

EtherCAT モーションスレーブ

# EtherCATシリーズ

## C400 INtime ライブラリマニユアル

INtime およびRSI-ECAT-Master 使用版

---

本マニュアル及びプログラムの全部又は一部の無断転載、コピーを禁止します。  
本製品の内容に関しましては、改良等により将来予告なしに変更することがあります。  
本製品の内容についてお気づきの点がございましたら、お手数ながら当社までご連絡ください。

EtherCAT ®は、ドイツのベッコフオートメーション株式会社がライセンスを供与した登録商標であり、特許取得済みの技術です。

INtime ®, TenAsys は TenAsys 社の登録商標です。

RSI-ECAT-Master ®、RSI-ECAT-Studio ®は株式会社マイクロネットの登録商標です。

Windows ®は Microsoft Corporation の米国及びその他の国における登録商標です。

その他、記載されている会社名、製品名は、各社の商標又は登録商標です。

株式会社 ハイバ - テック  
東京都江東区新大橋 1-8-11  
大樹生命新大橋ビル  
TEL 03-3846-3801  
FAX 03-3846-3773  
sales@hivertec.co.jp  
不許複製・転載

# 目次

<b>第 1 章</b>	<b>はじめに</b>	<b>4</b>
1.1	動作環境の準備 . . . . .	4
1.2	コンフィグレーションについて . . . . .	4
<b>第 2 章</b>	<b>ライブラリ使用準備</b>	<b>5</b>
2.1	ライブラリファイルの配置 . . . . .	5
2.2	Visual Studio へのリンクとインクルード . . . . .	5
<b>第 3 章</b>	<b>ライブラリ使用方法</b>	<b>6</b>
3.1	制御処理のフローについて . . . . .	6
3.1.1	処理 1 EtherCAT 通信開始~HES-C400 を Operational にするまで . . . . .	8
3.1.2	処理 2 HES-C400 の制御を行う . . . . .	9
3.1.3	処理 3 EtherCAT 通信の終了 . . . . .	10
3.2	ライブラリ使用上の注意事項 . . . . .	10
<b>第 4 章</b>	<b>API</b>	<b>11</b>
<b>第 5 章</b>	<b>サンプルプログラム</b>	<b>13</b>
5.1	サンプル概要 . . . . .	13

# 第1章

## はじめに

本書は、弊社 EtherCAT スレーブ製品 HES-C400 を INtime 上で制御できるライブラリの使用方法を記載したマニュアルです。

EtherCAT マスタとして、RSI-ECAT-Master(マイクロネット社製)を使用します。

本マニュアルは、C400 ユーザーズマニュアル<運用編>および<ソフトウェア編>と併せてご使用ください。

### 1.1 動作環境の準備

HES-C400 を INtime 上で本ライブラリを用いて制御する場合、INtime および EtherCAT マスタのインストールが必要となります。また、コンフィグレーションをするためのツールも必要となります。

表 1.1 は、制御するためにインストールが必要なものになります。これらがインストールされているか、また設定は正しいかを事前にご確認ください。インストール方法などは、各種製品のメーカー HP をご確認ください。

表 1.1 インストールソフトウェア一覧

No.	ソフトウェア	備考
1	リアルタイム OS INtime	サイクリック周期: 1000 $\mu$ s(弊社で動作確認した周期)
2	EtherCAT マスタ RSI-ECAT-Master	DC 機能無しでも使用可
3	Configuration ツール RSI-ECAT-Studio	

表 1.2 は、本ライブラリの推奨動作環境になります。

表 1.2 動作環境一覧

No.	ソフトウェア	バージョン等
1	OS	Windows10 64bit
2	RSI-ECAT-Master	ver.3.0.9
3	RSI-ECAT-Studio	ver.1.12.349.0
4	INtime	ver.6.4.20278.1
5	Network Interface Card	I210-T1 相当
6	開発ツール	VisualStudio2019

### 1.2 コンフィグレーションについて

INtime で RSI-ECAT-Master を使用する場合、RSI-ECAT-Studio で作成した ENI ファイルが必要となります。本ライブラリを使用するにあたって、ENI ファイル作成時には、以下の点について注意して作成してください。

