

INtime/EtherCAT/テクノ製品 のモーションソリューション

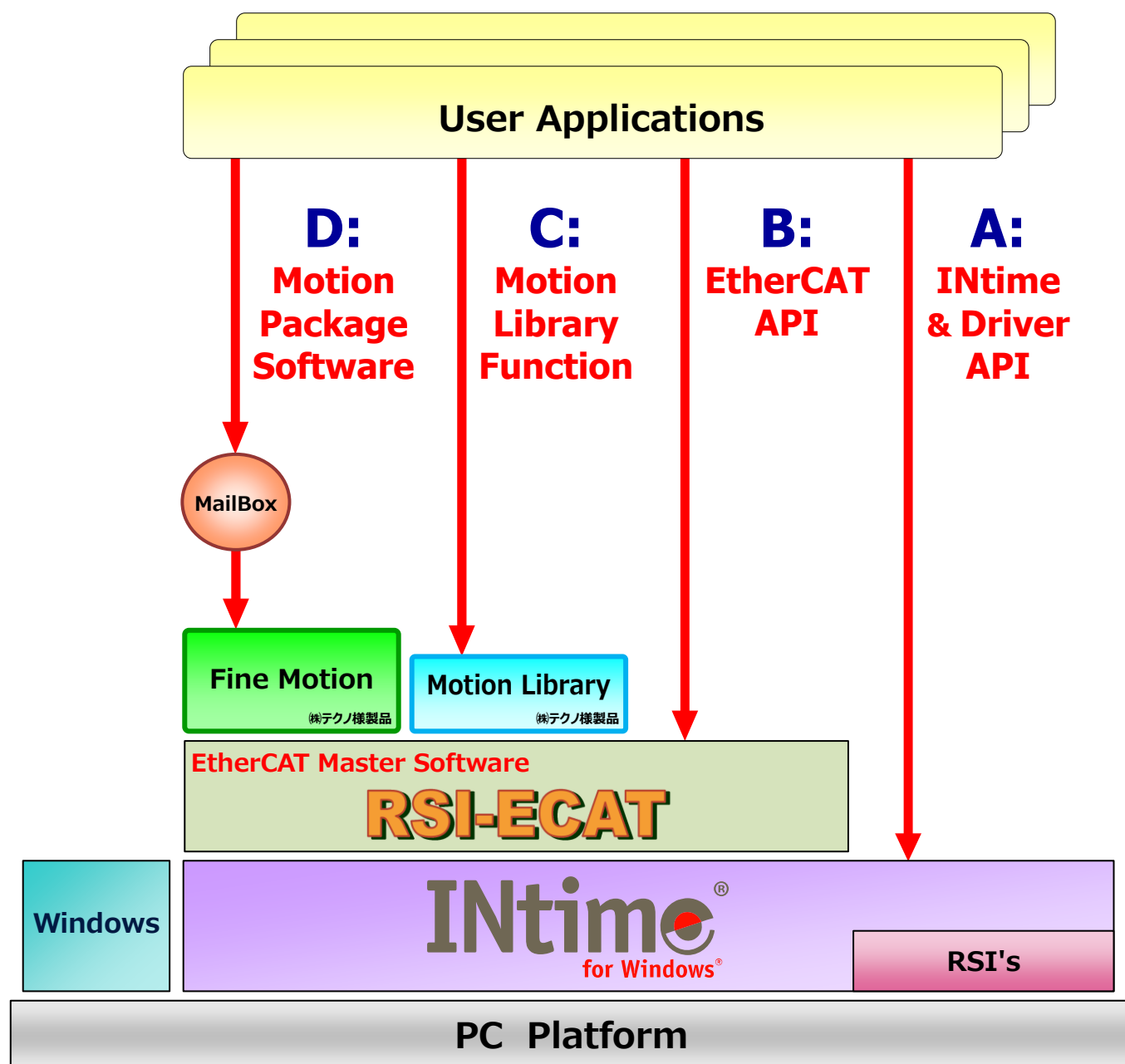


図 1 : INtime/EtherCAT/テクノ製品のモーションソリューション

A : INtime & Driver API

INtime は、PC を産業用制御に利用するためにリアルタイム性と信頼性を提供するリアルタイム OS です。

Window® operating system と共存/協調して動作し、リアルタイムアプリケーションも Microsoft® Visual Studio で開発が可能です。リアルタイムアプリケーション開発のために、リアルタイムアプリケーション用 API (※表 2) と、ドライバ (※表 3) が用意されています。

表 1 : INtime 仕様

プライオリティスケジューリング	256 段階
カーネルティック	50us, 100us, 125us, 200us, 250us, 500us, 1ms, 2ms, 5ms, 10ms
メールボックス	FIFO 式/優先度式
セマフォ	FIFO 式/優先度式
仮想セグメントサイズ	システムトータルで 4GB
割り込み	ハンドラ/スレッド構造 IRQ/MSI 対応
開発言語	C/C++
標準 I/O	TCP/IP, TCP/IP, USB, RS232C
最大オブジェクト数	8190 (システム含む)

表 2 : INtime API (主要抜粋)

割り込み管理
SignalEndOfRtInter, SetRtInterruptHandler, ResetRtInterruptHandler, SignalRtInterruptThread, WaitForRtInterrupt, DisableRtInterrupt, EnableRtInterrupt
メールボックス管理
CreateRtMailbox, DeleteRtMailbox, SendRtHandle, ReceiveRtHandle, SendRtData, ReceiveRtData
メモリ管理
AllocateRtMemory, FreeRtMemory, CreateRtMemoryHandle, DeleteRtMemoryHandle, MapRtShareMemory, MapRtPhysicalMemory, GetRtPhysicalAddress, GetRtSize
オブジェクト管理
