

The logo consists of the word "GLORY" in a bold, white, sans-serif font, centered within a solid black square.

GLORY

ISP-902E

OPOS つり銭機コントロールソフト

アプリケーション開発ガイド

ISP-902E OPOSつり銭機コントロールソフト
アプリケーション開発ガイド

はじめに

このたびは、グローリー製品をご採用いただきまして、誠にありがとうございます。

このアプリケーション開発ガイドには、製品を使用する際のアプリケーションの開発方法および使用上の注意事項を記載しております。

本書をお読みいただき、安全遵守事項、製品の性能、操作方法を十分にご理解のうえ、ご使用ください。また、お読みになった後も、お手元に置いてご確認ください。

本書を紛失、または汚損して確認できなくなったときは、最寄りの当社販売店または当社指定の取扱店でお求めください。

ご注意

この製品は、弊社つり銭機の制御を目的に開発しております。

他の目的で当社の許可なく使用したり、当社または当社委託の業者以外で改造し、使用した場合の損害については、その責任を負いかねます。

この製品の設置および移設に関しては、最寄りの当社販売店または当社指定の取扱店にご相談ください。

- 本書の記載事項は、事前通知なしに変更される場合もありますので、あらかじめご了承ください。
- 当社は、常に製品の改善のために努力しています。
したがって、実際の製品と本書とは多少の相違が生じることがあります。
- 本書のデータの使用に起因する第三者の特許権、およびその他の権利での損害については、その責任を負いかねます。
- 営利目的での本書の無断転載を禁止します。

目次

1. はじめに	1
1. 1. 開発ガイド概要	4
1. 2. プログラム開発時の留意点	5
2. コントロールソフト仕様	8
2. 1. 仕様条件	9
3. インストール	11
3. 1. インストール手順	12
3. 2. インストール後の設定 (OPoS Setup)	15
3.2.1. 設定項目	15
3.2.2. 操作方法	20
3. 3. アンインストール	20
3. 4. その他	21
3.4.1. レジストリ仕様	21
3.4.2. 定数の利用	25
3.4.3. サンプルアプリケーションについて	26
4. A P G仕様	31
4. 1. 一覧	32
4.1.1. プロパティ	32
4.1.2. メソッド	34
4.1.3. イベント	34
4.1.4. 初期値一覧	35
4. 2. プロパティ	37
4.2.1. 共通プロパティ	37
BinaryConversion プロパティ R/W	37
DataEventEnabled プロパティ R/W	38
DeviceEnabled プロパティ R/W	39
FreezeEvents プロパティ R/W	40
PowerNotify プロパティ R/W	41
4.2.2. 専用プロパティ	43

CurrencyCode プロパティ R/W	43
CurrentExit プロパティ R/W	44
CurrentService プロパティ R/W	45
RealTimeDataEnabled プロパティ R/W	46
4. 3. メソッド	47
4.3.1. 共通メソッド	47
Open メソッド	47
Close メソッド	49
ClaimDevice メソッド	50
ReleaseDevice メソッド	51
CheckHealthメソッド・(CheckHealthTextプロパティ)	52
ClearInput メソッド	56
CompareFirmWareVersion メソッド ※1	57
ResetStatistics メソッド ※1	58
RetrieveStatisticsメソッド ※1	59
UpdateFirmware メソッド ※1	60
UpdateStatistics メソッド ※1	61
4.3.2. 専用メソッド	62
BeginDeposit メソッド	62
DispenseCash メソッド	64
DispenseChange メソッド	66
EndDeposit メソッド	68
FixDeposit メソッド	70
PauseDeposit メソッド	74
ReadCashCounts メソッド	77
AdjustCashCounts メソッド ※1	79
DirectIO メソッド	80
リセット	82
メモリリード	83
メモリクリア	84
モード切り替え	85
SSW設定	86
日時設定	87
ENQ送信	88
文字列送信 ※1	89
回収	90
状態リード	91

補充枚数取得	99
精査	100
計数データリード	108
計数開始	109
計数終了	110
計数停止	111
計数再開	112
計数機／つり銭機モード（計数モード設定）	113
ログ吸い上げ	114
累計カウンタクリア	115
棒金ドロア制御	116
ローカル操作禁止	117
締め	118
補充開始	120
取引外入金開始	121
取引外金額指定出金	122
取引外枚数指定出金	123
現金戻し開始	124
交代データ取得	125
エラー解除ガイダンス起動	127
4. 4. イベント	129
DataEvent イベント	129
StatusUpdateEvent イベント	130
DirectIO イベント	131
貨幣抜き取り待ちイベント	132
補充動作検知イベント	133
計数中 エラー発生イベント	134
計数中 RJ発生イベント	135
計数中 回収カセットフルイベント	136
計数中 セット外れイベント	137
貨幣抜き取り完了イベント	138
計数待機中イベント	139
5. ログ仕様	142
5. 1. ログ仕様	143
5.1.1. 時系列ログ	143
5.1.2. 処理ログ	146

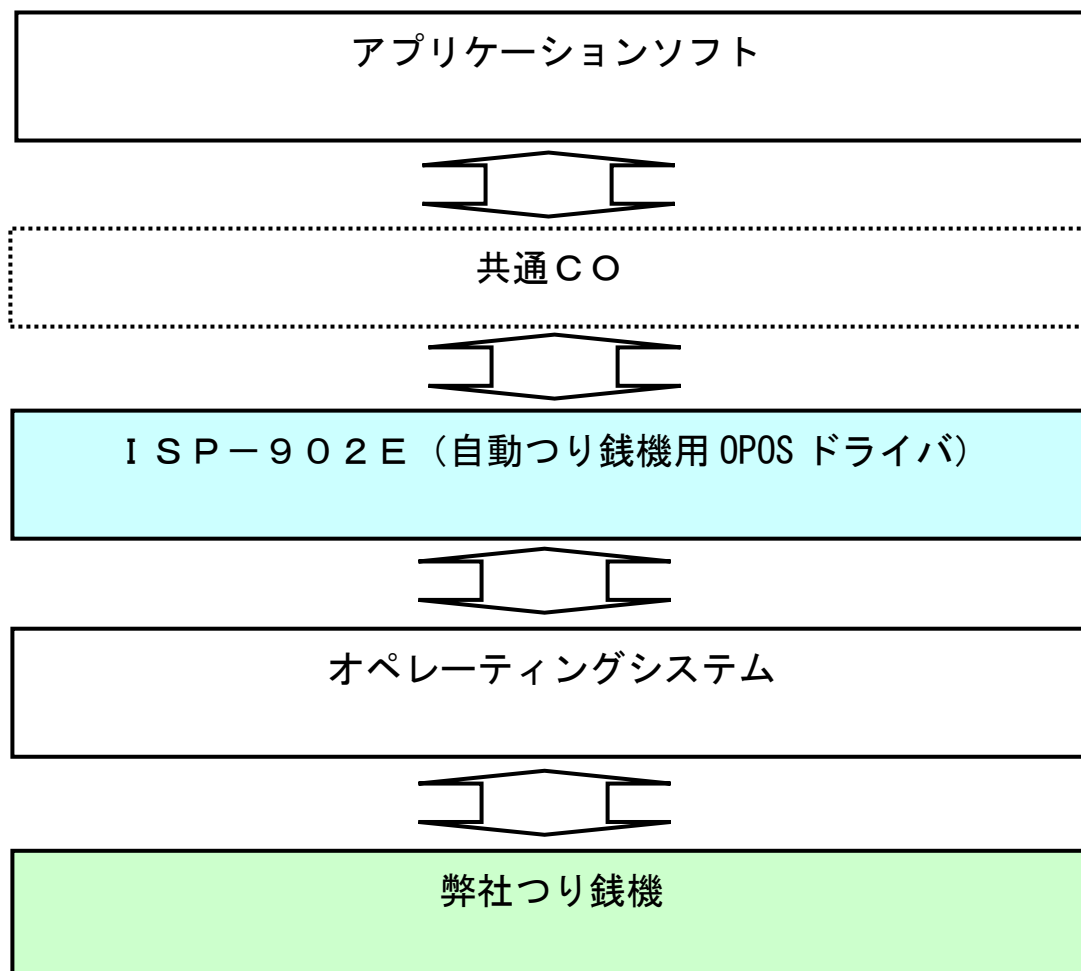
5.1.3.	通信ログ	147
5.1.4.	取引ログ	148
5.1.5.	処理再現ログ	151
5.2.	その他設定	152
6.	補足事項	155
6.1.	コード一覧	156
6.1.1.	ResultCode	156
6.1.2.	ResultCodeExtended	157
6.1.3.	OpenResult	158
6.2.	付録	159
6.2.1.	OPOSALL.BAS	159
6.2.2.	GLORYOPOS.BAS	163
6.3.	シーケンス図	165
6.3.1.	売上登録（釣銭あり）	165
6.3.2.	売上登録（釣銭なし）	166
6.3.3.	売上登録（取引中止・預り金返却）	167
6.3.4.	返品処理	168
6.3.5.	両替処理	169
6.3.6.	釣銭補充処理	170
6.3.7.	閉店時	171
6.3.8.	入金エラー時	172
6.3.9.	出金エラー時	173
6.4.	特記事項	174
6.5.	バージョン履歴	176

1. はじめに

本製品は、弊社製つり銭機を、OLE コントロールを用いて VisualBasic 等の開発ツールから制御できるソフトウェアです。本コントロールソフト仕様は OPOS 技術協議会発行の Application Programmer's Guide (以後 APG) の規定に沿っております。

本書では、本製品の基本機能及び、アプリケーション開発の方法について記載しています。

又、以後「Open Point of Service」を「OPOS」と称して説明します。



※本製品は、OPOS 仕様第 1.13 版に準拠しています。

OPOS 技術協議会の URL (2012 年 7 月時点)

<http://www.microsoft.com/japan/business/industry/retail/opos/default.msp>

※上記 URL から OPOS 技術協議会発行の仕様書も参照してください。

※つり銭機の仕様につきましては、営業担当へお問い合わせください。

Microsoft, Windows, WindowsNT, Windows2000, WindowsXP, Windows7, Windows8, Windows8.1, VisualStudio, VisualBasic および VisualC++は米国 Microsoft Corporation の米国およびその他の国における登録商標です。

その他、本書に記載した会社名, 商品名, 製品名などは、一般に各社の商標もしくは登録商標です。

本書に記載する開発・デバッグの為の仕様においては、予告無く改訂・仕様変更する場合があります。

1. 1. 開発ガイド概要

本書では、コントロールソフトを使用したアプリケーション開発の方法について説明致します。

1. 本書では、以下の製品での開発環境を前提として説明します。

- ◇ Microsoft Visual C++ 6.0 (SP5 適用)
- ◇ Microsoft Visual Basic 6.0 (SP5 適用)
- ◇ Microsoft Visual Studio 2005 (SP1 適用)

2. アプリケーションに組み込む為には事前に、以下のセットアップを完了しておく必要があります。

◆ 共通 CO (ControlObject)

共通 CO “OPOSCashChanger. OCX” は本ソフトのインストーラにてインストールされます。
又、共通 CO は以下の URL からダウンロードが可能です。(2012 年 7 月時点)

http://monroecs.com/oposccos_current.htm

又、共通 CO に関しての説明は、OPOS 技術協議会から発行されている技術解説マニュアルを参照して下さい



本書では Visual Basic/Visual C++ の操作方法、ActiveX の基礎知識や ActiveX を実装したプログラム作成を理解されている方を対象としています。
これらの用語、操作方法については各 WindowsOS、Visual Basic/Visual C++ のマニュアル、ヘルプなどをご参照ください。

1. 2. プログラム開発時の留意点

Visual Basic/Visual C++ でプログラム開発を行う場合の留意点を説明します。

◆イベント処理は最小限に押さえる

各イベント内に記述する処理（コード）を出来るだけ少なくし、処理のレスポンス向上を図る必要があります。

特にご注意くださいのは、「OLE for Retail POS Application Programmer's Guide」（日本語仕様書 第 1.13 版）にも記載の通り、イベントハンドラ内でのコントロールの終了処理（ReleaseDevice メソッドや Close メソッド）を行ってはいけません。

◆モーダル表示を避ける

フォームのモーダル表示やメッセージボックスを利用した場合、その部分で処理が一旦停止します。その為、連続して発生するイベントで使用する際には注意が必要です。

装置状態のリアルタイム表示等には使用しない事をお勧めします。

◆安易なアンロードは避ける

ActiveX コントロールのイベント内で、現在使用中のフォーム（ダイアログ）をアンロードすると、通常のイベント処理が完了しないことがあります。その為予期せぬトラブルが発生する可能性が高くなります。アンロードが必要な場合は、出来るだけコマンドボタン等の標準コントロールのイベントでアンロードを行う事をお勧めします。

特に、複数の画面（フォームやダイアログ）をもつアプリケーションや、プロセス間通信（メール・メッセージ・パイプ処理）を頻繁に行うような複雑なアプリケーションを構築する場合は、サービスのスイッチング（VB の場合 DoEvents 等）を反映させる等、終了処理には十分な注意が必要です。

（頻繁に発行されるブロードキャストメッセージ等に対するイベント処理が滞り、結果 OPOS 配下での終了処理が完了せずに予期せぬトラブルの原因となります）

◆開発環境と実行環境をあわせる

WindowsOS のソフトウェア環境は、そのバージョンやインストールされているアプリケーション、ハードウェアのスペックにより大きく左右されます。

開発環境と実行環境が異なる場合、予期せぬトラブルが発生する可能性が高くなります。

また、Internet Explorer 等のアプリケーションをインストールしたときに、既存の共有ライブラリ等が書き換えられ、環境が変わってしまう場合があるので注意が必要です。

◆長時間運用には注意が必要

アプリケーションの長時間運用中に、万が一 OS や VisualBasic、PowerTools 製品、いずれかの不正な動作によりシステムリソースが開放されない場合、システム全体が停止してしまうことがあります。できるだけ定期的にアプリケーションの再起動を行うなどして、システムリソースを開放するようにしましょう。

[illegible]

[illegible]

2. コントロールソフト仕様

2. 1. 仕様条件

① 本コントロールソフトの動作OS

- ◇ Microsoft Windows2000 (SP4 以降)
- ◇ Microsoft WindowsXP (SP1a 以降)
- ◇ Microsoft Windows Embedded POSReady 2009
- ◇ Microsoft Windows 7
- ◇ Microsoft Windows 8 (32bit 版のみ)
- ◇ Microsoft Windows 8.1 (64bit 版のみ)

② 動作環境による注意事項

本コントロールソフトを使用する際は、POS及びPC等の上位システムの動作環境下（マルチコア環境を含む）で、検証や確認を十分に実施してください。

③ 対応するつり銭機

機種名		備考
紙幣	RAD-300	詳細の型式及び製品情報については、販売窓口へお問合せ下さい。
硬貨	RT-300	
棒金ドロア	WD-300	

④ 対応する接続パターン

- ◇ RAD-300/RT-300
- ◇ RAD-300/RT-300/WD-300
- ◇ RT-300

⑤ 使用可能COMポート

本コントロールソフトを使用するPOS及びPC等にて、OSよりサポートされているCOMポート。使用するCOMポートの選択は、本コントロールソフトのインストール後、レジストリ内容を変更することで設定が可能です。

尚、本コントロールソフトは、POS及びPC内で1パターン分のつり銭機のみ制御できます（複数ポートの同時使用による複数のつり銭機の制御は出来ません）。

⑥ 本コントロールソフトインストールによる注意事項

本コントロールソフトのインストールにより、本コントロールソフト以外に表2. 1に示されるシステムファイルのインストールが行われます。システムファイルはインストール先PCにおいて、本コントロールオブジェクトと同じフォルダにインストールされます。

表2. 1 本コントロールソフトインストール時にインストールされるファイル一覧

ファイル名	ファイルバージョン	備考
ChangerS0.dll	本コントロールバージョンにより異なります	本サービスオブジェクト
err_restore.exe		エラー解除ガイダンス用アプリケーション
Cleanup.bat	バージョン情報なし	インストールしたファイルの削除
GLORY_Cleanup.bat		
OPOSCashChanger.ocx	1.13.0.0	共通CO
OposSetup.exe	1.0.0.0	レジストリ設定ツール
reg.exe	2.0.0.0	レジストリエントリ用ツール (Microsoft)
Install.log	バージョン情報なし	インストール結果（成功時）に作成される

[illegible]

3. インストール

3.1. インストール手順

本ドライバーのインストール方法を以下に示します。

バッチファイルによりシステムにファイルのコピー、レジストリへの登録を行います。

ファイルの構成は以下の通りです。

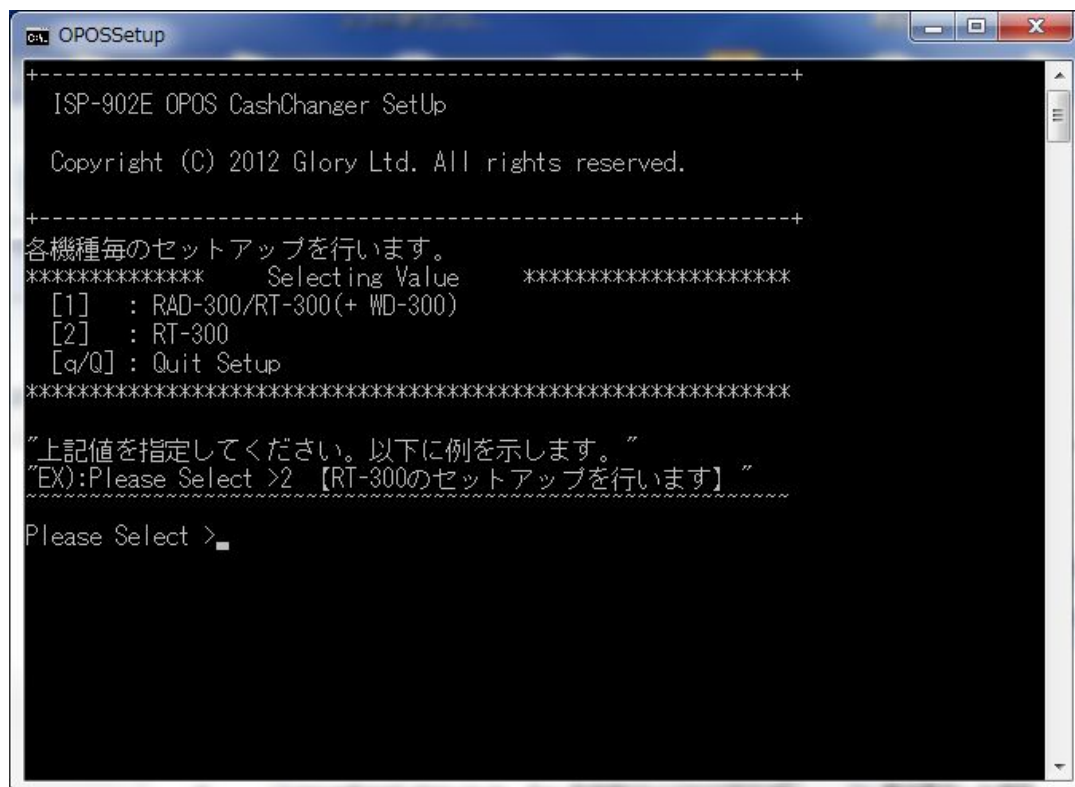
ファイル名	説明
ChangerS0.dll	本サービスオブジェクト (S0)
Cleanup.bat	インストールしたファイルの削除、レジストリ情報の解除
GLORY_Cleanupdir.bat	
err_restore.exe	エラー解除ガイダンス用アプリケーション
OPOSCashChanger.ocx	共通 C0
opossetup.exe	レジストリ設定ツール
reg.exe	レジストリエントリ用ツール
Regent.bat	レジストリにエントリするキー情報

(1) バッチファイル “OposSetup.bat” を実行します。

※バッチファイルの実行では、レジストリの登録処理が行われるため、

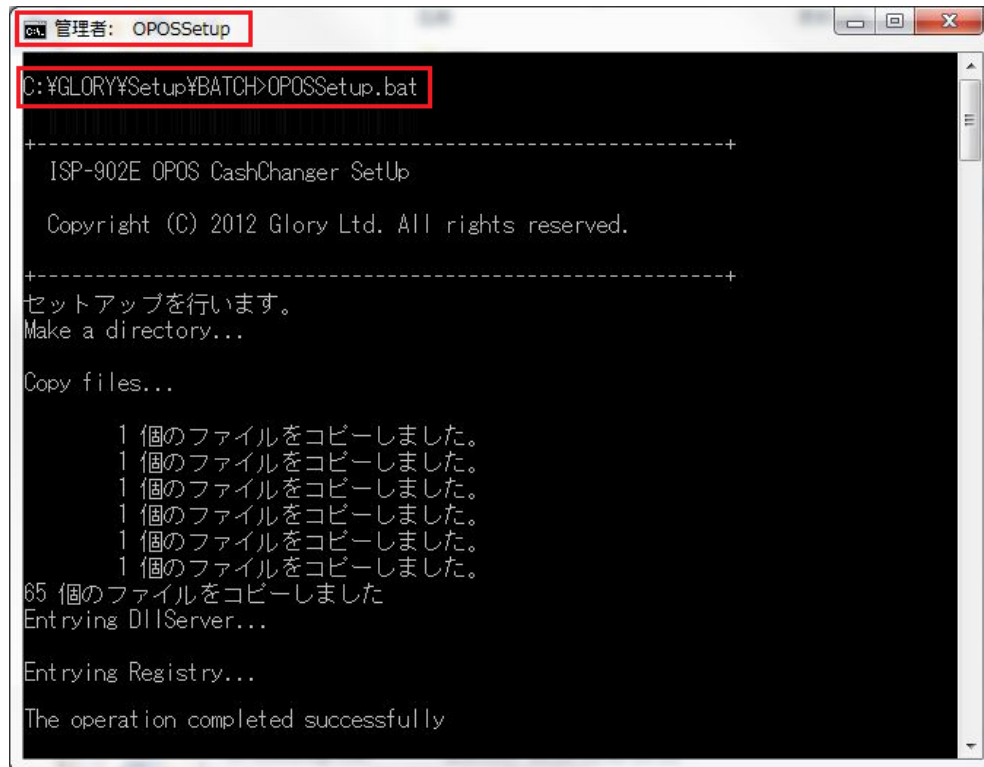
Windows 7 以降では「管理者として実行」で実行していただく必要があります。

- ・ 接続機種を選択します。([1] : RAD-300/RT-300(+WD-300)、[2] : RT-300 単体)
- ・ 機種選択後、バッチ処理にてファイルのコピーと、レジストリへのエントリを行います。
- ・ インストール先は “C:\¥OPOS¥GLORY¥RT300” (固定) 配下となります。
- ・ セットアップ正常終了後、インストール先に “install.log” (固定) が出力されます。



※インストールができない場合は、下記の手順をご確認ください。

- (1) コマンドプロンプトを管理者権限で起動します。
- (2) OPOSSetup.bat をコマンドプロンプトで指定し、実行します。



```
管理者: OPOSSetup
C:\%GLORY%\Setup\BATCH>OPOSSetup.bat

+-----+
ISP-902E OPOS CashChanger SetUp
Copyright (C) 2012 Glory Ltd. All rights reserved.
+-----+
セットアップを行います。
Make a directory...

Copy files...

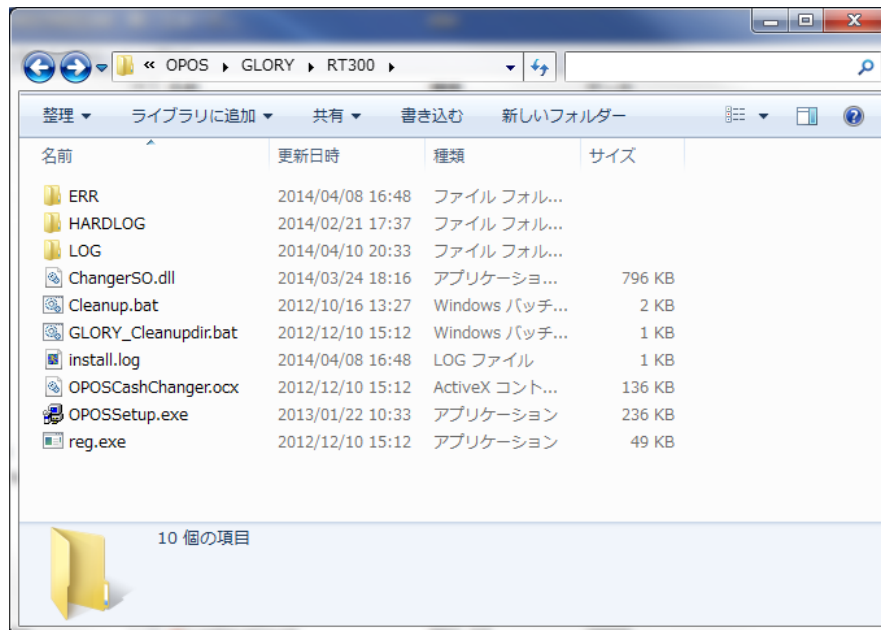
1 個のファイルをコピーしました。
1 個のファイルをコピーしました。
1 個のファイルをコピーしました。
1 個のファイルをコピーしました。
1 個のファイルをコピーしました。
1 個のファイルをコピーしました。
65 個のファイルをコピーしました
Entrying DIIServer...

Entrying Registry...

The operation completed successfully
```

※ 更新処理について

既に以前のバージョン (Ver1.0～4.0) がインストール済みの場合は、インストールフォルダ (C:\¥OPOS¥GLORY¥RT300) にある ChangerS0.dll をインストール CD 内の Dll¥ChangerS0.dll で上書きすることで更新可能です。
但し、接続パターンを変更する場合は、アンインストール後に再インストールを実施する必要があります。



◎バージョンの確認方法

インストールフォルダにある ChangerS0.dll のプロパティでバージョンが確認できます。



3.2. インストール後の設定 (OPOS Setup)

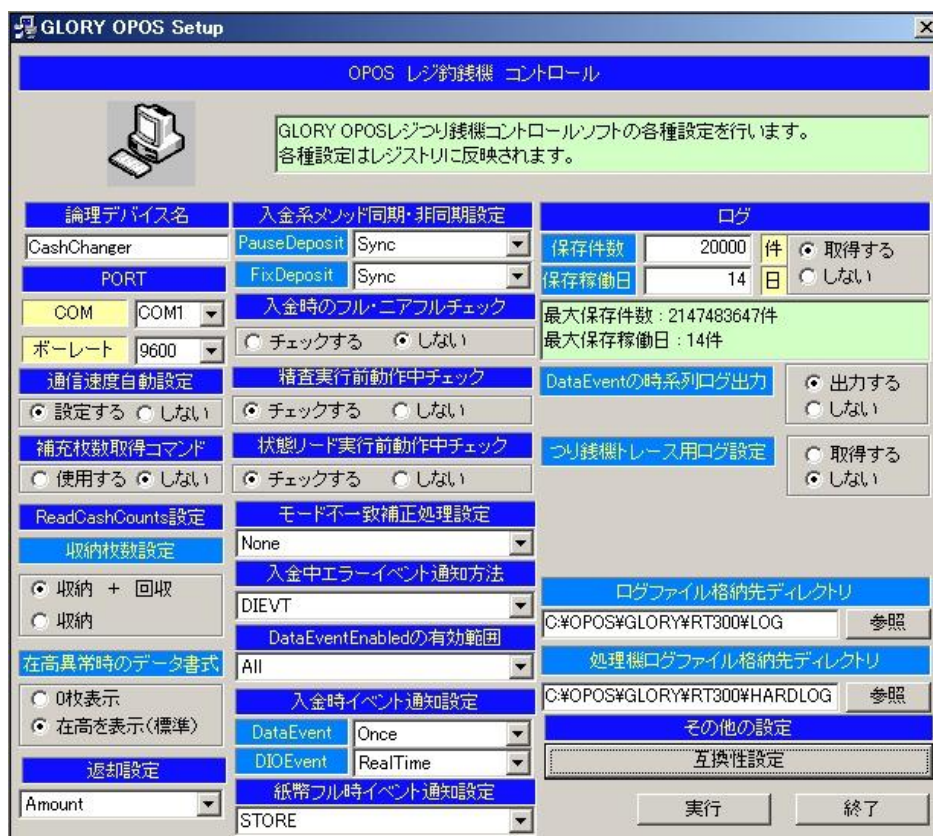
インストール完了後、設定ツール (OPOSSetup.exe) を使用し、引き続き各種設定を行います。

※設定ツールの実行では、レジストリの変更処理が行われるため、

Windows 7 以降では「管理者として実行」で実行していただく必要があります。

※実行環境により、「互換性アシスタント」メッセージが表示される場合がありますが、

2. 1 仕様条件で記載している OS 上では問題なく動作をします。



3.2.1. 設定項目

【論理デバイス名】

Open メソッド実行時引数として設定されます。

デフォルト値: "CashChanger"

【PORT】

◆COM: つり銭機を接続するポートを選択します。

デフォルト値: "COM1"

◆ボーレート: つり銭機との通信速度を選択します。

デフォルト値: "9600 (bps)"

【通信速度自動設定】

ボーレートがつり銭機側の通信速度と一致していなかった場合に、自動でつり銭機側の通信速度に合わせるかを選択します。

デフォルト値: "設定する"

[補充枚数取得コマンド]

DirectIO メソッド（補充枚数取得コマンド）の使用する／しないを選択します。

デフォルト値：[“使用しない”](#)

※補充枚数取得コマンドは設定と使用方法（タイミング）によっては、取得したデータが曖昧になります。
通常の運用では不要なコマンドであり、推奨しておりません。

使用の際はつり銭機のデータフローの熟知し、運用面からアプリケーション上の仕様（条件・タイミング）を明確にしておく必要があります。

[ReadCashCounts 設定]**◆収納枚数設定**

ReadCashCounts メソッドの実行により取得できるつり銭機内の現金在高の設定を選択します。

1) 収納＋回収：紙幣部）収納部在高 ＋ 回収カセット在高を現金在高とします。
硬貨部）収納部在高 を現金在高とします。

2) 収納：紙幣部・硬貨部共に、収納部在高を現金在高とします。

デフォルト値：[“収納＋回収”](#)

※ReadCashCounts メソッドの詳細については、別章「専用メソッド・プロパティ
— ReadCashCounts」を参照ください。

◆在高異常時のデータ書式

在高に異常が発生した場合本メソッドで通知する書式を選択します。

1) 0 枚表示：異常時は全金種 0 枚として通知します。

2) 在高を表示：現在つり銭機が保持するデータをそのまま通知します。

デフォルト値：[“在高を表示（標準）”](#)

[返却設定]

EndDeposit (CHAN_DEPOSIT_REPAY) 時の返却仕様を選択します。

1) Amount：金額指定放出で返却処理を行います。

2) CashList：枚数指定放出で返却処理を行います。

デフォルト値：[“Amount”](#)（金額指定放出で返却する）

[入金系メソッド同期・非同期設定]

FixDeposit, PauseDeposit の同期・非同期設定を選択します。

デフォルト値：

◆FixDeposit：[“Sync”](#)（同期処理の振る舞いで処理されます）

◆PauseDeposit：[“Sync”](#)（同期処理の振る舞いで処理されます）

[入金時のフル・ニアフルチェック有無]

BeginDeposit 実行時に、収納庫、回収カセットのフル・ニアフルチェックの実行可否を選択します。

デフォルト値：[“しない（フル・ニアフルチェックを行わない）”](#)

[精査実行前動作中チェック]

DirectIO メソッド-(CHAN_DI_SEISA) 実行時に、つり銭機の動作中プレチェックの実行有無を選択します。

プレチェックを行う場合は、動作中実行時はエラー (OPOS_ECHAN_BUSY) 応答します。

プレチェックを行わない場合は、動作中でも DirectIO メソッド-(CHAN_DI_SEISA) が実行可能です。

デフォルト値：[“する（プレチェックして、動作中はエラー応答とする）”](#)

[状態リード実行前動作中チェック]

DirectIO メソッド-(CHAN_DI_STATUSREAD) 実行時に、つり銭機の動作中プレチェックの
実行有無を選択します。

プレチェックを行う場合は、動作中実行時はエラー (OPOS_ECHAN_BUSY) 応答します。

プレチェックを行わない場合は、動作中でも DirectIO メソッド-(CHAN_DI_STATUSRAD) が
実行可能です。

デフォルト値 : “*する (プレチェックして、動作中はエラー応答とする)*”

[モード不一致補正処理設定]

OPOS-つり銭機間でモード不一致が発生した場合、補正行う為に DeviceEnabled → TRUE に
つり銭機に対して計数停止コマンドを自動で実行させるかの設定を行います。

ここで示すモード不一致とは、アプリケーション&OPOSの起動時に、つり銭機が計数中の状
態になっている事を表します。このような現象が発生するケースとして、入金中にアプリケーシ
ョンが再起動した場合などが挙げられます。

デフォルト値 : “*None (実行しない)*”

[入金中エラーイベント通知方法]

BeginDeposit 実行後の、入金中の事象発生を通知するイベントの種類を設定する。

イ) DIEVT : DirectIO イベントで通知します。

ロ) SUE : StatusUpdate イベントで通知します。

ハ) BOTH : DirectIO イベントと StatusUpdate イベントを併せて通知します。

デフォルト値 : “*DIEVT (DirectIO イベントで通知する)*”

[DataEventEnabled の有効範囲]

DataEventEnabled プロパティが「False」の状態、DataEvent がキューイングされた場合に、
以降に通知されるイベントもキューイングさせるかどうかを選択します。

イ) All : 全てのイベント (DirectIO イベント, StatusUpdate イベントを含む) を
キューイングさせる。

ロ) DataEvent : Data イベントのみキューイングさせる。(DirectIO イベント, StatusUpdate
イベントは通知する。)

デフォルト値 : “*All (全てのイベントをキューイング)*”

[入金イベント通知設定]

入金中に発生するイベント (DataEvent, DirectIOEvent) の通知タイミングを選択します。

◆DataEvent 通知設定

イ) Once : 計数動作停止のタイミングでイベント通知を行います。

ロ) Realtime : 計数枚数の変化を検知したタイミングでイベント通知を行います。

ハ) Realtime&Notify : 計数枚数の変化と計数動作停止を検知したタイミングで通知します。

デフォルト値 : “*Once*” (計数動作停止のタイミングで通知する)

◆DirectIOEvent 通知設定

イ) Delayed : 計数動作停止のタイミングで計数中に発生したイベントを通知します。

ロ) Realtime : 検知したタイミング (リアルタイム) でイベント通知を行います。

デフォルト値 : “*Realtime*” (リアルタイムに通知する)

[紙幣フル時イベント通知設定]

紙幣のフル発生時の StatusUpdate イベントの通知タイミングを選択します。

- 1) STORE : 収納庫がフルになったタイミングでイベント通知します。
- 0) NONE : フル時でもイベント通知しません。

(硬貨のフルイベント通知タイミングは、収納庫フル時のみです。)

デフォルト値 : “STORE (収納庫フルで SUE イベントを通知)”

[ログ]

- ◆時系列ログ・処理ログ・通信ログの処理の実行可否を選択します。

デフォルト値 : “する (ログ処理を行う)”

※ログの詳細については、別章「ログ仕様」を参照ください。

- ◆保存件数 : 1 稼働日分の時系列ログの最大保存件数を設定します。最大 2,147,483,647 件までのログの保存が可能です。

デフォルト値 : “20000 (件)”

- ◆保存稼働日 : 各ログファイルを何日分保存するかを設定します。最大 14 日稼働日までのログの保存が可能です。

デフォルト値 : “14 (日稼働分)”

[DataEvent の時系列ログ出力]

Data イベントの通知を時系列ログに出力する／しないを選択します。

デフォルト値 : “(ログに) 出力する”

※DataEvent の通知設定を Once 以外にしている場合は、大量に出力されますので、出力しない設定に変更することを推奨します。

[釣銭機トレース用ログ設定]

この設定が有効 (“する” を選択) な場合、本コントロールソフト稼働時 (つり銭機との通信開始～終了) にバックグラウンドで 処理再現ログ を取得し、ファイルを指定のローカルフォルダに生成します。

デフォルト値 : “(ログ生成) しない”

※処理機ログの詳細については、別章「ログ仕様 — 処理再現ログ」を参照ください。

※設定を有効にした場合、処理機との通信が頻繁に行われデフォルト設定の “しない” に比べてリソース (ハードディスクの使用量・CPU 占有) の消費が高くなります。

その為、適用するハードのスペックの検証を含め上位アプリでの入念な評価の上問題無き事を確認していただくことを前提とします。

※ログファイルのサイズは約 500K バイトです。

[ログファイル格納先ディレクトリ]

