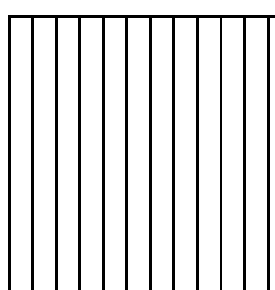
左右対角線網掛け



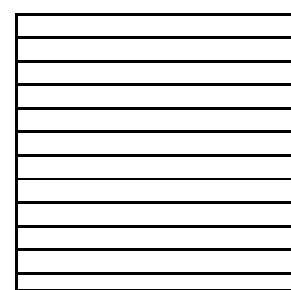
右下がり対角線



右上がり対角線



垂直線



水平線

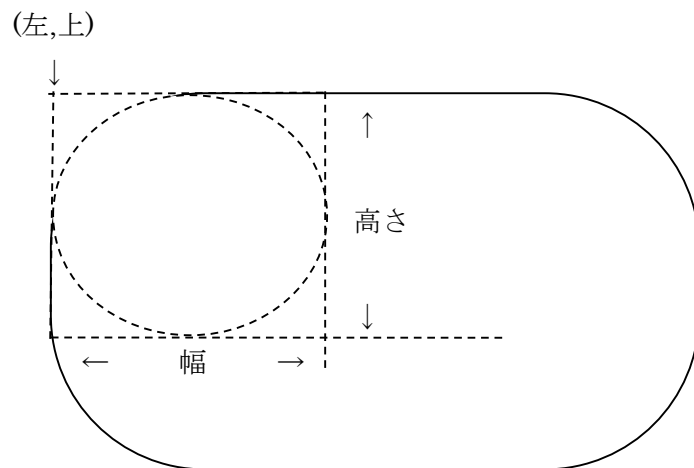
- ・ コーナー設定は「四角」「円形」「自動」から選択します。

「四角」はコンポーネント枠にそのまま描画されます。

「円形」はコンポーネント枠の上下左右を接点とし、円形を描画します。

「自動」は幅と高さの値を元に角落としの四角形を描画します。

角落としの四角形は Win32API の RoundRect 関数で描画される四角形の動作を元にしており、Win32API の RoundRect 関数で角は幅と高さの大きさの円を四角形の四隅に配置し、四角形と円の和となる部分を角として描画されています。



5. 5 イメージコンポーネント

指定したエリア内に選択したイメージ画像を描画します。



- ・ 誤差拡散のチェックを ON にすることで、イメージの全体の色の階調を滑らかにします。画像を構成する画素（ピクセル）の色情報を平均化して中間階調を表現します。

本ツールでは閾値(16 進数カラーコード)は 127 で設定されています。

閾値より高い場合は白(255) 、低い場合は黒(0)で表され中間階調の計算式は以下となります。

画像サイズ(幅×高さ)1pixel ごと閾値白黒判定の合計 ÷ 画像サイズ(幅×高さ) = 中間階調値

- ・ アスペクト比を有効にすることで、自動サイズを使用した拡大縮小時に元画像の縦横比率を維持します。無効にすると自動サイズの拡大縮小は領域幅・高さに合わせます。
- ・ イメージボタンを押下すると、画像ファイル選択ダイアログが立ち上がります。画像を選択すると、プレビュー画面と幅、高さが表示されます。

5. 6 テーブルコンポーネント

指定したエリア内に縦・横のセル数を指定したテーブル枠を描画します。

- ・線幅はテーブル外・内枠の太さの指定になります。

<テーブルの作りかた>

1. フレーム数の Row（縦セル数）、Col（横セル数）部分に数値を入力し、「作成→」ボタンを押下します。セル数はそれぞれ 1 ～ 20 の範囲で指定できます。
- ・テーブル作成後は、編集したいセルをフレーム編集の Row、Col に入力し、各値、設定を入力していきます。設定項目は Text コンポーネントと同様になります。
 - ・テーブル描画後に編集したいセルを右クリック、またはダブルクリックすることでクリックした場所のセル編集が行えます。
 - ・データ種類、フォーマット、增加值については巻末のデータ種別・フォーマットについてを参照してください。