CatalogRtHandle, LookupRtHandle, UncatalogRtHandle, GetRtHandleType, GetRtHandleTypeEx, InspectRtProcessDirectory
プロセス管理
CreateRtProcess, ExitRtProcess, RtNotifyEvent, RegisterRtDependency, UnregisterRtDependency, RegisterRtSponsor, UnregisterRtSponsor
スケジューラ管理
knRtSleep, knStartRtScheduler, knStopRtScheduler
セマフォ管理
CreateRtSemaphore, DeleteRtSemaphore, WaitForReSemaphore, ReleaseRtSemaphore
ステータス管理
GetLastRtError, SetLastRtError, CopyRtSystemInfo, ReportRtEvent
スレッド管理
CreateRtThread, DeleteRtThread, RtSleep, GetRtThreadPriority, GetRtThreadHandles, SetRtThreadPriority, SetRtProcessMaxPriority, GetRtThreadAccounting, SuspendRtThread, ResumeRtThread, GetRtThreadInfo
システムデータ管理
ntxGetLocationByName, ntxGetFirstLocation, ntxGetNextLocation, ntxGetNameOfLocation

表 3 : INtime ドライバ製品

シリアル通信バス
RSI-001, RSI-008, RSI-016, RSI-006, 標準シリアルコントローラドライバ
デジタル通信バス
RSI-002, 標準 USB ドライバ
A/D 変換ボード
AD12-16U(PCI)EH, AD16-16(LPCI)L ライブラリ
画像認識
AWP-332-04, AUP-SEARCH, AZP-7101-02
モーションコントロール
RSI-612, RSI-534, RSI-578, RSI-5212, RSI-708, RSI-SMC4DF, RSI-SMC8DF, RSI-SMC4DL, RSI-SMC8DL
リアルタイムネットワーク (FL-net)
RSI-040
フィールドバス
RSI-CCL, RSI-CCIE, RSI-060