ログファイルの出力先を指定します。

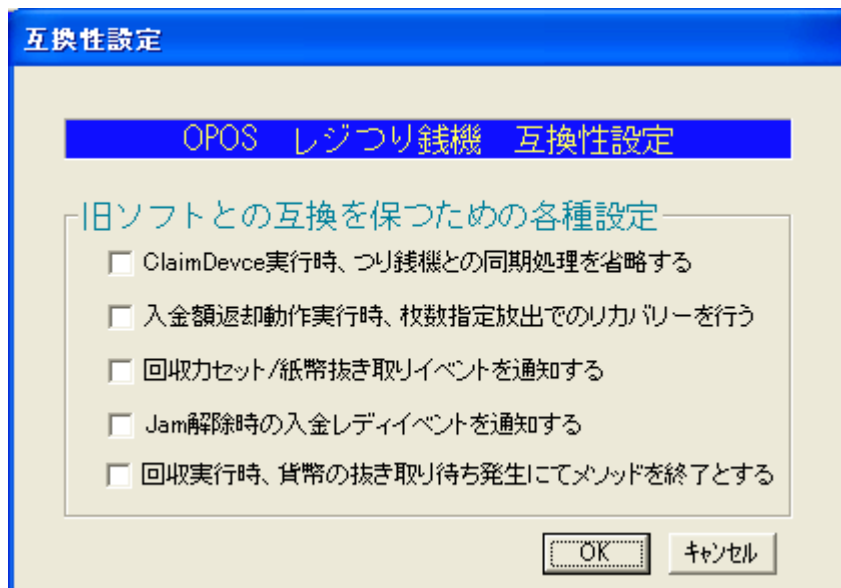
デフォルト値 : “インストール先パス¥LOG”

[処理機ログファイル格納先ディレクトリ] (未使用)

ログファイルの出力先を指定します。

デフォルト値 : “インストール先パス¥HARDLOG”

[その他設定]

**□ClaimDevice 実行時、つり銭機との同期処理を省略する**

: ClaimDevice メソッドの正常終了の条件につり銭機との同期処理の完了を含むか否か（省略する）を設定します。

同期処理とは、つり銭機との通信開始時に行う設定（SSW）の読み込みや、状態情報の取得を示します。

標準仕様はチェックを OFF（同期処理を含む）とします。

□入金額返却動作実行時、枚数指定放出でのリカバリーを行う

: EndDeposit (CHAN_DEPOSITREPAY) メソッドの返却処理が失敗した場合に、放出の指定方式を変更し、リカバリ処理を行うか否かを設定します。

標準仕様はチェックを OFF（リカバリ処理を行わない）とします。

□回収カセット/貨幣抜き取りイベントを通知する

: DirectIO イベントの貨幣抜き取り完了（CHAN_DIEVT_PULLOUT）の通知の可否を設定します。

標準仕様はチェックを OFF（通知しない）とします。

□Jam 解除時の入金レディイベントを通知する

: DirectIO イベントの入金中レディイベント（CHAN_DIEVT_DEPOSITREADY）の通知の可否を設定します。

標準仕様はチェックを OFF（通知しない）とします。

□回収実行時、貨幣の抜き取り待ち発生にてメソッドを終了とする

: 回収分割出金時のメソッド終了タイミングを設定します。

標準仕様はチェックを OFF（回収実行時、回収処理終了にてメソッドを終了とする）とします。

Bit2(回収カセット/紙幣抜き取りイベントを通知する)が OFF の場合、設定値にかかわらず回収処理終了にてメソッドを終了とします。

3.2.2. 操作方法

画面下部の「実行」ボタンの押下により変更した設定が反映されます。
「終了」ボタン押下時、設定は反映されず、本設定ツールは終了します。



以降、設定を変更したい場合はインストールCDにある *“opossetup.exe”* を実行してください。

3.3. アンインストール

“C:\¥OPOS¥GLORY¥RT300¥Cleanup.bat”を実行することにより、インストールしたファイルの削除、レジストリに登録した情報の削除・解除を行います。

※パッチファイルの実行では、レジストリの削除処理が行われるため、
Windows 7 以降では「管理者として実行」で実行していただく必要があります。

※Cleanup.bat 実行時に GLORY_Cleanupdir.bat が自動的に呼び出されます。

※インストール先配下にログフォルダ・ファイルがある場合、アンインストールでは全て削除されない場合があります。その場合は、手作業にてフォルダごと削除してください。

※本コントロール用に使用したレジストリについても一部削除されない場合があります。
その場合は、次項に示す構成を参照し、手作業にてフォルダごと削除してください。

3.4. その他

3.4.1. レジストリ仕様

本コントロールソフトインストール時、以下の内容（初期値）をレジストリに設定します。

《 レジストリ構成と説明 》

HKEY_LOCAL_MACHINE

SOFTWARE

Wow6432Node

OLEforRetail

ServiceInfo

GLORY

CashChanger

ServiceOPOS

キー

説明

Device

“C:¥OPOS¥GLORY¥RT300”

※ 1

InstallDir

“C:¥OPOS¥GLORY¥RT300”

※ 1

LogDevName

“CashChanger”

※ 1

CashChanger

キー

説明

CashChanger

論理デバイス名

※ 1

GloryRAD/RT-300

GloryRT-300

インストール時の接続機種選択によりいずれかのクラスがエントリされます。

キー

説明

(標準)

サービスオブジェクトのプログラムティック ID

※ 1

AutoNego

通信速度自動設定
TRUE : 設定する
FALSE : “しない”

※ 2

BaudRate

通信速度 2400/4800/9600

※ 2

BeginDeposit
PreCheck

メソッド開始時のフル・ニアフルチェックの可否
TRUE : チェックする
FALSE : “しない”

※ 2

BeginDeposit
Working

内部で使用する設定値（変更不可）

※ 1

Bill
_FullStatusEvent

紙幣フル時イベント通知設定
STORE : 収納庫フルで通知
NONE : フル通知しない

※ 2

Compatibility

互換/拡張性設定

※ 3

DataEventEnabled

DataEventEnabled プロパティの有効範囲
All : 全てのイベントが対象
DataEvent : Data イベントのみ

※ 2

DataEventLog

DataEvent の時系列ログ出力設定
TRUE : 出力する
FALSE : “しない”

※ 2

DataEventTiming

DataEvent の通知方法を設定する。
“Once” : つり銭機停止時に一回
“Realtime” : 計数データ変化の都度
“Realtime&Notify” : つり銭機停止時＋計数データ変化の都度

※ 2

インストール時の接続機種
選択によりいずれかのクラ
スがエントリされます。

	Description	DeviceDescription プロパティに設定される文字列	※ 1
	DepositErrorEvent	入金中の事象発生を通知するイベントの種類を設定する “DIEVT” : DirectIO イベントで通知 “SUE” : StatusUpdate イベントで通知 “BOTH” : DirectIO イベントと StatusUpdate イベントの両通知	※ 2
	DeviceExitValue	硬貨単体接続時の DeviceExit プロパティ値 1:CurrentExit=1 固定とする 2:CurrentExit=1 または 2 に切り替えを可能とする	※ 4
	DeviceName	接続機種を明示	※ 1
	DirectIOEvent Timing	“Delayed” : 入金中に発生したイベントは DataEven 通知後に発生させる “RealTime” : 検知すると即通知する	※ 2
	EnabledMode	“Adjust” : Enabled 時にモード不一致の補正処理を行う。 “None” : 何もしない “Repay” : モード不一致の補正処理後に、入金額を返却させる。	※ 2
	FixDeposit ReturnTiming	“Sync” : 同期処理の振る舞い “Async” : 非同期 1 (終了条件強) “ASyncStd” : 非同期 (終了条件標準) FixDeposit メソッド参照の事	※ 2
	HardLogSaveDir	処理機ログファイルの出力パス (未使用)	※ 2
	LogSaveCount	1 稼働日分でファイルに保持できる時系列ログの最大件数	※ 2
	LogSaveDate	ログファイルを保持できる日数	※ 2
	LogSaveDir	ログファイルの出力先パス	※ 2
	PauseDeposit ReturnTiming	“Sync” : 同期処理の振る舞い “Async” : 非同期処理の振る舞い PauseDeposit メソッド参照の事	※ 2
	PORT	通信ポート	※ 2
	ReadCashCounts	ReadCashCounts メソッドでの現金在高の取得方法 TRUE : 収納部 + 回収セット FALSE : 収納部のみ	※ 2
	ReadCashCounts Style	在高異常発生時のデータ書式の設定 0 : 0 枚表示 1 : 在高を表示	※ 2
	RealRead	つり銭機トレースログ設定 TRUE : 吐き出しを行う FALSE : “ ” をしない	※ 2
	RepayType	EndDeposit (CHAN_DEPOSIT_REPAY) の返却処理設定 “Amount” : 入金額 (最小構成枚数) での返却処理 “CashList” : 入金計数枚数での返却処理	※ 2

	SEISA_ CommandPreCheck	DirectIO メソッド-(CHAN_DI_SEISA) 実行時の動作中プレチェック TRUE : 動作中はエラー応答とする FALSE : 動作中も実行可能とする	※2
	Service	サービスオブジェクトファイル名	※1
	STATUSREAD_ CommandPreCheck	DirectIO メソッド -(CHAN_DI_STATUSREAD) 実行時の動作中プレチェック TRUE : 動作中はエラー応答とする FALSE : 動作中も実行可能とする	※2
	SupplyCountMode	補充枚数取得コマンドの許可 0 : コマンドを使用する 1 : " しない	※2
	WriteLog	TRUE : ログを行う FALSE : " をしない	※2

※ 実行環境によりレジストリの格納先（構成）が異なります。

32bit 版 : HKEY_LOCAL_MACHINE¥Software 配下に設定されます。

64bit 版 : HKEY_LOCAL_MACHINE¥Software¥Wow6432Node 配下に設定されます。

※1 : 値は固定とします（変更しないでください）。

※2 : インストーラで設定される値です。

※3 : 互換/拡張機能を設定します。

値は 16 進表記を文字列化したものを使用します。各設定に対して BIT 単位でのアサインを行います。

Bit0 : ClaimDevice の完了条件 ON : つり銭機との同期を省略 OFF : 同期させる（標準）

Bit1 : 予約

Bit2 : 貨幣抜き取り完了イベントの通知 ON : 通知する OFF : 通知しない（標準）

Bit3 : 入金中エラー解除時の入金レディイベントの通知 ON : 通知する OFF : 通知しない（標準）

Bit4 : 予約

Bit5 : 入金返却動作時、枚数指定放出でのリカバリー ON : 実行する OFF : 実行しない（標準）

Bit6 : 回収実行時のメソッド終了タイミング ON : 貨幣の抜き取り待ち発生にてメソッドを終了とする OFF : 回収処理終了にてメソッドを終了とする（標準）

Bit2(貨幣抜き取り完了イベントの通知)が OFF の場合、設定値にかかわらず回収処理終了にてメソッドを終了とします。

※4 : 硬貨単体接続時のみ使用します。

レジストリ初期値一覧

レジストリ項目	初期値
AutoNego	"TRUE"
BaudRate	"9600"
BeginDepositPreCheck	"FALSE"
BeginDepositWorking (変更不可)	"Pause"
Bill_FullStatusEvent	"STORE"
Compatibility	"00"
DataEventEnabled	"All"
DataEventLog	"TRUE"
DataEventTiming	Once
DepositErrorEvent	" DIEVT "
Description	"Glory RAD/RT-300 CashChanger" または、"Glory RT-300 CashChanger"
DeviceExitValue	"1"
DeviceName	"GloryRAD/RT-300"または、"GloryRT-300"
DirectIOEventTiming	"Realtime"
EnabledMode	"None"
FixDepositReturnTiming	"Sync"
HardLogSaveDir (未使用)	C:\¥OPOS¥GLORY¥RT300¥HARDLOG
LogSaveCount	"30000"
LogSaveDate	"14"
LogSaveDir	C:\¥OPOS¥GLORY¥RT300¥LOG
PauseDepositReturnTiming	"Sync"
PORT	"COM1"
ReadCashCounts	"TRUE"
ReadCashCountsStyle	"1"
RealRead	"FALSE"
RepayType	"Amount"
SEISA_CommandPreCheck	"TRUE"
Service	"ChangerS0.dll"
STATUSREAD_CommandPreCheck	"TRUE"
SupplyCountMode	"1"
WriteLog	"TRUE"

3.4.2. 定数の利用

「Application Programmer's Guide」で定義されている定数は、OPOS 技術協議会のサイトから取得可能な仕様書に記載されています。

このファイルにつきましては、Web 上の「OPOS 技術協議会ホームページ」からダウンロードすることが出来ます。

また、本コントロールソフトについて独自に定義している定数につきましては、"GLORYOPOS.BAS"等を作成し定義を行います。

プログラムを作成する場合、付録を参照し両ファイルをプロジェクトに追加してください。

注記：

共通 CO として提供されているサイト (http://monroecs.com/oposccos_current.htm) で取得可能な定義ファイル C 言語用 (OposChan.h)、VB 用 (OposAll.bas) で、PauseDeposit メソッドのパラメータが異なります。

弊社 OPOS では、C 言語用の定数値を採用し開発を行っております。

上位アプリケーションを VB にて開発する場合は、パラメータの値を変更してください。

【OposChan.h】

```

////////////////////////////////////
// "PauseDeposit" Method: "Control" Parameter Constants
////////////////////////////////////

const LONG CHAN_DEPOSIT_PAUSE   = 11;
const LONG CHAN_DEPOSIT_RESTART = 12;

```

【OposAll.bas】

```

Rem *////////////////////////////////////
Rem * "PauseDeposit" Property Constants
Rem *////////////////////////////////////

Public Const ChanDepositPause& = 1 11
Public Const ChanDepositRestart& = 2 12

```

値をそれぞれ 1 1 ,
1 2 に変更してご利用ください。

3.4.3. サンプルアプリケーションについて

本アプリケーションはあくまで参考としてご使用ください。同様の制御を行う場合でも、検証や確認を十分に実施してください。

※本アプリケーションの注意事項

・つり銭機の接続形態について

使用前に接続状態を確認しておく必要があります。

(RT-300 単体接続で使用する場合は「RT 単体接続」に接続状態を切り替えてください。)

・つり銭機のモードについて

本シミュレータは預かり金計数モードのみ対応しております。

※つり銭機モードのまま実行した場合、開局処理内で預かり金計数モードに変更されます。

・COM について

COM 番号の設定は、シミュレータではできません。

(OPOS 側のレジストリ設定値の変更で対応できます。)

・ログについて

本アプリケーションを起動すると、アプリケーションと同じディレクトリに

ログファイル（ファイル名：LOGyyyyymmdd. log）が生成されます。

ログファイルには、呼び出したメソッドとその実行結果が出力されます。

ボタン項目一覧

【開局処理】

開局処理を行います。

※アプリケーションからつり銭機を制御するためには、本処理を最初に実行する必要があります。
(本処理が異常の場合は、すべての処理が実行できません。)

処理手順を以下に示します。

- ①Open メソッドの実行
- ②ClaimDevice メソッド実行
- ③DeviceEnabled プロパティ (TRUE) の設定
- ④DirectIO メソッド-CHAN_DI_DEPOSITMODE-の実行

※つり銭機のモードを預かり金計数モードに切り替えます。

上記、全ての処理が成功して開局処理が正常終了となります。

【閉局処理】

閉局処理を行います。

※アプリケーションからつり銭機の制御を終える時に本処理を実行する必要があります。

処理手順を以下に示します。

- ①DeviceEnabled プロパティ (FALSE) の設定
- ②ReleaseDevice メソッド実行
- ③Close メソッドの実行

上記、全ての処理が成功して閉局処理が正常終了となります。

【スキャン開始】

取引を開始します。

処理手順を以下に示します。

- ①DataEventEnabled プロパティ (TRUE) の設定
- ②BeginDeposit メソッドの実行

※処理終了後、つり銭機が預かり金投入可能な状態になります。

【小計】

商品の合計金額を入力しておき、ボタン押下により確定します。

※ボタン押下前に商品合計額を入力しておく必要があります。

【現計】

預かり金額の確定を行い、つり銭の払出処理を行います。

商品合計額＞預かり金額の状態で行われた場合は、預かり金額を返却します。

処理手順を以下に示します。

①FixDeposit メソッドの実行

※入金額を取得する場合、メソッド正常終了後に DepositAmount プロパティを参照してから確定してください。

②EndDeposit メソッドの実行

※パラメータにより、動作が異なります。

- ・ CHAN_DEPOSIT_CHANGE

- ・ CHAN_DEPOSIT_NOCHANGE

⇒ 入金処理を終了します。(どちらのパラメータを指定しても EndDeposit メソッドの動作は同じです。)

- ・ CHAN_DEPOSIT_REPAY

⇒ 預かり金額を返却し、現計処理を終了します。

③CurrentExit プロパティの設定

※つり銭を払い出す場合は“1”（出金口）に設定してください。

④DispenChange メソッドの実行

つり銭を払い出し、現計処理を終了します。

【キャンセル】

取引をキャンセルし、預かり金額を返却します。

処理手順を以下に示します。

①FixDeposit メソッドの実行

※入金額を取得する場合、メソッド正常終了後に DepositAmount プロパティを参照してから確定してください。

②EndDeposit メソッドの実行

パラメータを CHAN_DEPOSIT_REPAY で実行し、預かり金額を返却後、キャンセル処理を終了します。

【返品処理】

ボタン押下で返品処理画面が起動し、金額を入力して返品処理を行います。

処理手順を以下に示します。

①CurrentExit プロパティの設定

※つり銭を払い出す場合は“1”（出金口）に設定してください。

②DispenChange メソッドの実行

つり銭を払い出し、返品処理を終了します。

【在高表示】

ボタン押下で在高確認画面が立ち上がり、現在のつり銭の機内在高を表示します。

処理手順を以下に示します。

①ReadCashCounts メソッドの実行

※ReadCashCounts メソッドでは、収納庫在高と回収カセット在高の合算値が取得できます。

在高を別々に取得したい場合は、DiectIO メソッドの CHAN_DI_SEISA コマンドを実行してください。

【全回収】

つり銭機内の貨幣を全て回収します。

※接続状態が異なった状態で実行した場合は、正常に動作しません。

処理手順を以下に示します。

①DirectIO メソッド-CHAN_DI_COLLECT-の実行

※パラメータにより、回収のモードと対象ユニット（紙幣のみ、硬貨のみ、両ユニット）が設定可能です。

【エラー解除】

つり銭機のエラー発生時にボタンを押下すると、アプリケーション画面の前面に

エラー解除ガイダンスを表示します。

処理手順を以下に示します。

①DirectIO メソッド-CHAN_DI_ERRGUIDANCE-の実行

※パラメータにより、表示位置とポップアップ表示有無が設定可能です。

【終了】

アプリケーションを終了します。

[illegible]

4. APG仕様

4.1. 一覧

プロパティ・メソッド・イベントの一覧を以下に記します。
本コントロールでサポートしない項目については“適用外”と記載しています。

尚、本章に詳細に記載されていない共通・専用プロパティの仕様については、以下のドキュメントを参照下さい。

『OLE for Retail POS Application Programmer's Guide 日本版仕様書 第1.13版』

- ・ 第2章 共通プロパティ、メソッド、イベント
- ・ 第15章 自動釣り銭機

4.1.1. プロパティ

共通	版数	型	アクセス	初期化	備考
AutoDisable	1.2	Boolean	R/W	適用外	
BinaryConversion	1.2	Long	R/W	Open	
CapCompareFirmwareVersion	1.9	Boolean	R	Open	
CapPowerReporting	1.3	Long	R	Open	
CapStatisticsReporting	1.8	Boolean	R	Open	
CapUpdateFirmware	1.9	Boolean	R	Open	
CapUpdateStatistics	1.8	Boolean	R	Open	
CheckHealthText	1.2	String	R	Open	
Claimed	1.2	Boolean	R	Open	
DataCount	1.5	Long	R	Open	
DataEventEnabled	1.5	Boolean	R/W	Open	
DeviceEnabled	1.2	Boolean	R/W	Open & Claim	
FreezeEvents	1.2	Boolean	R/W	Open	
OpenResult	1.5	Long	R	無し	
OutputID	1.2	Long	R	適用外	
PowerNotify	1.3	Long	R/W	Open	
PowerState	1.3	Long	R	Open	
ResultCode	1.2	Long	R	無し	
ResultCodeExtended	1.2	Long	R	Open	
State	1.2	Long	R	無し	
ControlObjectDescription	1.2	String	R	無し	
ControlObjectVersion	1.2	Long	R	無し	
ServiceObjectDescription	1.2	String	R	Open	
ServiceObjectVersion	1.2	Long	R	Open	
DeviceDescription	1.2	String	R	Open	
DeviceName	1.2	String	R	Open	

専用	版数	型	アクセス	初期化	備考
CapDeposit	1.5	Boolean	R	Open	
CapDepositDataEvent	1.5	Boolean	R	Open	
CapDiscrepancy	1.2	Boolean	R	Open	
CapEmptySensor	1.2	Boolean	R	Open	
CapFullSensor	1.2	Boolean	R	Open	
CapJamSensor	1.11	Boolean	R	Open	
CapNearEmptySensor	1.2	Boolean	R	Open	
CapNearFullSensor	1.2	Boolean	R	Open	
CapPauseDeposit	1.5	Boolean	R	Open	
CapRealTimeData	1.11	Boolean	R	Open	
CapRepayDeposit	1.5	Boolean	R	Open	
CurrentService	1.11	Long	R/W	Open	
AsyncMode	1.2	Boolean	R/W	Open	
AsyncResultCode	1.2	Long	R	Open, Claim, & Enable	
AsyncResultCodeExtended	1.2	Long	R	Open, Claim, & Enable	
CurrencyCashList	1.2	String	R	Open	
CurrencyCode	1.2	String	R/W	Open	
CurrencyCodeList	1.2	String	R	Open	
CurrentExit	1.2	Long	R/W	Open	
DepositAmount	1.5	Long	R	Open	
DepositCashList	1.5	String	R	Open	
DepositCodeList	1.5	String	R	Open	
DepositCounts	1.5	String	R	Open	
DepositStatus	1.5	Long	R	Open, Claim, & Enable	
DeviceExits	1.2	Long	R	Open	
DeviceStatus	1.2	Long	R	Open, Claim, & Enable	
ExitCashList	1.2	String	R	Open	
FullStatus	1.2	Long	R	Open, Claim, & Enable	
RealTimeDataEnabled	1.11	Boolean	R/W	Open, Claim, & Enable	
ServiceCount	1.11	Long	R	Open	
ServiceIndex	1.11	Long	R	Open	

4.1.2. メソッド

共通	版数	使用可能条件	備考
Open	1.2	無し	
Close	1.2	Open	
ClaimDevice	1.2	Open	
ReleaseDevice	1.2	Open & Claim	
CheckHealth	1.2	Open, Claim, & Enable	
ClearInput	1.5	Open & Claim, & Enable	
ClearOutput	1.2	適用外	
DirectIO	1.2	Open & Claim	
CompareFirmWareVersion	1.9	Open & Claim, & Enable	
ResetStatistics	1.8	Open & Claim, & Enable	
RetrieveStatistics	1.10	Open & Claim, & Enable	
UpdateFirmware	1.9	Open & Claim, & Enable	
UpdateStatistics	1.10	Open & Claim, & Enable	

専用	版数	使用可能条件	備考
BeginDeposit	1.5	Open, Claim, & Enable	
DispenseCash	1.2	Open, Claim, & Enable	
DispenseChange	1.2	Open, Claim, & Enable	
EndDeposit	1.5	Open, Claim, & Enable	
FixDeposit	1.5	Open, Claim, & Enable	
PauseDeposit	1.5	Open, Claim, & Enable	
ReadCashCounts	1.2	Open, Claim, & Enable	
AdjustCashCounts	1.11	Open & Claim, & Enable	

4.1.3. イベント

名称	版数	使用可能条件	備考
DataEvent	1.5	Open, Claim&Enable	
DirectIOEvent	1.2	Open, Claim	
ErrorEvent	1.2	適用外	
OutputCompleteEvent	1.2	適用外	
StatusUpdateEvent	1.2	Open, Claim, & Enable	

4.1.4. 初期値一覧

Open メソッド成功後に初期化されるプロパティ各初期値一覧を下表に示します。
尚、Open メソッドを呼び出す以前の値については特筆されているプロパティ以外は不定とします。

プロパティ	初期化	備考
BinaryConversion	OPOS_BC_NONE	OPOS_BC_NONE 固定
CapCompareFirmwareVersion	FALSE	変化しません
CapPowerReporting	OPOS_PR_ADVANCED	変化しません
CapStatisticsReporting	FALSE	変化しません
CapUpdateFirmware	FALSE	変化しません
CapUpdateStatistics	FALSE	変化しません
CheckHealthText	<CheckHealth・CheckHealthText>参照	
Claimed	FALSE	TRUE または FALSE
DataCount	0	計数処理が完了するたびにインクリメントする。
DataEventEnabled	FALSE	TRUE または FALSE
DeviceEnabled	FALSE	TRUE または FALSE
FreezeEvents	FALSE	TRUE または FALSE
OpenResult	OPOS_SUCCESS	<ResultCode 一覧>参照
PowerNotify	OPOS_PN_DISABLED	OPOS_PN_DISABLED または OPOS_PN_ENABLED
PowerState	OPOS_PS_UNKNOWN	OPOS_PS_UNKNOWN
		OPOS_PS_ONLINE
		OPOS_PS_OFF
		OPOS_PS_OFFLINE
		OPOS_PS_OFF_OFFLINE (未使用)
ResultCode	OPOS_SUCCESS	<ResultCode 一覧>参照
ResultCodeExtended	0	<ResultCode 一覧>参照
State	OPOS_S_IDLE	OPOS_S_CLOSED
		OPOS_S_IDLE
		OPOS_S_BUSY
		OPOS_S_ERROR (未使用)

ControlObjectDescription	使用するCOが設定を行います。	
ControlObjectVersion		
ServiceObjectDescription	“GLORY CashChanger OPOS Service Object, Copyright(C) 2012 GLORY LTD.”	
ServiceObjectVersion	“1013001”	
DeviceDescription	接続する機種によって値が変化します。 (部は接続機種により、設定されます。) “GLORY RAD/RT-300 CashChanger ” “RAD/RT-300”, “RT-300”	
DeviceName	接続する機種によって値が変化します。 (部は接続機種により、設定されます。) “ GloryRAD/RT-300 ” “RAD/RT-300”, “RT-300”	
CapDeposit	TRUE	変化しません
CapDepositDataEvent	TRUE	変化しません
CapDiscrepancy	TRUE	変化しません
CapEmptySensor	TRUE	変化しません
CapFullSensor	TRUE	変化しません
CapJamSensor	TRUE	変化しません

CapNearEmptySensor	TRUE	変化しません
CapNearFullSensor	TRUE	変化しません
CapPauseDeposit	TRUE	変化しません
CapRealTimeData	TRUE	変化しません
CapRepayDeposit	TRUE	変化しません
CurrentService	0	0 固定

AsyncMode	FALSE	TRUE または FALSE
AsyncResultCode	OPOS_SUCCESS	<ResultCode 一覧>参照
AsyncResultCodeExtended	0	<ResultCode 一覧>参照

CurrencyCashList	<ul style="list-style-type: none"> ・RAD/RT-300 接続時、収納部設定により異なります。 ・RT-300 単体接続時、“1, 5, 10, 50, 100, 500” をセットします。 	
CurrencyCode	“JPY”	変化しません
CurrencyCodeList	“JPY”	変化しません
CurrentExit	1	1 から DeviceExits までの値
DepositCashList	<ul style="list-style-type: none"> ・RAD/RT-300 接続時、収納部設定により異なります。 ・RT-300 単体接続時、“1, 5, 10, 50, 100, 500” をセットします。 	
DepositCodeList	“JPY”	変化しません
DepositCounts	0	計数データをセットします。
DepositAmount	0	計数データをセットします。
DeviceExits	2	
ExitCashList	つり銭機の設定により異なります。 <ul style="list-style-type: none"> ・RAD/RT-300 接続時、収納部設定により異なります。 ・RT-300 単体接続時、“1, 5, 10, 50, 100, 500” をセットします。 	
DepositStatus	初期化時のつり銭機の状態により異なります。 CHAN_STATUS_DEPOSIT_START CHAN_STATUS_DEPOSIT_END CHAN_STATUS_DEPOSIT_NONE CHAN_STATUS_DEPOSIT_COUNT CHAN_STATUS_DEPOSIT_JAM	
DeviceStatus	初期化時のつり銭機の状態により異なります。 CHAN_STATUS_OK CHAN_STATUS_EMPTY CHAN_STATUS_NEAREMPTY CHAN_STATUS_JAM	
FullStatus	初期化時のつり銭機の状態により異なります。 CHAN_STATUS_OK CHAN_STATUS_FULL CHAN_STATUS_NEARFULL	
RealTimeDataEnabled	TRUE	TRUE または FALSE
ServiceCount	0	0 固定
ServiceIndex	0	0 固定

4.2. プロパティ

4.2.1. 共通プロパティ

BinaryConversion プロパティ R/W

- ・ 形式 LONG BinaryConversion;

値	意味
<i>OPOS_BC_NONE</i>	データは変換されずに BString 一文字は 1 バイトで格納されます。
<i>OPOS_BC_NIBBLE</i>	本デバイスではサポートしません。
<i>OPOS_BC_DECIMAL</i>	本デバイスではサポートしません。

- ・ 説明 (*OPOS_BC_NONE* 以外を) 設定時、本プロパティに影響されるプロパティ・メソッドのデータフォーマット (文字列データ) に対して、以下の処理を行います。
 - ◆ プロパティの設定とパラメータの受け渡し前に、文字列データを BinaryConversion 値に指定されたフォーマットに変換する。
 - ◆ プロパティの設定とパラメータの受け渡し後に、文字列データを BinaryConversion 値に指定されたフォーマットから変換する。

- ・ 戻り値 本プロパティ設定時、次の値のいずれかが **ResultCode** プロパティに格納されます。

ResultCode	意味
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_ILLEGAL	不正な値が設定されました。 ※1

※1 *OPOS_BC_NONE* 以外が指定されていないか確認してください。

- ・ 備考 ※ 本コントロールソフトにおいては、各プロパティ、メソッドにて受け渡しされる文字列の変換はサポートしません (*OPOS_BC_NONE* 以外はサポートしません)。
- ・ 参照 ◇ APG 「OLE for Retail POS コントロール概要」

DataEventEnabled プロパティ R/W

- ・ 形式 BOOL DataEventEnabled;

値	意味
TRUE	入力データの受信で、即イベントが通知されます。
FALSE	即イベント通知はされずに、後で通知できるようにバッファリングされます。

- ・ 説明 イベントを即時通知させるか、バッファリングさせておくかを設定できます。
デバイスのオープン後、本プロパティは FALSE に初期化されます。
バッファリングされていた状態でプロパティを TRUE に変更すると、即座にイベント通知が行われます。

- ・ 戻り値 本プロパティ設定時、次の値のいずれかが **ResultCode** プロパティに格納されます。

ResultCode	意味
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。

- ・ 備考 ※ レジストリキー “DataEventEnabled” の値を “All” に設定した場合、全てのイベントに対してプロパティ設定値が有効になります。
（FALSE に設定した場合、DataEvent キューイング後は全てのイベントもキューイングされます。）
設定値を “DataEvent” に設定していた場合は、DataEvent のみが有効となります。
（FALSE に設定した場合、DataEvent のみがキューイングされ、StatusUpdateEvent、DirectIOEvent は即イベント通知されます。）

- ・ 参照 ◇ APG 「OLE for Retail POS コントロール概要」

DeviceEnabled プロパティ R/W

- ・ 形式 BOOL DeviceEnabled;

値 意味

TRUE デバイスは Enable（動作）状態です。TRUE を設定すると Enable にされます。
FALSE デバイスは Disable 状態です。FALSE を設定すると、Disable にされます。

- ・ 説明 デバイスのオープン後、デバイスは Disable 状態になっています。
 デバイスを使用する場合、Enable にする必要があります。
 また、ClaimDevice メソッドを実行した後でなければ Enable にすることはできません。

- ・ 戻り値 本プロパティ設定時、次の値のいずれかが ResultCode プロパティに格納されます。

ResultCode 意味

OPOS_SUCCESS	プロパティの設定に成功しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_FAILURE	プロパティの設定に失敗しました。	※2

※1 ClaimDevice メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 以下の状態でないか確認してください。
 ・ 釣り銭機電源 OFF（電源 OFF・ON 時のイニシャルリセット動作中も含む）
 ・ 釣り銭機が管理モードで実行中
 ・ 通信ケーブルの回線断

- ・ 備考 ※ レジストリキー “EnabledMode” の値を “Adjust” に設定した場合、Enable（TRUE）設定時、OPOS は釣り銭機の状態をチェックし、入金中状態なら入金終了を釣り銭機に指示し、待機中に戻します。
 これは、前回の入金取引の途中で強制的にアプリケーションが終了され（OPOS の Close メソッドが実行された）、次回起動時にアプリケーションがニュートラルで釣り銭機側が入金中状態といったモードずれを抑止する為に行います。
 計数途中の状態で釣り銭機に入金されていた貨幣は返却されません。
 ※ レジストリキーの “EnabledMode” 値が “Repay” の場合、“Adjust” と同様にモード不一致の補正処理を実行後、計数途中の状態で釣り銭機に入金されていた金額が返却されます。
 この時、返却する貨幣は EndDeposit（CHAN_DEPOSIT_REPAY）時と同様、レジストリの “RepayType” の設定値により、最小構成枚数で払い出し、又は、実際入金された金種・枚数が払い出されます。
 ※ レジストリ “EnabledMode” での入金解除（返却）処理は Open 実行後の初回のみ有効となります。
 ※ RT-300 単体接続では、DeviceEnabled→TRUE 設定時に、OPOS が内部的に硬貨の SSW 設定を実行し、動作不可をあらわすレスポンスキャラクタを変更しています。
 （各メソッド実行時に動作不可の場合、ResultCodeExtended に OPOS_ECHAN_IMPOSSIBLE をセットします。）
- ・ 参照 ◇ APG 「OLE for Retail POS コントロール概要」

FreezeEvents プロパティ R/W

- ・ 形式 BOOL FreezeEvents;

値	意味
---	----

TRUE	イベントを通知しません。(保持しておく。)
------	-----------------------

FALSE	イベントを通知します。
-------	-------------

- ・ 説明 イベントを通知させるか、バッファリングさせておくかを設定できます。
デバイスのオープン後、本プロパティは FALSE に初期化されます。
バッファリングされていた状態でプロパティを FALSE に変更すると、保持されたイベントが通知されます。

- ・ 戻り値 本プロパティ設定時、次の値のいずれかが **ResultCode** プロパティに格納されます。

ResultCode	意味
------------	----

OPOS_SUCCESS	プロパティの設定に成功しました。
--------------	------------------

OPOS_E_CLOSED	デバイスがオープンされていません。
---------------	-------------------

- ・ 備考

- ・ 参照 ◇ APG 「OLE for Retail POS コントロール概要」

PowerNotify プロパティ R/W

- ・ 形式 LONG PowerNotify;

値	意味
OPOS_PN_DISABLED	電源状態通知を行いません。 (StatusUpdateEvent での通知、PowerState プロパティへの設定を行いません)
OPOS_PN_ENABLED	電源状態通知を行います。 (StatusUpdateEvent での通知、PowerState プロパティの更新を行います)

・ 説明 デバイスのオープン後は、電源通知が行われない状態になっています。
本プロパティはデバイスがディセーブルの間、すなわち DeviceEnabled プロパティが FALSE の間のみ設定が可能です。

- ・ 戻り値 本プロパティ設定時、次の値のいずれかが ResultCode プロパティに格納されます。

ResultCode	意味
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_ILLEGAL	デバイスが Enable になっています。 不正な値が指定されました。

※ 1 DeviceEnabled プロパティを FALSE に設定してから実行してください。

※ 2 設定した値を確認してください。

- ・ 備考 ※ 電源通知を行う際の、PowerState プロパティと StatusUpdateEvent イベントの値を以下に示します。

[PowerState プロパティの値]

デバイス状態	プロパティ値	(参考) デバイスのレスポンス (伝送キャラクタ)
正常待機中	OPOS_PS_ONLINE	06H (ACK)
ニアエンプティ		17H (ETB)
計数動作中		01H (SOH)
計数待機中		19H (EM)
セット外れ	OPOS_PS_OFFLINE	13 (DC3)
補充中		14H (DC4)
動作中		1AH (SUB)
エラー中		18H (CAN)
貨幣抜き取り待ち		12H (DC2)
無応答	OPOS_PS_OFF	—

[StatusUpdateEvent イベントの通知タイミングとその値]

前回取得状態から変化があった場合、StatusUpdateEvent を通知します。
その時、PowerState プロパティの内容も更新します。

		今回取得した状態		
		OPOS_PS_ONLINE	OPOS_PS_OFFLINE	OPOS_PS_OFF
前回 状態	OPOS_PS_ONLINE	無し	OPOS_SUE_POWER_OFFLINE	OPOS_SUE_POWER_OFF
	OPOS_PS_OFFLINE	OPOS_SUE_POWER_ONLINE	無し	OPOS_SUE_POWER_OFF
	OPOS_PS_OFF	OPOS_SUE_POWER_ONLINE	OPOS_SUE_POWER_OFFLINE	無し

- ・ 参照
 - ◇ APG 「デバイス電源通知モデルについて」
 - ◇ PowerState プロパティ

4.2.2. 専用プロパティ

CurrencyCode プロパティ R/W

- ・形式 BSTR CurrencyCode;
- ・説明 つり銭機の操作で使用されている通貨コードが設定されます。
有効な値は、CurrencyCodeList プロパティに記される通貨の中の一つです。
- ・戻り値 本プロパティ設定時、次の値のいずれかが ResultCode プロパティに格納されます。