6 データ種別・フォーマットについて

テキスト、バーコード、テーブルコンポーネントには、データ種別、フォーマット、増加値の3つのプロパティを持っています。

フォーマットを使用する場合は自動でデータが生成されるため、Serial Count 以外は直接コンポーネントデータを入れることはできません。

Serial Count はカウントアップされるデータとしてコンポーネントデータが必要です。

データ種別の指定によってフォーマットの記述方法が異なります。

フォーマットで要素を生成する処理順序は決まっており、5段階に分けて処理が行われます。

処理順序	データ種別
1	Standard, SerialCount, DateTime
2	DataSearch
3	Copy, Substring, Compute
4	Join
5	Attribute

フォーマットで指定するコンポーネント名は、使用するコンポーネントのデータ種別の処理順序よりも小さいデータ種別のコンポーネント名を指定する必要があります。

例えばデータ種別が Copy のコンポーネントは、Join で使用することが出来ますが、Substring や Compute で使用することはできません。

フォーマットの記述を簡単にするために、プロパティ画面にフォーマット編集ボタンが用意されています。以下にフォーマットの一覧と編集画面の説明を記述します。

6. 1 Standard

説明：フォーマットの記述による変換を行わずデータを出力します。

6. 2 Serial Count

The screenshot shows a dialog box titled 'フォーマット編集' (Format Edit) with a 'Serial Count' dropdown menu. The 'Component Name' is 'Component1'. The 'Increment' is '1', 'Format' is '%4d', and 'Data' is '12345'. Below these fields are several options: 'Left-align' (unchecked), 'Sign' (unchecked), 'Zero-padding' (set to 'Do not pad with zeros'), 'Field width' (set to '4 characters'), 'Input size' (set to 'None'), 'Conversion specifier' (set to 'd (signed decimal)'), and 'Increment after printing' (set to '1').

説明：フォーマットの書式に合わせて数値を出力します。

書式の動作はC言語のprintf()等で使用されるフォーマット指定子と同じです。

印字後に増加値分カウントアップを行います。

左詰め：印字幅が指定されていた際に値の左詰めを行います。

符号付与：値の頭に+が付与されます。

変換指定文字に符号なしのもの(o, u, x, X)が指定された場合は付与されません。

余白処理：印字幅指定時に余白を0で埋めるか空白で埋めるかを選択します。

印字幅：何文字分のスペースに値を収めるかを指定します。

印字幅を超えるデータの場合、指定は無視されます。

入力サイズ：データを指定された型に変換します。

変換指定文字：出力形式を指定します。

d = 符号付き 10 進整数

i = 符号付き 10 進整数

o = 符号なし 8 進整数

u = 符号なし 10 進整数

x = 符号なし 16 進 int (a, b, c, d, e, f を使用)

X = 符号なし 16 進 int (A, B, C, D, E, F を使用)

印字後加算値：印字後に増加する値です。

6. 3 Date Time

The screenshot shows a dialog box titled 'フォーマット編集' (Format Edit). It has a 'データ種別' (Data Type) dropdown set to 'Date Time'. There are 'キャンセル' (Cancel) and 'OK' buttons. Below, 'コンポーネント名' (Component Name) is 'Component2'. '増加値' (Increment Value) is a text box with '0'. 'フォーマット' (Format) is a text box with 'YYYY年MM月DD日 HH時NN分SS秒'. 'データ' (Data) is an empty text box. At the bottom, there are buttons for '年' (Year), '月' (Month), '日' (Day), '時' (Hour), '分' (Minute), '秒' (Second), and 'マイクロ秒' (Microsecond). Below these is '経過時間' (Elapsed Time) with a text box containing '0' and a spinner.

説明：フォーマットの書式に合わせて日時を表示します。

増加値を指定すると増加値時間先の日時を表示します。

年月日ボタン：書式がフォーマットに追加されます。

Y:年 M:月 D:日 H:時 N:分 S:秒 Z:マイクロ秒

1つのフォーマットで同じ書式を複数回含める(MMを2回含める等)ことはできません。

年月日の値の間に文字を入れる場合は直接フォーマット欄に入力してください。

6. 4 Copy

The screenshot shows a dialog box titled 'フォーマット編集' (Format Edit). It has a 'データ種別' (Data Type) dropdown set to 'Copy'. There are 'キャンセル' (Cancel) and 'OK' buttons. Below, 'コンポーネント名' (Component Name) is 'Component2'. '増加値' (Increment Value) is an empty text box. 'フォーマット' (Format) is a text box with 'Component1'. 'データ' (Data) is an empty text box. At the bottom, there is a 'コピー元コンポーネント' (Copy Source Component) dropdown set to 'Component1'.