- HES-C400 の ENI ファイルは、添付ファイルにある最新版をご使用ください。
- PDO マッピングは編集せずにデフォルト設定のままお使いください。デフォルト設定でない場合の動作保証はできません。

## 第2章

# ライブラリ使用準備

### 2.1 ライブラリファイルの配置

C400 INtime ライブラリを使用するために、添付フォルダ内のライブラリファイルを PC 内に配置します。

表 2.1 は、PC 内に配置するファイルとその場所の一覧です。

INtime インストールディレクトリパスは、ユーザが INtime をインストールした場所に依存するためご注意ください。

表 2.1 PC 内に配置するファイルとその場所の一覧

No.	ファイル名	場所
1	c400intime.rsl	(INtime インストールディレクトリパス)\bin デフォルトパスでインストールした場合、 C:\Program Files (x86)\INtime\bin

### 2.2 Visual Studio へのリンクとインクルード

C400 INtime ライブラリを使用するために、Visual Studio の INtime プロジェクトにファイルをリンクします。

表 2.1 は、リンク・インクルードするファイル一覧です。

表 2.2 Visual Studio の INtime プロジェクトにリンクするファイル一覧

No.	ファイル名	説明
1	c400intime.lib	プロジェクトにリンクさせます。
2	c400intime.h	プロジェクトにインクルードさせます。

## 第3章

# ライブラリ使用方法

本ライブラリを使用する上での必要な処理手順や注意点について説明します。

INtime で HES-C400 の制御を行うためには、本ライブラリ以外に RSI-ECAT-Master の API を使用する必要があります。そのため、それらによる制御方法についても記述していますが、マイクロネット様の技術 HP で詳しい API の使用方法等をあらかじめご確認いただき、ご理解の上でご使用ください。