<i>ResultCode</i>	<i>意味</i>
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_ILLEGAL	CurrencyCodeList 以外の値が指定されました。 ※1

※1 CurrencyCodeList プロパティの内容を確認してください。

- ・備考 ※ 本ソフトは CurrencyCodeList = “JPY” のみをサポートします。
- ・参照 ◇ CurrencyCodeList プロパティ

CurrentExit プロパティ R/W

- ・ 形式 LONG CurrentExit;
- ・ 説明 現在の払い出し口を示します。
適切な値は 1 から DeviceExits までの間です。
- ・ 戻り値 本プロパティ設定時、次の値のいずれかが ResultCode プロパティに格納されます。

ResultCode	意味
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_ILLEGAL	無効な CurrentExit の値が指定されました。 ※1

※1 DeviceExits プロパティの内容を確認してください。

- ・ 備考
 - ※ 本ソフトの DeviceExits の最大値は = 2 です。
 - ※ 以下のようにプロパティの値とメソッドの組み合わせにより、つり銭機に対して発行されるコマンドと払い出し口が異なります。

CurrentExits			DispenseCash		DispenseChange	
			コメント	払出口	コメント	払出口
	1	紙幣 硬貨	枚数指定放出	出金口 放出トレイ	金額指定放出	出金口 放出トレイ
	2	紙幣 硬貨	回収	回収セット 放出トレイ	回収	回収セット 放出トレイ

- ・ 参照 ◇ DeviceExits プロパティ

CurrentService プロパティ R/W

- ・ 形式 LONG CurrentService;
- ・ 説明 現在のサービスを示します。設定値 0 なら基本サービスを示します。
0 より大きく ServiceCount 以下なら統合サービスの情報要求に使用されます。
- ・ 戻り値 本プロパティ設定時、次の値のいずれかが **ResultCode** プロパティに格納されます。

<i>ResultCode</i>	<i>意味</i>
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_ILLEGAL	ServiceCount 以外の値が指定されました。。 ※1

※ 1 **ServiceCount** プロパティの内容を確認してください。

- ・ 備考 ※ **本ソフトは ServiceCount = 0 のみをサポートします。**
- ・ 参照 ◇ ServiceCount プロパティ
 ◇ ServiceIndex プロパティ

RealTimeDataEnabled プロパティ R/W

- ・ 形式 BOOL RealTimeDataEnabled;
- ・ 説明 計数データ（DepositAmount、DepositCounts プロパティ）の更新タイミングを設定します。
TRUE の場合、計数変化の都度、プロパティが更新されます。
FALSE の場合、計数終了時（FixDeposit 実行時）のみ、プロパティが更新されます。
本プロパティにより、BeginDeposit-EndDeposit 間の状態変化に対する混乱を防ぎます。
- ・ 戻り値 本プロパティ設定時、次の値のいずれかが **ResultCode** プロパティに格納されます。

ResultCode	意味
OPOS_SUCCESS	プロパティの設定に成功しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
- ・ 備考 ※ **CapRealTimeData** プロパティ=TRUE 時のみサポートします。
- ・ 参照 ◇ CapRealTimeData プロパティ

4.3. メソッド

4.3.1. 共通メソッド

※DirectIO メソッドは共通メソッドに分類されますが、標準コントロールオブジェクトのサポートしていないデバイスの機能をサポートしている特性より、専用メソッドの項目に記載しています。

Open メソッド

- ・形式 LONG Open (BSTR DeviceName);

パラメータ 意味

DeviceName オープンするデバイス名を指定します
アプリケーション側では、“CashChanger” を指定します。

DeviceName パラメータは、オープンするデバイス名（通常“CashChanger”）を指定します。“CashChanger” キーに設定されている値（文字列）を参照してどの接続パターンで起動するかを決定します。

※本コントロールソフトでは、以下の何れかの値（文字列）がインストール時に設定されます。

“GloryRAD/RT-300”
“GloryRT-300”

- ・説明 デバイスをオープンするときに呼び出します。
デバイス名は、このコントロールオブジェクトがサポートするデバイスのどれを使用するかを指定します。DeviceName はこのデバイスクラスのシステムレジストリに存在しなければなりません。デバイス名と物理デバイスの関係は、オペレーティングシステムレジストリ内のエントリで決まります。これらのエントリはインストーラ又は設定ツール（OPOSSetup.exe）にて更新します。

Open メソッドが成功すると、Claimed、DeviceEnabled、DataEventEnabled、FreezeEvents のプロパティと、OPOS コピーライト記述およびバージョン番号が設定されます。その他のクラス固有プロパティも初期化されます。

- ・戻り値 次の値のいずれかが戻され、ResultCode プロパティにも格納されます。
さらに詳細の結果が、OpenResult プロパティに格納され、参照する事ができます。

ResultCode	OpenResult	意味
OPOS_SUCCESS	0	オープンに成功しました。
OPOS_E_ILLEGAL	OPOS_OR_ALREADYOPEN	該当のコントロールはすでにオープンしています。
OPOS_E_NOEXIST	OPOS_OR_REGBADNAME	<ul style="list-style-type: none"> 指定された DeviceName のレジストリ情報が見つかりません。（空の文字列で指定されている） ※1 レジストリキーがオープンできません。
OPOS_E_NOSERVICE	OPOS_OR_REGPROGID	<ul style="list-style-type: none"> 論理デバイス名のデータが設定されていません。 ※2 レジストリキーがオープンできません。 ※3 S0 のプログラマティック ID が取得できません。（又は NULL の状態である） ※4 取得した S0 のプログラマティック ID に対応した S0 のクラス ID が取得できません。

OPOS_OR_CREATE	取得した S0 のプログラマティック ID に対応した S0 と接続 できません。	※4
OPOS_OR_BADIF	取得した S0 のプログラマティック ID に対応した S0 とメソッドを実行する為のインタフェースで異常が発生しました。	※4
OPOS_ORS_CONFIG	ログ 書き込み件数を示す数値が不正です ログファイル保存件数を示す数値が不正です レジストリの各値が正しく設定されていません 各値がレジストリに存在しません	※5
OPOS_ORS_NOPORT	ポートをオープンすることができません。他のプロセスが既にオープンしています。	※6

※ 先述「3.4 その他 - レジストリ仕様」を参照し、レジストリの設定値を確認してください。

¥HKEY_LOCAL_MACHINE	└	
¥SOFTWARE		※1
¥OLEforRETAIL		
¥ServiceOPOS	└	
¥CashChanger	---	[CashChanger]キーの内容 ※2
¥GloryXXXXX		※3
	---	[標準]キーの内容 ※4
	---	その他各キーの内容 ※5～7

- ※1 レジストリの構成を確認し、クラス及びキーがある事を確認してください。
- ※2 上図に示す [GloryXXXXX](#) はインストール時の接続機種の選択により以下の何れかがクラス/キー値がエントリされていることを確認してください。
- ※3 “GloryRAD/RT-300”
“GloryRT-300”
- ※4 プログラマティック ID は “CHANGERSO.CASHCHANGER” となっている事を確認下さい。
- ※5 各キーの設定値を確認してください。
- ※6 ・ [PORT] キーの設定を確認して下さい。
・ アプリケーションを実行する端末で設定されている COM ポートが正常に認識されているか確認してください。
・ 他のプロセスで設定されている COM ポートとして既に使用中となっていないか確認してください。

- ・ 備考
- ※ [OpenResult](#) プロパティが、OPOS_OR_～ 示すエラーについては C0 がセットします。
(OPOS_OR_ALREADYOPEN を除く)
[OPOS_ORS_～を示すエラーについては、S0 がセットします。](#)
- ※ レジストリ “AutoNego” の設定値が “TRUE” の場合のみ、Open メソッド正常終了後、ドライバ-つり銭機間で通信確立チェックを行います。
もし、ドライバの通信速度 (レジストリ:Baudrate) とつり銭機の通信速度が一致していない場合は、ドライバの通信速度を変更し再チェックを行います。それでも一致しなかった場合は通信未確立となります。
- ※ つり銭機の入替えを行った場合は、内部での設定値更新処理が必要な為、一度 Close メソッドで切断後、再度本メソッドを実行してください。
- ・ 参照
- ◇ APG 「OLE for Retail POS コントロール概要」
- ◇ APG 付録 A 「OPOS レジストリの利用法」

Close メソッド

- ・ 形式 LONG Close ();
- ・ 説明 デバイスとそのリソースを解放するときに呼び出します。
これによりアプリケーションとデバイスが切断され、これ以後このデバイスは次回の Open 処理を行うまで使用できません。
Close メソッドを実行した結果は、Close メソッドの戻り値として返されます。
また、**ResultCode** にも格納されていますので参照することができます。
- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

<i>ResultCode</i>	<i>意味</i>
OPOS_SUCCESS	デバイスがディセーブルにされ、クローズしました。
OPOS_E_CLOSED	デバイスがオープンされていません。
その他	ResultCode の項目を参照してください。

- ・ 備考 **Close メソッドの実行前に、以下の項目について注意が必要です。**
 - ※ **アプリケーション側での処理が滞留していないこと。**
プロセス間、スレッド間通信による（ブロードキャスト等の）メッセージの滞留は OLE へのサービスの障害要因となり、正常終了できない場合があります。

[対処法] VB での **DoEvents** に相当する処理を行う等。
 - ※ **デバイスの動作の完了後に実行すること。**

[対処法] **State** プロパティ、**DeviceStatus** プロパティ等を参照して本コントロールソフト及びデバイス状態が正常（レディ状態）であること確認してください。
- ・ 参照 ◇ APG 「OLE for Retail POS コントロール概要」
 ◇ Open メソッド

ClaimDevice メソッド

- ・ 形式 `LONG ClaimDevice (LONG Timeout);`

パラメータ	意味
Timeout	排他アクセス件を獲得するまでの最大待ち時間（ミリ秒単位）を指定します。 ゼロの場合、メソッドはデバイスの排他アクセスができなかった場合でも、その結果を直ちに返します。 OPOS_FOREVER(-1)が設定されている場合は、メソッドは排他アクセスが獲得できるまで必要なだけ待ちます。

- ・ 説明 アプリケーションは、デバイスに対して排他アクセスを要求する時に、このメソッドを使用します。
本デバイスは、排他アクセス権を獲得していなければ使用できません。
成功すると、Claimed プロパティは TRUE に設定されます。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_CLOSED	デバイスがオープンされていません。 ※1
OPOS_E_TIMEOUT	他のプロセスがデバイスに排他アクセスしています。 ※2
	指定時間内に排他アクセス権の獲得に失敗しました。 ※3
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。
	OPOS_ECHAN_BUSY 処理機動作中です。 ※4

※ 1 **Open** メソッド成功後に実行してください。

※ 2 他のデバイスがアクセス権を解放後実行してください

※ 3 **Timeout** の値を大きめに設定してください。
本コントロールは **Open** メソッド完了後からデバイス（つり銭機）のイニシャル（メモリリード等）を内部的に開始します。その完了を ClaimDevice メソッドの成功条件の一つとしている為、指定時間を考慮する必要があります。
推奨値は 5秒以上（注）とします。
注）2400BPS で通信手順のマージを考慮した最大値となります。
又、指定された時間分待つわけではありません。処理が成功すればその時点でメソッドは終了します。

※ 4 処理機の状態を確認してください。

- ・ 備考 ※ **ClaimDevice** されているか否かの情報は **Claimed** プロパティで参照ができます。
TRUE 排他アクセス権を獲得しました
FALSE 排他アクセス権を獲得していません
- ※ 設定ツール（OPOSSetup.exe）の [その他] **ClaimDevice 実行時、レジつり銭機との同期処理を省略する** の設定により正常終了の条件が異なります。
- ※ OPOS 内部でつり銭機に時間設定コマンドを送信してます。
- ・ 参照 ◇ APG 「デバイス共有モデルについて」
◇ ReleaseDevice メソッド

ReleaseDevice メソッド

- ・ 形式 LONG ReleaseDevice ();
- ・ 説明 アプリケーションは、デバイスの排他的なアクセス権を開放する時に、このメソッドを使用します。
- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

<i>ResultCode</i>	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_ILLEGAL	排他アクセスを取得していません。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。 ※1

※1 処理が終了してから実行してください。

- ・ 備考 ※ *ReleaseDevice* されているか否かの情報は *Claimed* プロパティで参照ができます。
TRUE 排他アクセス権を獲得しています
FALSE 排他アクセス権を解放しました
- ・ 参照 ◇ APG 「デバイス共有モデルについて」
 ◇ ClaimDevice メソッド

CheckHealthメソッド・(CheckHealthTextプロパティ)

- ・ 形式 `LONG CheckHealth (LONG Level);`

パラメータ	意味
Level	デバイスで実行する自己診断のレベルを指定します。
OPOS_CH_INTERNAL	デバイスを物理的に使用しないヘルスチェックを行います。 DeviceStatus と FullStatus のみをチェックし、結果を CheckHealthText プロパティに格納します。
OPOS_CH_EXTERNAL	デバイスを使用してテストを実行します。 出金口から出金可能な金種(ExitCashList の金種)各1枚ずつの出金を行います(注)。 その結果を CheckHealthText プロパティに格納します。
OPOS_CH_INTERACTIVE	本コントロールはモーダルダイアログボックス表示させて、テストオプションと結果を表示させます。 ダイアログボックスに入力された金額／各種金種枚数を放出します(注)。 その出金処理の結果を CheckHealthText プロパティに格納します。

注) **CurrentExit=1** の条件で **DispenseCash** メソッドを内部的に実行します。
1金種でも0枚の場合、エラーとなります。

- ・ 説明 デバイスの状態をテストするときに呼び出します。
このメソッドの結果は**CheckHealthText**プロパティに格納されます。
本メソッドは常に同期です。

[OPOS_CH_INTERNAL] 時に格納される文字列

DeviceStatus	FullStatus	CheckHealthText
CHAN_STATUS_OK	CHAN_STATUS_OK	"Internal Hcheck:Successful"
	CHAN_STATAUS_NEARFULL	"Internal Hcheck:NearFull"
	CHAN_STATAUS_FULL	"Internal Hcheck:Full"
CHAN_STATUS_EMPTY	無視されます	"Internal Hcheck:Empty"
CHAN_STATUS_NEAREMPTY	無視されます	"Internal Hcheck:NearEmpty"
CHAN_STATUS_JAM	無視されます	"Internal Hcheck:Jam"

※ **FullStatus** プロパティは**DeviceStatus**がCHAN_STATUS_OKの場合のみ参照されます。

[OPOS_CH_EXTERNAL] 時に格納される文字列

<i>resultcode</i>	<i>resultcodeextended</i>	<i>CheckHealthText</i>
OPOS_SUCCESS		"External Hcheck:Successful"
OPOS_E_BUSY		"External Hcheck:ControlBusy"
OPOS_E_ILLEGAL		"External Hcheck:Illegal" ※ 1
OPOS_E_NOHARDWARE		"External Hcheck:NoHardWare"
OPOS_E_NOTCLAIMED		"External Hcheck:NotClaimed"
OPOS_E_DISABLED		"External Hcheck:Disabled"
OPOS_E_OFFLINE		"External Hcheck:Offline" ※ 1
OPOS_E_EXTENDED	OPOS_ECHAN_OVERDISPENSE	"External Hcheck:OverDispense"
	OPOS_ECHAN_TOTALOVER	"External Hcheck:TotalOver" ※ 1
	OPOS_ECHAN_CHANGEERROR	"External Hcheck:ChangeError" ※ 1
	OPOS_ECHAN_OVER	"External Hcheck:Over" ※ 1
	OPOS_ECHAN_IFERROR	"External Hcheck:IFError"
	OPOS_ECHAN_SETError	"External Hcheck:SetError"
	OPOS_ECHAN_ERROR	"External Hcheck:Error"
	OPOS_ECHAN_CHARGING	"External Hcheck:Charging"
	OPOS_ECHAN_NEAREMPTY	"External Hcheck:NearEmpty" ※ 1
	OPOS_ECHAN_EMPTY	"External Hcheck:Empty" ※ 1
	OPOS_ECHAN_NEARFULL	"External Hcheck:NearFull" ※ 1
	OPOS_ECHAN_FULL	"External Hcheck:Full"
	OPOS_ECHAN_OVERFLOW	"External Hcheck:OverFlow" ※ 1
	OPOS_ECHAN_REJECT	"External Hcheck:Reject" ※ 1
	OPOS_ECHAN_BUSY	"External Hcheck:Busy"
	OPOS_ECHAN_ASYNCBUSY	"External Hcheck:AsyncBusy" ※ 1
	OPOS_ECHAN_CASSETTEWAIT	"External Hcheck:CassetteWait"
	OPOS_ECHAN_COLLECTWAIT	"External Hcheck:CollectWait" ※ 1
	OPOS_ECHAN_COUNTERERROR	"External Hcheck:CounterError" ※ 1
	OPOS_ECHAN_AMOUNTERROR	"External Hcheck:AmountError" ※ 1
	OPOS_ECHAN_IMPOSSIBLE	"External Hcheck:Impossible"
	OPOS_ECHAN_CANNOTPAY	"External Hcheck:Cannotpay" ※ 1
	OPOS_ECHAN_NOTSTORE	"External Hcheck:NotStore" ※ 1
	OPOS_ECHAN_NEAUTRAL	"External Hcheck:Neutral" ※ 1
	OPOS_ECHAN_DEPOSIT	"External Hcheck:Deposit"
	OPOS_ECHAN_PAUSEDEPOSIT	"External Hcheck:PauseDeposit"

- ※ 上表で示した *resultcode*, *resultcodeextended* は本コントロール内部の情報を示すものであり、ResultCode プロパティ・ResultCodeExtended プロパティを表すものではありません。
- ※ 各金種を 1 枚ずつ放出する際に、エンプティ金種がある場合は放出来ません。
- ※ 1 未使用の項目です。

[OPOS_CH_INTERACTIVE] 時に格納される文字列

<i>resultcode</i>	<i>resultcodeextended</i>	<i>CheckHealthText</i>
OPOS_SUCCESS		"Interactive Hcheck:Successful"
OPOS_E_BUSY		"Interactive Hcheck:ControlBusy"
OPOS_E_ILLEGAL		"Interactive Hcheck:Illegal"
OPOS_E_NOHARDWARE		"Interactive Hcheck:NoHardWare"
OPOS_E_NOTCLAIMED		"Interactive Hcheck:NotClaimed"
OPOS_E_DISABLED		"Interactive Hcheck:Disabled"
OPOS_E_OFFLINE		"Interactive Hcheck:Offline"
OPOS_E_EXTENDED	OPOS_ECHAN_OVERDISPENSE	"Interactive Hcheck:OverDispense"
	OPOS_ECHAN_TOTALOVER	"Interactive Hcheck:TotalOver" ※ 1
	OPOS_ECHAN_CHANGEERROR	"Interactive Hcheck:ChangeError" ※ 1
	OPOS_ECHAN_OVER	"Interactive Hcheck:Over" ※ 1
	OPOS_ECHAN_IFERROR	"Interactive Hcheck:IFError"
	OPOS_ECHAN_SETERROR	"Interactive Hcheck:SetError"
	OPOS_ECHAN_ERROR	"Interactive Hcheck:Error"
	OPOS_ECHAN_CHARGING	"Interactive Hcheck:Charging"
	OPOS_ECHAN_NEAREMPTY	"Interactive Hcheck:NearEmpty" ※ 1
	OPOS_ECHAN_EMPTY	"Interactive Hcheck:Empty" ※ 1
	OPOS_ECHAN_NEARFULL	"Interactive Hcheck:NearFull" ※ 1
	OPOS_ECHAN_FULL	"Interactive Hcheck:Full"
	OPOS_ECHAN_OVERFLOW	"Interactive Hcheck:OverFlow" ※ 1
	OPOS_ECHAN_REJECT	"Interactive Hcheck:Reject" ※ 1
	OPOS_ECHAN_BUSY	"Interactive Hcheck:Busy"
	OPOS_ECHAN_ASYNCBUSY	"Interactive Hcheck:AsyncBusy" ※ 1
	OPOS_ECHAN_CASSETTEWAIT	"Interactive Hcheck:CassetteWait"
	OPOS_ECHAN_COLLECTWAIT	"Interactive Hcheck:CollectWait" ※ 1
	OPOS_ECHAN_COUNTERERROR	"Interactive Hcheck:CounterError" ※ 1
	OPOS_ECHAN_AMOUNTERERROR	"Interactive Hcheck:AmountError" ※ 1
	OPOS_ECHAN_IMPOSSIBLE	"Interactive Hcheck:Impossible"
	OPOS_ECHAN_CANNOTPAY	"Interactive Hcheck:Cannotpay" ※ 1
	OPOS_ECHAN_NOTSTORE	"Interactive Hcheck:NotStore" ※ 1
	OPOS_ECHAN_NEUTRAL	"Interactive Hcheck:Neutral" ※ 1
	OPOS_ECHAN_DEPOSIT	"Interactive Hcheck:Deposit"
	OPOS_ECHAN_PAUSEDEPOSIT	"Interactive Hcheck:PauseDeposit"

※ 上表で示した *resultcode*, *resultcodeextended* は本コントロール内部の情報を示すものであり、ResultCode プロパティ・ResultCodeExtended プロパティを表すものではありません。

※ 1 未使用の項目です。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

<i>ResultCode</i>	意味	
OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_ILLEGAL	指定されたパラメータをサポートしていません。 キャンセルされました。	※1
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。	※2
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※3
OPOS_E_DISABLED	デバイスが Disable 状態です。	※4

- ※ 1 Interactive モードにて **CheckHealth** メソッドを実行しましたが、ダイアログボックスのキャンセルボタンを押下したため、処理が行われませんでした、
- ※ 2 処理が終了してから実行してください。
- ※ 3 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※ 4 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考 ※ 本メソッドは *CurrentExit* プロパティに関係なく放出口に放出します。

- ・ 参照 ◇ CheckHealthText プロパティ

ClearInput メソッド

- ・ 形式 LONG ClearInput ();
- ・ 説明 アプリケーションは、バッファリングされているデバイス入力データをすべてクリアする時に、本メソッドを使用します。
保留になっていたデータイベントや入力エラーイベントはクリアされます。
- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

<i>ResultCode</i>	<i>意味</i>	
OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_DISABLED	デバイスが Disable 状態です。	※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。	

※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考
- ・ 参照 ◇ APG 「入力モデルについて」

CompareFirmWareVersion メソッド ※1

- ・ 形式 `LONG CompareFirmWareVersion (STRING FirmWareFileName, LONG Result);`

パラメータ *意味*

<i>FirmWare</i>	デバイスのファームウェアとバージョンを比較させる為、
<i>FileName</i>	対象のファームウェアファイル名を指定します。
<i>Result</i>	比較結果がセットされます。

- ・ 説明 引数で示されるファイルに含まれるファームウェアのバージョンと物理デバイスに含まれるファームウェアのバージョンの比較を行います。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode *意味*

OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_DISABLED	デバイスが Disable 状態です。	※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。 (CapCompareFirmwareVersion が false)	

※ 1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※ 2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考

- ・ 参照 ◇ CapCompareFirmwareVersion プロパティ

※1 本メソッドは未サポートです (CapCompareFirmwareVersion プロパティが FALSE 固定です)。

ResetStatistics メソッド ※1

- ・ 形式 `LONG ResetStatistics (BSTR StatisticsBuffer);`

パラメータ *意味*

<i>Statistics</i>	リセットするスタティスティクスを定義したデータバッファ。
<i>Buffer</i>	

- ・ 説明 デバイスの中で定義されたリセット可能なスタティスティクスをリセットします。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode *意味*

OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_DISABLED	デバイスが Disable 状態です。	※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。 (CapStatisticsReporting, CapUpdateStatistics プロパティの いずれかが false)	

※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考

- ・ 参照
 - ◇ CapStatisticsReporting プロパティ
 - ◇ CapUpdateStatistics プロパティ

※1 本メソッドは未サポートです (CapStatisticsReporting, CapUpdateStatistics プロパティが共に FALSE 固定です)。

RetrieveStatisticsメソッド ※1

- ・ 形式 `LONG RetrieveStatistics (BSTR *pStatisticsBuffer);`

パラメータ	意味
<i>pStatisticsBuffer</i>	取得するスタティスティクスを定義したデータバッファであり、かつ通知されるスタティスティクスを格納する場所。

- ・ 説明 デバイスから指定されたスタティスティクスを取得します。
- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。 ※1
OPOS_E_DISABLED	デバイスが Disable 状態です。 ※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。 (CapStatisticsReporting プロパティが false)

※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考
- ・ 参照
 - ◇ APG 「入力モデルについて」
 - ◇ CapStatisticsReporting プロパティ

※1 本メソッドは未サポートです (CapStatisticsReporting プロパティが FALSE 固定です)。

UpdateFirmware メソッド ※1

- ・ 形式 `LONG UpdateFirmware (STRING FirmWareFileName);`

パラメータ *意味*

<i>FirmWare</i>	デバイスにダウンロードするファームウェアファイルを指定します。
<i>FileName</i>	

- ・ 説明 引数で示されるファイルに含まれるファームウェアのバージョンにデバイスのファームウェアを更新します。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode *意味*

OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_DISABLED	デバイスが Disable 状態です。	※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。 (CapCompareFirmwareVersion が false)	

※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考

- ・ 参照 ◇ CapCompareFirmwareVersion プロパティ

※1 本メソッドは未サポートです (CapComperFirmwareVersion プロパティが FALSE 固定です)。

UpdateStatistics メソッド ※1

- ・ 形式 `LONG UpdateStatistics (BSTR StatisticsBuffer);`

パラメータ *意味*

<i>Statistics</i>	デバイスのファームウェアとバージョンを比較させる為、
<i>Buffer</i>	対象のファームウェアファイル名を指定します。

- ・ 説明 引数で示されるファイルに含まれるファームウェアのバージョンと物理デバイスに含まれるファームウェアのバージョンの比較を行います。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode *意味*

OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※1
OPOS_E_DISABLED	デバイスが Disable 状態です。	※2
OPOS_E_ILLEGAL	本メソッドはサポートしていません。 (CapStatisticsReporting, CapUpdateStatistics プロパティが false)	

※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考

- ・ 参照 ◇ CapCompareFirmwareVersion プロパティ

※1 本メソッドは未サポートです (CapStatisticsReporting, CapUpdateStatistics プロパティが共に FALSE 固定です)。

4.3.2. 専用メソッド

BeginDeposit メソッド

- ・ 形式 LONG BeginDeposit ();
- ・ 説明 計数処理を行う場合、このメソッドを使用します。
このメソッドの呼び出しで以下のプロパティ値は初期化されます。
 - ・ DepositCounts プロパティは各金種の枚数値が0にセットされます。
 - ・ DepositAmount プロパティは0がセットされます。
(ResultCode が OPOS_SUCCESS の場合のみとなります。)

CapDepositDataEvent プロパティ値が TRUE なら、このメソッドの呼び出し後につり銭機が現金を受け付けるたびに **DataEvent** イベントにて入金イベントが報告されます。但し、入金処理が一時停止状態の場合は入金イベントは通知されません。

レジストリ”BeginDepositPreCheck”が”TRUE”の場合、入金開始前に以下の部位の装填状態をチェックします。

フル／ニアフル状態の場合、つり銭機に対する計数開始は指示せずにエラー応答します。(FullStatus プロパティとは連動しておりません。)

- ・ 紙幣カセットフル／ニアフル
- ・ 硬貨収納庫フル／ニアフル
- ・ 紙幣 RJ 部フル

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

<i>ResultCode</i>	意味	
OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_ILLEGAL	呼び出しシーケンスが正しくありません（既に計数処理中です）。	※1
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※2
OPOS_E_DISABLED	デバイスが Disable 状態です。	※3
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。	
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。	
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。	
	<i>OPOS_ECHAN_SETERERROR</i>	セット外れ状態です。 ※4
	<i>OPOS_ECHAN_ERROR</i>	処理機異常です。 ※4
	<i>OPOS_ECHAN_BUSY</i>	処理機動作中です。 ※4
	<i>OPOS_ECHAN_IFERROR</i>	通信異常。
	<i>OPOS_ECHAN_IMPOSSIBLE</i>	処理機がつり銭機モードです ※4
	<i>OPOS_ECHAN_CASSETTEWAIT</i>	抜取待ちです。 ※4
	<i>OPOS_ECHAN_FULL</i>	貨幣がフル・ニアフル状態です。 ※4
	<i>OPOS_ECHAN_CHARGING</i>	釣銭補充中です。 ※4
OPOS_E_FAILURE	デバイスが本メソッドを受付ません。	

- ※ 1 計数処理のシーケンス通りに処理を実行してください。
この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。
- | | |
|---------------------------------|----------|
| <i>OPOS_ECHAN_DEPOSIT</i> | 計数中です。 |
| <i>OPOS_ECHAN_PAUSEDDEPOSIT</i> | 計数停止中です。 |
- ※ 2 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※ 3 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※ 4 処理機の状態を確認してください。

・ 備考

DispenseCash メソッド

- ・ 形式 LONG DispenseCash (BSTR CashCounts);

パラメータ 意味

CashCounts

払い出す金種と枚数を指定します。

“金種:枚数, 金種:枚数, . . .” の書式で指定します。

- ・ 金種と枚数の間は、' :' (半角コロン) を使用
- ・ 上記の項目の区切りには、' ,' (半角カンマ) を使用

Ex1) 50 円 2 枚、100 円 4 枚、500 円 1 枚、1,000 円 5 枚を払出す場合

CashCounts=" 50:2,100:4,500:1,1000:5"

Ex2) 1,000 円 4 枚、5,000 円 1 枚、(紙幣のみ)を払出す場合

CashCounts=" ;1000:4,5000:1" (先頭に半角コロンが必要になります。)

- ・ 説明 CurrentExit プロパティで指定された払い出し口に、現金を払い出します。
CurrentExit=1 の場合、つり銭機に対して枚数指定放出コマンドを指示します。
CurrentExit=2 の場合、つり銭機に対して回収コマンドを指示します。

- ・ 戻り値 次の値のいずれかが戻され、ResultCode プロパティにも格納されます。

ResultCode 意味

OPOS_SUCCESS

本メソッドは正常に終了しました。

(正常に払い出されました。又はメソッドは非同期に実行されました)

OPOS_E_ILLEGAL

CashCounts パラメータの指定が不正です。

※1

現在の ExitCashList プロパティ値では払い出しできません。

呼び出しシーケンスが正しくありません (既に計数処理中です)。

※2

OPOS_E_CLOSED

デバイスがオープンされていません。

OPOS_E_NOTCLAIMED

排他アクセス権を獲得していません。

※3

OPOS_E_DISABLED

デバイスが Disable 状態です。

※4

OPOS_E_NOHARDWARE

デバイスが本体に接続されていないか、電源が入っていません。

OPOS_E_BUSY

非同期にて実行中のため、本メソッドが実行できません。

OPOS_E_EXTENDED

詳細内容が ResultCodeExtended プロパティに格納されています。

OPOS_ECHAN_SETERORR

セット外れ状態です。

※5

OPOS_ECHAN_ERROR

処理機異常です。

※5

OPOS_ECHAN_BUSY

処理機動作中です。

※5

OPOS_ECHAN_IFERROR

通信異常。

OPOS_ECHAN_OVERDISPENSE

指定の金種枚数が機内在高を超えています。

OPOS_ECHAN_OVER

指定された金種枚数が範囲を超えています。

※6

OPOS_ECHAN_IMPOSSIBLE

払出処理ができません。

※5

リジェクト部がフル状態です。

※5

OPOS_ECHAN_CASSETTEWAIT

抜き取り待ちです。

※5

OPOS_E_FAILURE

- ※1
- ・ 金種に ExitCashList プロパティで指定されていない値が指定されました。
 - ・ 枚数データにマイナス値が指定されました。
 - ・ その他、書式が不正です。

- ※2
- 払い出し処理のシーケンス通りに処理を実行してください。
この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。

OPOS_ECHAN_DEPOSIT

計数中です。

OPOS_ECHAN_PAUSEDDEPOSIT

計数停止中です。

- ※3
- ClaimDevice メソッドを実行し、排他アクセス権を取得してから実行してください。

- ※4 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※5 処理機の状態を確認してください。
- ※6 処理機の SSW 等の設定も確認してください（備考参照）。

- ・備考
 - ※ **CurrentExit** プロパティの値（払い出し口の選択）によってつり銭機に指示するコマンドが異なる為、結果レジつり銭機側が参照する SSW の項目も異なります。

【CurrentExit=1】 枚数指定放出コマンド

- ・紙幣部の **SSW24** の“枚数指定放出フォーマット”設定を反映します。

[2桁設定] 99枚

[3桁設定] 999枚

(硬貨部も上記設定値が有効となります。)

- ・硬貨部は各金種別に収納枚数が上限となります。
放出トレイフル検知時に動作を一旦停止し、トレイフル解除で動作を再開します。
尚、フルを検知する枚数については機種に依存します。

【CurrentExit=2】 回収コマンド

- ・紙幣部は各金種 999 枚となります。

- ・硬貨部は放出トレイフル検知時に動作を一旦停止し、トレイフル解除で動作を再開します。
尚、フルを検知する枚数については機種に依存します。

注 上記の指定可能枚数は、設定に基づく電文のフォーマット上の値を示します。
実際、つり銭機で払い出しが可能な枚数は各処理機の仕様に依存します。

- ※ **AsyncMode** プロパティが TRUE の場合は、非同期にて動作します。
非同期で実行した場合、非同期完了イベント (CHAN_STATUS_ASYNC) をトリガに AsyncResultCode, AsyncResultCodeExtended プロパティを参照すれば、実行結果が確認できます。
- ※ **RT-300** の出金トレイが残留を検知している状態で本メソッドを実行すると、メソッドは成功しますがトレイの残留を解除しない限り、釣銭機は払い出し動作を行いません。
- ※ DispenceCash メソッドでは、ユニット別に制御しているため、エラー応答時でもエラー状態でないユニットからは正常に払い出されます。

- ・参照

DispenseChange メソッド

- ・ 形式 LONG DispenseChange (LONG Amount);

パラメータ

Amount

意味

払い出すつり銭の合計を指定します。

- ・ 説明 CurrentExit で指定された払い出し口に、指定合計金額を払い出します。
 CurrentExit=1 の場合、つり銭機に対して金額指定放出コマンドを指示します。
 CurrentExit=2 の場合、つり銭機に対して回収コマンドを指示します。
 つり銭機は利用できる金種から最小構成にて払い出しを行います。
- ・ 戻り値 次の値のいずれかが戻され、ResultCode プロパティにも格納されます。

ResultCode

意味

OPOS_SUCCESS	本メソッドは正常に終了しました。 (正常に払い出されました。又はメソッドは非同期に実行されました)	
OPOS_E_ILLEGAL	Amount パラメータの指定が不正です。 現在の ExitCashList プロパティ値では Amount を払い出しできません。 呼び出シーケンスが正しくありません (既に計数処理中です)。	※1 ※2
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※3
OPOS_E_DISABLED	デバイスが Disable 状態です。	※4
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。	
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。	
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。	
	OPOS_ECHAN_SETERERROR	セット外れ状態です。 ※5
	OPOS_ECHAN_ERROR	処理機異常です。 ※5
	OPOS_ECHAN_BUSY	処理機動作中です。 ※5
	OPOS_ECHAN_IFERROR	通信異常。
	OPOS_ECHAN_OVERDISPENSE	指定の金額が機内在高を超えています。
	OPOS_ECHAN_OVER	指定された金額が範囲を超えています。 ※6
	OPOS_ECHAN_IMPOSSIBLE	払出処理ができません。 ※5
		リジェクト部がフル状態です。 ※5
	OPOS_ECHAN_CASSETTEWAIT	抜き取り待ちです。 ※5
OPOS_E_FAILURE		

- ※ 1 ・ 金額にマイナス値が指定されました。
 ・ その他、書式が不正です。
- ※ 2 払い出し処理のシーケンス通りに処理を実行してください。
 この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。
 OPOS_ECHAN_DEPOSIT 計数中です。
 OPOS_ECHAN_PAUSEDDEPOSIT 計数停止中です。
- ※ 3 ClaimDevice メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※ 4 DeviceEnabled プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※ 5 処理機の状態を確認してください。
- ※ 6 処理機の SSW 等の設定も確認してください (備考参照)。

- ・ 備考
 - ※ **CurrentExit** プロパティの値（払い出し口の選択）によってつり銭機に指示するコマンドが異なる為、結果つり銭機側が参照する SSW の項目も異なります。
 - 【CurrentExit=1】 金額指定放出コマンド
 - ・ 紙幣部は **SSW24** の“金額指定放出フォーマット”設定を反映します。
 - [5 桁設定] 99,999 円
 - [6 桁設定] 999,999 円
 - ・ 硬貨部は 999 円まで払い出しが可能です。
 - 【CurrentExit=2】 回収コマンド
 - 利用できる高額金種より最小構成枚数にて払い出されます。
 - ※ **AsyncMode** プロパティが TRUE の場合は、非同期にて動作します。
非同期で実行した場合、非同期完了イベント（CHAN_STATUS_ASYNC）をトリガに AsyncResultCode, AsyncResultCodeExtended プロパティを参照すれば、実行結果が確認できます。
 - ※ **RT-300** の出金トレイが残留を検知している状態で本メソッドを実行すると、メソッドは成功しますがトレイの残留を解除しない限り、釣銭機は払い出し動作を行いません。
 - ※ DispenceChange メソッドでは、ユニット別に制御しているため、エラー応答時でもエラー状態でないユニットからは正常に払い出されます。
- ・ 参照