B : EtherCAT マスターソフトウェア「RSI-ECAT」

INtime用のEtherCATマスターソフトウェア製品として、RSI-ECAT-Masterがあります。

RSI-ECAT-Masterは、表4に示す機能を実装し、表5のAPIを使って、DIO・AIO等の基本IO制御および基本的なサーボモータの制御を行うことができます。

表 4 : RSI -ECAT Master 機能仕様

機能	RSI-ECAT-Master		
	-	DC	RED
Basic spec			
Protocol	IEC61158-2/3/4/5/6-12		
Max Slaves	65,535		
Baudrate	100Mbps		
Data transport layer	100BASE-TX		
Duplex	Full		
Max cable length slave-slave	100m		
Master cycle			
Bus cycle time(max)	100us		
Bus cycle time(default)	500us		
Priority	Configurable		
Special feature			
DC support		○	○
Cable redundancy support			○
Hot connect support			○
64bit Windows support	○	○	○
INtime multi kernel support	○	○	○
Master class specification			
Master-Class	Class B	Class A	Class A + FP

表 5 : RSI -ECAT Master API 一覧

カテゴリ	API	内容
API 対話 関連	EhGetEhNodeStatus	EcatHandlerの動作状態を確認します。
	EhOpen	API対話を開設し、他APIを使用できる状態にします。
	EhClose	API対話を閉鎖します。
マスター関連 API	EhRqState	マスターステートの変更要求を行います。
	EhGetState	現在要求中のマスターステートと現在のマスターステートを取得します。
	EhWaitForCyclic	Masterサイクリック処理の開始を待機します。
	EhGetSystemInfo	EcatHandlerの情報を取得します。
	EhRequestAsyncOutputs	サイクリック周期とは別に、非同期に出力を開始します。
	EhSetOverrunCheck	サイクリック破綻機能の設定を行います。
	EhGetOverrunCheck	サイクリック破綻機能の設定を取得します。
スレーブ関連 API	EhGetSlaveCount	コンフィグレーション(XML)定義上のスレーブ数を取得します。
	EhGetOnlineSlaveCount	接続しているスレーブ数を取得します。
	EhFindSlave	VendorID, ProductCode, Instance を持つスレーブのスレーブ詳細情報を取得します。
	EhFindSlaveByAlias	指定エイリアスを持つスレーブのスレーブ詳細情報を取得します。
	EhFindSlaveBySlaveNo	指定スレーブ接続順番番号に位置するスレーブ詳細情報を取得します。
	EhGetSlaveStatus	対象スレーブのステータス状態を取得します。
	EhGetALStatus	対象スレーブのALステータスを取得します。
	EhGetALStatusCode	対象スレーブのALステータスコードを取得します。
	EhGetDLStatus	対象スレーブのDLステータスコードを取得します。
	EhReadRegister	対象スレーブのESCLレジスタからデータを読み取ります。
	EhWriteRegister	対象スレーブのESCLレジスタへデータを書き込みます。
	EhReadEEPROM	対象スレーブのEEPROMからデータを読み取ります。
	EhWriteEEPROM	対象スレーブのEEPROMへデータを書き込みます。
	EhRecalcChecksum	対象スレーブのEEPROMのチェックサムを再計算し、チェックサムを更新します。
イベント関連 API	EhSetEventFilter	診断イベントのフィルタを設定します。
	EhGetEventFilter	設定されている診断イベントのフィルタ情報を取得します。

C : モーション制御用ソフトウェア関数群("MotionLibrary")(株テクノ製品)

MotionLibrary は、モーション制御の高度な機能を部品化したソフト関数群です。
本ソフト関数群をユーザーアプリケーションに組み込むことで、自由に独自のモーション制御を開発ができます。

MotionLibrary を使ったソフトウェア構成図 :