### 3.1 制御処理のフローについて

HES-C400 の基本的な制御フローは以下ようになります。この処理フローについて、それぞれ解説します。

## C400の基本的な制御フロー

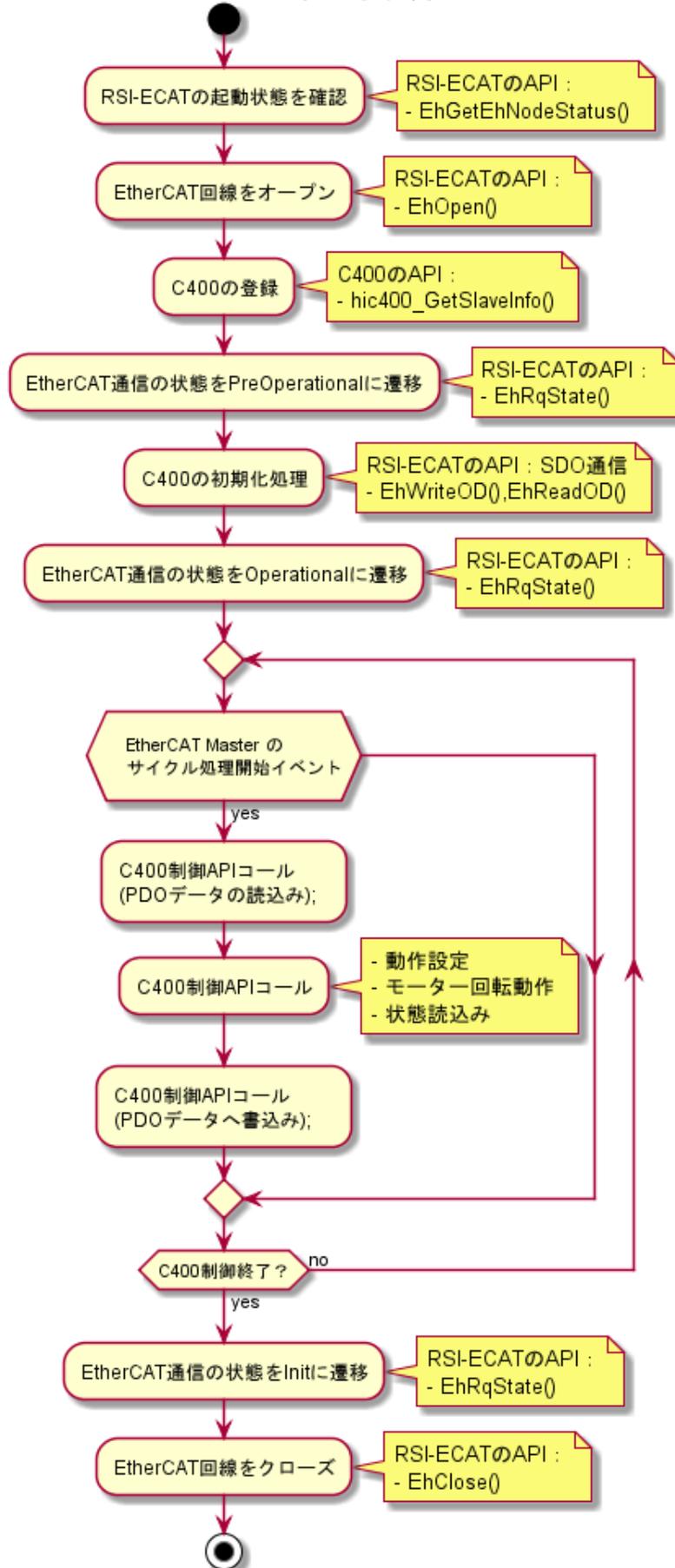


図 3.1 HES-C400 INtime ライブラリ基本処理フロー図

### 3.1.1 処理 1 EtherCAT 通信開始~HES-C400 を Operational にするまで

#### EtherCAT 回線のオープン

最初に RSI-ECAT-Master の API を使用して、EtherCAT 回線をオープンします。

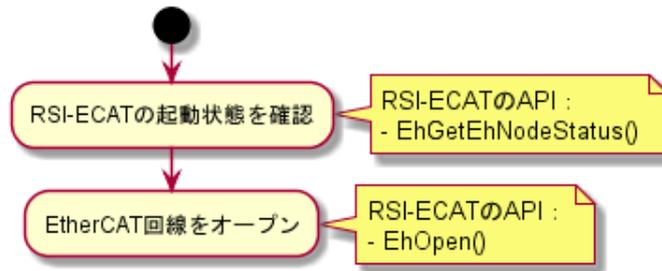


図 3.2 EtherCAT 回線のオープン

#### HES-C400 の登録

オープンに成功したら、ライブラリの API をコールして制御したい HES-C400 を登録します。

本ライブラリで提供されているスレーブ情報構造体 (SlaveRef) が、HES-C400 一つにつき 1 個必要になります。API の使用方法は API リファレンスまたはサンプルをご確認ください。

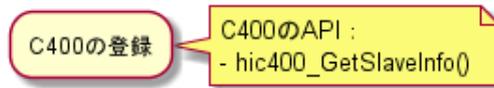


図 3.3 HES-C400 の登録

#### スレーブを Operational に遷移

登録後、スレーブを Operational に遷移させます。PreOperational になった段階でスレーブへの書き込みができるようになるため、HES-C400 の初期化設定ができます。ただし、ここでは RSI-ECAT-Master の API による SDO 通信でのみ設定が可能となります。

初期化は、SDO 通信を使わずに、本ライブラリ API によるレジスタ・ポート書き込みで設定することができます。この方法はサイクリック周期中の処理になるため、後述の「HES-C400 の制御を行う」をご参照ください。