EndDeposit メソッド

- ・形式 **LONG EndDeposit (LONG Success);**

パラメータ	意味
Success	入金(計数)した現金の処理を設定します。
CHAN_DEPOSIT_CHANGE	計数処理を終了します。釣銭出金を行います(動作なし)。
CHAN_DEPOSIT_NOCHANGE	計数処理を終了します。釣銭出金はありません(動作なし)。
CHAN_DEPOSIT_REPAY	計数した現金を返却します。

- ・説明 つり銭機の入金処理を終了します。
アプリケーションは、本メソッドを呼び出す前に釣銭額の計算をする必要があります。
入金合計 > 販売合計 ・・・ **CHAN_DEPOSIT_CHANGE** をパラメータとします。
入金合計 ≤ 販売合計 ・・・ **CHAN_DEPOSIT_NOCHANGE** をパラメータとします。
(上記のようなパラメータの切り分けを推奨しますが、実際動作をしませんので 2 つのパラメータ (CHAN_DEPOSIT_CHANGE, CHAN_DEPOSIT_NOCHANGE) のうち、どちらを指定しても同じ振る舞いです。)

又、このメソッドを呼び出す前には必ず **FixDeposit** メソッドにて入金の確定を行う必要があります。

- ・戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_ILLEGAL	呼び出しシーケンスが正しくありません。 ※1 指定されたパラメータをサポートしていません。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。 ※2
OPOS_E_DISABLED	デバイスが Disable 状態です。 ※3
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。
	<i>OPOS_ECHAN_ERROR</i> 処理機異常です。 ※4
	<i>OPOS_ECHAN_BUSY</i> 処理機動作中です。 ※4
	<i>OPOS_ECHAN_IFERROR</i> 通信異常。
	<i>OPOS_ECHAN_SETERORR</i> セット外れ状態です。 ※4
	<i>OPOS_ECHAN_OVER</i> 返却金額が範囲を超えています。
	<i>OPOS_ECHAN_OVERDISPENSE</i> 返却金額が機内在高を超えています。
	<i>OPOS_ECHAN_IMPOSSIBLE</i> 実行できません。 ※4
	<i>OPOS_ECHAN_CASSETTEWAIT</i> 抜き取り待ちです。 ※4
OPOS_E_FAILURE	

- ※1 **FixDeposit** の後に実行してください。
この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。
OPOS_ECHAN_DEPOSIT 計数中です。
OPOS_ECHAN_PAUSEDDEPOSIT 計数停止中です。
- ※2 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※3 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

※ 4 処理機の状態を確認してください。

・備考

- ※ レジストリ “**RepayType**” について
このレジストリの設定により、Success = CHAN_DEPOSIT_REPAY（返却）の返却処理の振る舞いを切り替えることが可能です。
- “**Amount**” : DepositAmount プロパティ（入金合計金額）の値を金額指定放出コマンドのパラメータとして実行しています。(DispenseChange メソッドと同様の仕様となります)
つまり、つり銭機からは最小構成枚数（不可ならば紙幣 2 金種・硬貨 1 金種下まで代替）で現金が放出されます。
従って、返却時の金種枚数構成は計数した時と異なる可能性があります。
- “**CashList**” : DepositCount プロパティ（入金合計金種枚数）の値を枚数指定放出コマンドのパラメータとして実行しています。(DispenseCash メソッドと同様の仕様となります)
つまり、つり銭機からは実際の計数枚数と同様の現金が放出されます。
収納庫に無い金種（例：2000 円）が入金された場合は、最小構成枚数（不可ならば、1 金種下まで代替）で返却処理が行われます。
- ※ RT-300 単体接続の場合は、レジストリ “RepayType” の値にかかわらず、“CashList” の振る舞いになります。（実際の計数枚数と同様の金種構成で放出されます。）
- ※ 2000 円札の返却はできません。
- ※ CHAN_DEPOSIT_CHANGE/CHAN_DEPOSIT_NOCHANGE のつり銭機の動作を伴わないパラメータと CHAN_DEPOSIT_REPAY では、メソッド実行の結果コードが違う場合があります。
例えば、メソッド実行時に回線断の場合、先のパラメータが正常終了するのに対して、CHAN_DEPOSIT_REPAY の場合は異常終了します。
これは、CHAN_DEPOSIT_REPAY の場合、本メソッドの内部でつり銭機に対してコマンドを発行している為です。
- ※ **RT-300 の出金トレイが残留を検知している状態で本メソッドで (CHAN_DEPOSIT_REPAY : 返却) を実行すると、メソッドは成功しますがトレイの残留を解除しない限り、つり銭機は払い出し動作を行いません。**
- ※ 返却処理 (CHAN_DEPOSIT_REPAY) は同期のみをサポートしております。
非同期で返却処理を行いたい場合は、EndDeposit を CHAN_DEPOSIT_REPAY 以外で実行後、出金処理 (AsyncMode=TRUE、DispenseCash() メソッド等) で代用が可能です。

・参照

レジストリキー : Compatibility

FixDeposit メソッド

- ・ 形式 LONG FixDeposit ();
- ・ 説明 つり銭機の入金確定を行います。
この時、つり銭機は入金処理を停止し、DepositCounts プロパティと DepositAmount プロパティの値が確定します。
- ・ 戻り値 次の値のいずれかが戻され、ResultCode プロパティにも格納されます。

ResultCode	意味	
OPOS_SUCCESS	本メソッドは正常に終了しました。	
OPOS_E_ILLEGAL	呼び出しシーケンスが正しくありません。	※1
OPOS_E_CLOSED	デバイスがオープンされていません。	
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。	※2
OPOS_E_DISABLED	デバイスが Disable 状態です。	※3
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。	
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。	
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。	
OPOS_ECHAN_ERROR	処理機異常です。	※4
OPOS_ECHAN_BUSY	処理機動作中、または、内部処理中です。	※4
OPOS_ECHAN_IFERROR	通信異常。	
OPOS_ECHAN_SETERROR	セット外れです。	※4
OPOS_ECHAN_CASSETTEWAIT	抜き取り待ちです。	※4
OPOS_ECHAN_IMPOSSIBLE	実行できません。	※4
OPOS_ECHAN_CHARGING	補充中です。	※4
OPOS_ECHAN_FULL	フル状態です。	※4
OPOS_ECHAN_DEPOSIT_MOVE_BILL	紙幣部) 動作中です。	※5
OPOS_ECHAN_DEPOSIT_MOVE_COIN	硬貨部) 動作中です。	※5
OPOS_ECHAN_DEPOSIT_ERR_BILL	紙幣部) エラーが発生しています。	※6
OPOS_ECHAN_DEPOSIT_ERR_COIN	硬貨部) エラーが発生しています。	※6
OPOS_ECHAN_DEPOSIT_RJ_BILL	紙幣部) RJ 紙幣を抜き取ってください。	※5
OPOS_ECHAN_DEPOSIT_RJ_COIN	硬貨部) RJ 硬貨を抜き取ってください。	※5
OPOS_ECHAN_DEPOSIT_CAS_BILL	紙幣部) カセットがフルです、要交換。	※5
OPOS_ECHAN_DEPOSIT_SET_BILL	紙幣部) セット外れです。	※6
OPOS_ECHAN_DEPOSIT_SET_COIN	硬貨部) セット外れです。	※6
OPOS_ECHAN_DEPOSIT_RESET_BILL	紙幣部) リセット中です。	※6
OPOS_ECHAN_DEPOSIT_RESET_COIN	硬貨部) リセット中です。	※6
OPOS_E_FAILURE	デバイスが本メソッドを受付ません。	

- ※1 **BeginDeposit** による計数処理が開始されていません。
又は計数処理中です。
この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。
- | | |
|--------------------|---------------|
| OPOS_ECHAN_DEPOSIT | 計数中です (備考参照)。 |
| OPOS_ECHAN_NEUTRAL | 待機中です。 |
- ※2 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。

- ※3 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※4 処理機の状態を確認してください。
- ※5 レジストリ “FixDepositReturnTiming” の値が “Async” / “AsyncStd” の場合に使用されます。
- ※6 レジストリ “FixDepositReturnTiming” の値が “Async” の場合でのみ使用されます。

・備考

- ※ レジストリ “**FixDepositReturnTiming**” について
このレジストリを設定により FixDeposit メソッドを実行した際の終了条件を変更することが可能です。

- “**Sync**” : 同期処理 または「**キー設定無し**」
本メソッドは入金停止・終了コマンドを送出し、つり銭機の状態を監視します。
入金終了状態になるとアプリケーションに制御を戻します（リターンします）。

例えば、エラー・RJ（リジェクト）等が発生している場合に本メソッドを実行すると、OPOS はつり銭機がその状態が解除されるまでアプリケーションに制御を戻しません（リターンしません）。

メリットとしては FixDeposit メソッドの完了を待ち合わせて、次に EndDeposit を呼び出すといった平易なアプリケーション設計が可能となります。
但し、つり銭機が入金終了状態になるまでアプリケーションに制御を戻さない為、画面のリフレッシュが滞ったりアプリケーション側で次の処理が出来ない等を留意する必要があります。

- “**Async**” : 非同期処理（終了条件チェック強）
本メソッドは入金停止コマンドを送出する前につり銭機の状態をチェックします。
入金待機中（入金停止・終了コマンド送付で直ぐに入金プロセスが解除できる状態）や、硬貨部収納庫フルといった入金プロセスを一旦終了させなければ解除できない事象については入金停止・終了コマンドを送出して正常終了します。
それ以外はアプリケーションにエラー応答します。

- “**AsyncStd**” : 非同期処理（終了条件チェック標準）
本メソッドは入金停止コマンドを送出する前につり銭機の状態をチェックします。
つり銭機が入金停止コマンドを受付可能な状態であれば、コマンド送付後正常終了します。
それ以外はアプリケーションにエラー応答します。

※ 以下に“FixDepositReturnTiming” 設定選択のポイントを記載します。

・ “Sync”

完全同期処理で簡易な振る舞いを期待する場合に有効です。

・ “Async”

1 取引毎の入金確定の精度を優先したい場合に有効です。

セット外れやジャムエラー発生時、本メソッドはエラー応答し入金できません。

その為、入金プロセス内での解除が前提となります。

メリットはエラー解除を取引最中に取り組むため、入金終了時の取引内容の確定度が向上する事ですが、エラー解除作業によるお客様の待ち時間を増大させる可能性があります。

・ “AsyncStd”

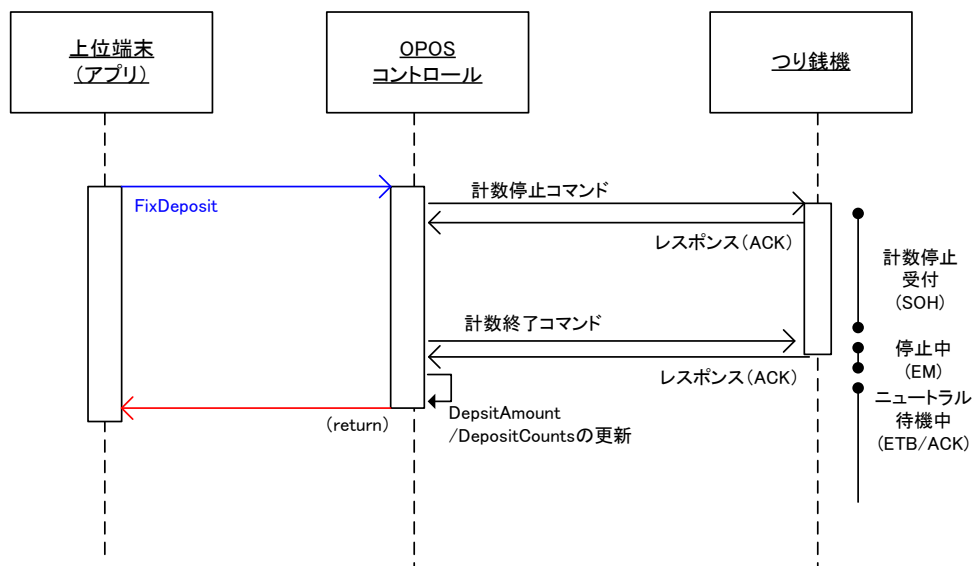
入金プロセスを終了させ、客さばきを優先させたい場合に有効です。

セット外れやジャムエラーの状態でもとりあえず入金プロセスは終了させ、つり銭についてはドロア・手元管理の貨幣で手払いした後に、つり銭機のエラー解除に取り掛かる運用が想定されます。

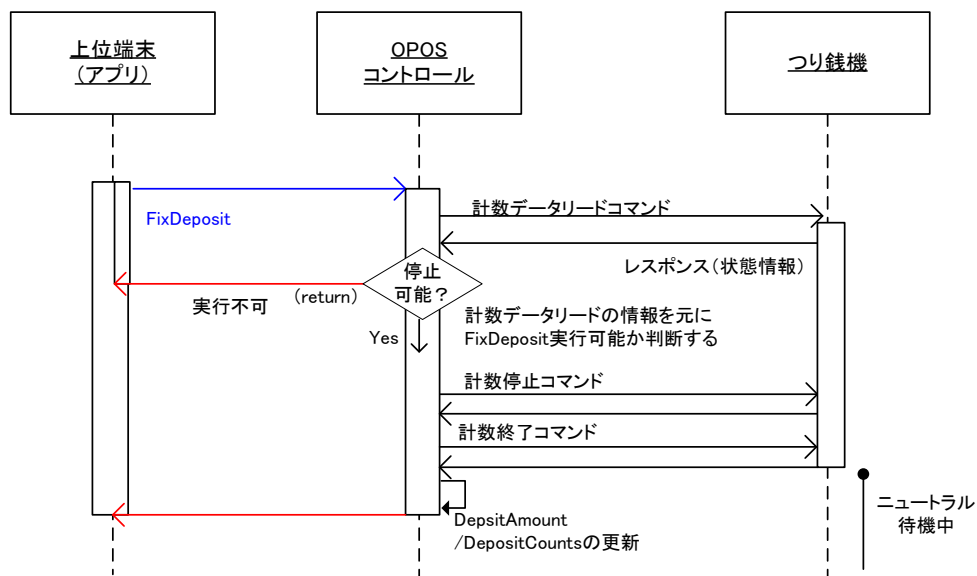
メリットとしてはお客様を待たせる時間が最小限ですむ事ですが、機外の貨幣による手払いの運用や現金管理の不整合への対策を十分に検討する必要があります。

※ シーケンス図

・ FixDepositReturnTiming = “Sync”の場合 またはキー無しの場合



・ FixDepositReturnTiming = “Async”, “AsyncStd”



・ 参照

PauseDeposit メソッド

- ・ 形式 **LONG PauseDeposit (LONG Control);**

パラメータ

意味

Control

計数処理の一時停止／再開を指定します。

CHAN_DEPOSIT_PAUSE 計数処理の一時停止

CHAN_DEPOSIT_RESTART 計数処理の再開

- ・ 説明 計数処理の一時停止または再開を行います。
入金処理が一時停止の状態で、**FixDeposit** を呼び出すことが出来ます。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode

意味

OPOS_SUCCESS

本メソッドは正常に終了しました。

OPOS_E_ILLEGAL

呼び出シーケンスが正しくありません。

※1

指定されたパラメータをサポートしていません。

OPOS_E_CLOSED

デバイスがオープンされていません。

OPOS_E_NOTCLAIMED

排他アクセス権を獲得していません。

※2

OPOS_E_DISABLED

デバイスが Disable 状態です。

※3

OPOS_E_NOHARDWARE

デバイスが本体に接続されていないか、電源が入っていません。

OPOS_E_BUSY

非同期にて実行中のため、本メソッドが実行できません。

OPOS_E_EXTENDED

詳細内容が ResultCodeExtended プロパティに格納されています。

OPOS_ECHAN_ERROR

処理機異常です。

※4

OPOS_ECHAN_BUSY

動作中です。

※4

OPOS_ECHAN_IFERROR

通信異常。

OPOS_ECHAN_SETERORR

セット外れです。

※4

OPOS_ECHAN_CASSETTEWAIT

抜き取り待ちです。

※4

OPOS_ECHAN_IMPOSSIBLE

実行できません。

※4

OPOS_ECHAN_CHARGING

補充中です。

OPOS_ECHAN_FULL

フル状態です。

※4

OPOS_E_FAILURE

- ※1 **BeginDeposit** による計数処理が開始されていません。
この場合、デバイス状態が ResultCodeExtended プロパティにて確認出来ます。
OPOS_ECHAN_NEUTRAL 待機中です。
- ※2 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※3 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※4 処理機の状態を確認してください。

・ 備考

- ※ レジストリ “**PauseDepositReturnTiming**” について
このレジストリの設定により PauseDeposit メソッド (CHAN_DEPOSIT_PAUSE: 停止) を実行した際の終了タイミングを変更することが可能です。

□ “**Sync**” : 同期処理

本メソッドは入金停止コマンドを送出し、つり銭機の状態を監視します。停止状態になるとアプリケーションに制御を戻します (リターンします)。

例えば、エラー・RJ (リジェクト) 等が発生している場合に本メソッドを実行すると、OPOS はつり銭機がその状態が解除されるまでアプリケーションに制御を戻しません (リターンしません)。

この設定は、メソッドの終了とその結果報告を待てばよい為、OPOS の制御が平易に行えるメリットがあります。

但し、つり銭機が停止状態になるまでアプリケーションに制御を戻さない為、画面のリフレッシュが滞ったりアプリケーション側で次の処理が出来ない可能性がある事を留意する必要があります。

□ “**Async**” : 非同期処理

本メソッドは、入金停止コマンドを送出後、直ちにアプリケーションに制御を戻します (リターンします)。

例えば、メソッド実行時に計数中の場合、OPOS はつり銭機に対して停止コマンドを指示した後アプリケーションに制御を戻します (リターンします)。

つり銭機は投入された貨幣が計数し終わると停止状態になります。

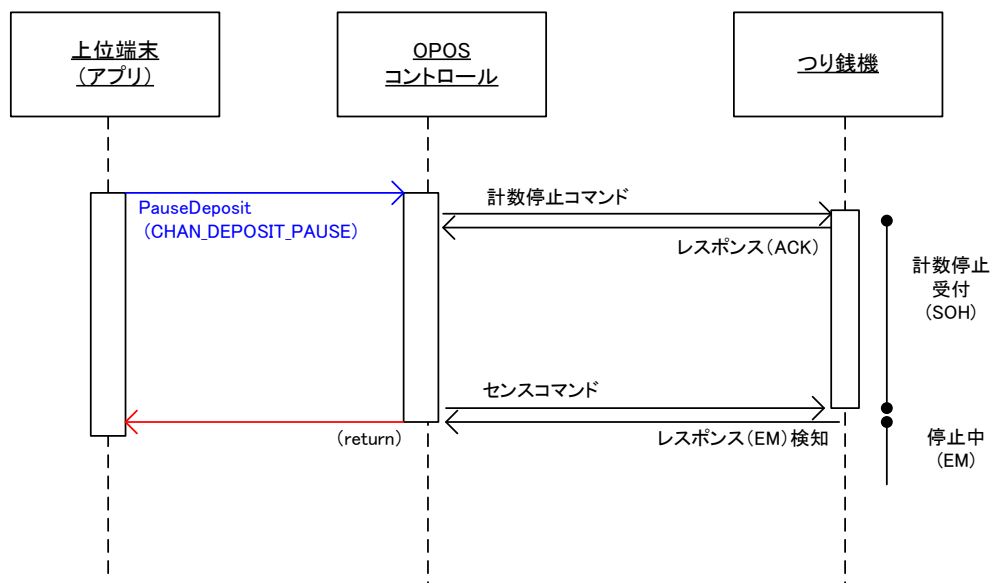
エラー・RJ (リジェクト) 等が発生している場合についても、OPOS は停止コマンドを指示した後アプリケーションに制御を戻します (リターンします)。

つり銭機側は上記状態が解除されると停止状態になります。

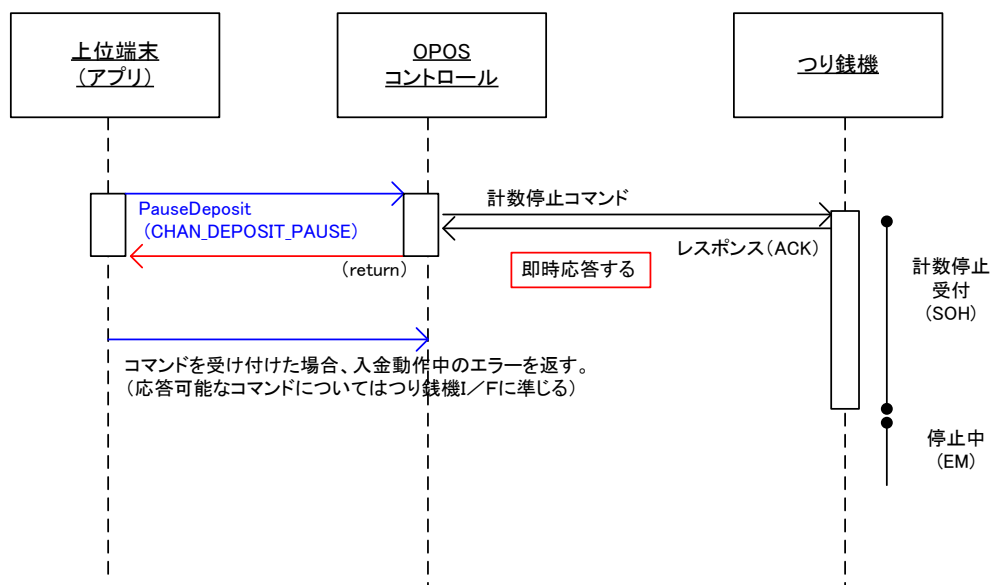
この設定は、直ちにアプリケーションに制御を戻すためアプリケーション側の待ち合わせ時間が短縮されるメリットがあります。

※ シーケンス図

・ PauseDepositReturnTiming = “Sync”の場合 またはキー無しの場合



・ PauseDepositReturnTiming = “Async”



・ 参照

ReadCashCounts メソッド

- ・ 形式 **LONG ReadCashCounts (BSTR* pCashCounts, BOOL* pDiscrepancy);**

パラメータ	意味
<i>pCashCounts</i>	現金枚数データを格納するエリアへのポインタ。
<i>pDiscrepancy</i>	機内の在高状態を示します。
TRUE	<i>pCashCounts</i> で報告された現金枚数に含まれない現金がつり銭機内にあります。
FALSE	現金在高は <i>pCashCounts</i> の内容と一致します。

- ・ 説明 つり銭機内の現金在高を取得します。
現金在高は設定によって取得できる内容が異なります。
- ① 収納+回収 : 紙幣部) 収納部在高 + 回収カセット在高を現金在高とします。
硬貨部) 収納部在高 を現金在高とします。
- ② 収納 : 紙幣部・硬貨部共に、収納部在高を現金在高とします。

pCashCounts で指定される文字列のフォーマットは **DispenseCash** メソッド内の *CashCounts* と同じです。

pCashCounts 内の金種は、**CurrencyCashList** プロパティと同じ順序で並んでいます。

例) *pCashCounts* = " 10:56, 50:77, 100:55, 500:38:1000:98, 5000:45"

pDiscrepancy = TRUE の場合

¥10- 56 枚 ¥50- 77 枚 ¥100- 55 枚
¥500- 38 枚 ¥1000- 98 枚 ¥5000- 45 枚

◇ 報告した現金枚数以外にも機内に現金があります。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。 ※1
OPOS_E_DISABLED	デバイスが Disable 状態です。 ※2
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。
	<i>OPOS_ECHAN_BUSY</i> 処理機動作中です。 ※3
	<i>OPOS_ECHAN_IFERROR</i> 通信異常。 包装硬貨部が開いています。
	<i>OPOS_ECHAN_CHARGING</i> 補充中です。 ※3
OPOS_E_FAILURE	

- ※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※3 処理機の状態を確認してください。(備考参照)

- ・ 備考 ※ リジェクト貨幣がある場合、*pDiscrepancy* には TRUE がセットされます。
(紙幣出金 RJ を判定させるには、紙幣 SSW23 の精査フォーマットを標準仕様以外にする必要が有ります。)
- ※ レジストリ (**SEISA_CommandPreCheck**、**STATUSREAD_CommandPreCheck**) を FALSE に設定することで、つり銭機が動作中でも本メソッドが実行可能となります。
つり銭機が動作中に本メソッドが呼び出された場合、*pDiscrepancy* には TRUE がセットされます。

- ※ 硬貨単体接続時、硬貨 SSW50 の精査フォーマットに紙幣つり銭機互換以外が設定されていた場合は、つり銭機が動作中でも *pDiscrepancy* には FALSE がセットされます。
- ※ 包装硬貨部 (WD-300) 接続時にドロアを開けた状態で実行した場合、エラー応答になります。(ResultCodeExtended = OPOS_ECHAN_IFERROR が返ります。)

・参照

AdjustCashCounts メソッド ※1

- ・ 形式 LONG AdjustCashCounts (BSTR CashCounts);

パラメータ

意味

CashCounts

初期化する現金の金種・枚数を指定します。

- ・ 説明 つり銭機内の現金在高の設定または、修正を行います。

CashCounts で指定する文字列のフォーマットは DispenseCash メソッド内の CashCounts と同じです。

pCashCounts 内の金種は、CurrencyCashList プロパティと同じ順序で並んでいます。

例) CashCounts = " 10:56, 50:77, 100:55, 500:38, 1000:98, 5000:45"

で本メソッドを実行すると、

¥10- 56 枚 ¥50- 77 枚 ¥100- 55 枚

¥500- 38 枚 ¥1000- 98 枚 ¥5000- 45 枚

が、つり銭機内の現金在高として設定されます。

- ・ 戻り値 次の値のいずれかが戻され、ResultCode プロパティにも格納されます。

ResultCode

意味

OPOS_SUCCESS

本メソッドは正常に終了しました。

OPOS_E_CLOSED

デバイスがオープンされていません。

OPOS_E_NOTCLAIMED

排他アクセス権を獲得していません。

※1

OPOS_E_DISABLED

デバイスが Disable 状態です。

※2

OPOS_E_ILLEGAL

本メソッドはサポートしていません。

※1 ClaimDevice メソッドを実行し、排他アクセス権を取得してから実行してください。

※2 DeviceEnabled プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。

- ・ 備考

- ・ 参照

※1 本メソッドは未サポートです。

DirectIO メソッド

- ・ 形式 `LONG DirectIO (LONG Command, LONG* pData, BSTR* pString);`

パラメータ	意味
<i>Command</i>	コマンド番号。サービスオブジェクト（本コントロール）が割り当てた特定値です。
<i>pData</i>	数値データへのポインタ。 値はコマンド番号とサービスオブジェクトによって変化します。
<i>pString</i>	文字列データへのポインタ。 値はコマンド番号とサービスオブジェクトによって変化します。 このデータのフォーマットは BinaryConversion プロパティの値に依存します。

- ・ 説明 サービスオブジェクトと直接通信する時に使用します。

このメソッドにより、標準コントロールオブジェクトではサポートしていないデバイスクラスの機能を、サービスオブジェクトが提供できるようになります。

- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。
OPOS_E_ILLEGAL	呼び出しシーケンスが正しくありません。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。 ※1
OPOS_E_DISABLED	デバイスが Disable 状態です。 ※2
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。 ※3
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。
<i>OPOS_ECHAN_ERROR</i>	処理機異常です。 ※3
<i>OPOS_ECHAN_BUSY</i>	動作中です。 ※3
<i>OPOS_ECHAN_IFERROR</i>	通信異常。
<i>OPOS_ECHAN_SETERORR</i>	セット外れです。 ※3
<i>OPOS_ECHAN_CASSETTEWAIT</i>	抜き取り待ちです。 ※3
<i>OPOS_ECHAN_IMPOSSIBLE</i>	実行できません。 ※3
<i>OPOS_ECHAN_CHARGING</i>	補充中です。 ※3
<i>OPOS_ECHAN_FULL</i>	フル状態です ※3
<i>OPOS_ECHAN_NEUTRAL</i>	待機中です
<i>OPOS_ECHAN_DEPOSIT</i>	計数中です
<i>OPOS_ECHAN_PAUSEDEPOSIT</i>	計数停止中です
OPOS_E_FAILURE	本メソッドで要求した処理をデバイスが実行できません。

- ※1 **ClaimDevice** メソッドを実行し、排他アクセス権を取得してから実行してください。
- ※2 **DeviceEnabled** プロパティを TRUE に設定し、デバイスを Enable 状態にしてから実行してください。
- ※3 処理機の状態を確認してください。

- ・備考 ※ 本コントロールでサポートする機能は以下の通りです。
 尚、詳細については以降に記載します。

機能	Command (コマンド番号)	
リセット	CHAN_DI_RESET	(=1)
メモリリード	CHAN_DI_MEMREAD	(=2)
メモリクリア	CHAN_DI_MEMCLR	(=3)
モード切替	CHAN_DI_CHGMODE	(=4) ※ 2
SSW 設定	CHAN_DI_SSWSET	(=5)
日時設定	CHAN_DI_TIMESSET	(=6)
ENQ 送信	CHAN_DI_ENQ	(=7)
文字列送信	CHAN_DI_STRING	(=8) ※ 1
回収	CHAN_DI_COLLECT	(=9)
状態リード	CHAN_DI_STATUSREAD	(=10)
補充枚数取得	CHAN_DI_SUPPLYCOUNTS	(=11)
精査	CHAN_DI_SEISA	(=12)
計数データリード	CHAN_DI_DEPOSITDATAREAD	(=13)
計数開始	CHAN_DI_BEGINDEPOSIT	(=14)
計数終了	CHAN_DI_ENDDEPOSIT	(=15)
計数停止	CHAN_DI_PAUSEDDEPOSIT	(=16)
計数再開	CHAN_DI_RESTARTDEPOSIT	(=17)
計数機／つり銭機モード	CHAN_DI_DEPOSITMODE	(=18)
累計カウンタクリア	CHAN_DI_COUNTCLR	(=19)
ログ吸い上げ	CHAN_DI_GETLOG	(=20)
棒金ドロア制御	CHAN_DI_OPENDRAWER	(=21) ※ 2
ローカル操作禁止	CHAN_DI_CHILDLCK	(=22)
取引締め	CHAN_DI_CLOSE	(=25)
補充開始	CHAN_DI_SUPPLY	(=26)
取引外入金開始	CHAN_DI_BEGINDEPOSITOUTSIDE	(=27)
取引外金額指定出金	CHAN_DI_DISPENSECHANGEOUTSIDE	(=28)
取引外枚数指定出金	CHAN_DI_DISPENSECASHOUTSIDE	(=29)
現金戻し開始	CHAN_DI_BEGINCASHRETURN	(=30)
交代データ取得	CHAN_DI_CLOSEDATAREAD	(=31)
エラー解除ガイダンス起動	CHAN_DI_ERRGUIDANCE	(=101)

※ 1 : 未使用です。

※ 2 : RT-300 単体接続時サポートしていません。

- ・参照 ◇ DirectIO イベント

リセット

・形式

CHAN_DI_RESET

パラメータ

意味

Command

CHAN_DI_RESET

リセットコマンド

pData

—未使用—

pString

—未使用—

・説明

- ・メカイニシャル動作を実行します。
- ・装置が異常状態の場合はその解除を行います。

・備考

- ※ 貨幣が詰まった場合等の多くは、エラーの要因を解除しない限りエラー状態は解除されません。
又、リセットコマンドを送信することによりつり銭機の表示パネル上に示されたエラーコードがクリアされる場合があります。
表示パネルでエラー情報を確認する際は、コマンドの送信のタイミング等注意が必要です。
- ※ AsyncMode プロパティが TRUE の場合は、非同期にて動作します。
非同期で実行した場合、非同期完了イベント (CHAN_STATUS_ASYNC) をトリガに AsyncResultCode, AsyncResultCodeExtended プロパティを参照すれば、実行結果が確認できます。
- ※ 処理機が異常状態時のみ、イニシャル動作を行います。
例えば紙幣部のみのエラー発生の場合は、コマンド実行後紙幣部のみイニシャル動作を行い硬貨部は動作しません。

・参照

メモリリード

- ・ 形式 **CHAN_DI_MEMREAD**

パラメータ	意味
<i>Command</i>	CHAN_DI_MEMREAD メモリリード
<i>pData</i>	—
<i>pString</i>	—

- ・ 説明 詳細については未掲載です。
- ・ 備考
- ・ 参照

メモリクリア

- ・ 形式 **CHAN_DI_MEMCLR**

パラメータ	意味
Command	CHAN_DI_MEMCLR メモリクリアコマンド
pData	クリアする内容を指定します。
	1 累計補充枚数等をクリアする
	2 前回抜き取りカセット枚数等をクリアする（紙幣部のみ）
pString	---未使用---

- ・ 説明
 - ・ 指定されたメモリの内容をクリアします。
 - ・ 開店時等に使用してください。
 - ・ メカ動作は行いません。
- ・ 備考 ※ **pData に指定する値と処理機へ送信されるコマンドについて下表に示します。**

紙幣処理機	0x01	<ul style="list-style-type: none"> ・ 累計抜き取りセット枚数 ・ 累計抜き取りセット在高異常フラグ ・ 累計補充枚数
	0x02	<ul style="list-style-type: none"> ・ 前回抜き取りセット枚数 ・ 前回抜き取りセット在高異常フラグ
硬貨処理機	0x01	累計補充枚数
	0x02	---未使用---

- ・ 参照

モード切り替え

- ・ 形式 **CHAN_DI_CHGMODE**

パラメータ	意味
<i>Command</i>	CHAN_DI_CHGMODE モード切り替え
<i>pData</i>	モードを指定します。下記値の組み合わせにて指定します。
	0x00 (紙幣・硬貨) 接続
	0x10 (硬貨) 切り離し
	0x20 (紙幣) 切り離し
	ex) 硬貨 : 接続、紙幣接続 pData = 0x00 + 0x10
<i>pString</i>	——未使用——

- ・ 説明
 - ・ モードを切り替えます。
 - ・ 縮退運用等に使用します。
 - ・ メカ動作は行いません。
- ・ 備考
 - ※ **RAD/RT-300 接続時**、本メソッドはサポートしていません。
コマンドを受信しても、何もせずに正常応答を返します。
(ResultCode = OPOS_SUCCESS)
 - ※ **RT-300 単体接続時**、本メソッドはサポートしていません。
(ResultCode = OPOS_E_ILLEGAL)
 - ※ このコマンドは **DeviceEnabled** プロパティが FALSE の場合でも実行可能です。
紙幣-硬貨接続にて硬貨異常で通信できない場合、硬貨部を切り離す事が可能です。
- ・ 参照

SSW設定

・形式

CHAN_DI_SSWSET

パラメータ

意味

Command	CHAN_DI_SSWSET	ソフトSW設定
pData	設定する部位を指定します。	
	0x00	硬貨部
	0x01	紙幣部
pString	SSWNo.と設定値を指定します。 “SSWNo, 設定値” のカンマ区切りの書式にて指定します。	
	SSWNo	10 進表記の文字列
	設定値	16 進表記の文字列

・説明

- ・装置の初期設定時に使用します。
- ・設定データは電源 OFF/ON しても消去されません。
- ・1 回のコマンド送信で 1 つの SSW の設定を行います。複数の SSW の設定が必要な場合は、その回数分実行してください。
- ・メカ動作は行いません。

・備考

※ パラメータの設定例を下記に示します。

Ex1) 紙幣部 SSW:5 を 96H に設定する場合、

pData = 1

pString = “05, 96”

Ex2) 硬貨部 SSW:101 を 01H に設定する場合、

pData = 1

pString = “101, 01”

- ※ 本メソッドは、無条件につり銭機に対して送信します。
つり銭機の機種によって SSW の仕様・制限事項が異なる為、十分な理解の上使用する必要があります。
- ※ このコマンドは DeviceEnabled プロパティが FALSE の場合でも実行可能です。
- ※ 設定完了後、内部で設定値の更新処理を行います。更新中に上位から指示された場合は、OPOS 内部で更新処理の完了を待ってから実行します。
- ※ RT-300 硬貨単体接続時、紙幣部のパラメータは指定できません。

・参照

日時設定

・ 形式

CHAN_DI_TIMESET

パラメータ	意味
Command	CHAN_DI_TIMESET 日時設定コマンド
pData	——未使用——
pString	年月日時分秒(秒あり)又は、年月日時分(秒なし)の2パターンのフォーマットで指定できます。 ① “YYMMDDhhmmss” の書式にて指定します。 ② “YYMMDDhhmm” の書式にて指定します。 ex) 2008/01/18 19:45.30 の場合 ①の場合 pString = “080118194530” ②の場合 pString = “0801181945”

・ 説明

- ・ つり銭機の内部時計を設定します。
- ・ メカ動作は行いません。

・ 備考

※ 設定可能範囲

年 : “00” ~ “99”
 月 : “01” ~ “12”
 日 : “01” ~ “31”
 時 : “00” ~ “23”
 分 : “00” ~ “59”
 秒 : “00” ~ “59”