説明：フォーマットで指定されたコンポーネントのデータと同じデータを表示します。

コピー元コンポーネント：コンボボックスからコピー元とするコンポーネント名を指定します。

テキスト、バーコード、テーブルコンポーネントのみが対象です。

6. 5 Substring

The screenshot shows a dialog box titled 'フォーマット編集' (Format Edit) with a standard Windows window frame. It contains several input fields and buttons. At the top right are 'キャンセル' (Cancel) and 'OK' buttons. The 'データ種別' (Data Type) is set to 'Substring'. The 'コンポーネント名' (Component Name) is 'Component2'. Below this are three empty text boxes for '増加値' (Increment), 'フォーマット' (Format), and 'データ' (Data). The 'フォーマット' field contains the text '002,004,Component1'. A section at the bottom contains three more fields: '分割元コンポーネント' (Split Source Component) set to 'Component1', '分割開始位置(0~999)' (Split Start Position) set to '2', and '印字バイト数(1~999)' (Print Byte Count) set to '4'.

データ種別	Substring	キャンセル	OK
コンポーネント名	Component2		
増加値			
フォーマット	002,004,Component1		
データ			
分割元コンポーネント	Component1		
分割開始位置(0~999)	2		
印字バイト数(1~999)	4		

説明： 指定したコンポーネントの開始位置から印字バイト数分の文字列を印字します。

上記画像を例にすると、Component1 のデータが「123456789」だった場合、

2 バイト目から 4 バイト取得し、結果は「3456」となります。

分割元コンポーネント：コンボボックスから分割元とするコンポーネント名を指定します。

テキスト、バーコード、テーブルコンポーネントのみが対象です。

分割開始位置：データの分割開始位置を指定します。

印字バイト数：分割開始位置から取得するバイト数です。

6. 6 Compute

フォーマット編集

データ種別: Compute

コンポーネント名: Component2

増加値: 1

フォーマット: 000,004,A,R,1,00,0,N,S,0,Component1

データ:

分割元コンポーネント: Component1

開始位置(0~999): 000

取得バイト数(1~999): 004

演算種類: A(足し算)

小数点処理: R(四捨五入)

小数点処理桁数(1~9): 1

整数出力桁数(0~99): 00

小数点以下出力桁数(0~9): 0

正数桁埋め文字: N(空白埋め 左寄せ)

小数点以下桁埋め文字: S(空白埋め)