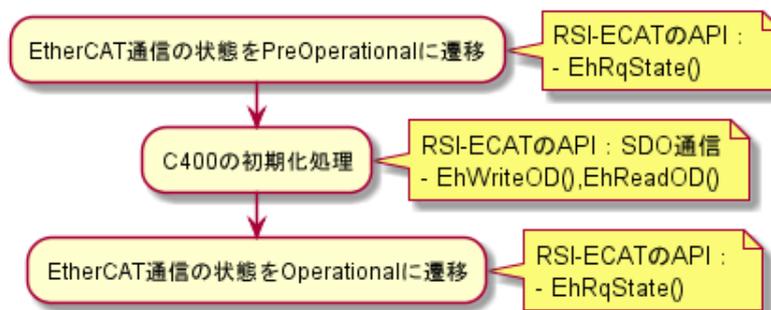


図 3.4 スレーブを Operational に遷移

### 3.1.2 処理 2 HES-C400 の制御を行う

1. RSI-ECAT-Master による EtherCAT 通信では、まずサイクリック周期のイベント待ちをします。
2. イベント発生後、PDO データの読み込みを行います。hic400\_rPdo をコールしてください。
3. 本ライブラリの API をコールします。直前で読み込んだ PDO の情報を基に処理されます。
4. 最後に PDO データの書き込みを行います。hic400\_wPdo をコールしてください。手順 3 で処理されたデータが EtherCAT 通信のパケットに書き込まれます。

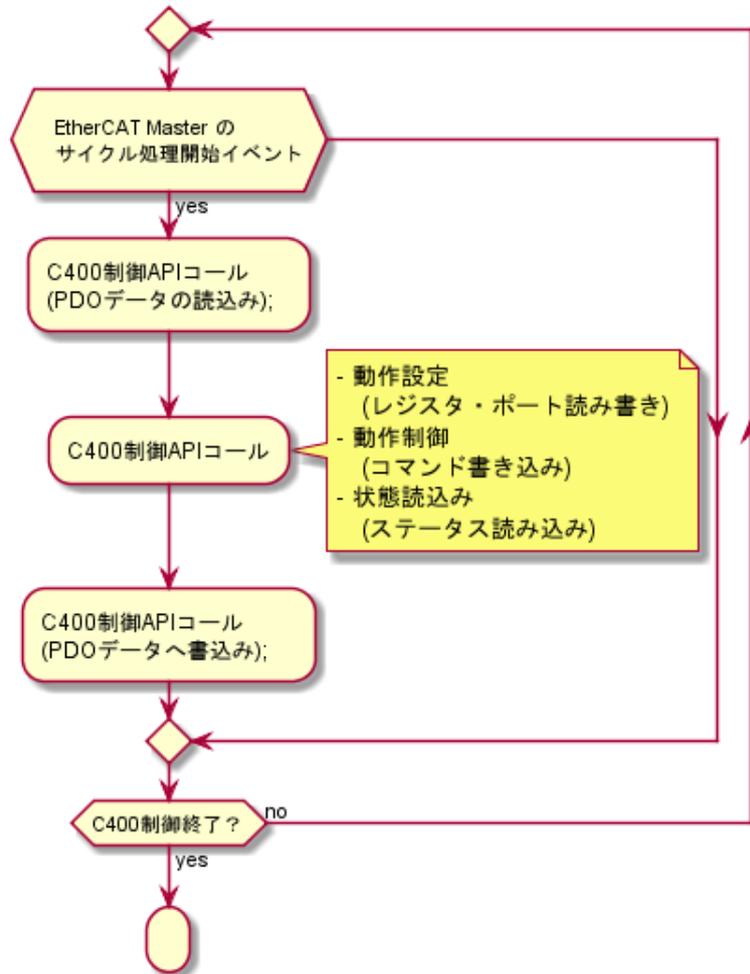


図 3.5 HES-C400 の制御を行う

この処理を 1 周期内に実行します。EtherCAT 通信では、スレーブに送りたい情報を PDO データに書き込みますが、1 周期でスレーブがそのデータを取得できるとは限らないため、ループ処理をして絶えず読み書きします。本ライブラリ API は、次のようにループ処理をします。