- ※ カレンダー日付でない日時設定に対しては、エラーとなります。
 (2月30日等)

・ 参照

E N Q 送信

・ 形式

CHAN_DI_ENQ

パラメータ

意味

Command

CHAN_DI_ENQ

E N Q コマンド

pData

—未使用—

pString

E N Q 送信後のレスポンス格納

レスポンスキャラクタ	説明
0x01 (SOH)	計数処理中
0x06 (ACK)	正常終了
0x12 (DC2)	貨幣抜き取り待ち
0x13 (DC3)	セット外れ
0x14 (DC4)	補充中 ※ 1
0x17 (ETB)	ニアエンブティ
0x18 (CAN)	異常終了
0x19 (EM)	計数停止中
0x1A (SUB)	動作中
0xFF -	回線断・電源断・通信不可

※ 1 計数機／つり銭機モードコマンド(CHAN_DI_DEPOSITMODE)でつり銭機モード設定になっている場合のみセットされます。

・ 説明

- ・ デバイスに対してE N Qコマンドを送信してレスポンスデータを ASCII 形式の 16 進表記文字列に変換後、pStringに格納します。
- ・ データリンクの確立及びつり銭機への問合せにE N Qキャラクタを使用します。つり銭機は、E N Qに対し装置状態を表すレスポンスコードを送信します。
- ・ メカ動作は行いません。

・ 備考

- ※ このコマンドは DeviceEnabled プロパティが FALSE の場合でも実行可能です。
- ※ DeviceEnabled プロパティへ TRUE のセットがエラーとなる場合等の要因を調査する場合、本メソッドを実行し、pString にセットされる内容を確認すると通信状態等が把握できます。

・ 参照

文字列送信 ※ 1

- ・ 形式 **CHAN_DI_STRING**

パラメータ	意味
<i>Command</i>	CHAN_DI_STRING 文字列送信
<i>pData</i>	送信タイムアウト値を指定します。(0 以下はエラー : OPOS_E_ILLEGAL)
<i>pString</i>	送信したい文字列を指定します。 コマンド、データ長、データ部の順に指定します。 ex) ・リセットコマンドの場合 ・コマンド : 0x30 ・データ長 : " 00" ・データ部 : 無し <i>pString</i> = "303030" となります。

- ・ 説明
 - ・ つり銭機に対して *pString* で指定された文字列を電文として送信します。
 - ・ 送信結果は *pString* に格納されます。

- ・ 備考 ※ コマンドフォーマットを以下に示します。

<データ部あり>

S	D	D	L		データ	E	B
T	C	H	1	0		T	C
X	1	1	16	16		X	C

pString に指定する部分

<データ部なし>

S	D	D	L		E	B
T	C	H	1	0	T	C
X	1	1	16	16	X	C

pString に指定する部分

- ・ 参照

※ 1 DirectIO メソッドにより網羅されている為、本コマンドをユーザが使用する必要はありません。

回収

・ 形式 CHAN_DI_COLLECT

パラメータ

意味

Command

CHAN_DI_COLLECT

回収コマンド

pData

回収モード（備考欄参照）

pString

——未使用——

・ 説明 ・ 指定されたモードにより回収処理を行います。

・ 備考 ※ pData に指定する値と処理機へ送信されるコマンドについて下表に示します。

紙幣処理機	0x00000000	指示無し
	0x00000001	千 券収納部+回収カセット+RJ 部の紙幣を全て回収します
	0x00000002	五千券収納部+回収カセット+RJ 部の紙幣を全て回収します
	0x00000003	紙幣を全て回収します
	0x00000004	残置回収として回収します ※ 現金管理機モードでは RJ 部の回収なし その他のモードでは RJ 部の回収あり
	0x00000005	回収カセット+RJ 部の紙幣を全て回収します（回収動作なし）
	0x00000006	未使用
	0x00000007	万券収納部+回収カセット+RJ 部の紙幣を全て回収します
	0x00000008	二千券収納部+回収カセット+RJ 部の紙幣を全て回収します
硬貨処理機	0x00000000	指示無し
	0x00010000	5 0 0 円収納部の硬貨を全て回収します
	0x00020000	1 0 0 円収納部の硬貨を全て回収します
	0x00030000	5 0 円収納部の硬貨を全て回収します
	0x00040000	1 0 円収納部の硬貨を全て回収します
	0x00050000	5 円収納部の硬貨を全て回収します
	0x00060000	1 円収納部の硬貨を全て回収します
	0x00070000	全ての収納硬貨を一括で回収します
	0x00080000	残置回収として回収します
	0x000A0000	簡易回収を行います

※ 紙幣処理機について回収後の回収カセットを抜取る事により、カセット在高をクリアします。

※ AsyncMode プロパティが TRUE の場合は、非同期にて動作します。
非同期で実行した場合、非同期完了イベント（CHAN_STATUS_ASYNC）をトリガに AsyncResultCode, AsyncResultCodeExtended プロパティを参照すれば、実行結果が確認できます。

※ 残置回収は金額残置回収と枚数残値回収、連動残置回収が可能です。
（回収機能の変更は、事前につり銭機の SSW 設定が必要です。）
連動残置設定時は、紙幣部のみに対して残置回収を指定（0x00000004）で実行可能です。（硬貨は端数分（999 円以下）回収されます。）
（紙幣回収モードで連動残置回収を選択時は硬貨モードを無視します。）

※ 入金中（計数待機中、計数停止中）では、すべての回収処理が実行できません。

※ RT-300 硬貨単体接続時、紙幣部のパラメータは指定できません。

・ 参照

状態リード

・形式 CHAN_DI_STATUSREAD

パラメータ	意味
<i>Command</i>	CHAN_DI_STATUSREAD 状態リードコマンド
<i>pData</i>	リードNo. 0x01 硬貨部の状態を取得します。 0x04 0x05 0x80 紙幣部の状態を取得します。 0x82 0x83 ※リードNo.の詳細については以降に記載します。
<i>pString</i>	取得した状態データを格納するポインタ データの詳細は以降に記載します。

- ・説明
 - ・ 指定されたリードNo.で状態リードを発行してそのレスポンスデータを ASCII16 進表記文字列に変換して *pString* に指定されたエリアにセットします。
 - ・ メカ動作は行いません。
- ・備考
 - ※ RT-300 硬貨単体接続時、紙幣部のパラメータ (0x80~0x83) は指定できません。
- ・参照

リードNo.

0x01

桁数	内容	詳細		
4	硬貨部エラーコード	エラーコード（10 ³ の位～10 ⁰ の位）		
2	硬貨部メカセット状態	Bit	内容	
		7	メンテキー（1でメンテ位置）	
		6	投入口残留（1で残留検知）	
		5	トレイフル（1でトレイフル）	
		4	トレイ残留（1で残留検知）	
		3	収納庫扉セット外れ（1でセット外れ）	
		2	予約（0 固定）	
		1	予約（1 固定）	
		0	装置内部セット外れ（1でセット外れ）	
1	収納状態	Bit	内 容	データ
2-3		500円	0:エンプティ 1:ニアエンプティ 2:適量 3:フル	
0-1		100円		
2-3		50円		
0-1		10円		
1		2-3	5円	リジェクトは1枚以上で2:適量、抜き取りで0:エンプティとなる。
0-1		1円		
2-3		リジェクト		
0-1		---未使用---		
2		解除手順	データ	
	‘00’ :解除完了			
	‘01’ :リセット待ち			
	‘02’ :リセット中			
	‘03’ :要因解除待ち			
	‘04’ :要因解除中			
2	在高状態フラグ ※1	‘00’ : 在高異常の可能性なし ‘01’ : 在高異常の可能性あり		
1	包装硬貨部状態 ※2 （包装硬貨部接続時）	Bit	内 容	データ
2-3		500円	0:エンプティ 1:ニアエンプティ 2:適量	
0-1		100円		
2-3		50円		
0-1		10円		
1		2-3	5円	0:なし 1:リジェクト 0:ドロア閉 1:ドロア開
0-1		1円		
2-3		---未使用---		
1		リジェクト情報		
0		ドロア開情報		
14	RJ 部在高 （オーバーフロー枚数・ リジェクト枚数）	500円枚数 ～ 1円枚数、リジェクト枚数 各金種・リジェクト 2桁（16 ¹ の位～16 ⁰ の位） ＜例＞100円4枚、5円12枚、リジェクト3枚の場合 00 04 00 00 0C 00 03 となります		
1	---未使用---			

			データ	
1	エラー貨幣の取り扱い	‘0’：判定不可 ‘1’：機外貨幣 ‘2’：機内貨幣（在高に含む）		
14	一計数カウンタ	上記“RJ 部在高と”同様		
1	収納状態	内 容	データ	
		5 0 0 円	0:エンプティ 1:ニアエンプティ 2:適量 3:ニアフル 4:フル	
1		1 0 0 円		
1		5 0 円		
1		1 0 円		
1		5 円		
1		1 円		
1		リジェクト		
1		---未使用---		
2	硬貨部ローカル操作禁止モード	‘00’：ローカル操作許可 ‘01’：ローカル操作禁止		
1	硬貨搬送路情報	Bit	内容	データ
		3	PS34	0:透光 1:遮光
		2	PS24	
		1	PS22 or PS33	
		0	PS21 or PS32	
1		3	PS20 or PS31	PS17～PS22 と PS28～PS33 は いずれかの遮光で 1 となる
		2	PS19 or PS30	
		1	PS18 or PS29	
	0	PS17 or PS28		
1	在高異常解除方法 ※3	Bit	内容	データ
		2-3	5 0 0 円	0:在高異常の可能性なし 1:回収必要 2:戻し入れ必要
		0-1	1 0 0 円	
1		2-3	5 0 円	
		0-1	1 0 円	
1		2-3	5 円	
		0-1	1 円	
1		2-3	未使用	
		0-1	未使用	

※1：在高異常フラグについて

在高異常フラグが「異常あり」の場合、収納部の在高データと実際の硬貨枚数が異なっている可能性が有ります。

特に在高確定が必要な場合（残置運用等）では、上位機の締上げ時に本フラグを確認する事を推奨します。

※2：包装硬貨部未接続時は0固定になります。

※3：在高異常の解除誘導について

上位で在高異常解除の誘導を行う際は、在高異常解除方法を元に行ってください。

対応例

- ・回収が必要な場合： 回収コマンド(CHAN_DI_COLLECT pData=0x00000007)の送出。
「全回収を促すメッセージ」の表示など。
- ・戻し入れが必要な場合： 現金戻しコマンド、または回収コマンドの送出。
「現金戻しを促すメッセージ」または「全回収を促すメッセージ」の表示など。

リードNo. 0x04

※RAD/RT-300 接続時は0 固定

桁数	内容	内容詳細
3	前回計数枚数	500円 (10 ² の位～10 ⁰ の位)
3		100円 (")
3		50円 (")
3		10円 (")
3		5円 (")
3		1円 (")
3	前回放出枚数	500円 (10 ² の位～10 ⁰ の位)
3		100円 (")
3		50円 (")
3		10円 (")
3		5円 (")
3		1円 (")
28	---未使用---	

リードNo. 0x05

桁数	内容	内容詳細
4	累計補充枚数	500円 (10 ³ の位～10 ⁰ の位)
4		100円 (")
4		50円 (")
4		10円 (")
4		5円 (")
4		1円 (")
40	---未使用---	

リードNo.

0x80

桁数	内容	詳細		
4	紙幣部エラーコード	エラーコード（10 ³ の位～10 ⁰ の位）		
2	紙幣メカセット状態	Bit	内容	
		7	予約（0 固定）	
		6	投入口残留（1で残留検知）	
		5	払出部残留（1で残留検知）	
		4	未使用	
		3	カセットセット外れ（1でセット外れ）	
		2	収納庫扉セット外れ（1でセット外れ）	
		1	前カバーセット外れ（1でセット外れ）	
		0	ユニットセット外れ（1でセット外れ）	
2	紙幣部解除手順	‘00’：解除完了 ‘01’：リセット待ち ‘02’：リセット中 ‘03’：要因解除待ち ‘04’：要因解除中		
1	紙幣部収納状態	内 容		データ
1		1万円	0:エンプティ	
1		5千円	1:ニアエンプティ	
1		2千円	2:適量	
1		1千円	3:ニアフル	
1		カセット	4:フル	
1		R J 部	5:収納部無し	
		金庫別収納庫は0～5のみ カセット、RJ部は2～4のみ		
1	紙幣部在高異常解除方法	Bit	内容	データ
1		3-2	1 0 0 0 0 円	0:在高異常の可能性なし 1:回収必要 2:戻し入れ必要
		1-0	5 0 0 0 円	
1		3-2	2 0 0 0 円	
		1-0	1 0 0 0 円	
1		3-2	回収カセット	
		1-0	未使用	
1		3-2	未使用	
		1-0	未使用	
2	処理機状態フラグ	‘00’：処理機待機中 ‘01’：処理機省電力中 ‘02’：処理機動作中		
2	紙幣在高状態フラグ ※1	‘00’：在高異常の可能性なし ‘01’：収納部又は収納セットに在高異常の可能性あり		
2	紙幣硬貨状態フラグ	Bit	内容	データ
		7	未使用	
		6	未使用	
		5	エラー表示ユニット	0:エラーなし
		4		1:硬貨エラー表示中 2:紙幣エラー表示中 3:棒金エラー表示中
		3	未使用	

			2	未使用	
			1	紙幣部運用フラグ	0:接続 1:切り離し
			0	硬貨部運用フラグ	0:接続(固定)
2	紙幣部ローカル操作禁止モード	‘00’ :ローカル操作許可 ‘01’ :ローカル操作禁止			
2	紙幣搬送路情報	Bit	内容	データ	
		7	PS7	0:透光 1:遮光	
		6	PS3		
		5	PS4	待機中、ユニット開時、要因解除完了時にセンサを自動消灯する為、センサ OFF 時の情報は更新されません	
		4	PS2		
		3	PS5		
		2	PS9		
		1	PS8		
		0	PS6		
1	エラー貨幣の取り扱い 紙幣	データ			
1	エラー貨幣の取り扱い 硬貨	‘0’ : 判定不可 ‘1’ : 機外貨幣 ‘2’ : 機内貨幣（在高に含む）			
1	硬貨部在高異常解除方法	Bit	内容	データ	
		3-2	5 0 0 円	0:在高異常の可能性なし 1:回収必要 2:戻し入れ必要	
		1-0	1 0 0 円		
		3-2	5 0 円		
		1-0	1 0 円		
		3-2	5 円		
		1-0	1 円		
		3-2	未使用		
1-0	未使用				
2	硬貨搬送路情報	Bit	内容	データ	
		7	PS34	0:透光 1:遮光	
		6	PS24		
		5	PS22 or PS33	PS17～PS22 と PS28～PS33 は いずれかの遮光で 1 となる	
		4	PS21 or PS32		
		3	PS20 or PS31		
		2	PS19 or PS30		
		1	PS18 or PS29		
		0	PS17 or PS28		
2	硬貨部ローカル操作禁止モード	‘00’ :ローカル操作許可 ‘01’ :ローカル操作禁止			
2	硬貨メカセット状態	Bit	内容		
		7	メンテキー（1でメンテ位置）		
		6	投入口残留（1で残留検知）		
		5	トレイフル（1でトレイフル）		
		4	トレイ 残留（1で残留検知）		
		3	収納庫扉セット外れ（1でセット外れ）		
		2	予約（0 固定）		
		1	予約（1 固定）		
		0	装置内部セット外れ（1でセット外れ）		
4	硬貨部エラーコード	エラーコード（1 0 ³ の位～1 0 ⁰ の位）			
2	硬貨部解除手順	‘00’ :解除完了 ‘01’ :リセット待ち ‘02’ :リセット中 ‘03’ :要因解除待ち ‘04’ :要因解除中			

	2	硬貨在高状態フラグ ※1	‘00’ : 在高異常の可能性なし ‘01’ : 収納部に在高異常の可能性あり		
	1	包装硬貨部状態 (包装硬貨部接続時) ※2	bit	内 容	データ
			3-2	5 0 0 円	0:エンプティ 1:ニアエンプティ 2:適量
			1-0	1 0 0 円	
	1		3-2	5 0 円	
			1-0	1 0 円	
	1		3-2	5 円	
			1-0	1 円	
			3-2	---未使用---	
	1		1	リジェクト状態	0:なし 1:リジェクト
			0	ドロア開状態	0:ドロア閉 1:ドロア開
	1	硬貨部収納状態 (RJ 部)	内 容		データ
			5 0 0 円		0:エンプティ 1:ニアエンプティ 2:適量 3:ニアフル 4:フル
	1		1 0 0 円		
	1		5 0 円		
	1		1 0 円		
	1		5 円		
	1		1 円		
	1		RJ		RJ 部は 2, 4 のみ
	1		予備		
	4	“ffff” 固定			

※1:在高異常フラグについて

在高異常フラグが「異常あり」の場合、収納部の在高データが異なっている可能性が有ります。

特に在高確定が必要な場合(残置運用等)では、上位機の締上げ時に本フラグを確認する事を推奨します。

※2:包装硬貨部未接続時は0固定になります。

※3:在高異常の解除誘導について

上位で在高異常解除の誘導を行う際は、在高異常解除方法を元に行ってください。

対応例

- ・回収が必要な場合: 回収コマンド(CHAN_DI_COLLECT pData=0x00000007(硬貨部), 0x00040000(紙幣部))の送出。
「全回収を促すメッセージ」の表示など。
- ・戻し入れが必要な場合: 現金戻しコマンド、または回収コマンドの送出。
「現金戻しを促すメッセージ」または「全回収を促すメッセージ」の表示など。

リードNo. 0x82

桁数	内容	内容詳細
3	前回計数枚数	1 0 0 0 0 円 (1 0 ² の位 ~ 1 0 ⁰ の位)
3		5 0 0 0 円 (")
3		2 0 0 0 円 (")
3		1 0 0 0 円 (")
3	前回放出枚数	1 0 0 0 0 円 (1 0 ² の位 ~ 1 0 ⁰ の位)
3		5 0 0 0 円 (")
3		2 0 0 0 円 (")
3		1 0 0 0 円 (")
40	---未使用---	

リードNo. 0x83

桁数	内容	内容詳細
4	累計補充枚数	1 0 0 0 0 円 (1 0 ³ の位 ~ 1 0 ⁰ の位)
4		5 0 0 0 円 (")
4		2 0 0 0 円 (")
4		1 0 0 0 円 (")
48	---未使用---	

補充枚数取得

- ・ 形式

パラメータ	意味
<i>Command</i>	CHAN_DI_SUPPLYCOUNTS 補充枚数取得コマンド
<i>pData</i>	補充枚数指定モード
	0x00 補充枚数を pString で指定されたエリアにセットします。
	0x01 補充枚数を pString で指定されたエリアにセットするとともに、レジつり銭機は保持している補充枚数のクリアを行います。
<i>pString</i>	取得した補充枚数を格納するポインタ 補充された金種枚数を次のフォーマットで書式化してこのエリアにセットします。 “合計金額=(硬貨)金種:枚数, 金種:枚数, …; (紙幣)金種:枚数, 金種:枚数= 紙幣リジェクト件数; 硬貨リジェクト件数”の順に格納されます ex) 10円:4枚, 50円:1枚, 500円:1枚, 1000円:2枚, 5000円:1枚 硬貨リジェクト件数:5件の場合 pString= “7590=1:0, 5:0, 10: 4, 50: 1, 100:0, 500: 1; 1000: 2, 5000: 1, 10000:0=0:5”
- ・ 説明
 - ・ つり銭機に補充された金種枚数を読み取り、pStringに指定されたエリアにセットします。
 - ・ このコマンド実行直後につり銭機が保持している補充枚数のクリアを行うか否かの指示を行います。
 - ・ メカ動作は行いません。
- ・ 備考
 - ※ つり銭機にて保持する補充枚数の1金種あたり記憶枚数は最大255枚です。このコマンドを使用する際はアプリケーション側で適時、“pData=0x01”を実行し補充枚数のクリアを行わないと正しい補充枚数が取得できません。
- ・ 参照

精査

・ 形式 CHAN_DI_SEISA

パラメータ

意味

Command

CHAN_DI_SEISA

精査コマンド

pData

—未使用—

pString

取得した精査データを格納するポインタ
データの詳細は以降に記載します。

- ・ 説明
 - ・ つり銭機の在高枚数を読み取りそのデータを ASCII10 進表記文字列に変換して *pString* で指定されたエリアにセットします。
 - ・ メカの動作は行いません。

- ・ 備考
 - ※ 棒金ドロア (WD-300) 接続時にドロアを開けた状態で実行した場合、エラー応答になります。(ResultCodeExtended = OPOS_ECHAN_IFERROR が返ります。)
 - ※ 棒金ドロア (WD-300) 接続時に、SSW 設定にてドロア内在高を硬貨収納部在高に加算する設定としていた場合、実際の硬貨収納部在高が判断できなくなるため、払い出し指示時にエンptyエラーになる（払い出せない）場合があります。

・ 参照

標準仕様（紙幣接続時）

桁数	内容	内容詳細
	収納部	
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
	回収カセット	
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
	収納部	
3	5 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
	前回抜き取りカセット	
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
	累積抜き取りカセット	
4	2 0 0 0 円	(10 ³ の位～10 ⁰ の位)
4	1 0 0 0 0 円	(")
4	5 0 0 0 円	(")

4	1 0 0 0 円	(")
1	紙幣在高状態フラグ	
	Bit	内 容
	0	収納部
	1	回収カセット
	2	前回抜取りカセット
	3	累積抜取りカセット
		‘0’ : 在高異常なし ‘1’ : 在高異常あり
1	紙幣フル状態フラグ、処理機動作中フラグ	
	Bit	内 容
	0	R J 部
	1	回収カセット
	2	処理機動作中フラグ
	3	‘0’ 固定
		‘0’ : フル以外 ‘1’ : フル状態 ‘0’ : 処理機動作無し ‘1’ : 処理機動作中
1	硬貨在高状態フラグ、省電力中フラグ	
	Bit	内 容
	0	収納部
	1	‘0’ 固定
	2	‘0’ 固定
	3	省電力中フラグ
		‘0’ : 在高異常なし ‘1’ : 在高異常あり ‘0’ : 処理機通常動作中 ‘1’ : 処理機省電力中
1	硬貨フル状態フラグ	
	Bit	内 容
	0	‘0’ 固定
	1	収納部
	2	放出トレイ残留
	3	投入部硬貨あり
		‘0’ : その他の状態 ‘1’ : フル状態 (Bit2,3 の場合 あり)
合計 7 4		

※ 紙幣データは紙幣切離し時には‘0’固定

拡張仕様（紙幣接続時）

桁数	内容	内容詳細
収納部		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
回収カセット		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
収納部		
3	5 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
前回抜き取りカセット		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
累積抜き取りカセット		
4	2 0 0 0 円	(10 ³ の位～10 ⁰ の位)
4	1 0 0 0 0 円	(")
4	5 0 0 0 円	(")
4	1 0 0 0 円	(")
紙幣在高状態フラグ		
1	Bit	内 容
	0	収納部
	1	回収カセット
	2	前回抜き取りカセット
	3	累積抜き取りカセット
<div> <div>'0' : 在高異常なし</div> <div>'1' : 在高異常あり</div> </div>		
紙幣フル状態フラグ、処理機動作中フラグ		
1	Bit	内 容
	0	R J 部
	1	回収カセット
	2	処理機動作中フラグ
	3	'0' 固定
<div> <div>'0' : フル以外</div> <div>'1' : フル状態</div> <div>'0' : 処理機動作無し</div> <div>'1' : 処理機動作中</div> </div>		
硬貨在高状態フラグ、省電力中フラグ		
1	Bit	内 容
	0	収納部
	1	'0' 固定
	2	'0' 固定
	3	省電力中フラグ
<div> <div>'0' : 在高異常なし</div> <div>'1' : 在高異常あり</div> <div>'0' : 処理機通常動作中</div> <div>'1' : 処理機省電力中</div> </div>		

1	硬貨フル状態フラグ		
	Bit	内 容	
	0	‘0’ 固定	‘0’ : その他の状態 ‘1’ : フル状態 (Bit2,3 の場合 あり)
	1	収 納 部	
	2	放 出 ト レ イ 残 留	
3	投 入 部 硬 貨 有 り		
3	紙幣R J 件数		(1 0 ² の位～1 0 ⁰ の位)
3	硬貨R J 件数		(1 0 ² の位～1 0 ⁰ の位)
3	オーバーフロー 5 0 0 円		(1 0 ² の位～1 0 ⁰ の位)
3	オーバーフロー 1 0 0 円		(“)
3	オーバーフロー 5 0 円		(“)
3	オーバーフロー 1 0 円		(“)
3	オーバーフロー 5 円		(“)
3	オーバーフロー 1 円		(“)
合計 9 8			

※ 紙幣データは紙幣切離し時には‘0’ 固定

棒金分拡張仕様（紙幣接続時）

桁数	内容	内容詳細
収納部		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
回収カセット		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
収納部		
3	5 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
前回抜き取りカセット		
3	2 0 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 0 0 円	(")
3	5 0 0 0 円	(")
3	1 0 0 0 円	(")
累積抜き取りカセット		
4	2 0 0 0 円	(10 ³ の位～10 ⁰ の位)
4	1 0 0 0 0 円	(")
4	5 0 0 0 円	(")
4	1 0 0 0 円	(")
紙幣在高状態フラグ		
1	Bit	内 容
	0	収納部
	1	回収カセット
	2	前回抜き取りカセット
	3	累積抜き取りカセット
<div> <div>'0' : 在高異常なし</div> <div>'1' : 在高異常あり</div> </div>		
紙幣フル状態フラグ、処理機動作中フラグ		
1	Bit	内 容
	0	R J 部
	1	回収カセット
	2	処理機動作中フラグ
	3	'0' 固定
<div> <div>'0' : フル以外</div> <div>'1' : フル状態</div> <div>'0' : 処理機動作無し</div> <div>'1' : 処理機動作中</div> </div>		
硬貨在高状態フラグ、省電力中フラグ		
1	Bit	内 容
	0	収納部
	1	'0' 固定
	2	'0' 固定
	3	省電力中フラグ
<div> <div>'0' : 在高異常なし</div> <div>'1' : 在高異常あり</div> <div>'0' : 処理機通常動作中</div> <div>'1' : 処理機省電力中</div> </div>		

1	硬貨フル状態フラグ		
	Bit	内 容	
	0	‘0’ 固定	‘0’ : その他の状態 ‘1’ : フル状態 (Bit2,3 の場合 あり)
	1	収 納 部	
	2	放 出 ト レ イ 残 留	
	3	投 入 部 硬 貨 あ り	
3	紙幣RJ件数		(10 ² の位～10 ⁰ の位)
3	硬貨RJ件数		(10 ² の位～10 ⁰ の位)
3	オーバーフロー 500円		(10 ² の位～10 ⁰ の位)
3	オーバーフロー 100円		(“)
3	オーバーフロー 50円		(“)
3	オーバーフロー 10円		(“)
3	オーバーフロー 5円		(“)
3	オーバーフロー 1円		(“)
1	予備		
※1	1	包装硬貨部トレイ状態	‘0’ : トレイ閉 ‘1’ : トレイ開
	3	包装硬貨 500円	(10 ² の位～10 ⁰ の位)
	3	包装硬貨 100円	(“)
	3	包装硬貨 50円	(“)
	3	包装硬貨 10円	(“)
	3	包装硬貨 5円	(“)
	3	包装硬貨 1円	(“)
合計 118			

※1 : 拡張仕様に包装硬貨情報が付加された内容となります。

※ 包装硬貨部開中はコマンドはエラーとなります。

(ResultCodeExtended = OPOS_ECHAN_IFERROR が返ります。)

標準仕様（硬貨単体）

桁数	内容	内容詳細
	収 納 部	
3	5 0 0 円	(10^2 の位～ 10^0 の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
合 計 1 8		

拡張仕様（硬貨単体）

桁数	内容	内容詳細
	収 納 部	
3	5 0 0 円	(10^2 の位～ 10^0 の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
3	硬貨RJ件数	(10^2 の位～ 10^0 の位)
3	オーバーフロー 500円	(10^2 の位～ 10^0 の位)
3	オーバーフロー 100円	(")
3	オーバーフロー 50円	(")
3	オーバーフロー 10円	(")
3	オーバーフロー 5円	(")
3	オーバーフロー 1円	(")
合 計 3 9		

紙幣つり銭機互換仕様（硬貨単体接続）

桁数	内容	内容詳細
24	(予備)	
3	収 納 部 5 0 0 円	(10 ² の位～10 ⁰ の位)
3	1 0 0 円	(")
3	5 0 円	(")
3	1 0 円	(")
3	5 円	(")
3	1 円	(")
30	(予備)	
1	硬貨在高状態フラグ、省電力中フラグ	
	Bit 内 容	
	0 収 納 部	‘0’ : 在高異常なし
	1 ‘0’ 固定	‘1’ : 在高異常あり
	2 ‘0’ 固定	
	3 省電力中フラグ	‘0’ : 処理機通常動作中
		‘1’ : 処理機省電力中
1	硬貨フル状態フラグ	
	Bit 内 容	
	0 ‘0’ 固定	
	1 収 納 部	‘0’ : その他の状態
	2 放出トレイ残留	‘1’ : フル状態 (Bit2,3 の場合 残留あり)
	3 投入部硬貨あり	
3	(予備)	
3	硬貨RJ件数	(10 ² の位～10 ⁰ の位)
3	オーバーフロー 500円	(10 ² の位～10 ⁰ の位)
3	オーバーフロー 100円	(")
3	オーバーフロー 50円	(")
3	オーバーフロー 10円	(")
3	オーバーフロー 5円	(")
3	オーバーフロー 1円	(")
合計	98	

計数データリード

- ・形式 **CHAN_DI_DEPOSITDATAREAD**

パラメータ

意味

Command

CHAN_DI_DEPOSITDATAREAD

計数データリードコマンド

pData

—未使用—

pString

取得した処理結果データを格納するポインタ

- ・説明
 - ・つり銭機で処理された結果データを *pString* で指定されたエリアにセットします。
 - ・メカの動作は行いません。

- ・備考 ※ *pString* に格納されるデータの書式を以下に示します（合計39）。

桁数	説明	
1	計数情報	'1' 計数処理中 '1以外' 計数処理中以外
1	計数停止コマンド	'0' 未受信 '1' 受信済み
1	装置状態	'0' その他 '1' 計数動作中(SOH と同等) '2' 計数停止中 (EM と同等)
1	紙幣投入部情報	'0' 紙幣無し '1' 紙幣有り
1	紙幣部詳細情報	'0' その他 '1' 計数待機中 '2' 計数動作中 '3' エラー中 '4' RJ中 '5' カセットフル '6' セット外れ '7' リセット中 '8' 回収中
1	硬貨投入部情報	'0' 硬貨無し '1' 硬貨有り
1	硬貨部詳細情報	'0' その他 '1' 計数待機中 '2' 計数動作中 '3' エラー中 '4' RJ中 '5' 収納庫フル '6' セット外れ '7' リセット中
3	10000 円処理枚数	10 ⁴ ~ 10 ⁰
3	5000 円処理枚数	
3	2000 円処理枚数	
3	1000 円処理枚数	
3	500 円処理枚数	
3	100 円処理枚数	
3	50 円処理枚数	
3	10 円処理枚数	
3	5 円処理枚数	
3	1 円処理枚数	
1	装置接続情報	Bit0 : 0 固定 Bit1 : 0:紙幣接続 1:切り離し Bit2~7 : 0 固定
1	予備	'0' 固定

- ・参照

計数開始

・形式

CHAN_DI_BEGINDEPOSIT

パラメータ

意味

Command

CHAN_DI_BEGINDEPOSIT

計数開始コマンド

pData

——未使用——

pString

——未使用——

・説明

- ・計数処理を開始します。
- ・貨幣が投入された場合に計数動作へ移ります。
- ・つり銭機が計数開始コマンドを受け付けたタイミングでメソッドはリターンします。

・備考

- ※ このコマンドの呼び出しで以下のプロパティ値は初期化されます。
- ・ **DepositCounts** プロパティは各金種の枚数値が0にセットされます。
 - ・ **DepositAmount** プロパティは0がセットされます。
- (ResultCode が OPOS_SUCCESS の場合のみとなります。)

・参照

計数終了

・形式

CHAN_DI_ENDDEPOSIT

パラメータ

意味

Command

CHAN_DI_ENDDEPOSIT

計数終了コマンド

pData

——未使用——

pString

——未使用——

・説明

- ・ 計数処理を終了します。
- ・ つり銭機が計数処理を終了してニュートラル状態に戻ったタイミングでメソッドがリターンします。
- ・ 正常終了した後、投入口に貨幣を投入（挿入）した場合でも計数処理を行いません。

・備考

・参照

計数停止

・形式

CHAN_DI_PAUSEDDEPOSIT

パラメータ

意味

Command

CHAN_DI_PAUSEDDEPOSIT

計数停止コマンド

pData

——未使用——

pString

——未使用——

・説明

- ・ 計数処理を停止します。
- ・ 投入口に貨幣が無くなるまで計数動作を行い、以降計数動作を行わないようにします。
- ・ メソッドは計数が終了して計数停止状態に遷移したことを検知してからリターンします。
- ・ 正常終了した後、投入口に貨幣を投入（挿入）した場合でも計数処理を行いません。

・備考

- ※ 計数処理中にメソッドを呼出し、回収カセットフルが発生した場合は、その状態が解除されるまでメソッドがリターンしません。
- ※ DirectIO メソッド 計数再開コマンドを実行すると、再び計数処理を開始します。

・参照

計数再開

・形式

CHAN_DI_RESTARTDEPOSIT

パラメータ

意味

Command

CHAN_DI_RESTARTDEPOSIT

計数再開コマンド

pData

——未使用——

pString

——未使用——

・説明

- ・計数停止中状態にのみ、計数処理を再開します。
- ・貨幣が投入（挿入）された場合に計数動作へ移ります。
- ・つり銭機が計数再開コマンドを受け付けたタイミングでメソッドはリターンします。

・備考

・参照

計数機／つり銭機モード（計数モード設定）

・形式 CHAN_DI_DEPOSITMODE

パラメータ

Command	CHAN_DI_DEPOSITMODE	計数機／つり銭機モードコマンド															
pData	<p>実行するモードを指定します。</p> <p>以下の組み合わせ（論理和・OR 演算）にてモードを指定します。</p> <table> <tr> <td>設定フラグ</td> <td>0x0000</td> <td>モード照会</td> </tr> <tr> <td></td> <td>0x0100</td> <td>モード設定</td> </tr> <tr> <td>設定内容</td> <td>0x00</td> <td>つり銭機モード（預かり金手入力モード）</td> </tr> <tr> <td></td> <td>0x01</td> <td>計数機モード（預かり金計数モード）</td> </tr> <tr> <td></td> <td>0x02</td> <td>現金管理機モード</td> </tr> </table> <p>※設定フラグがモード照会の場合、設定内容は無効です。 設定内容は 0x00 を設定して下さい。</p> <p>ex) 設定フラグ：モード設定，設定内容：計数機モード（預かり金計数モード）， pData = 0x0100 + 0x01 = 0x0101 を指定します。</p>		設定フラグ	0x0000	モード照会		0x0100	モード設定	設定内容	0x00	つり銭機モード（預かり金手入力モード）		0x01	計数機モード（預かり金計数モード）		0x02	現金管理機モード
設定フラグ	0x0000	モード照会															
	0x0100	モード設定															
設定内容	0x00	つり銭機モード（預かり金手入力モード）															
	0x01	計数機モード（預かり金計数モード）															
	0x02	現金管理機モード															
pString	<p>実行後の設定内容を格納するポインタ</p> <p>書式は以下の通りです。</p> <table> <tr> <th>内容</th> <th>桁数</th> <th>説明</th> </tr> <tr> <td>設定フラグ</td> <td>2</td> <td> <p>実行した設定フラグがそのままセットされます。</p> <p>“00”：モード照会 “01”：モード設定</p> </td> </tr> <tr> <td>設定内容</td> <td>2</td> <td> <p>現在の設定内容がセットされます。</p> <p>“00”：つり銭機モード “01”：計数機モード</p> <p>“02”：現金管理機モード</p> </td> </tr> </table> <p>ex) pString = “0001” の場合 設定フラグ … “00” モード照会で実行された 設定内容 … “01” 計数機モードで設定されている</p>		内容	桁数	説明	設定フラグ	2	<p>実行した設定フラグがそのままセットされます。</p> <p>“00”：モード照会 “01”：モード設定</p>	設定内容	2	<p>現在の設定内容がセットされます。</p> <p>“00”：つり銭機モード “01”：計数機モード</p> <p>“02”：現金管理機モード</p>						
内容	桁数	説明															
設定フラグ	2	<p>実行した設定フラグがそのままセットされます。</p> <p>“00”：モード照会 “01”：モード設定</p>															
設定内容	2	<p>現在の設定内容がセットされます。</p> <p>“00”：つり銭機モード “01”：計数機モード</p> <p>“02”：現金管理機モード</p>															