※この MotionLibrary は、複数のユーザーアプリケーションから呼び出すことができます。

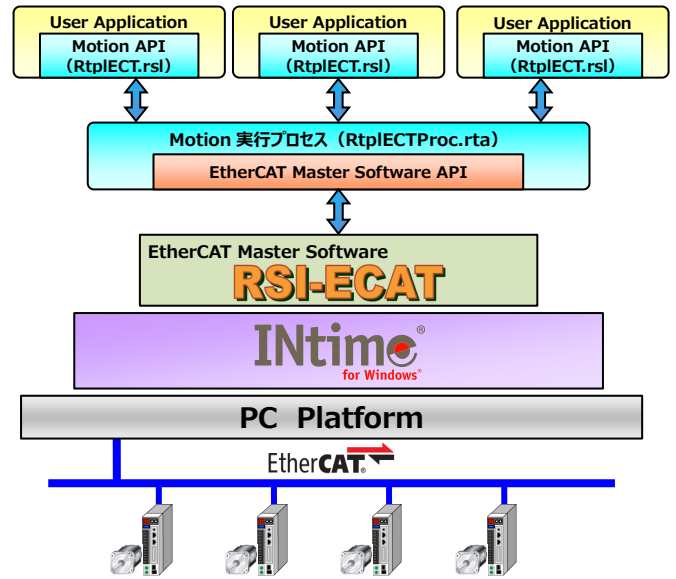


図 2 : MotionLibrary を使ったソフトウェア構成

モーションライブラリ API 一覧 :

カテゴリ	API	内容
初期化処理 API	RtplECTInitializeLib	ライブラリ初期化
	RtplECTCloseLib	ライブラリ終了
	RtplECTResetLib	ライブラリリセット
	RtplECTInitializeMaster	マスター初期化
	RtplECTCmdActive	コマンド実行
	RtplECTResWait	応答待機
	RtplECTInitializeAPIThread	スレッド毎の API リソースの初期化
RtplECTReleaseAPIThread	スレッド毎の API リソースの破棄	
EtherCAT 通信処理 API	RtplECTSetSDO	SDO 書き込み
	RtplECTGetSDO	SDO 読み込み
	RtplECTSetPDO	PDO 書き込み
	RtplECTGetPDO	PDO 読み込み
	RtplECTSetAllPDO	全 PDO 書き込み
モニタ系 API	RtplECTGetAllPDO	全 PDO 読み込み
	RtplECTGetPosition	位置モニタ
	RtplECTGetVelocity	速度モニタ
	RtplECTGetStatus	ステータスモニタ
システム設定 API	RtplECTGetTorque	トルクモニタ
	RtplECTDefinePosition	ポジションプリセット
	RtplECTServoON	サーボ ON
	RtplECTServoOFF	サーボ OFF
	RtplECTClearAlarm	アラームクリア
	RtplECTStopPositioning	移動中断
	RtplECTHoldAxis	一時停止
	RtplECTResumeAxis	一時停止解除
	RtplECTSetGantryAxis	同一指令 2 軸設定
	RtplECTResetGantryAxis	同一指令 2 軸解除
	RtplECTSetOverride	オーバーライド設定
	RtplECTClearCmdBuff	バッファコマンドクリア処理
	RtplECTClearScsError	システムエラークリア処理
移動系 API	RtplECTHomePosition	原点復帰
	RtplECTPositioning	位置決め
	RtplECTLinInterpolate	直線補間
	RtplECTLcIrInterpolate	円弧補間
	RtplECTLatchPositioning	ラッチ位置決め
	RtplECTJOGStart	JOG 開始
	RtplECTJOGStop	JOG 停止
	RtplECTTorqueCtrlStart	トルク制御開始
	RtplECTTorqueCtrlStop	トルク制御停止
その他 API	RtplECTGetVersion	バージョン取得