1. コールが成功すると API は「Busy 中」となり、スレーブ情報構造体のカレント API の状態が Busy になります。PDO データに HES-C400 に送るコマンドデータを載せた状態になります。
2. 毎周期同じ API をコールして、HES-C400 にコマンドが書き込まれたかを、API の状態から確認します。
3. HES-C400 への書き込みが成功すると、スレーブ情報構造体にあるカレント API の状態が Done になり実行完了になります。それを見て次の API を実行してください。
4. API の状態が Busy のままで終わらない場合、エラーハンドリングしてください。HES-C400 に書き込めない状態が発生しているため、EtherCAT 通信を終了し、電源断して再起動します。
5. API の状態が Error となった場合、コマンド書き込みが失敗しています。書き込むレジスタ・ポートのアドレスやデータの内容をご確認ください。ただし、この状態であっても次の API の実行は可能です。

### 第3章 ライブラリ使用方法

1 コマンド送信エラーは、エラークリアする必要はありません。次のコマンドが正常に書き込まれると自動的にクリアされます。

2 戻り値が 0 以外るとき、エラーが発生しています。エラー内容を確認し、エラーの解消を行ってください。

3 コマンド書き込みの応答時間の目安は、弊社の動作確認では 30 周期以内で Done になることを確認しています。ただし、動作環境によって異なりますのであらかじめご確認の上動作させるようにしてください。

4 API の状態が変化しない API 関数があります。詳細は API リファレンスをご確認ください。

#### 3.1.3 処理 3 EtherCAT 通信の終了

動作制御を終了し EtherCAT 通信の終了をするときは、RSI-ECAT-Master の API を使用します。

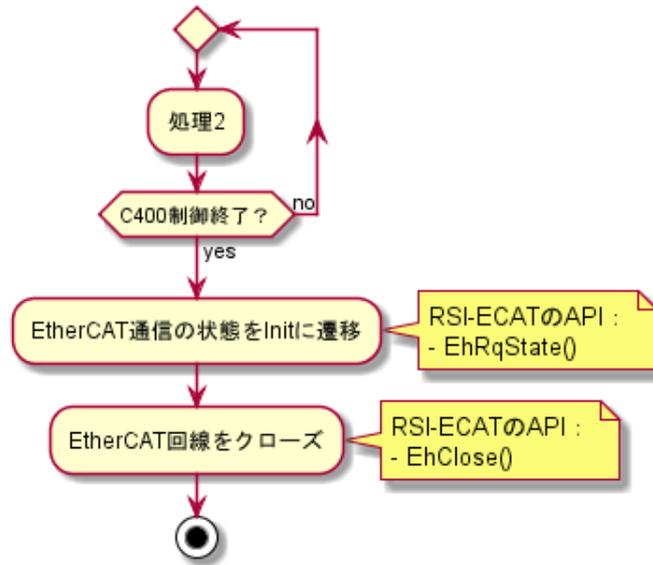


図 3.6 EtherCAT 通信の終了

### 3.2 ライブラリ使用上の注意事項

- HES-C400 の PDO マッピングは、ESI ファイルを読み込んだ時のデフォルトのままお使いください。
- ライブラリ API は排他制御されていません。複数の HES-C400 を使用する、あるいは並列処理を行うときは排他制御をするようにしてください。
- ライブラリ API が実行されてスレーブ情報構造体の API 状態が Busy となったとき、他の API をコールしても実行されません。現在 Busy となっている API が終わって (Done もしくは Error となって) から次の処理をしてください。

## 第 4 章

# API

本ライブラリの API 一覧です。API 一覧の詳細は、API リファレンスをご確認ください。

API には、複数軸指定が可能な API と、単軸指定のみ動作できる API があります。複数軸指定可能な API は、コマンド書き込みを行うカテゴリが動作制御指令の API になります。それ以外の API で複数軸指定してコールするとパラメータエラーとなります。ご注意ください。