- ・説明
 - ・計数モードの設定または照会を行います。
 - ・実行後、設定内容が報告されます。
 - ・メカ動作は行いません。
- ・備考
 - ※ つり銭機内に在高がある状態、または、在高異常の状態では現金管理機モードへの設定変更はできません。
 - ※ 設定完了後、内部で設定値の更新処理を行います。更新中に上位から指示された場合は、OPOS 内部で更新処理の完了を待ってから実行します。
- ・参照

ログ吸い上げ

・形式 CHAN_DI_GETLOG

パラメータ	意味
Command	CHAN_DI_GETLOG ログ吸い上げコマンド
pData	ログを取得する処理機を指定します。 0x00 紙幣つり銭機のログ取得を指定します。 0x01 硬貨つり銭機のログ取得を指定します。
pString	指定するログアドレスを格納する先のポインタ

値) 取得するメモリアドレスの範囲を ASCII16 進 4 桁にて
“開始アドレス; 終了アドレス” の形式で指定して下さい。

ex) 0x0000~0x000F (機種コード) の範囲を指定する場合、
“0000; 000F” となります。

- ・説明
 - ・接続されているつり銭機を対象にログを吸い上げテキスト形式でファイル作成を行います。
 - ・ファイルの保存先ディレクトリはレジストリ情報“LogSaveDir”になります。

- ・ログファイル名称は以下の書式にて作成します。

紙幣処理機 : ' B_+ " 開始アドレス" + ' _ ' + " 終了アドレス" .txt
硬貨処理機 : ' C_+ " 開始アドレス" + ' _ ' + " 終了アドレス" .txt

ex) 紙幣処理機のアドレス 0000H~1234H のログを取得した場合のログファイル
名称
“B_0000_1234.txt”

- ・備考
 - ※ 本コマンドで生成されたログファイルは、アプリケーション又は手動にて削除する必要があります。
 - ※ RT-300 硬貨単体接続時、紙幣部のパラメータは指定できません。
- ・参照

累計カウンタクリア

- ・ 形式 **CHAN_DI_COUNTCLR**

パラメータ	意味
<i>Command</i>	CHAN_DI_COUNTCLR 累計カウンタクリアコマンド
<i>pData</i>	——未使用——
<i>pString</i>	——未使用——

- ・ 説明 ・ 実行後、累計カウンタがクリアされます。
- ・ 備考
- ・ 参照

棒金ドロア制御

- ・ 形式 **CHAN_DI_OPENDRAWER**

パラメータ	意味
<i>Command</i>	CHAN_DI_OPENDRAWER 棒金ドロア制御コマンド
<i>pData</i>	制御する内容を指定します。
	0x01 包装硬貨部開許可
	0x02 包装硬貨部開禁止
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 包装硬貨部（WD-300）の開放／ロックの（許可・禁止）制御を行います。
 - POS側は包装硬貨部を開けたい場合、開許可を発行します。
 - ロックする場合には開禁止を発行します。
 - 包装硬貨部が開中に開禁止が発行された場合、包装硬貨部を閉めたタイミングで開禁止状態になります。
 - POSから開許可がきている状態では、在高ボタンを押しても包装硬貨部は開きません。
- ・ 備考
 - ※ 本コマンドを使用せずに包装硬貨部を制御する場合は、紙幣部 SSW13 のドロア開設定を 2:無効以外にする必要があります。
- ・ 参照

ローカル操作禁止

- ・ 形式 **CHAN_DI_CHILDLCK**

パラメータ	意味
<i>Command</i>	CHAN_DI_CHILDLCK ローカル操作禁止コマンド
<i>pData</i>	装置に対するローカル操作の禁止／許可を設定します。 0x00 ローカル操作禁止 0x01 ローカル操作許可
<i>pString</i>	——未使用——

- ・ 説明 ・ つり銭機と包装硬貨部（WD-300）に対して、ローカル操作禁止の制御を行います。

- ・ 備考

- ・ 参照

締め

・形式

CHAN_DI_CLOSE

パラメータ

意味

Command	CHAN_DI_CLOSE	締めコマンド
pData	モードを指定します。	
	0x00	交代処理
	0x01	締め処理
pString	送信時：レスポンスの ID 項目に入れる値 指定可能な値は、0~99999999 です。 受信時：取得した処理結果データを格納するポインタ	

・説明

- 交代データをシフトします。

・備考

- ※ 回収の場合はカセットを抜き取り、セット後に累積出金金額に加算されます。
- ※ 交代データをシフトすると交代処理もしくは締め処理実行前の交代データは再取得できません。必ず交代データを取得してから交代処理もしくは締め処理をしてください。
- ※ pString に格納されるデータの書式を以下に示します（合計 8 8）。

桁数	説明			
2	月	開始時間 (1 0 ¹ ~ 1 0 ⁰ の位・各 BCD2 桁)		
2	日			
2	時			
2	分			
2	月	現在時間 (1 0 ¹ ~ 1 0 ⁰ の位・各 BCD2 桁)		
2	日			
2	時			
2	分			
10	開始時在高金額	1 0 ⁹ ~ 1 0 ⁰ の位・BCD10 桁		
10	累積入金系金額	(")		
10	累積出金系金額	(")		
10	現在在高金額	(")		
10	(予備)			
2	紙幣在高異常	Bit	内容	データ
		7	収納庫開	0 : 正常 1 : 在高異常の可能性あり
		6-5	---未使用---	
		4	カセット	
		3	1 万円	
		2	5 千円	
		1	2 千円	
		0	千円	
2	硬貨在高異常	Bit	内容	データ
		7	収納庫開	0 : 正常 1 : 在高異常の可能性あり
		6	---未使用---	
		5	5 0 0 円	
		4	1 0 0 円	
		3	5 0 円	
		2	1 0 円	
		1	5 円	
		0	1 円	
2	紙幣区間ステータス	Bit	内容	データ
		7-4	---未使用---	0 : 発生していない
		3	紙幣 RJ	1 : 発生した

		2	現在紙幣 RJ	
		1	在高異常	
		0	収納庫開による 在高異常	
2	硬貨区間ステータス	Bit	内容	データ
		7-3	---未使用---	
		2	包装硬貨矯正開	0：発生していない
		1	在高異常	1：発生した
		0	収納庫開による 在高異常	
2	シリアルナンバー	1 6 ¹ ～ 1 6 ⁰ の位		
8	ID	1 0 ⁷ ～ 1 0 ⁰ の位・BCD8 桁		
4	(予備)			

・参照

補充開始

・形式

CHAN_DI_SUPPLY

パラメータ

意味

Command

CHAN_DI_SUPPLY

補充開始コマンド

pData

未使用

pString

未使用

・説明

- ・ 補充処理を開始します。
- ・ POS のつり銭補充処理等に用いることで、取引データ（入金）との管理データ区別が可能です。
- ・ 貨幣が投入（挿入）された場合に計数動作へ移ります。
- ・ 釣銭機が補充開始コマンドを受け付けたタイミングでメソッドはリターンします。

・備考

- ※ 補充処理の停止、再開、終了する場合は、DirectIO メソッド-計数停止、計数再開、計数終了コマンドを使用します。（PauseDeposit、FixDeposit、EndDeposit でも可能です。）
- ※ このコマンドの呼び出しで以下のプロパティ値は初期化されます。
 - ・ DepositCounts プロパティは各金種の枚数値が 0 にセットされます。
 - ・ DepositAmount プロパティは 0 がセットされます。（ResultCode が OPOS_SUCCESS の場合のみとなります。）
- ※ ローカル操作での補充処理と同じ動作をします。
（例：つり銭機モードでリジェクト発生時はリジェクト BOX に格納されます。
預かり金モードでリジェクト発生時は払い出し口に払い出されます。）

・参照

取引外入金開始

- ・ 形式 **CHAN_DI_BEGINDEPOSITOUTSIDE**

パラメータ	意味
<i>Command</i>	CHAN_DI_BE EGINDEPOSITOUTSIDE 取引外入金開始コマンド
<i>pData</i>	未使用
<i>pString</i>	未使用

- ・ 説明
 - ・ 取引外入金動作を開始します。
 - ・ POS の取引以外のお金処理（両替等）に用いることで、取引データ（入金）との管理データ区別が可能です。
 - ・ 貨幣が投入（挿入）された場合に計数動作へ移ります。
 - ・ つり銭機が取引外入金開始コマンドを受け付けたタイミングでメソッドはリターンします。
- ・ 備考
 - ※ 取引外入金処理の停止、再開、終了する場合は、DirectIO メソッド-計数停止、計数再開、計数終了コマンドを使用します。（PauseDeposit、FixDeposit、EndDepositでも可能です。）
 - ※ このコマンドの呼び出しで以下のプロパティ値は初期化されます。
 - ・ **DepositCounts** プロパティは各金種の枚数値が 0 にセットされます。
 - ・ **DepositAmount** プロパティは 0 がセットされます。（ResultCode が OPOS_SUCCESS の場合のみとなります。）
- ・ 参照

取引外金額指定出金

- ・ 形式 **CHAN_DI_DISPENSECHANGEOUTSIDE**

パラメータ	意味
<i>Command</i>	CHAN_DI_DISPENSECHANGEOUTSIDE 取引外出金コマンド（金額指定放出）
<i>pData</i>	払い出すつり銭の合計を指定します。
<i>pString</i>	未使用

- ・ 説明
 - ・ 取引外出金処理を行います。
 - ・ POS の取引以外の出金処理（両替処理等）に用いることで、取引データ（入出金）との管理データ区別が可能です。
- ・ 備考
 - ※ 本メソッドは、CurrentExit=1 のみをサポートします。
（CurrentExit=2 を指定時は、メソッドは OPOS_E_ILLEGAL）応答します。）
それ以外の動作仕様は専用メソッド-DispenseChange () と同じです。
- ・ 参照

取引外枚数指定出金

- ・ 形式 **CHAN_DI_DISPENSECASHOUTSIDE**

パラメータ	意味
<i>Command</i>	CHAN_DI_DISPENSECASHOUTSIDE 取引外出金コマンド（枚数指定放出）
<i>pData</i>	未使用
<i>pString</i>	払い出す金種と枚数を指定します。（DispenseCash と同じ。） “金種:枚数, 金種:枚数, . . .” の書式で指定します。 ・ 金種と枚数の間は、’ : ’（半角コロン）を使用 ・ 上記の項目の区切りには、’ , ’（半角カンマ）を使用

Ex) 50 円 2 枚、100 円 4 枚、500 円 1 枚、1,000 円 5 枚を払出す場合
” 50:2,100:4,500:1:1000:5”

- ・ 説明
 - ・ 取引外出金処理を行います。
 - ・ POS の取引以外の出金処理（両替処理等）に用いることで、取引データ（入出金）との管理データ区別が可能です。
- ・ 備考
 - ※ 本メソッドは、CurrentExit=1 のみをサポートします。
（CurrentExit=2 を指定時は、メソッドは OPOS_E_ILLEGAL）応答します。）
それ以外の動作仕様は専用メソッド-DispenseCash() と同じです。
- ・ 参照

現金戻し開始

- ・ 形式 **CHAN_DI_BEGINCASHRETURN**

パラメータ	意味
<i>Command</i>	CHAN_DI_BEGINCASHRETURN 現金戻し開始コマンド
<i>pData</i>	未使用
<i>pString</i>	未使用

- ・ 説明
 - ・ 現金戻し(戻し入れ)処理を開始します。
- ・ 備考
 - ※ 現金戻し処理の停止、再開、終了する場合は、DirectIO メソッド-計数停止、計数再開、計数終了コマンドを使用します。(PauseDeposit、FixDeposit、EndDepositでも可能です。)
 - ※ このコマンドの呼び出しで以下のプロパティ値は初期化されます。
 - ・ **DepositCounts** プロパティは各金種の枚数値が0にセットされます。
 - ・ **DepositAmount** プロパティは0がセットされます。
(ResultCode が OPOS_SUCCESS の場合のみとなります。)
 - ※ 本メソッドは、つり銭機が現金管理機モード時のみをサポートします。
(それ以外のモード時に実行した場合は、OPOS_ECHAN_IMPOSSIBLE を返します。)
- ・ 参照

交代データ取得

- ・ 形式 CHAN_DI_CLOSEDATAREAD

パラメータ

意味

Command CHAN_DI_CLOSEDATAREAD 交代データ取得コマンド

pData 参照するインデックスを指定します。(0~23)

pString 取得した処理結果データを格納するポインタ

- ・ 説明
 - ・ 0~23 件までの指定された任意の交代データを取得します。

- ・ 備考 ※ pString に格納されるデータの書式を以下に示します (合計 90)。

桁数	説明			
2	データインデックス	10 ¹ ~ 10 ⁰ の位 (0~23) ※ 参照インデックス指定と同じ		
2	月	開始時間 (10 ¹ ~ 10 ⁰ の位・各 BCD2 桁)		
2	日			
2	時			
2	分			
2	月	現在時間 (10 ¹ ~ 10 ⁰ の位・各 BCD2 桁)		
2	日			
2	時			
2	分			
10	開始時在高金額	10 ⁹ ~ 10 ⁰ の位・BCD10 桁		
10	累積入金系金額	(")		
10	累積出金系金額	(")		
10	現在在高金額	(")		
10	(予備)			
2	紙幣在高異常	Bit	内容	データ
		7	収納庫開	0: 正常 1: 在高異常の可能性あり
		6-5	---未使用---	
		4	カセット	
		3	1 万円	
		2	5 千円	
		1	2 千円	
		0	千円	
2	硬貨在高異常	Bit	内容	データ
		7	収納庫開	0: 正常 1: 在高異常の可能性あり
		6	---未使用---	
		5	500 円	
		4	100 円	
		3	50 円	
		2	10 円	
		1	5 円	
		0	1 円	
2	紙幣区間ステータス	Bit	内容	データ
		7-4	---未使用---	0: 発生していない 1: 発生した
		3	紙幣 RJ	
		2	現在紙幣 RJ	
		1	在高異常	
		0	収納庫開による 在高異常	
2	硬貨区間ステータス	Bit	内容	データ
		7-3	---未使用---	0: 発生していない
		2	包装硬貨強制開	1: 発生した

		1	在高異常
		0	収納庫開による 在高異常
2	シリアルナンバー	1 6 ¹ ~ 1 6 ⁰ の位	
8	ID	1 0 ⁷ ~ 1 0 ⁰ の位・BCD8 桁	
4	(予備)		

・ 参照

エラー解除ガイダンス起動

- ・ 形式 **CHAN_DI_ERRGUIDANCE (=101)**

パラメータ	意味
<i>Command</i>	CHAN_DI_ERRGUIDANCE エラー解除誘導ガイダンスの起動
<i>pData</i>	自動ポップアップ（常に手前に表示）するかどうかを指定します。 0x00 自動ポップアップしません。 0x01 自動ポップアップします。
<i>pString</i>	表示座標 (X, Y) を格納する先のポインタ 値) 表示するガイダンス（ウィンドウサイズは X: 640, Y: 480 です）の左上座標を ASCII10 進で “X 座標, Y 座標” の形式にて指定してください。 ex) 解像度が 1024×768 のディスプレイに対して、中央にガイダンスウィンドウを表示させたい場合、 “192, 144” となります。

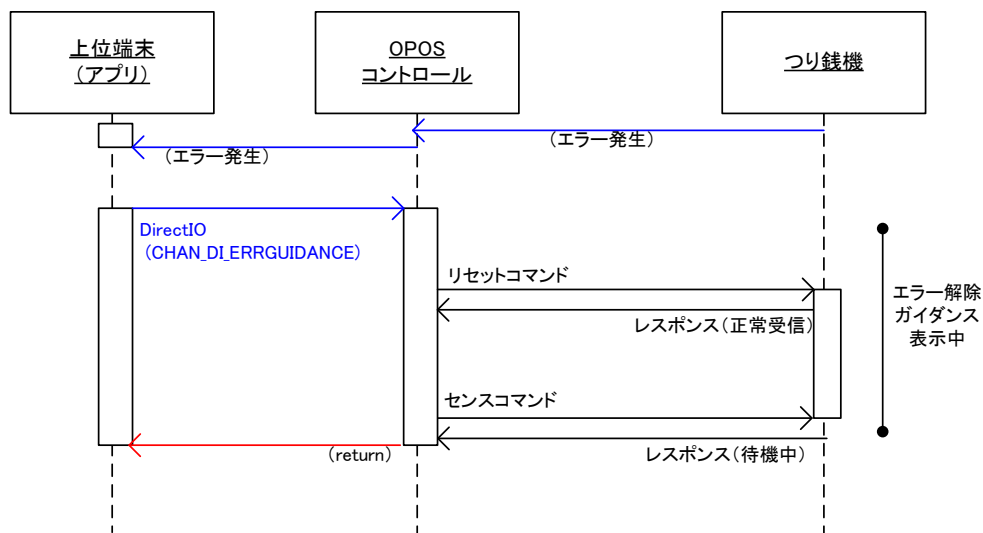
- ・ 説明
 - ・ 上位から与えられたパラメータの情報を元にエラー解除ガイダンスの画面を表示します。
 - ・ 解除ガイダンス画面ではリセット動作完了までを監視します。
 - ・ 処理機にエラーが発生していない時に本メソッドをコールされた場合は、ガイダンスを起動せずに OPOS_SUCCESS を返します。
 - ・ 処理機の電源 OFF/ON が必要なエラーの場合は、ガイダンスが起動しません。
- ・ 戻り値 次の値のいずれかが戻され、**ResultCode** プロパティにも格納されます。

ResultCode	意味
OPOS_SUCCESS	本メソッドは正常に終了しました。 (エラー解除完了しました。又はメソッドは非同期に実行されました) 処理機はエラーではありません。
OPOS_E_ILLEGAL	パラメータの指定が不正です。
OPOS_E_CLOSED	デバイスがオープンされていません。
OPOS_E_NOTCLAIMED	排他アクセス権を獲得していません。
OPOS_E_DISABLED	デバイスが Disable 状態です。
OPOS_E_NOHARDWARE	デバイスが本体に接続されていないか、電源が入っていません。
OPOS_E_BUSY	非同期にて実行中のため、本メソッドが実行できません。
OPOS_E_EXTENDED	詳細内容が ResultCodeExtended プロパティに格納されています。 OPOS_ECHAN_IFERROR 通信異常。 OPOS_ECHAN_ERROR エラー解除を中断しました。

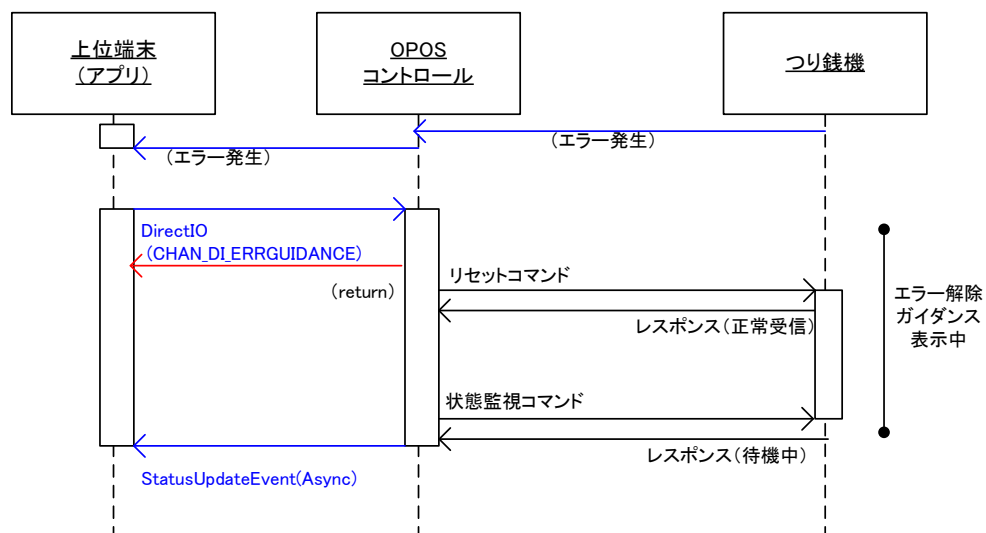
- ・ 備考
 - ※ **AsyncMode** プロパティが TRUE の場合は、非同期にて動作します。
非同期で実行した場合、非同期完了イベント (CHAN_STATUS_ASYNC) をトリガに AsyncResultCode, AsyncResultCodeExtended プロパティを参照していただければ、実行結果が確認できます。

※ シーケンス図

・ AsyncMode = FALSE の場合



・ AsyncMode = TRUE の場合



・ 参照

4. 4. イベント

DataEvent イベント

- ・ 形式 `void DataEvent (LONG Status);`

パラメータ

意味

Status

0（ゼロ）が固定でセットされます。

- ・ 説明 釣り銭機の状態が、【計数動作中】からそれ以外（計数待機中・計数停止中・エラー発生等）に変化したタイミングで本イベントの発行を行います。
これにより、アプリケーションは計数動作の終了を知ることができます。

状態変化によりイベントの発行を行う為、計数枚数が0枚の場合でも発行されます。

- ・ 備考 ※ **本イベントは入金処理時のみ発行されます。**
 ※ イベントの発生については、*DataEventEnabled*と*FreezeEvents*に依存します。
発生しない場合は先のプロパティの設定を確認してください。
 ※ レジストリ “*DataEventTiming*” により DataEvent の通知条件を変更する事が出来ます。

“Once”	上記説明の通りの条件で本イベントの発行を行います。 レジストリ値が無い場合、この設定をデフォルト値とします。
“Realtime”	計数枚数の変化を検知するとリアルタイムにイベントを通知します。 計数データが0（ゼロ）の場合、このイベントは通知されません。
“Realtime&Notify”	計数枚数の変化を検知するとリアルタイムにイベントを通知します。 計数データが0（ゼロ）の場合でも動作終了時にイベントを通知します。
※ レジストリ “ <i>DataEventLog</i> ” により DataEvent を時系列ログに出力する事が出来ます。	
“TRUE”	イベントの発行を時系列ログに出力します。 レジストリ “ <i>DataEventTiming</i> ” を “Once” 以外に設定した場合は、計数の都度イベントが通知され、膨大にログが出力されるため、本設定は推奨できません。
“FALSE”	イベントの発行を時系列ログに出力しません。

- ・ 参照 ◇ APG 「入力モデルについて」
 ◇ DataEventEnabled プロパティ
 ◇ FreezeEvents プロパティ

StatusUpdateEvent イベント

- ・ 形式 `void StatusUpdateEvent (LONG Status);`

パラメータ	意味
Status	デバイスの状態が変化した時、その状態が設定されます。
OPOS_SUE_POWER_ONLINE	電源オンでかつレディ状態です。
OPOS_SUE_POWER_OFF	電源オフまたは本体に接続されていません。
OPOS_SUE_POWER_OFFLINE	電源オンですがノットレディ状態です。
CHAN_STATUS_JAM	機器障害が生じました。
CHAN_STATUS_JAMOK	機器障害が解消しました。
CHAN_STATUS_EMPTY	エンプティの金種があります。
CHAN_STATUS_NEAREMPTY	ニアエンプティの金種があります。
CHAN_STATUS_EMPTYOK	エンプティ、ニアエンプティの状態が解除されました。
CHAN_STATUS_FULL	フルの金種があります。
CHAN_STATUS_NEARFULL	ニアフルの金種があります。
CHAN_STATUS_FULLOK	フル、ニアフルの状態が解除されました。
CHAN_STATUS_ASYNC	非同期動作が終了しました。

- ・ 説明 つり銭機の充填レベル（フル～エンプティ）・障害等の状態が発生した時に本イベントの発行を行います。

- ・ 備考 ※ 1 金種が既にエンプティ状態の時、他の金種がエンプティになってもイベントは発生しません。

※ 電源通知機能は、PowerNotify = *OPOS_ENABLED* の場合に発生します。

※ エンプティステータス (*CHAN_STATUS_EMPTY*) 及び、ニアエンプティステータス (*CHAN_STATUS_NEAREMPTY*) について

- ・ つり銭機から収納庫のエンプティ、ニアエンプティ情報を取得した場合に通知を行います。但し、収納庫金種未設定の金種については通知を行いません。
- ・ 各金種毎のエンプティ情報通知の設定を無しとした場合、金種毎の通知を行いません。

※ フルスステータス (*CHAN_STATUS_FULL*)

- ・ つり銭機から収納庫のフル情報を取得した場合に通知を行います。但し、収納庫金種未設定の金種については通知を行いません。
- ・ レジストリ “*Bill_Full/StatusEvent*” により *CHAN_STATUS_FULL* のイベント通知条件を変更する事が出来ます。(紙幣部のみ有効)

“STORE”	いずれかの金種の収納庫がフルになったタイミングで通知します。この設定をデフォルト値とします。
“NONE”	収納庫フル、回収カセットフルのどちらが発生してもイベントは通知しません。

※ ニアフルステータス (*CHAN_STATUS_NEARFULL*)

- ・ つり銭機から収納庫のニアフル情報を取得した場合に通知を行います。但し、収納庫金種未設定の金種については通知を行いません。

※ イベントの発生については、*FreezeEvents* に依存します。
イベントの発生を一時的に停める場合は先のプロパティを TRUE に、再開する場合は FALSE をセットしてください。

- ・ 参照 ◇ APG 「入力モデルについて」
- ◇ DataEventEnabled プロパティ
- ◇ FreezeEvents プロパティ

DirectIO イベント

- ・ 形式 `VOID DirectIOEvent (LONG EventNumber, LONG* pData, BSTR* pString);`

パラメータ	意味
<i>EventNumber</i>	イベント番号がセットされます。
<i>pData</i>	数値データへのポインタ。 値はイベント番号とサービスオブジェクトによって変化します。
<i>pString</i>	文字列データへのポインタ。 値はコマンド番号とサービスオブジェクトによって変化します。 このデータのフォーマットは BinaryConversion プロパティの値に依存します。

- ・ 説明 アプリケーションと直接通信する為に、サービスオブジェクトが通知します。

このイベントにより、標準コントロールオブジェクトではサポートされないイベントを、サービスオブジェクトがアプリケーションに提供する事ができるようになります。

- ・ 備考 ※ **本コントロールでサポートする機能は以下の通りです。**
尚、詳細については以降に記載します。

機能	EventNumber (イベント番号)	
貨幣抜き取り待ち	CHAN_DIEVT_CASSETTEWAIT	(=1)
補充動作検知	CHAN_DIEVT_CHARGING	(=2)
計数中エラー発生	CHAN_DIEVT_DEPOSITERROR	(=3) ※1
計数中RJ (リジェクト) 発生	CHAN_DIEVT_DEPOSITRJ	(=4)
計数中回収カセットフル	CHAN_DIEVT_DEPOSITCASSETTEFULL	(=5) ※1
計数中セット外れ	CHAN_DIEVT_DEPOSITSETERROR	(=6) ※1
貨幣抜き取り完了	CHAN_DIEVT_PULLOUT	(=7)
計数待機中	CHAN_DIEVT_DEPOSITREADY	(=8)

※1：レジストリ “DepositErrorEvent” により DirectIO イベントの通知条件を変更する事が出来ます。

“DIEVT”	入金中に事象発生時に、DirectIO イベントで通知します。
“SUE”	入金中に発生した DirectIO イベントを全て StatusUpdate イベント (CHAN_STATUS_JAM、CHAN_STATUS_FULL) に変換して通知します。
“BOTH”	DirectIO イベントと同時に StatusUpdate イベントを通知します。

- ※ レジストリ “DirectIOEventTiming” の値が “Delayed” の場合、入金中に発生したイベントは DataEvent 通知後 (計数停止後) に通知します。但し、計数動作停止前に事象を解除するとイベントは通知されません。(紙幣部：RJ 発生・硬貨部：計数動作中の場合に計数動作終了前に RJ 紙幣を抜き取ると、イベントは通知しません。)

“Realtime” の場合は検知したタイミングで通知します。

- ・ 参照 ◇ DirectIO メソッド

貨幣抜き取り待ちイベント

- ・ 形式 `CHAN_DIEVT_CASSETTEWAIT`

パラメータ	意味
<i>Command</i>	<code>CHAN_DI_CASSETTEWAIT</code> 貨幣抜き取り待ちイベント
<i>pData</i>	0 (固定)
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 紙幣または硬貨のどちらかが貨幣抜き取り待ち状態の開始タイミングの報告を行います。
 - ・ ただし `AsyncMode = TRUE` でメソッド実行中などアプリケーション側でイベントの受け取りが直ちに行われない状態ではイベントの発行は行われません。
- ・ 備考
- ・ 参照

補充動作検知イベント

- ・ 形式 **CHAN_DIEVT_CHARGING**

パラメータ	意味
<i>Command</i>	CHAN_DIEVT_CHARGING 補充動作検知イベント
<i>pData</i>	0 (固定)
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 補充動作が行われた事を報告します。
 - ・ つり銭機の状態が補充中から他の状態へ遷移した事を検知したタイミングでイベントを発行します。
 - ・ ただしSSW設定—計数確定機モード設定がつり銭機モード になっている場合のみイベントは発行可能です。
- ・ 備考
 - ※ **DirectIO メソッド-補充開始により補充処理を行った場合は、つり銭機モードであっても本イベントは通知されません。**
- ・ 参照

計数中 エラー発生イベント

- ・ 形式 **CHAN_DIEVT_DEPOSITERROR**

パラメータ	意味
<i>Command</i>	CHAN_DIEVT_DEPOSITERROR 計数中エラー発生イベント
<i>pData</i>	0 (固定)
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 計数処理中にエラーが発生した事を報告します。
 - ・ エラーを解除しない限り、処理を終了する事ができません。
- ・ 備考
 - ※ **DirectIO メソッド-補充開始による補充処理中にエラーが発生した場合も、本イベントは通知されます。**
- ・ 参照

計数中 RJ 発生イベント

- ・ 形式 CHAN_DIEVT_DEPOSITRJ

パラメータ	意味
Command	CHAN_DIEVT_DEPOSITRJ 計数中RJ発生イベント
pData	0 (固定)
pString	——未使用——

- ・ 説明
 - ・ 計数処理中に貨幣がリジェクトされた事を報告します。
 - ・ リジェクト貨幣を取り除かない限り、処理を継続する事ができません。
- ・ 備考
 - ※ DirectIO メソッド-補充開始による補充処理中にリジェクトが発生した場合も、本イベントは通知されます。
- ・ 参照

計数中 回収カセットフルイベント

- ・ 形式 `CHAN_DIEVT_DEPOSITCASSETTEFULL`

パラメータ	意味
<i>Command</i>	<code>CHAN_DIEVT_DEPOSITCASSETTEFULL</code> 計数中カセットフル発生イベント
<i>pData</i>	0 (固定)
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 計数中に回収カセットフルになった事を報告します。
 - ・ 回収カセットの紙幣を回収しない限り、処理を継続する事ができません。
- ・ 備考
 - ※ `DirectIO` メソッド-補充開始による補充処理中に回収カセットフルが発生した場合も、本イベントは通知されます。
- ・ 参照

計数中 セット外れイベント

- ・ 形式 **CHAN_DIEVT_DEPOSITSETERROR**

パラメータ	意味
<i>Command</i>	CHAN_DIEVT_DEPOSITSETERROR 計数中セット外れ発生イベント
<i>pData</i>	0 (固定)
<i>pString</i>	——未使用——

- ・ 説明
 - ・ 計数中にセット外れになった事を報告します。
 - ・ セット外れを正常に戻さない限り、処理を継続する事ができません。
- ・ 備考
 - ※ **DirectIO メソッド-補充開始による補充処理中にセット外れになった場合も、本イベントは通知されます。**
- ・ 参照

貨幣抜き取り完了イベント

- ・ 形式 **CHAN_DIEVT_PULLOUT**

パラメータ	意味
Command	CHAN_DIEVT_PULLOUT 貨幣抜き取り完了発生イベント
pData	0 (固定)
pString	——未使用——

- ・ 説明
 - ・ 装置の出金口から貨幣が抜き取られた事を報告します。
 - ・ 装置抜き取り待ち状態（DC2）からの変化（ACK/ETB）をトリガに通知します。
- ・ 備考
 - ※ レジストリ “Compatibility” の設定値（初期値 “00”）により、本イベントの通知・非通知を切り替えることが可能です。
Bit2-ON：条件を満たした場合に通知する。
Bit2-OFF：通知しない。
- ・ 参照

計数待機中イベント

- ・ 形式 **CHAN_DIEVT_DEPOSITREADY**

パラメータ	意味
Command	CHAN_DIEVT_DEPOSITREADY 計数待機中遷移イベント
pData	0 (固定)
pString	——未使用——

- ・ 説明
 - ・ 装置が計数待機中状態に変化した事を報告します。
 - ・ 装置エラー中状態からの変化をトリガに通知します。
- ・ 備考
 - ※ レジストリ “Compatibility” の設定値（初期値 “00”）により、本イベントの通知・非通知を切り替えることが可能です。
Bit3-ON : 条件を満たした場合に通知する。
Bit3-OFF : 通知しない。
 - ※ DirectIO メソッド補充開始による補充処理中での状態変化でも、本イベントは通知されます。
- ・ 参照

[illegible]

[illegible]

5. ログ仕様

5.1. ログ仕様

※ログ処理を行うのは、Open() メソッド ~ Close() メソッド間です。

※本書に記載する内容は、開発用・デバッグ用を目的としており上位アプリケーションでの製品仕様として使用した場合の責任は負いかねます。

5.1.1. 時系列ログ

メソッドを実行する毎にロギングされます。2行/1件（実行時と実行結果）のテキスト形式で1ファイル20,000件（デフォルト）とします。

14稼働日分（デフォルト）のファイルを保持しそれ以前のファイルは削除します。

【ファイル名】

`J_yymmdd.log`

※ `yymmdd` … ファイルを保存した日付

【ファイル名パス】

レジストリにて指定されたパス。

【容 量】

1ファイルの容量は、処理されるコマンドおよび件数により異なります。

【フォーマット】

・メソッド／プロパティセット（開始時）

年月日	,	処理開始 時間	,	“START” (5byte)	,	メソッド名称 (プロパティ名称)	,	パラメータ/データ
-----	---	------------	---	--------------------	---	---------------------	---	-----------

・メソッド／プロパティセット（完了時）

年月日	,	処理完了 時間	,	“END ” (5byte)	,	メソッド名称 (プロパティ名称)	,	Result Code	,	ResultCode Extended	,
-----	---	------------	---	-------------------	---	---------------------	---	----------------	---	------------------------	---

AsyncResultCode	,	AsyncResultCode Extended	,	パラメータ/データ
-----------------	---	-----------------------------	---	-----------

・イベント

年月日	,	イベント セット時間	,	“ ” (空白) (5byte)	,	イベント 名称	,	パラメータ/データ
-----	---	---------------	---	---------------------	---	------------	---	-----------

【年月日】

yyyymmdd 形式 (8byte)

【処理開始時間】 / 【処理完了時間】

hh:mm:ss.ms

(hh…時 24 時間表記 2 桁、mi…分 2 桁、ss…秒 2 桁、ms…ミリ秒 3 桁 計 12byte)

[メソッド（プロパティ）名称]

実行したメソッド（プロパティ）名称が記述されます。（MAX 19byte）

ex) **DispenseChage** メソッドの場合 … “DispenseChange”

DirectIO メソッドの場合、コマンド名称が記述されます。

また、プロパティのセット時もログされます。

[イベント名称]

発行したイベント名称が記述されます。（MAX 19byte）

ex) **DirectIOEvent** イベントの場合 … “DirectIOEvent”

[ResultCode] / [AsyncResultCode]

ResultCode・**AsyncResultCode** 値名称が記述されます。（MAX 17byte）

ex) **OPOS_SUCCESS** の場合 … “OPOS_SUCCESS”

AsyncResultCode については、非同期動作を行った場合のみ値は有効です。

それ以外の場合、常に **OPOS_SUCCESS** で記述されます。

[ResultCodeExtended] / [AsyncResultCodeExtended]

ResultCodeExtended・**AsyncResultCodeExtended** 値名称が記述されます。

（MAX 23byte）

ex) **OPOS_ECHAN_OVERDISPENSE** の場合 … “OPOS_ECHAN_OVERDISPENSE”

AsyncResultCode については、非同期動作を行った場合のみ値は有効です。

それ以外の場合、常に **OPOS_SUCCESS** で記述されます。

ResultCodeExtended 値については、＜ResultCode 一覧＞を参照

[パラメータ/データ]

各メソッドによって異なります。

内容は以下の通りです。

No.	メソッド/プロパティ/イベント	パラメータ/データ
1	BinaryConversion	設定要求値
2	DataEventEnabled	”
3	DeviceEnabled	”
4	FreezeEvents	”
5	PowerNotify	”
6	AsyncMode	”
7	CurrencyCode	”
8	CurrentExit	”
9	CurrentService	”
10	RealTimeDataEnabled	”
11	Open	OpenResult 値、レジストリ読み取りデータ、エラー要因（エラー時）
12	Close	（無し）
13	ClaimDevice	Timeout パラメータ値
14	ReleaseDevice	（無し）
15	CheckHealth ※ 1	自己診断レベル値
16	ClearInput	（無し）
17	CompareFirmWareVersion	ファイル名、結果
18	ResetStatistics	スタティスティクス格納先
19	RetrieveStatistics	スタティスティクス格納先
20	UpdateFirmware	ファイル名