カテゴリ	API	内容
イベント関連 API	EhWaitForEvent	診断イベントを待機します。
DI カテゴリ API	EhDiRead	指定の DI チャンネルから 16 ビットデータを読み込みます。
	EhDiBlock	DI カテゴリの任意のオフセット位置から任意サイズ分のデータを読み込みます。
	EhDiGetChNums	DI カテゴリのチャンネル数を取得します。
DO カテゴリ API	EhDoWrite	指定の DO チャンネルへ 16 ビットデータを書き込みます。
	EhDoRead	指定の DO チャンネルから 16 ビットデータを読み込みます。
DO カテゴリ API	EhDoBlock	DO カテゴリの任意のオフセット位置から任意サイズ分のデータを書き込みます。
	EhDoGetChNums	DO カテゴリのチャンネル数を取得します。
AI カテゴリ API	EhAiRead	指定の AI チャンネルから 32 ビットデータを読み込みます。
	EhAiBlock	AI カテゴリの任意のオフセット位置から任意サイズ分のデータを読み込みます。
	EhAiSetRange	指定の AI スレーブのレンジ設定を行います。
	EhAiGetRange	指定の AI スレーブのレンジを取得します。
	EhAiGetChNums	AI カテゴリのチャンネル数を取得します。
AO カテゴリ API	EhAoWrite	指定の AO チャンネルへ 32 ビットデータを書き込みます。
	EhAoRead	指定の AO チャンネルから 32 ビットデータを読み込みます。
	EhAoBlock	AO カテゴリの任意のオフセット位置から任意サイズ分のデータを書き込みます。
	EhAoSetRange	指定の AO スレーブのレンジ設定を行います。
	EhAoGetRange	指定の AO スレーブのレンジを取得します。
	EhAoGetChNums	AO カテゴリのチャンネル数を取得します。
OD アクセス API	EhReadOD	指定スレーブの指定インデックスからデータを読み取ります。
	EhWriteOD	指定スレーブの指定インデックスヘデータを書き込みます。
	EhReadODByAlias	指定エイリアスを持つスレーブの指定インデックスからデータを読み取ります。
	EhWriteODByAlias	指定エイリアスを持つスレーブの指定インデックスヘデータを書き込みます。
VIOS アクセス API	EhGetViosInOffset	指定エイリアスを持つスレーブの VIOS IN 領域オフセット値を取得します。
	EhGetViosOutOffset	指定エイリアスを持つスレーブの VIOS OUT 領域オフセット値を取得します。
	EhReadByte	VIOS IN 領域から 8 ビットデータを読み取ります。
	EhReadWord	VIOS IN 領域から 16 ビットデータを読み取ります。
	EhReadDword	VIOS IN 領域から 32 ビットデータを読み取ります。
	EhWriteByte	VIOS OUT 領域へ 8 ビットデータを書き込みます。
	EhWriteWord	VIOS OUT 領域へ 16 ビットデータを書き込みます。
	EhWriteDword	VIOS OUT 領域へ 32 ビットデータを書き込みます。
	EhReadbackByte	VIOS OUT 領域から 8 ビットデータを読み取ります。
	EhReadbackWord	VIOS OUT 領域から 16 ビットデータを読み取ります。
	EhReadbackDword	VIOS OUT 領域から 32 ビットデータを読み取ります。
	EhGetViosInAddress	VIOS IN 領域の先頭アドレスを取得します。
EhGetViosOutAddress	VIOS OUT 領域の先頭アドレスを取得します。	