表 4.1 API 一覧

No.	関数名	説明	備考
1	hic400_AccStart	加速スタート	複数軸指定可
2	hic400_CalAccRate	加減速レートの計算	複数軸指定可
3	hic400_CnstStartFH	FH 定速スタート	複数軸指定可
4	hic400_DecStop	減速停止	複数軸指定可
5	hic400_EmgStop	非常停止	複数軸指定可
6	hic400_ReadCtr	カウンタの読出し	-
7	hic400_ReadErrorSts	エラーステータスの読出し	-
8	hic400_ReadEventSts	イベントステータスの読出し	-
9	hic400_ReadMainSts	メインステータスの読出し	-
10	hic400_ReadSysErr	スレーブエラーステータスの読み込み	-
11	hic400_ReadSpd	指令速度の読出し	-
12	hic400_ReadSubSts	サブステータスの読出し	-
13	hic400_rReg	レジスタ読出し	-
14	hic400_SetAccRate	加速レートの設定	-
15	hic400_SetAuxSpd	補助速度の設定	-
16	hic400_SetCmdPulse	指令パルスの出力形式の設定	-
17	hic400_SetDecRate	減速レートの設定	-
18	hic400_SetEls	ELS の設定	-
19	hic400_SetEventMask	イベントマスクの設定	-
20	hic400_SetFLSpd	ベース速度の設定	-
21	hic400_SetMult	倍率設定値 (MG) の設定	-
22	hic400_SetOls	OLS の設定	-
23	hic400_SetOrgMode	原点復帰モードの設定	-
24	hic400_SetSls	ソフトリミットの設定	-
25	hic400_SetSvAlm	SVALM の設定	-
26	hic400_wCmdW	制御コマンド書込み	複数軸指定可
27	hic400_wReg	レジスタ書込み	-
28	hic400_WritCtr	カウンタプリセット	-
29	hic400_WritFHSpd	動作速度の設定	-
30	hic400_WritOpeMode	動作モードの設定	-
31	hic400_WritPos	位置決め移動量の設定	-
32	hic400_GetSlaveInfo	スレーブ情報の取得	-
33	hic400_rPdo	PDO データ (IN 側 (C400 RSI-ECAT) ) の読込み	-
34	hic400_wPdo	PDO データ (Out 側 (RSI-ECAT C400) ) の書込み	-
35	hic400_SysErrClr	エラークリア	-
36	hic400_rPortW	オプションポート読出し	-
37	hic400_wPortW	オプションポート書込み	-

## 第 5 章

# サンプルプログラム

sample フォルダにはサンプルが添付されています。こちらのサンプルは、接続した HES-C400 1 台 (インスタンス番号 0) について、ポート書き込み・レジスタ初期化を行ったうえで動作実行をするプログラムになっています。



### サンプルソフトウェアの責任範囲

本サンプルプログラムの動作不良、エラー若しくは他の不具合が生じた場合、いかなる保証も致しません。責任も負いません。また、本ソフトウェアを使用した結果又は本ソースコードを利用した結果について一切責任を負いません。

位置決め Start 時の動作速度は、ソースコード内で固定の値が入力されています。あらかじめソースコード内の速度をご確認ください。

## 5.1 サンプル概要

下記のモーション制御を可能とするサンプルプログラムです。

- 1 軸の位置決め制御
- 2 軸の円弧補間
- JOG 動作
- 原点復帰
- 停止動作

[サンプルの仕様について]

- サンプルプログラムを起動すると、自動的に EtherCAT 通信の状態が Operational まで遷移します。Operational になった後、スレーブの初期化を行います。
- 初期化後は、コンソール画面にコマンドを入力することでモーション制御や軸のステータス表示ができます。
- モーション機能を実行しているときは、停止動作・ステータス表示以外のモーション機能は受け付けません。
- 本サンプルプログラムでは、コマンドを受け付けるメインスレッドと、モーション制御を行うモーションスレッドのマルチスレッドになっています。ただし、優先度はモーション制御の方を高くしています。