21	UpdateStatistics	スタティスティクス格納先
22	BeginDeposit	(無し)
23	DispenseCash	CurrentExit 値、出金指示文字列
24	DispenseChange	CurrentExit 値、出金指示金額
25	EndDeposit	パラメータ値
26	FixDeposit	(無し)
27	PauseDeposit	パラメータ値
28	ReadCashCounts	pCashCounts パラメータ値、pDiscrepancy パラメータ値、
29	CHAN_DI_RESET	(無し)
30	CHAN_DI_MEMREAD	指定アドレス、リードデータ
31	CHAN_DI_MEMCLR	指定メモリNo.
32	CHAN_DI_CHGMODE	設定モード値
33	CHAN_DI_SSWSET	設定機種、SSWNo., 設定値
34	CHAN_DI_TIMESET	設定時間
35	CHAN_DI_ENQ	レスポンス値
36	CHAN_DI_STRING	タイムアウト値、送信文字列
37	CHAN_DI_COLLECT	指定回収モード
38	CHAN_DI_STATUSREAD	リードNo.、リードデータ
39	CHAN_DI_SUPPLYCOUNT	補充枚数データ、モード、データ
40	CHAN_DI_SEISA	精査データ
41	CHAN_DI_DEPOSITDATAREAD	リードした処理データ
42	CHAN_DI_BEGINDEPOSIT	(無し)
43	CHAN_DI_ENDDEPOSIT	(無し)
44	CHAN_DI_PAUSEEPOSIT	(無し)
45	CHAN_DI_RESTARTDEPOSIT	(無し)
46	CHAN_DI_DEPOSITMODE	設定フラグ、内容、レスポンス
47	CHAN_DI_GETLOG	(無し)
48	CHAN_DI_COUNTCLR	(無し)
49	CHAN_DI_OPENDRAWER	パラメータ値
50	CHAN_DI_CHILDLCK	設定機種
51	CHAN_DI_CLOSE	(無し)
52	CHAN_DI_SUPPLY	(無し)
53	CHAN_DI_BEGINDEPOSITOUTSIDE	(無し)
54	CHAN_DI_DISPENSECHANGEOUTSIDE	CurrentExit 値、出金指示金額
55	CHAN_DI_DISPENSECASHOUTSIDE	CurrentExit 値、出金指示文字列
56	CHAN_DI_BEGINCASHRETURN	(無し)
57	CHAN_DI_CLOSEDATAREAD	パラメータ値
58	CHAN_DI_ERRGUIDANCE	パラメータ値
59	DataEvent	0 固定
60	DirectIOEvent	EventNo に対するキャプション(名称)、pData 値
61	StatusUpdateEvent	Status に対するキャプション (名称)、pData(0 固定)
	COFreezeEvent	”

※ 1 : 実行されるパラメータにより、“CH_Internal” , “CH_External” , “CH_Interactive” と表現します。

5.1.2. 処理ログ

メソッドの実行回数を累積します。また、**ResultCode**、**ResultCodeExtended** が更新される毎にカウントアップします。9,999 件を超えると 1 件となります。

14 稼働日分(デフォルト)のファイルを保持しそれ以前のファイルは削除します。

【ファイル名】

Ryymmdd. log

※ yymmdd ... ファイルを保存した日付

【容 量】

1 ファイルは 2,592 バイト。

14 稼働日分の場合

2,592 バイト × 14 稼働日 = 36,288 バイト

【フォーマット】

```
*** ResultCode 件数 ***
OPOS_SUCCESS                0097件
OPOS_E_CLOSED                0000件
OPOS_E_CLAIMED               0000件
OPOS_E_NOTCLAIMED            0000件
OPOS_E_NOSERVICE             0000件
OPOS_E_DISABLED              0000件
OPOS_E_ILLEGAL               0001件
OPOS_E_NOHARDWARE            0000件
OPOS_E_OFFLINE               0000件
OPOS_E_NOEXIST               0000件
OPOS_E_EXISTS                0000件
OPOS_E_FAILURE               0000件
OPOS_E_TIMEOUT               0001件
OPOS_E_BUSY                  0000件
OPOS_E_EXTENDED              0002件

*** ResultCodeExtended 件数 ***
OPOS_ECHAN_OVERDISPENSE      0000件
OPOS_ECHAN_TOTALOVER         0000件
OPOS_ECHAN_CHANGEERROR       0000件
OPOS_ECHAN_OVER              0000件
OPOS_ECHAN_IFERROR           0000件
OPOS_ECHAN_SETError          0000件
OPOS_ECHAN_ERROR             0002件
OPOS_ECHAN_CHARGING           0000件
OPOS_ECHAN_NEAREMPTY          0000件
OPOS_ECHAN_EMPTY             0000件
OPOS_ECHAN_NEARFULL          0000件
OPOS_ECHAN_FULL              0000件
OPOS_ECHAN_OVERFLOW          0000件
OPOS_ECHAN_REJECT            0000件
OPOS_ECHAN_BUSY              0000件
OPOS_ECHAN_ASYNCBUSY         0000件
OPOS_ECHAN_CASSETTEWAIT      0000件
OPOS_ECHAN_COLLECTWAIT       0000件
OPOS_ECHAN_COUNTERERROR      0000件
OPOS_ECHAN_AMOUNTERROR       0000件
OPOS_ECHAN_IMPOSSIBLE         0000件
OPOS_ECHAN_CANNOTPAY         0000件
OPOS_ECHAN_NOTSTORE          0000件
OPOS_ECHAN_NEUTRAL            0000件
OPOS_ECHAN_DEPOSIT           0001件
OPOS_ECHAN_PAUSEDDEPOSIT     0000件
OPOS_ECHAN_UNMATCH           0000件
OPOS_ECHAN_DEPOSIT_ELSE_BILL  0000件
OPOS_ECHAN_DEPOSIT_ELSE_COIN  0000件
OPOS_ECHAN_DEPOSIT_MOVE_BILL  0000件
OPOS_ECHAN_DEPOSIT_MOVE_COIN  0000件
OPOS_ECHAN_DEPOSIT_ERR_BILL   0000件
OPOS_ECHAN_DEPOSIT_ERR_COIN   0000件
OPOS_ECHAN_DEPOSIT_RJ_BILL    0000件
OPOS_ECHAN_DEPOSIT_RJ_COIN    0000件
OPOS_ECHAN_DEPOSIT_CAS_BILL   0000件
OPOS_ECHAN_DEPOSIT_OVF_COIN   0000件
OPOS_ECHAN_DEPOSIT_SET_BILL   0000件
OPOS_ECHAN_DEPOSIT_SET_COIN   0000件
OPOS_ECHAN_DEPOSIT_RESET_BILL 0000件
OPOS_ECHAN_DEPOSIT_RESET_COIN 0000件
```

5.1.3. 通信ログ

処理機と通信しコマンド・レスポンスに変化があった時にロギングされます。1件/1行で時系列ログファイルがMAX件数ロギングされると通信ログの書き込みを停止します。14稼働日分（デフォルト）のファイルを保持しそれ以前のファイルは削除します。

【ファイル名】

T_yymmdd.log

※ yymmdd ... ファイルを保存した日付

【容 量】

1件、1ファイルの容量は処理されるコマンドおよび時系列ログのロギング件数により異なります。

正常に動作していた場合、時系列ログの約4~5倍、途中でエラーになりエラー解除等を行った場合の容量は不明です。

【ファイル名パス】

レジストリにて指定されたパス。

【容 量】

1ファイルの容量は、処理されるコマンドおよび件数により異なります。

【フォーマット】

・コマンドデータ送信時

年月日		時分秒	W->	コマンド名称		コマンドデータ
8byte	,	8byte	,		,	

・レスポンスデータ受信時

年月日		時分秒	R->	コマンド名称		レスポンスデータ
8byte	,	8byte	,		,	

5.1.4. 取引ログ

つり銭機に計数された金種枚数およびコントロールソフトから放出／回収指示された枚数がロギングされます。

1件/1行で1日に処理されるすべての取引をロギングします。

14稼働日分（デフォルト）のファイルを保持しそれ以前のファイルは削除します。

【ファイル名】

Hyymmdd.log

※ yymmdd … ファイルを保存した日付

【容 量】

1件は124バイト。

1ファイルの容量は1日の取引件数により異なります。

【フォーマット】

※「合計金額」以降の各金種欄の「(3)」等はその項目の桁数を示します。

1. 計数の場合

年月日 8byte	.	時分秒 6byte	.	入金処理枚数 12byte	.	予備 1byte	.	合計金額 (8)	=	1 (1)	:	枚数 (3)
--------------	---	--------------	---	------------------	---	-------------	---	-------------	---	----------	---	-----------

※1

5 (1)	:	枚数 (3)	,	10 (2)	:	枚数 (3)	,	50 (2)	:	枚数 (3)	,	100 (3)	:	枚数 (3)	,	500 (3)	:	枚数 (3)
----------	---	-----------	---	-----------	---	-----------	---	-----------	---	-----------	---	------------	---	-----------	---	------------	---	-----------

1000 (4)	:	枚数 (3)	,	2000 (4)	:	枚数 (3)	,	5000 (4)	:	枚数 (3)	,	10000 (5)	:	枚数 (3)	空白 (8)
-------------	---	-----------	---	-------------	---	-----------	---	-------------	---	-----------	---	--------------	---	-----------	--------

2. 補充の場合

年月日 8byte	.	時分秒 6byte	.	補充計数枚数 12byte	.	予備 1byte	.	合計金額 (8)	=	1 (1)	:	枚数 (3)
--------------	---	--------------	---	------------------	---	-------------	---	-------------	---	----------	---	-----------

※1

5 (1)	:	枚数 (3)	,	10 (2)	:	枚数 (3)	,	50 (2)	:	枚数 (3)	,	100 (3)	:	枚数 (3)	,	500 (3)	:	枚数 (3)
----------	---	-----------	---	-----------	---	-----------	---	-----------	---	-----------	---	------------	---	-----------	---	------------	---	-----------

1000 (4)	:	枚数 (3)	,	2000 (4)	:	枚数 (3)	,	5000 (4)	:	枚数 (3)	,	10000 (5)	:	枚数 (3)	空白 (8)
-------------	---	-----------	---	-------------	---	-----------	---	-------------	---	-----------	---	--------------	---	-----------	--------

※ 補充動作停止直後に追加補充が行われた場合、つり銭機側で保持するログと本ログで書き込みタイミングが異なります。

取引ログ（本ログ）：追加補充を別取引としてログします。

つり銭機側のログ：追加補充終了時点をも1取引としてログします。

3. 金額指定放出の場合

年月日 8byte	時分秒 6byte	金額指定放出 12byte	予備 1byte	合計金額 (8)	空白 87byte
--------------	--------------	------------------	-------------	-------------	--------------

4. 枚数指定放出の場合

年月日 8byte	時分秒 6byte	枚数指定放出 12byte	予備 1byte	合計金額 (8)	=	1 (1)	:	枚数 (3)
--------------	--------------	------------------	-------------	-------------	---	----------	---	-----------

5 (1)	:	枚数 (3)	,	10 (2)	:	枚数 (3)	,	50 (2)	:	枚数 (3)	,	100 (3)	:	枚数 (3)	,	500 (3)	:	枚数 (3)
----------	---	-----------	---	-----------	---	-----------	---	-----------	---	-----------	---	------------	---	-----------	---	------------	---	-----------

1000 (4)	:	枚数 (3)	,	2000 (4)	:	枚数 (3)	,	5000 (4)	:	枚数 (3)	,	10000 (5)	:	枚数 (3)	空白 (8)
-------------	---	-----------	---	-------------	---	-----------	---	-------------	---	-----------	---	--------------	---	-----------	-----------

5. 回収の場合

年月日 8byte	時分秒 6byte	回収 (+ 空白 8byte) 12byte	予備 1byte	合計金額 (8)	=	1 (1)	:	枚数 (3)
--------------	--------------	------------------------------	-------------	-------------	---	----------	---	-----------

※2

※3

5 (1)	:	枚数 (3)	,	10 (2)	:	枚数 (3)	,	50 (2)	:	枚数 (3)	,	100 (3)	:	枚数 (3)	,	500 (3)	:	枚数 (3)
----------	---	-----------	---	-----------	---	-----------	---	-----------	---	-----------	---	------------	---	-----------	---	------------	---	-----------

1000 (4)	:	枚数 (3)	,	2000 (4)	:	枚数 (3)	,	5000 (4)	:	枚数 (3)	,	10000 (5)	:	枚数 (3)	空白 8byte
-------------	---	-----------	---	-------------	---	-----------	---	-------------	---	-----------	---	--------------	---	-----------	-------------

※4

※1 つり銭機の持つ補充枚数エリアのクリアを指示するパラメータをセットしています。

0 : クリアしない 1 : クリアする

※2 枚数指定された部位についてのみ、金額算出を行います。

※3 硬貨部の回収モードにより異なります。

ex) 上記のフォーマットは、モード9：硬貨部指示された金種：枚数を回収する場合です。

それ以外のモードに関しては、下の表を参照してください。

モード	内 容 (41byte)
0	硬貨部回収指示なし + 空白 (23byte)
1	硬貨部 500 円の回収指示 + 空白 (20byte)
2	硬貨部 100 円の回収指示 + 空白 (20byte)
3	硬貨部 50 円の回収指示 + 空白 (21byte)
4	硬貨部 10 円の回収指示 + 空白 (21byte)
5	硬貨部 5 円の回収指示 + 空白 (22byte)
6	硬貨部 1 円の回収指示 + 空白 (22byte)
7	硬貨部全ての回収指示 + 空白 (21byte)
8	硬貨部残置回収指示 + 空白 (23byte)
9	硬貨部指示された金種：枚数 + 空白 (なし)

※4 紙幣部の回収モードにより異なります。

例) 上記のフォーマットは、モード6：紙幣部指示された金種：枚数を回収する場合です。

それ以外のモードに関しては、下の表を参照してください。

モード	内 容 (36byte)
0	紙幣部回収指示なし + 空白 (18byte)
1	紙幣部 1000 円の回収指示 + 空白 (10byte)
2	紙幣部 5000 円の回収指示 + 空白 (10byte)
3	紙幣部全ての回収指示 + 空白 (14byte)
4	紙幣部残置回収指示 + 空白 (16byte)
5	紙幣部回収カセット在高クリア指示 + 空白 (2byte)
6	紙幣部指示された金種：枚数 + 空白 (なし)
7	紙幣部 10000 円の回収指示 + 空白 (なし)
8	紙幣部 2000 円の回収指示 + 空白 (なし)

5.1.5. 処理再現ログ

コントロールソフトが処理機のログを処理と処理の間の IDLE 時間を利用して、吸い上げおよびログファイルに書き込みます。

14 稼働日分（デフォルト）のファイルを保持しそれ以前のファイルは削除します。

【ファイル名】

M_y_y m m d d . l o g

※ y y m m d d … ファイルを保存した日付

【容 量】

ログは可変長で処理機の動作状況により 1 ファイルの容量は異なります。

【フォーマット】

つり銭機の入出金・エラー・ユニット開閉などの履歴がログされます。

- ※ 設定ツール（OPOSSetup.exe）にて、[つり銭機トレース用ログ設定] を“する”に設定している場合のみログされます。
- ※ つり銭機の接続パターンを変更する場合、必ず LOG フォルダ配下にある“logdat.ini”ファイル削除してください。削除されない場合、正しくログファイルが出力されなくなります。
- ※ 添付のログ変換ツールを使用して、ログ内容を確認できます。

5.2. その他設定

以下の設定は本コントロールソフトのインストール時に設定されます。

1. ログ停止／再開

時系列ログ・処理ログ・通信ログ・補充枚数ログをロギングする／しないを設定します。

2. 保存稼働日の設定

ログファイルを何稼働日分保持するかを設定します。デフォルトは 14 稼働日です。

MAX14 稼働日分

3. 時系列ログ件数設定

時系列ログを何件保持するかを設定します。デフォルトは 20,000 件です。

MAX 2147483,647 件

※一取引でのメソッドコール数や一日の取引件数が多い場合は、デフォルト値を越える可能性があります。

[illegible]

[illegible]

6. 補足事項

6.1. コード一覧

6.1.1. ResultCode

値		意味
0	OPOS_SUCCESS	正常動作
101	OPOS_E_CLOSED	State = OPOS_S_CLOSED の時にアクセスしようとした
102	OPOS_E_CLAIMED	Claimed = TRUE の時にアクセスしようとした
103	OPOS_E_NOTCLAIMED	Claimed = FALSE の時にメソッドまたはプロパティ設定処理をしようとした
104	OPOS_E_NOSERVICE	COがSOと通信できません
105	OPOS_E_DISABLED	DeviceEnabled = FALSE の時に動作しようとした
106	OPOS_E_ILLEGAL	デバイスに無効な動作か、サポートされていない動作を実行しようとしたか、無効なパラメータを使用しました
107	OPOS_E_NOHARDWARE	デバイスがシステムに接続されていないか、電源 OFF の状態です
108	OPOS_E_OFFLINE	デバイスがオフラインです
109	OPOS_E_NOEXIST	ファイル名 (他の指定値) が存在しません
110	OPOS_E_EXISTS	ファイル名 (他の指定値) がすでに存在しています
111	OPOS_E_FAILURE	リクエストされた処理をデバイスが実行できません
112	OPOS_E_TIMEOUT	デバイスからの応答を待っているSOがタイムアウトしたか、SOからの応答を待っているCOがタイムアウトしました
113	OPOS_E_BUSY	State = OPOS_S_BUSY の時処理を実行しようとした
114	OPOS_E_EXTENDED	クラス固有のエラーが発生しました

6.1.2. ResultCodeExtended

値		意味
201	OPOS_ECHAN_OVERDISPENSE	機内在高を超える金額・枚数が指定されました
202	OPOS_ECHAN_TOTALOVER	---未使用---
203	OPOS_ECHAN_CHANGEERROR	---未使用---
204	OPOS_ECHAN_OVER	放出限度額、枚数データが指定されました。
205	OPOS_ECHAN_IFERROR	データ異常・通信異常
206	OPOS_ECHAN_SETERROR	セットはずれ
207	OPOS_ECHAN_ERROR	異常終了・エラー中
208	OPOS_ECHAN_CHARGING	補充中
209	OPOS_ECHAN_NEAREMPTY	---未使用---
210	OPOS_ECHAN_EMPTY	---未使用---
211	OPOS_ECHAN_NEARFULL	---未使用---
212	OPOS_ECHAN_FULL	フルを検知しました
213	OPOS_ECHAN_OVERFLOW	---未使用---
214	OPOS_ECHAN_REJECT	---未使用---
215	OPOS_ECHAN_BUSY	処理機動作中にコマンド発行した時
216	OPOS_ECHAN_ASYNCBUSY	---未使用---
217	OPOS_ECHAN_CASSETTEWAIT	回収カセット抜取待ち・貨幣抜取待ち
218	OPOS_ECHAN_COLLECTWAIT	---未使用---
219	OPOS_ECHAN_COUNTERERROR	---未使用---
220	OPOS_ECHAN_AMOUNTERROR	---未使用---
221	OPOS_ECHAN_IMPOSSIBLE	コマンド受付不可状態
222	OPOS_ECHAN_CANNOTPAY	---未使用---
223	OPOS_ECHAN_NOTSTORE	---未使用---
224	OPOS_ECHAN_NEUTRAL	待機中 (ACK/ETB)
225	OPOS_ECHAN_DEPOSIT	計数計数中 (SOH)
226	OPOS_ECHAN_PAUSEDDEPOSIT	計数停止中 (EM)
227	OPOS_ECHAN_UNMATCH	モード不一致
228	OPOS_ECHAN_DEPOSIT_ELSE_BILL	入金中詳細状態 0: その他 (紙幣)
229	OPOS_ECHAN_DEPOSIT_ELSE_COIN	入金中詳細状態 0: その他 (硬貨)
230	OPOS_ECHAN_DEPOSIT_MOVE_BILL	入金中詳細状態 2: 動作中 (紙幣)
231	OPOS_ECHAN_DEPOSIT_MOVE_COIN	入金中詳細状態 2: 動作中 (硬貨)
232	OPOS_ECHAN_DEPOSIT_ERR_BILL	入金中詳細状態 3: エラー (紙幣)
233	OPOS_ECHAN_DEPOSIT_ERR_COIN	入金中詳細状態 3: エラー (硬貨)
234	OPOS_ECHAN_DEPOSIT_RJ_BILL	入金中詳細状態 4: RJ 中 (紙幣)
235	OPOS_ECHAN_DEPOSIT_RJ_COIN	入金中詳細状態 4: RJ 中 (硬貨)
236	OPOS_ECHAN_DEPOSIT_CAS_BILL	入金中詳細状態 5: カセットフル (紙幣)
237	OPOS_ECHAN_DEPOSIT_OVF_COIN	入金中詳細状態 5: 収納庫フル (硬貨)
238	OPOS_ECHAN_DEPOSIT_SET_BILL	入金中詳細状態 6: セット外 (紙幣)
239	OPOS_ECHAN_DEPOSIT_SET_COIN	入金中詳細状態 6: セット外 (硬貨)
240	OPOS_ECHAN_DEPOSIT_RESET_BILL	入金中詳細状態 7: リセット (紙幣)
241	OPOS_ECHAN_DEPOSIT_RESET_COIN	入金中詳細状態 7: リセット (硬貨)

6.1.3. OpenResult

値		意味
301	OPOS_OR_ALREADYOPEN	コントロールはすでにオープンされています。
302	OPOS_OR_REGBADNAME	レジストリに指定したデバイス名称キーが存在しません。
303	OPOS_OR_REGPROGID	デバイス名キーのデフォルト値が読めないか、そこに保持されたプログラマティック ID を有効なクラス ID に変換できませんでした。
304	OPOS_OR_CREATE	サービスオブジェクトインスタンスを生成できなかったか、IDispatch インターフェースを取得できませんでした。
305	OPOS_OR_BADIF	サービスオブジェクトは指定版数で要求されるひとつ、もしくはひとつ以上のメソッドをサポートしていません。
306	OPOS_OR_FAILEDOPEN	サービスオブジェクトはオープン呼び出しでエラーのステータスを返しました、しかしながらそれ以上の定義されたエラーコードを返しませんでした。
307	OPOS_OR_BADVERSION	サービスオブジェクトのメジャーバージョンはコントロールオブジェクトのメジャーバージョンに適合していません
401	OPOS_ORS_NOPORT	サービスオブジェクトで使用する COM ポートがオープンできませんでした。
402	OPOS_ORS_NOTSUPPORTED	サービスオブジェクトは指定されたデバイスをサポートしていません。
403	OPOS_ORS_CONFIG	構成情報エラー
451	OPOS_ORS_EVENTCLASS	イベント処理用オブジェクトクラスの作成に失敗しました。
452	OPOS_ORS_COCREATE	サービスオブジェクト側からコントロールオブジェクト側のインターフェイスを取得できませんでした。
453	OPOS_ORS_PORTCONTROL	COM ポート設定時にエラーが発生しました。
454	OPOS_ORS_EVENTTHREAD	イベント処理用スレッドの起動に失敗しました。
455	OPOS_ORS_SENSETHREAD	センススレッドの起動に失敗しました。

6. 2. 付録

6. 2. 1. O P O S A L L . B A S

※ 自動つり銭機関連箇所を抜粋

```

REM *////////////////////////////////////
REM *////////////////////////////////////
REM *////////////////////////////////////
REM *
REM * OPOSALL.BAS
REM *
REM * Includes all OPOS Device Classes.
REM *
REM * Modification history
REM * -----
REM * 96-03-18 OPOS Release 1.01 CRM
REM * 96-04-22 OPOS Release 1.1 CRM
REM * 97-06-04 OPOS Release 1.2 CRM
REM * 98-03-06 OPOS Release 1.3 CRM
REM * 99-06-18 OPOS Release 1.4 and 1.5 CRM
REM * 00-09-24 OPOS Release 1.5 BKS
REM * 01-07-15 OPOS Release 1.6 THH/BKS
REM *
REM * Note: This file is not actively maintained.
REM * The recommended source of up-to-date constants is the
REM * file Opos_Constants.dll. Add a reference to it, then
REM * use VB's Object Browser to find the constants.
REM *
REM *////////////////////////////////////
REM *////////////////////////////////////
REM *////////////////////////////////////

REM *////////////////////////////////////
REM *
REM * Opos.h
REM *
REM * General header file for OPOS Applications.
REM *
REM * Modification history
REM * -----
REM * 95-12-08 OPOS Release 1.0 CRM
REM * 97-06-04 OPOS Release 1.2 MtD CRM
REM * Add OPOS_FOREVER.
REM * Add BinaryConversion values.
REM * 98-03-06 OPOS Release 1.3 CRM
REM * Add CapPowerReporting, PowerState, and PowerNotify values.
REM * Add power reporting values for StatusUpdateEvent.
REM * 00-09-24 OPOS Release 1.5 CRM
REM * Add OpenResult status values.
REM * 00-07-18 OPOS Release 1.5 BKS
REM * Add many values and Point Card Reader Writer and
REM * POS Power sections.
REM *
REM *////////////////////////////////////

REM *////////////////////////////////////
REM * OPOS "State" Property Constants

```

REM *///

Public Const OposSClosed& = 1
Public Const OposSIdle& = 2
Public Const OposSBusy& = 3
Public Const OposSError& = 4

REM *///

REM * OPOS "resultCode" Property Constants

REM *///

Public Const OposSuccess& = 0
Public Const OposEClosed& = 101
Public Const OposEClaimed& = 102
Public Const OposENotclaimed& = 103
Public Const OposENoservice& = 104
Public Const OposEDisabled& = 105
Public Const OposEIllegal& = 106
Public Const OposENohardware& = 107
Public Const OposEOffline& = 108
Public Const OposENoexist& = 109
Public Const OposEExists& = 110
Public Const OposEFailure& = 111
Public Const OposETimeout& = 112
Public Const OposEBusy& = 113
Public Const OposEExtended& = 114

REM *///

REM * OPOS "OpenResult" Property Constants

REM *///

Public Const Oposopenerr& = 300

Public Const OposOrAlreadyopen& = 301
Public Const OposOrRegbadname& = 302
Public Const OposOrRegprogid& = 303
Public Const OposOrCreate& = 304
Public Const OposOrBadif& = 305
Public Const OposOrFailedopen& = 306
Public Const OposOrBadversion& = 307

Public Const Oposopenerrso& = 400

Public Const OposOrsNoport = 401
Public Const OposOrsNotSupported = 402
Public Const OposOrsConfig = 403
Public Const OposOrsSpecific = 450

REM *///

REM * OPOS "BinaryConversion" Property Constants

REM *///

Public Const OposBcNone& = 0
Public Const OposBcNibble& = 1
Public Const OposBcDecimal& = 2

REM *///

REM * "CheckHealth" Method: "Level" Parameter Constants

REM *///

```

Public Const OposChInternal&          = 1
Public Const OposChExternal&          = 2
Public Const OposChInteractive&       = 3

REM *////////////////////////////////////////
REM * OPOS "CapPowerReporting", "PowerState", "PowerNotify" Property
REM * Constants
REM *////////////////////////////////////////

Public Const OposPrNone&              = 0
Public Const OposPrStandard&          = 1
Public Const OposPrAdvanced&          = 2

Public Const OposPnDisabled&          = 0
Public Const OposPnEnabled&           = 1

Public Const OposPsUnknown&           = 2000
Public Const OposPsOnline&            = 2001
Public Const OposPsOff&               = 2002
Public Const OposPsOffline&           = 2003
Public Const OposPsOffOffline&        = 2004

'//*** 略 ***

REM *////////////////////////////////////////
REM * "StatusUpdateEvent" Event: Common "Status" Constants
REM *////////////////////////////////////////

Public Const OposSuePowerOnline&       = 2001
Public Const OposSuePowerOff&          = 2002
Public Const OposSuePowerOffline&      = 2003
Public Const OposSuePowerOffOffline&   = 2004

REM *////////////////////////////////////////
REM * General Constants
REM *////////////////////////////////////////

Public Const OposForever&              = -1

'//*** 略 ***

REM *////////////////////////////////////////
REM *
REM * OposChan.h
REM *
REM * Cash Changer header file for OPOS Applications.
REM *
REM * Modification history
REM * -----
REM * 97-06-04 OPOS Release 1.2                      CRM
REM * 00-09-24 OPOS Release 1.5                      OPOS-J
REM * Add DepositStatus Constants.
REM * Add EndDeposit Constants.
REM * Add PauseDeposit Constants.
REM *
REM *////////////////////////////////////////

```



```

REM *////////////////////////////////////
REM * "DeviceStatus" and "FullStatus" Property Constants
REM * "StatusUpdateEvent" Event Constants
REM *////////////////////////////////////

Public Const ChanStatusOk&          = 0          ' DeviceStatus, FullStatus

Public Const ChanStatusEmpty&        = 11         ' DeviceStatus, StatusUpdateEvent
Public Const ChanStatusNearempty&    = 12         ' DeviceStatus, StatusUpdateEvent
Public Const ChanStatusEmptyok&      = 13         ' StatusUpdateEvent

Public Const ChanStatusFull&         = 21         ' FullStatus, StatusUpdateEvent
Public Const ChanStatusNearfull&     = 22         ' FullStatus, StatusUpdateEvent
Public Const ChanStatusFullok&       = 23         ' StatusUpdateEvent

Public Const ChanStatusJam&          = 31         ' DeviceStatus, StatusUpdateEvent
Public Const ChanStatusJamok&        = 32         ' StatusUpdateEvent

Public Const ChanStatusAsync&        = 91         ' StatusUpdateEvent

REM *////////////////////////////////////
REM * "DepositStatus" Property Constants
REM *////////////////////////////////////

Public Const ChanStatusDepositStart& = 1
Public Const ChanStatusDepositEnd&   = 2
Public Const ChanStatusDepositNone&  = 3
Public Const ChanStatusDepositCount& = 4
Public Const ChanStatusDepositJam&   = 5

REM *////////////////////////////////////
REM * "EndDeposit" Property Constants
REM *////////////////////////////////////

Public Const ChanDepositChange&      = 1
Public Const ChanDepositNochange&    = 2
Public Const ChanDepositrepay&       = 3

REM *////////////////////////////////////
REM * "PauseDeposit" Property Constants
REM *////////////////////////////////////

Public Const ChanDepositPause&       = 11
Public Const ChanDepositRestart&     = 12

REM *////////////////////////////////////
REM * "ResultCodeExtended" Property Constants for Cash Changer
REM *////////////////////////////////////

Public Const OposEchanOverdispense& = 201

' /*** 略 ***/

REM *End of OPOSALL.BAS*

```

6.2.2. GLORY OPOS. BAS

```

' ///////////////////////////////////////////////////
' // "ResultCodeExtended" Property Constants for Cash Changer
' ///////////////////////////////////////////////////
Public Const OPOSERREXT = 200

Public Const OPOS_ECHAN_OVERDISPENSE As Long = 1 + OPOSERREXT
Public Const OPOS_ECHAN_TOTALOVER As Long = 2 + OPOSERREXT
Public Const OPOS_ECHAN_CHANGEERROR As Long = 3 + OPOSERREXT
Public Const OPOS_ECHAN_OVER As Long = 4 + OPOSERREXT
Public Const OPOS_ECHAN_IFERROR As Long = 5 + OPOSERREXT
Public Const OPOS_ECHAN_SETERORR As Long = 6 + OPOSERREXT
Public Const OPOS_ECHAN_ERROR As Long = 7 + OPOSERREXT
Public Const OPOS_ECHAN_CHARGING As Long = 8 + OPOSERREXT
Public Const OPOS_ECHAN_NEAREMPTY As Long = 9 + OPOSERREXT
Public Const OPOS_ECHAN_EMPTY As Long = 10 + OPOSERREXT
Public Const OPOS_ECHAN_NEARFULL As Long = 11 + OPOSERREXT
Public Const OPOS_ECHAN_FULL As Long = 12 + OPOSERREXT
Public Const OPOS_ECHAN_OVERFLOW As Long = 13 + OPOSERREXT
Public Const OPOS_ECHAN_REJECT As Long = 14 + OPOSERREXT
Public Const OPOS_ECHAN_BUSY As Long = 15 + OPOSERREXT
Public Const OPOS_ECHAN_ASYNCBUSY As Long = 16 + OPOSERREXT
Public Const OPOS_ECHAN_CASSETTEWAIT As Long = 17 + OPOSERREXT
Public Const OPOS_ECHAN_COLLECTWAIT As Long = 18 + OPOSERREXT
Public Const OPOS_ECHAN_COUNTERORR As Long = 19 + OPOSERREXT
Public Const OPOS_ECHAN_AMOUNTERROR As Long = 20 + OPOSERREXT
Public Const OPOS_ECHAN_IMPOSSIBLE As Long = 21 + OPOSERREXT
Public Const OPOS_ECHAN_CANNOTPAY As Long = 22 + OPOSERREXT
Public Const OPOS_ECHAN_NOTSTORE As Long = 23 + OPOSERREXT
Public Const OPOS_ECHAN_NEAUTRAL As Long = 24 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT As Long = 25 + OPOSERREXT
Public Const OPOS_ECHAN_PAUSEDDEPOSIT As Long = 26 + OPOSERREXT
Public Const OPOS_ECHAN_UNMATCH As Long = 27 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_ELSE_BILL As Long = 28 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_ELSE_COIN As Long = 29 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_MOVE_BILL As Long = 30 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_MOVE_COIN As Long = 31 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_ERR_BILL As Long = 32 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_ERR_COIN As Long = 33 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_RJ_BILL As Long = 34 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_RJ_COIN As Long = 35 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_CAS_BILL As Long = 36 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_OVF_COIN As Long = 37 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_SET_BILL As Long = 38 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_SET_COIN As Long = 39 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_RESET_BILL As Long = 40 + OPOSERREXT
Public Const OPOS_ECHAN_DEPOSIT_RESET_COIN As Long = 41 + OPOSERREXT

' ///////////////////////////////////////////////////
' // DirectIO Method Constants
' ///////////////////////////////////////////////////
Public Const CHAN_DI_RESET As Long = 1&
Public Const CHAN_DI_MEMREAD As Long = 2&
Public Const CHAN_DI_MEMCLEAR As Long = 3&
Public Const CHAN_DI_CHGMODE As Long = 4&
Public Const CHAN_DI_SSWSET As Long = 5&
Public Const CHAN_DI_TIMESET As Long = 6&
Public Const CHAN_DI_ENQ As Long = 7&
Public Const CHAN_DI_STRING As Long = 8&
Public Const CHAN_DI_COLLECT As Long = 9&
Public Const CHAN_DI_STATUSREAD As Long = 10&