カンマ付与: 無し

演算値

☒ 固定値を使用: 1

☐ コンポーネント値を使用

説明: 開始位置から取得バイト数のデータを指定したコンポーネントから抜き出し、抜き出した値と増加値の値で演算を行います。

抜き出した値に数値とカンマ以外が含まれていた場合は処理を行えません。

抜き出した値の頭が0から始まっていた場合は頭の0を削除してから処理します。

演算結果に小数点が出た場合は指定された小数点処理を行い、出力桁数が指定されていた場合は桁埋め文字で桁数を合わせます。

カンマ付与が有だった場合は3桁ごとにカンマを追加します。

カンマは整数出力桁数に入るためカンマを考慮した文字数の指定が必要です。

計算結果が出力桁数指定を超えた場合は空白データとなります。

浮動小数点数の丸め誤差により小数点以下の値は誤差が生じる恐れがあります。

分割元コンポーネント：演算対象の値を取得するコンポーネント名を指定します。

開始位置：データの分割開始位置を指定します。

取得バイト数：分割開始位置から取得するバイト数です。

演算種類：四則演算と割引パーセントを選べます。

割引パーセントは演算値の%分引いた値を求めます。（10 の場合 10%オフの値。最大 100）

小数点処理：小数点が出た際の小数点処理を選択できます。

小数点処理桁数：小数点処理の対象とする小数点以下桁数を指定します。

指定した小数点以下のデータは削除されるため、小数点以下を出力する場合は
2 以上を指定します。

整数出力桁数：整数部の桁数を指定します。

小数点以下出力桁数：小数部の桁数を指定します。

正数桁埋め文字：整数部が出力桁数に満たない際の桁埋め動作を指定します。

小数点以下桁埋め文字：小数部が出力桁数に満たない際の桁埋め動作を指定します。

カンマ付与：3 桁区切りのカンマ付与の有無です。

演算値：この値で演算を行います。コンポーネント値を使用することも可能です。

6. 7 DataSearch / DataSearch2

フォーマット編集

データ種別: DataSearch

コンポーネント名: Component3

増加値:

フォーマット: C:\Users\%EasyPrint %easy.csv,3,1,Component1,2,C

データ:

CSVファイル名: C:\Users\%EasyPrint %

取得フィールド番号: 3

サーチ条件フィールド番号1: 1

サーチ条件コンポーネント名1: Component1

サーチ条件フィールド番号2: 2

サーチ条件コンポーネント名2: Component2

サーチ条件フィールド番号3:

サーチ条件コンポーネント名3:

説明: 指定した CSV ファイルのサーチ条件フィールド値が同一のレコードから取得フィールド番号の値を取得します。

サーチ対象が存在しない/条件に一致するレコードが複数存在する場合はエラーとなり印字されません。

上記画像を例にすると、指定した CSV ファイルの 1 列目が Component1 の値と同一で、2 列目が Component2 の値と同一の行から 3 列目のデータを取得します。

DataSearch2 を選択した場合は、出力形式が XML の際にプリンタ側で 2 分探索を行います。

DataSearch2 の場合はサーチ条件のフィールドがソートされている必要があります。

CSV ファイル名: サーチを行う CSV ファイル名を指定します。

XML 出力やプリンタにテンプレート登録する場合、CSV ファイル名はプリンタに保存されている CSV ファイルパス (E:\%~ または U:\%~) を指定してください。(ファイルパスが異なる為、EasyPrint 上ではデータサーチエラーとなります。)

取得フィールド番号: 取得したいフィールド番号 (列番号) を指定します。

サーチ条件フィールド番号 1~6: サーチ条件とするフィールド番号 (列番号) を指定します。

サーチ条件コンポーネント名 1~6: サーチする値を取得するコンポーネント名を指定します。

6. 8 Join

The screenshot shows the 'Format Edit' dialog box with the 'Join' data type selected. The 'Component Name' is 'Component3'. The 'Format' field contains 'Component1,Component2'. Below these fields is a list of nine 'Join Component' dropdowns. The first two are set to 'Component1' and 'Component2' respectively, while the others are empty.

結合コンポーネント	選択されたコンポーネント
結合コンポーネント1	Component1
結合コンポーネント2	Component2
結合コンポーネント3	
結合コンポーネント4	
結合コンポーネント5	
結合コンポーネント6	
結合コンポーネント7	
結合コンポーネント8	
結合コンポーネント9	

説明： 複数コンポーネントのデータを結合した内容を印字します。

結合コンポーネント 1～9：1 から順番に指定したコンポーネントのデータを結合していきます。

6. 9 Attribute

The screenshot shows the 'Format Edit' dialog box with the 'Attribute' data type selected. The 'Component Name' is 'Component3'. The 'Format' field contains 'Component1,Rotation,Component2'. Below these fields is a section for attribute settings. The 'Attribute Change Component' is 'Component1', the 'Attribute Name' is '回転' (Rotation), and the 'Attribute Value Acquisition Component' is 'Component2'.

属性変更コンポーネント	属性名	変更値取得コンポーネント
Component1	回転	Component2

説明： 指定されたコンポーネントの属性をコンポーネントのデータで変更します。

上記画像を例にすると、Component1 の回転値を Component2 のデータに変更します。

属性変更コンポーネント：属性値を変更したいコンポーネント名を指定します。

属性名：変更する属性名を指定します。

変更値取得コンポーネント：変更値を取得するコンポーネント名を指定します。