図 5.1 は本サンプルのフロー図です。ソースコードと併せてお読みください。

アプリケーション処理フロー

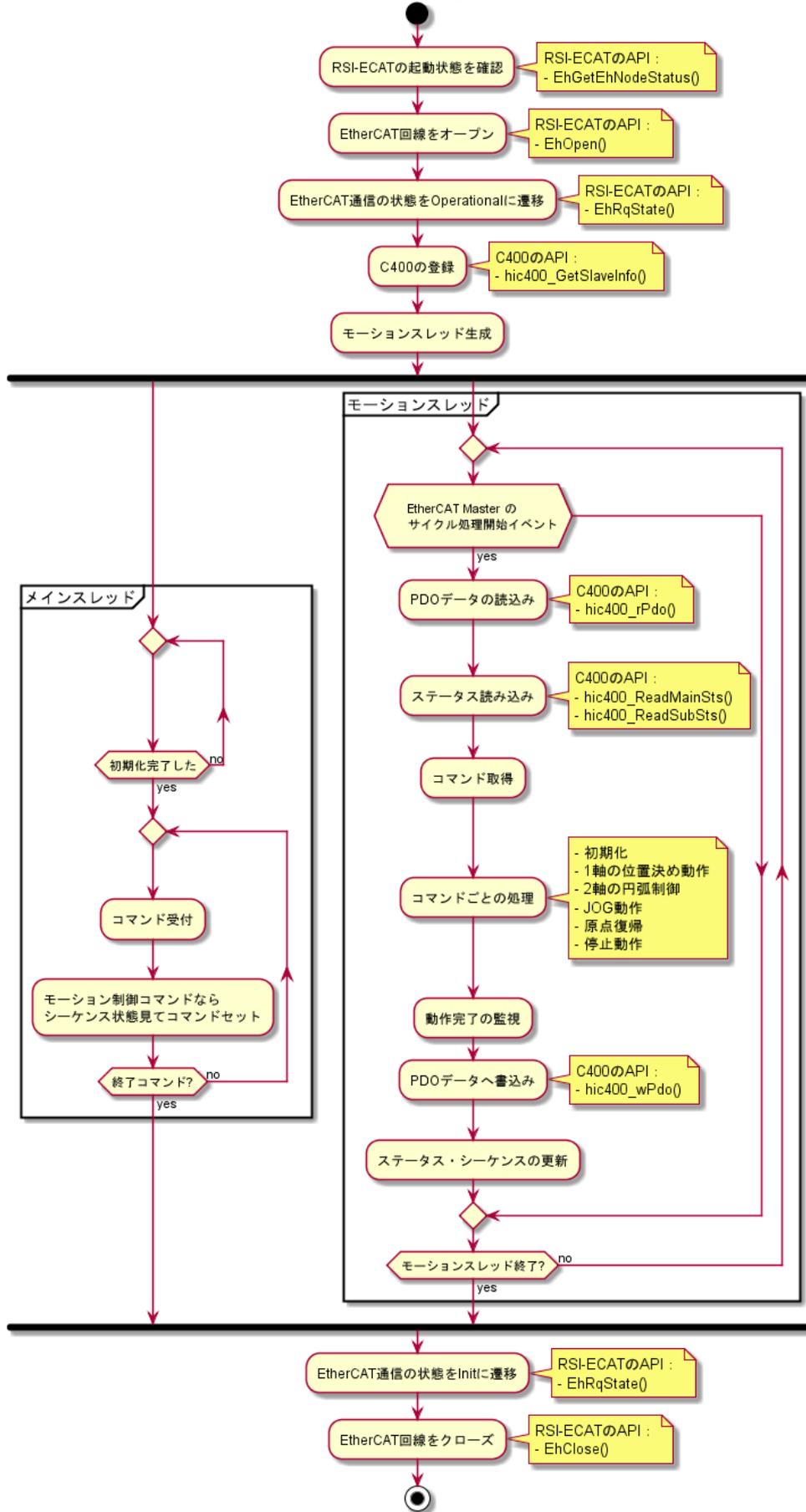


図 5.1 サンプルプログラムのフロー図



---

2022年4月25日 C400 INtime 対応ライブラリ版新規作成

発行所 株式会社ハイパーテック

連絡先 株式会社 ハイパ - テック、東京都江東区新大橋 1-8-11 大樹生命新大橋ビル、TEL 03-3846-3801、FAX  
03-3846-3773、sales@hivertec.co.jp

---

(C) Hivertec, Inc.