```

```
Public Const CHAN_DI_SUPPLYCOUNTS As Long = 11&
Public Const CHAN_DI_SEISA As Long = 12&
Public Const CHAN_DI_DEPOSITDATAREAD As Long = 13&
Public Const CHAN_DI_BEGINDEPOSIT As Long = 14&
Public Const CHAN_DI_ENDDEPOSIT As Long = 15&
Public Const CHAN_DI_PAUSEDDEPOSIT As Long = 16&
Public Const CHAN_DI_RESTARTDEPOSIT As Long = 17&
Public Const CHAN_DI_DEPOSITMODE As Long = 18&
Public Const CHAN_DI_COUNTCLR As Long = 19&
Public Const CHAN_DI_GETLOG As Long = 20&
Public Const CHAN_DI_OPENDRAWER As Long = 21&
Public Const CHAN_DI_CHILDLCK As Long = 22&
Public Const CHAN_DI_CLOSE As Long = 25&
Public Const CHAN_DI_SUPPLY As Long = 26&
Public Const CHAN_DI_BEGINDEPOSITOUTSIDE As Long = 27&
Public Const CHAN_DI_DISPENSECHANGEOUTSIDE As Long = 28&
Public Const CHAN_DI_DISPENSECASHOUTSIDE As Long = 29&
Public Const CHAN_DI_BEGINCASHRETURN As Long = 30&
Public Const CHAN_DI_CLOSEDATAREAD As Long = 31&
Public Const CHAN_DI_ERRGUIDANCE As Long = 101&

' ////////////////////////////////////////////////////////////////////
' // DirectIO Event Constants
' ////////////////////////////////////////////////////////////////////
Public Const CHAN_DIEVT_CASSETTEWAIT As Long = 1
Public Const CHAN_DIEVT_CHARGING As Long = 2
Public Const CHAN_DIEVT_DEPOSITERROR As Long = 3
Public Const CHAN_DIEVT_DEPOSITRJ As Long = 4
Public Const CHAN_DIEVT_DEPOSITCASSETTEFULL As Long = 5
Public Const CHAN_DIEVT_DEPOSITSETERROR As Long = 6
Public Const CHAN_DIEVT_PULLOUT As Long = 7
Public Const CHAN_DIEVT_DEPOSITREADY As Long = 8

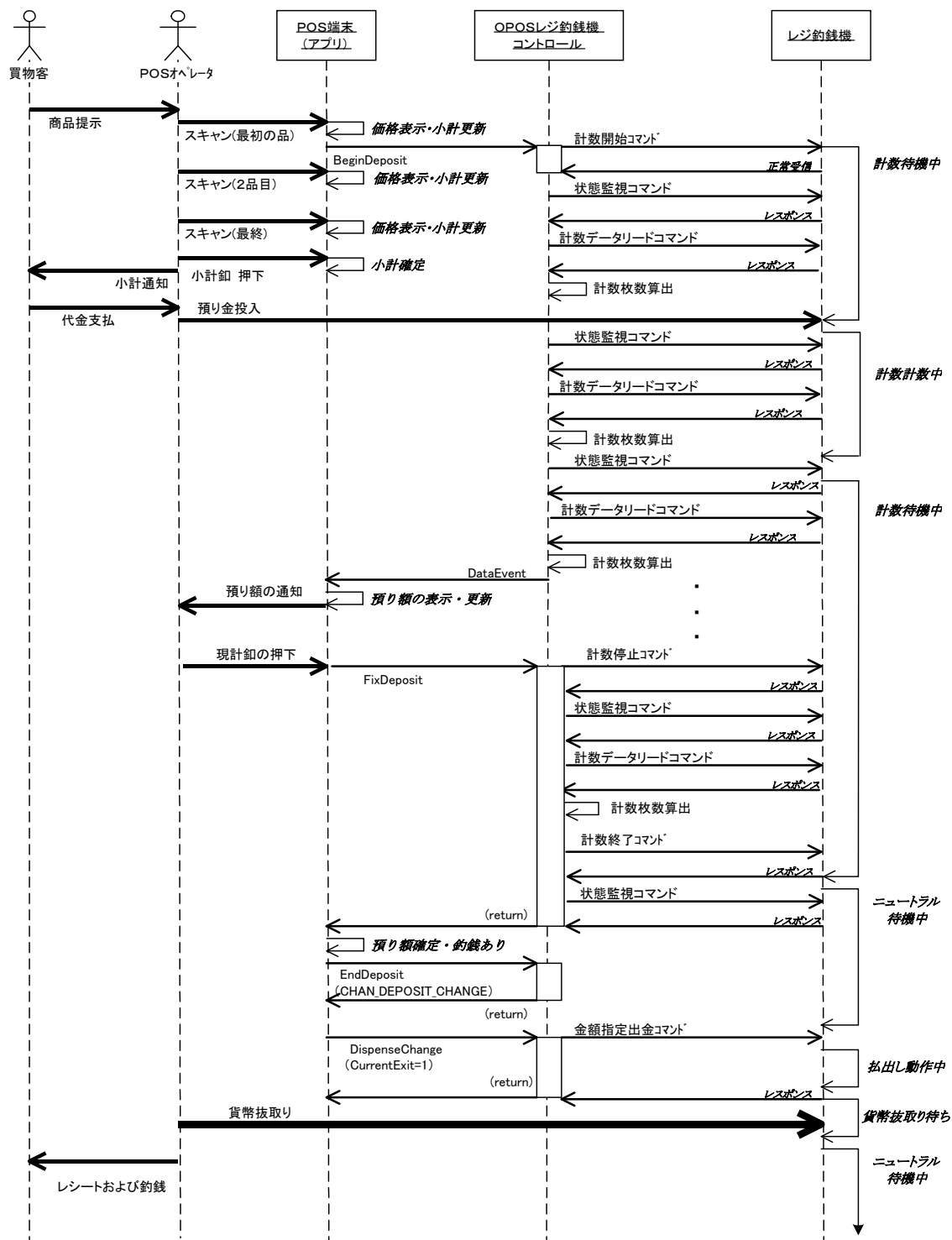
' ////////////////////////////////////////////////////////////////////
' // OpenResult Constants
' ////////////////////////////////////////////////////////////////////
Public Const OPOS_ORS_EVENTCLASS As Long = OPOSORSSPECIFIC + 1
Public Const OPOS_ORS_COCREATE As Long = OPOSORSSPECIFIC + 2
Public Const OPOS_ORS_PORTCONTROL As Long = OPOSORSSPECIFIC + 3
Public Const OPOS_ORS_EVENTTHREAD As Long = OPOSORSSPECIFIC + 4
Public Const OPOS_ORS_SENSETHREAD As Long = OPOSORSSPECIFIC + 5
```

6.3. シーケンス図

ここに掲載するシーケンス図は平易な同期処理を中心にまとめております。
下図に沿った運用方法と異なった制御を行う場合、検証や確認を十分に実施してください。

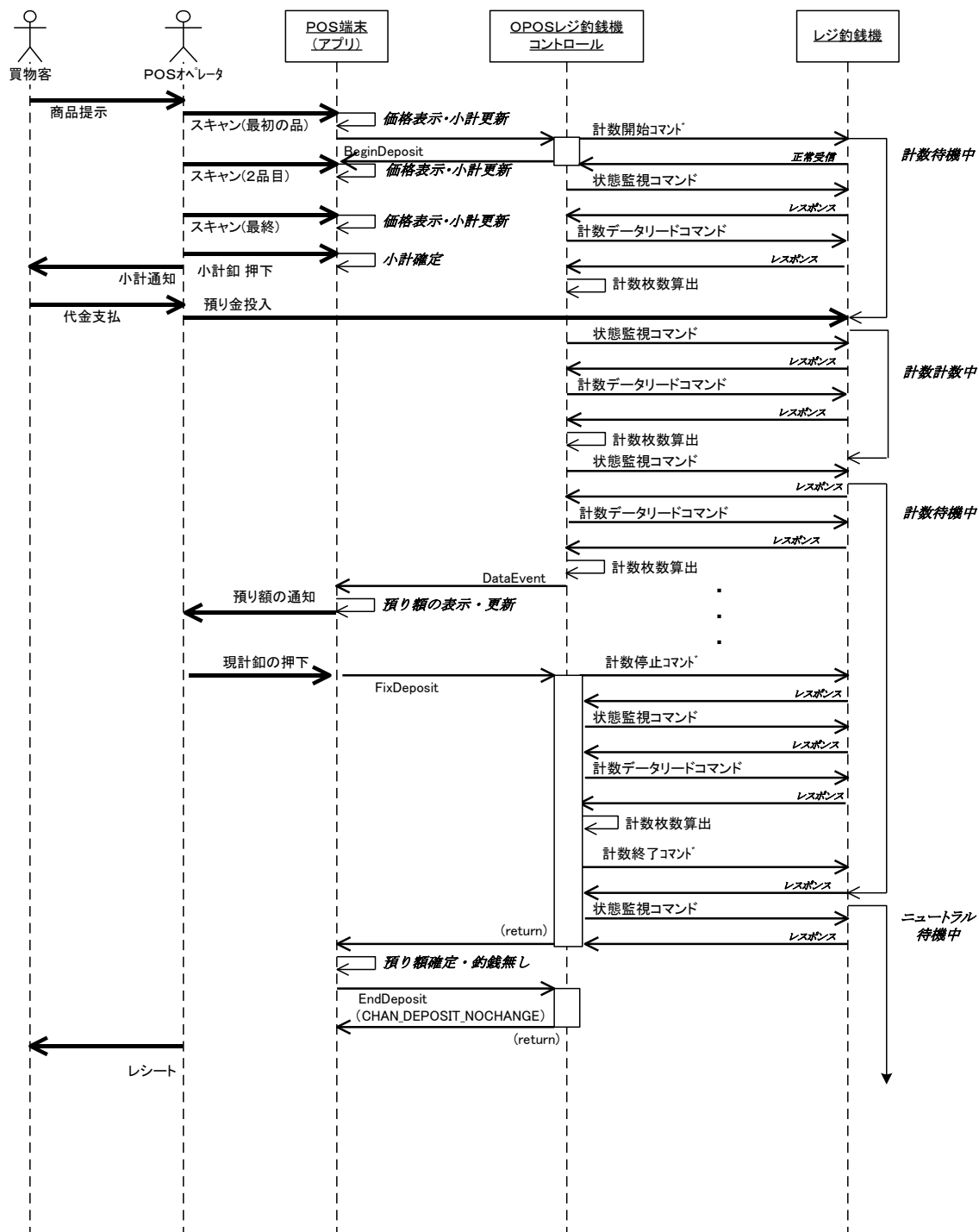
6.3.1. 売上登録（釣銭あり）

売上登録(釣銭あり)



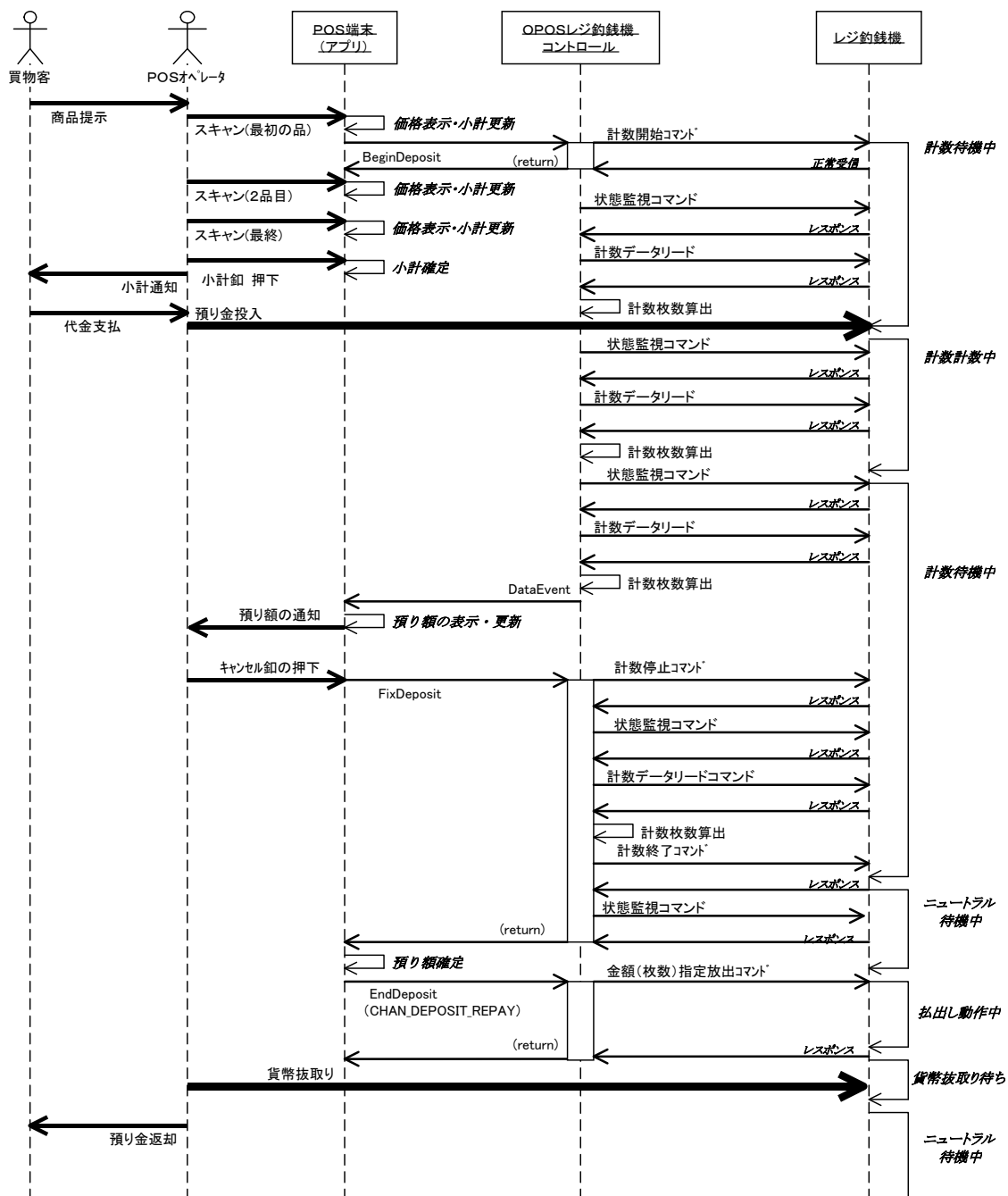
6.3.2. 売上登録（釣銭なし）

売上登録（釣銭なし）



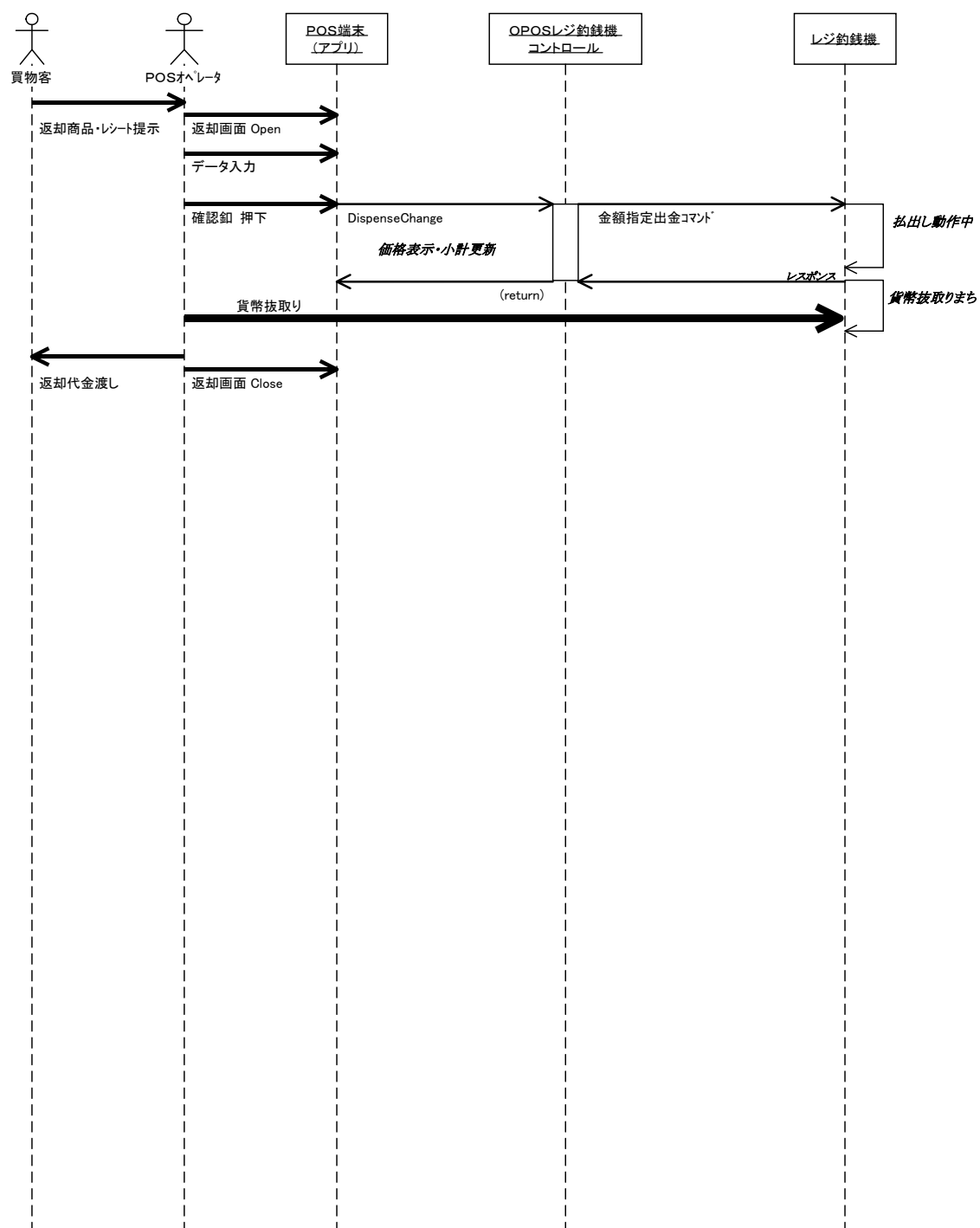
6.3.3. 売上登録（取引中止・預り金返却）

売上登録(取引中止 預り金返却)



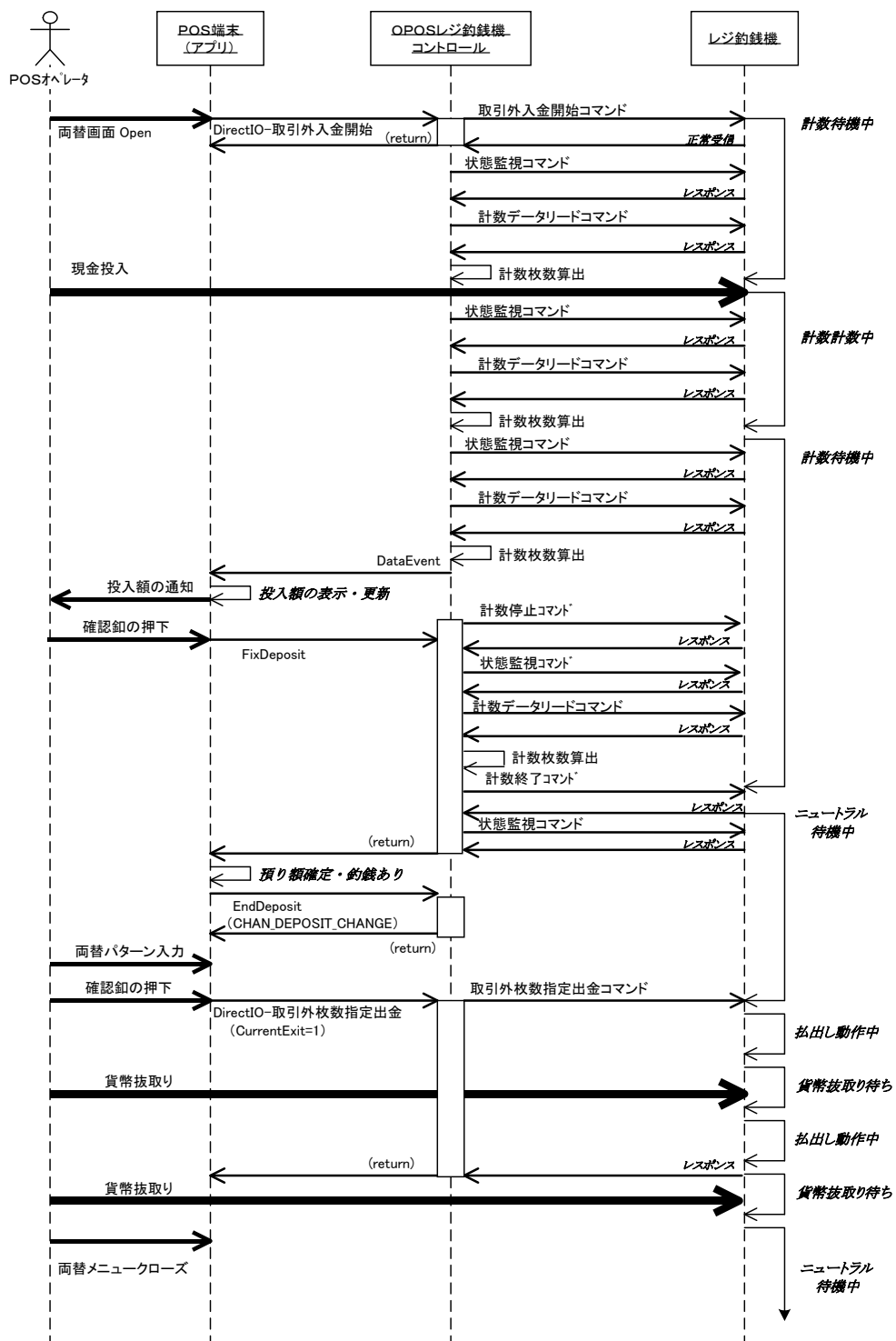
6.3.4. 返品処理

返品処理



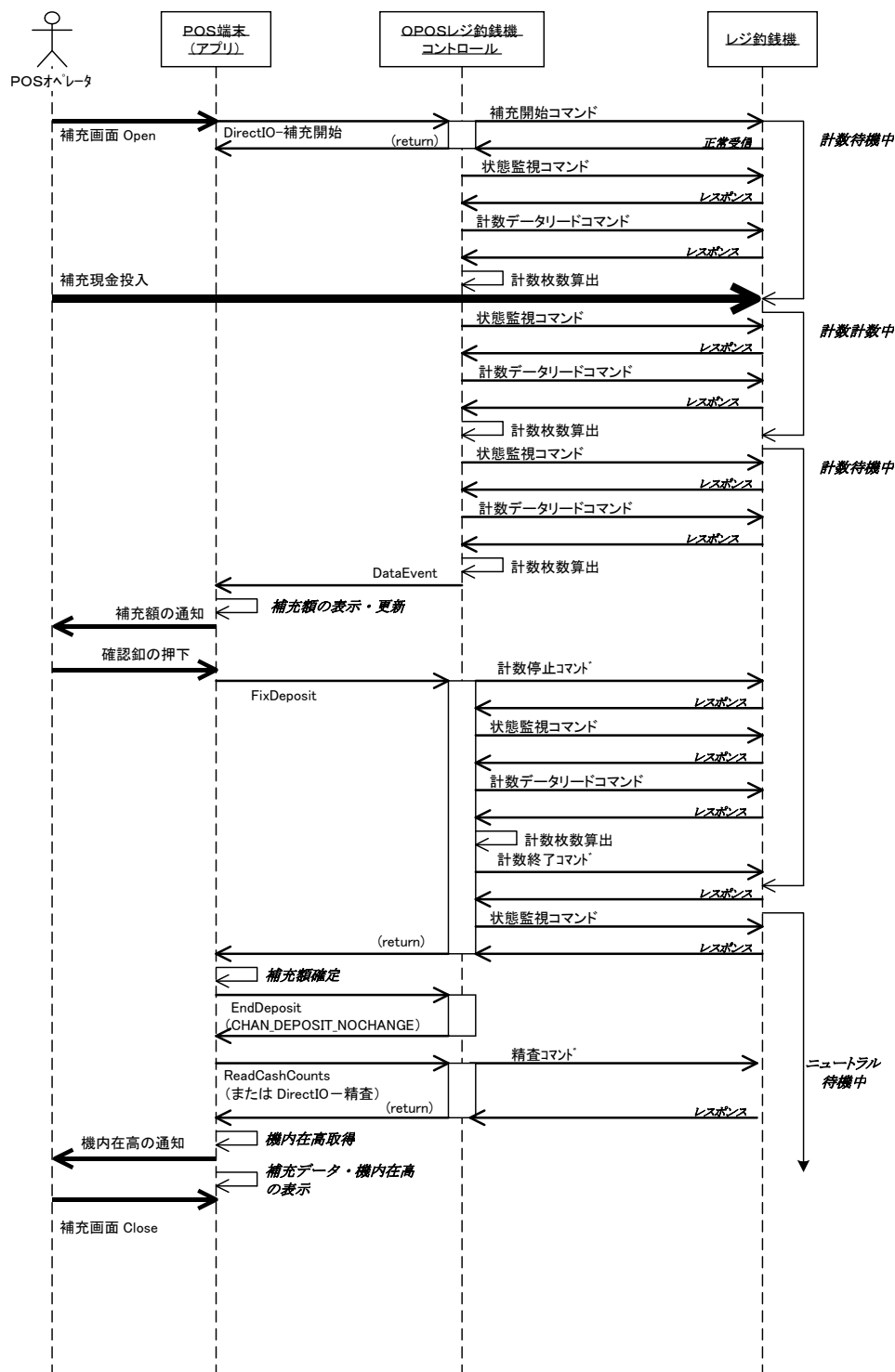
6.3.5. 両替処理

両替処理



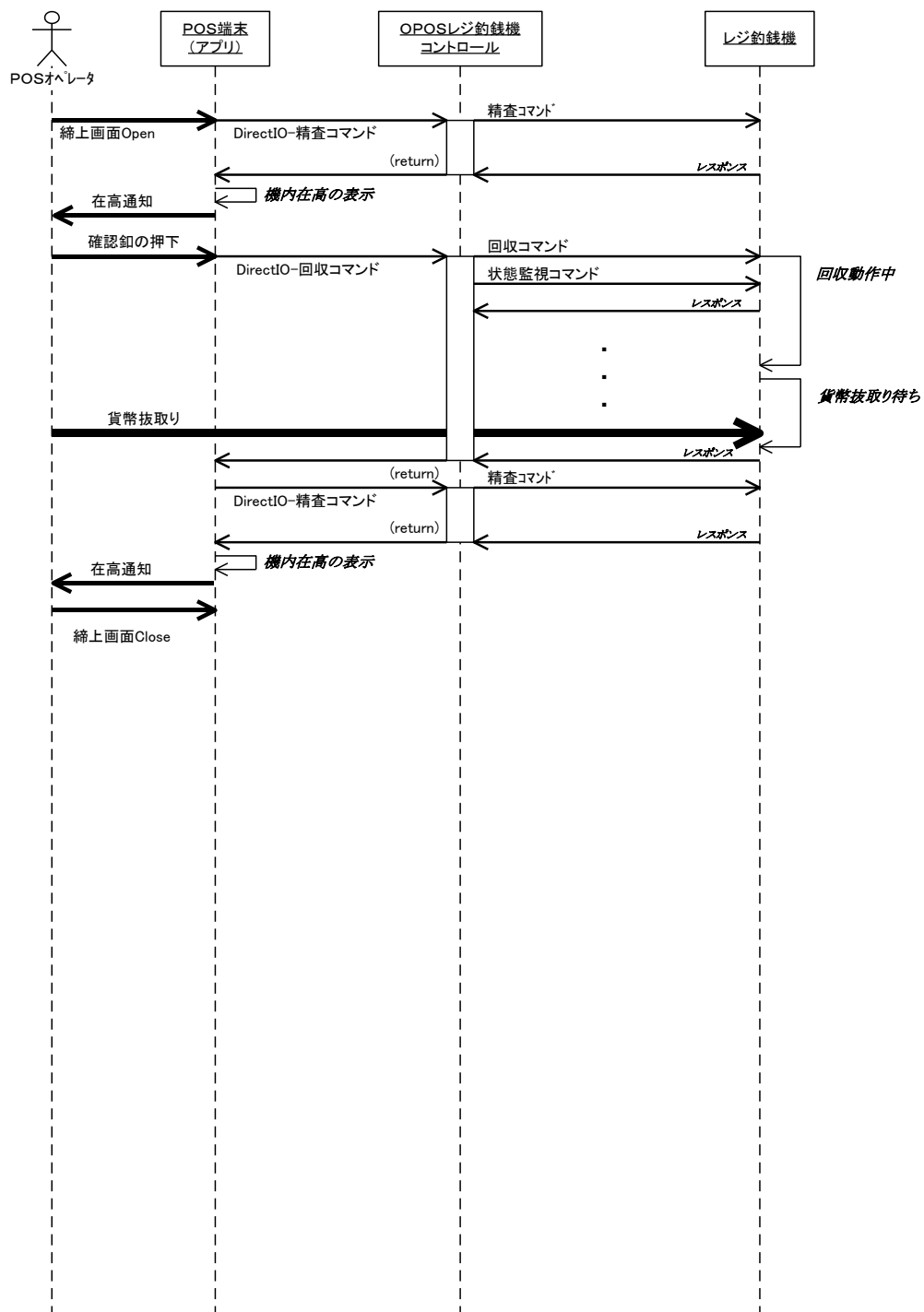
6.3.6. 釣銭補充処理

釣銭補充処理

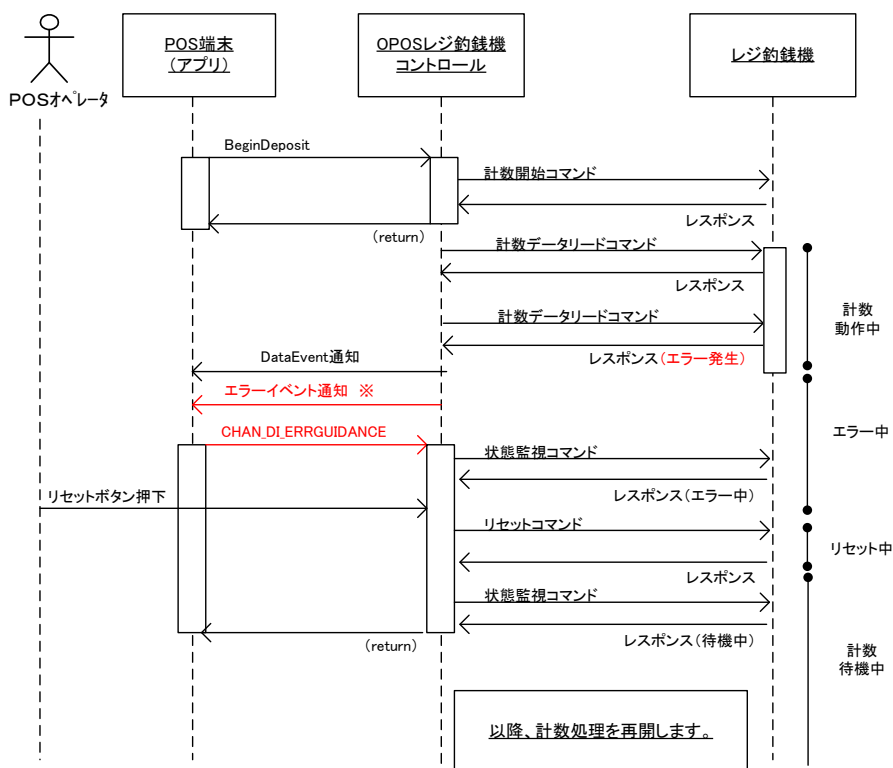


6.3.7. 閉店時

閉店時

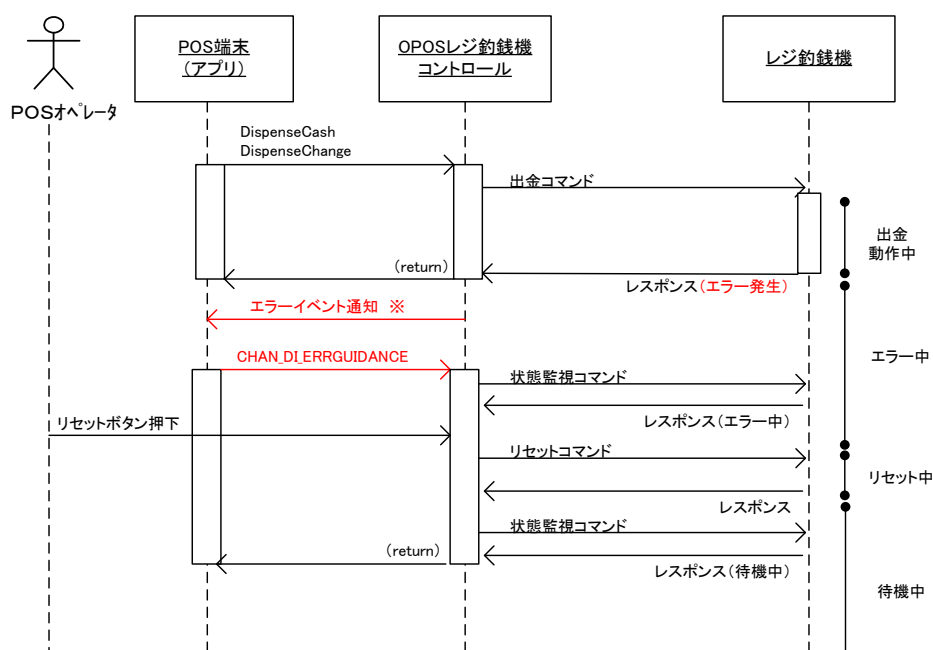


6.3.8. 入金エラー時



※レジストリ DepositErrorEvent の設定値により、通知するイベントを DirectIO イベント (CHAN_DIEVT_DEPOSITERROR、CHAN_DIEVT_DEPOSITSETERROR) または StatusUpdate イベント (CHAN_STATUS_JAM) に切り替えることができます。(デフォルト: "DIEVT" DirectIO イベントで通知する)

6.3.9. 出金エラー時



※ 通知されるイベントは StatusUpdate イベント (CHAN_STATUS_JAM) になります。

エラー解除後は出金処理を継続しないので、運用方法に合わせて再出金やドロア運用等を行ってください。

6. 4. 特記事項

◆ DepositAmount プロパティ、DepositCashCounts プロパティの更新タイミングについて

本コントロールソフトは、処理機に対し一定間隔で行う通信結果をリアルタイムに各プロパティ反映します。よって、計数処理中は常時値の更新を行っています。

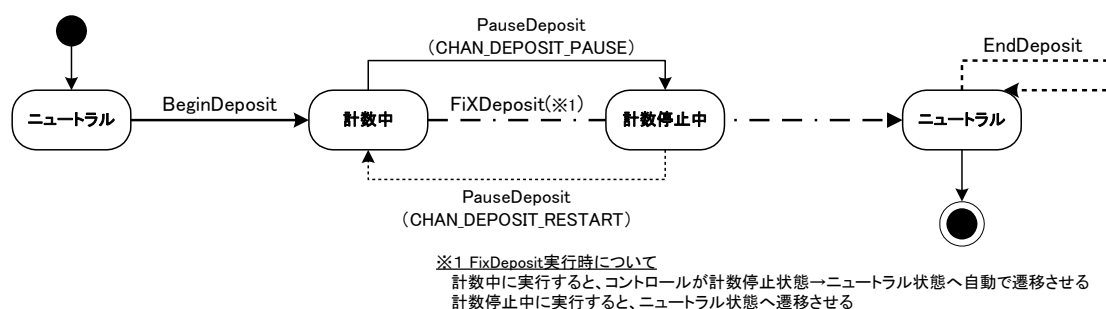
※ DataEvent イベントは、処理機の計数処理が終了したタイミングを通知します。
このため、計数終了後の値が必要な場合は、DataEvent イベントを受けた後に各プロパティを参照してください。

◆ 計数処理に関して

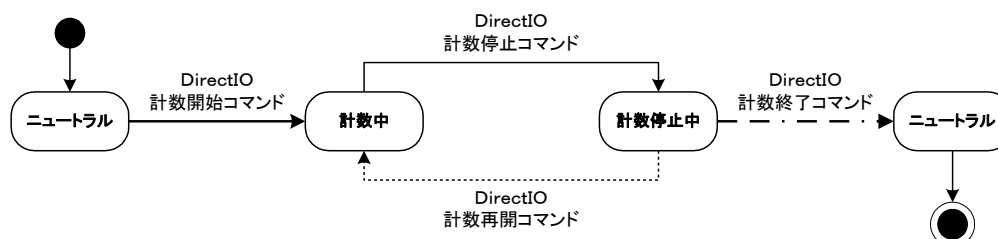
計数処理に関するつり銭機の状態は下記のとおりです。

No.	処理機の計数状態	説 明
1	ニュートラルレディ	・ 待機中状態 ・ 出金、回収などの処理を実行する事が可能 ・ 投入口へ入れた紙幣・硬貨は計数しない
2	計数中	・ 投入口の紙幣・硬貨を計数する。また計数終了後に投入口への紙幣・硬貨の再投入で計数を再開する。 ・ 出金、回収などの処理を実行する事は不可能
3	計数停止中	・ 投入口へ入れた紙幣・硬貨は計数しない ・ 出金、回収などの処理を実行する事は不可能 ・ この状態で計数計数データの確定を行うことができます

・ 専用メソッドを APG で規定されたシーケンスで呼び出す際は以下の様に遷移を行います。



・ DirectIO メソッドの各コマンドを利用する場合、以下の様に遷移します。



処理機の各計数状態と実行可能なメソッドを下記に記述します。

No.	処理機の計数状態	実行可能なメソッド	遷移先
1	ニュートラルレディ (ACK/ETB)	BeginDeposit	2
		DirectI0 - 計数開始コマンド	2
		DirectI0 - 補充開始コマンド	2
		DirectI0 - 取引外入金開始コマンド	2
		DirectI0 - 現金戻し開始コマンド	2
		EndDeposit	1
2	計数中 (SOH)	FixDeposit	3→1 ※
		PauseDeposit (CHAN_DEPOSIT_PAUSE)	3
		PauseDeposit (CHAN_DEPOSIT_RESTART)	2
		DirectI0 - 計数停止コマンド	3
		DirectI0 - 計数再開コマンド	2
3	計数停止中 (EM)	FixDeposit	1
		PauseDeposit (CHAN_DEPOSIT_PAUSE)	3
		PauseDeposit (CHAN_DEPOSIT_RESTART)	2
		DirectI0 - 計数停止コマンド	3
		DirectI0 - 計数再開コマンド	2
		DirectI0 - 計数終了コマンド	1

※ コントロールが自動的に計数停止中 → ニュートラル状態へ遷移を行います。

6. 5. バージョン履歴

Version	内容
1.0	・ 初版リリース
2.0	・ 取引外出金時の抜き取り待ちイベント通知不具合対応
3.0	・ 出金指示で直前の入金枚数が返却されない不具合対応
4.0	・ Windows8 (32bit 版) サポート ・ Windows8.1 (64bit 版) サポート
5.0	・ RT-300 単体接続のサポート
6.0	・ 入金した金額よりも少ない金額で確定される不具合対応 (レジストリ FixDepositReturnTiming が “Async” or “AsncStd” いずれかの設定時のみ)

[illegible]

[illegible]

ISP-902E

アプリケーション 開発ガイド

初 版 日 2013 年 04 月 05 日

版 数 第 5 版

発 行 元 グローリー株式会社

●アフターサービス

- * 全国にまたがったグローリーのサービス網は、サービスに万全を期しております。
- * この製品の保守、操作についてのお問い合わせは、最寄りの当社販売店または当社指定の取扱店にご連絡ください。

2015. 1. 15