D : モーション制御プロセス製品 ("Fine Motion") (株)テクノ製品

Fine Motion は、高度なモーション制御機能を内包したモーション制御プロセス製品です。

それ自身が独立したモーション制御プロセスとして機能し、ユーザーアプリケーションはメールボックスを通じてインターフェースを行い、モーション制御を実現します。

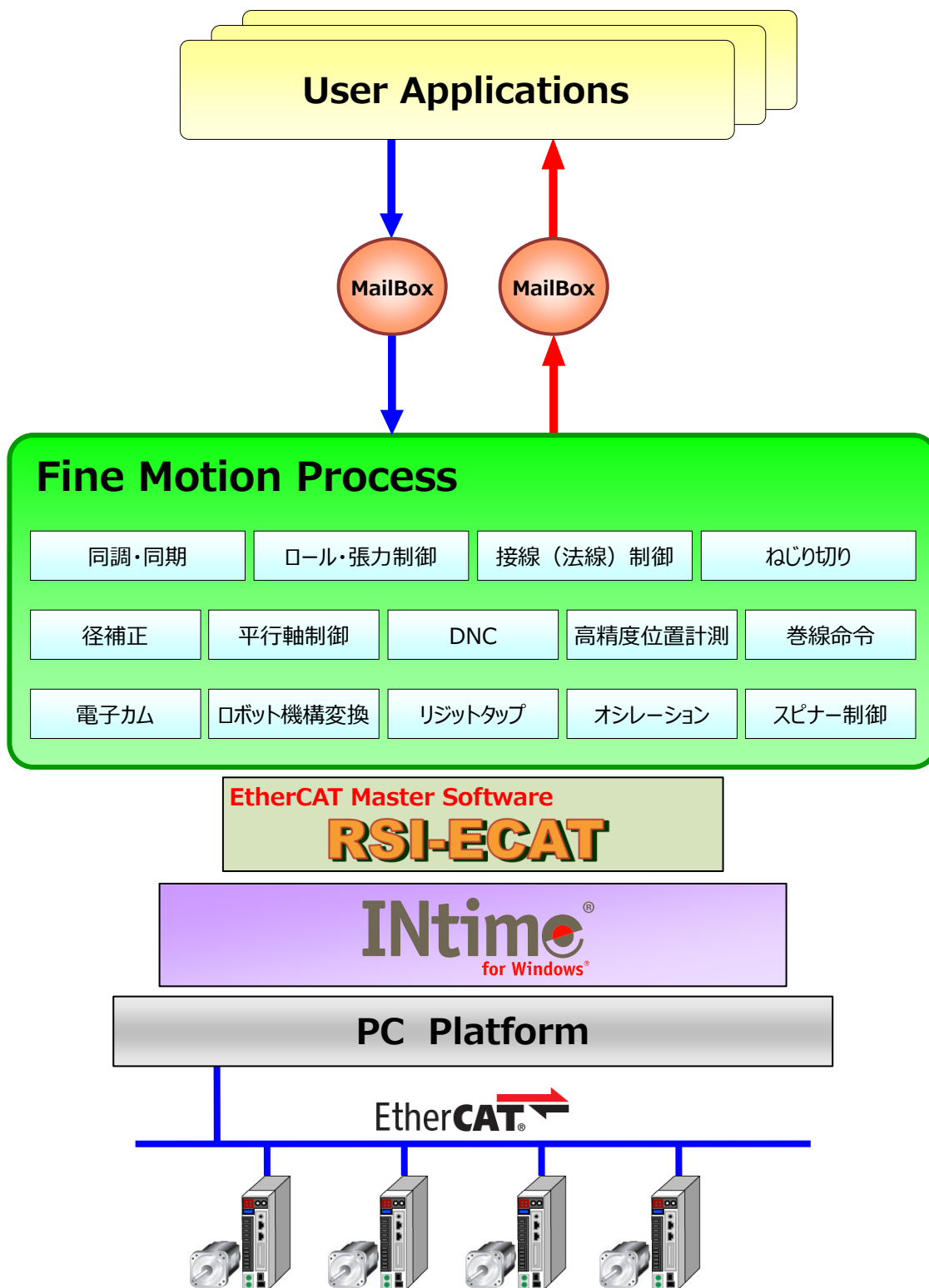


図3 : Fine Motion 製品を利用したソフト構成図

株式会社 **マイクロネット**

営業部 : 〒104-0032 東京都中央区八丁堀 3-17-13 TMYビル 9F
Tel : 03-6909-3371 Fax : 03-6909-3373
E-mail : bcd@mnc.co.jp URL : <http://www.mnc.co.jp/>