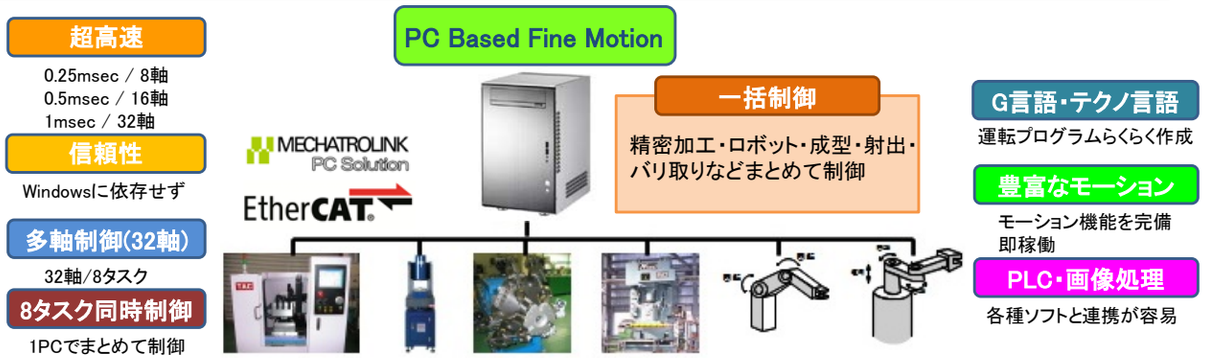
PC Based Fine Motion (RTMC-M3/EC)

MECHATROLINK / EtherCAT対応のモーションコントローラソフトです。
PCが最高性能のモーションコントローラになります。一般NCやロボットコントローラの数倍の
能力により最大8台の繊密マシンを1台のPCで一括制御できます。
INtime とFAPC(Factory Automation PC)によりコントローラの信頼性を向上できます。

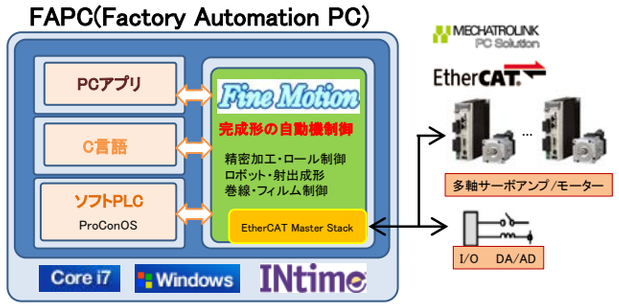
Feature



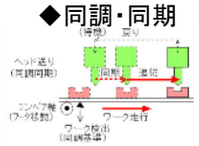
Specification

- 高性能CPUによる高速動作
  (0.25msec/8軸　0.5msec/16軸　1msec/32軸)
- INtimeによりWindowsに依存しない高信頼性
- FAPC使用(FANレス・SSD)による高信頼性
- マルチタスク(最大8台のマシンを同時制御)
- G言語/テクノ言語(ロボット言語)により
  運転プログラムを簡単に作成・運転
- 豊富なモーション機能の完備により即稼働
- ユーザのソフト開発なしに繊密なマシンを制御
- ソフトPLCや画像処理とも容易に連係

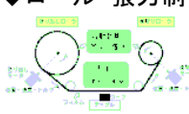
System Configuration



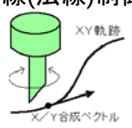
◆同調・同期



◆ロール・張力制御

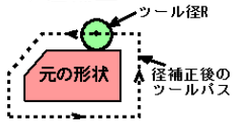


◆接線(法線)制御

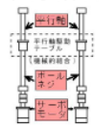


◆ねじ切り

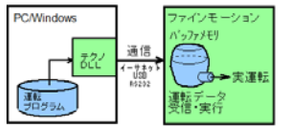
◆径補正



◆平行軸制御

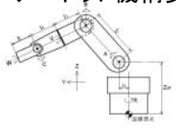


◆DNC



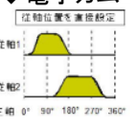
◆高精度位置計測

◆ロボット機構変換

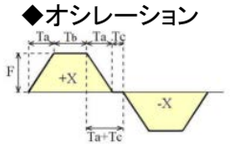


◆リジッドタップ

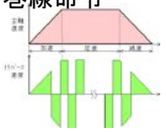
◆電子カム



◆オシレーション



◆巻線命令



◆スピナー制御

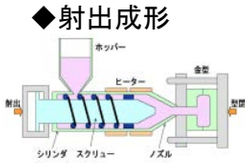
◆真円度解析

Application Sample

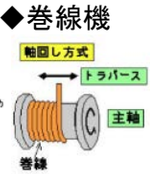
◆精密加工

◆ロボット

◆射出成形

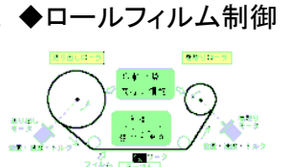


◆巻線機



◆レーザー加工

◆ロールフィルム制御



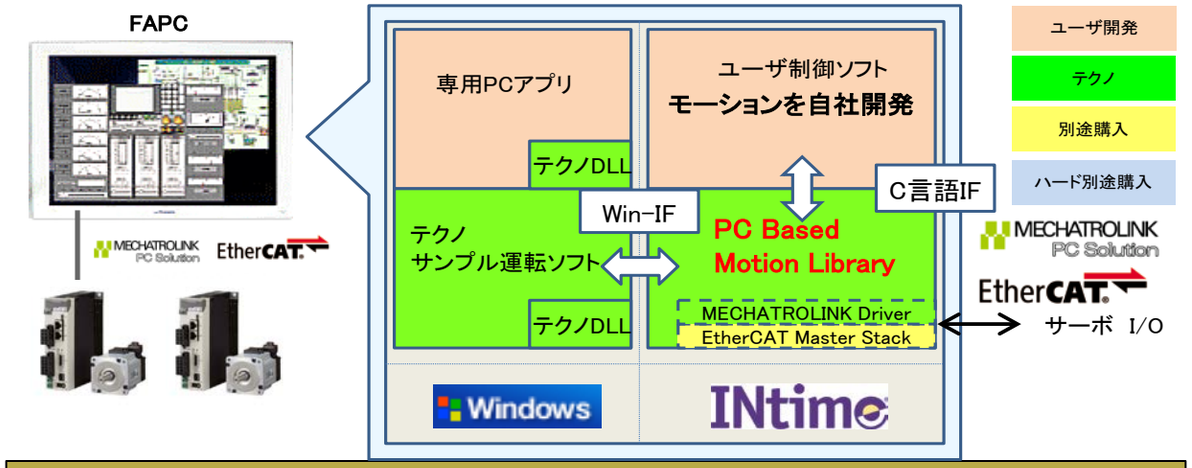
For more information

PC Based Fine Motion : http://www.open-mc.com/products/pdt05.html
INtime : http://www.mnc.co.jp/INtime/

PC Based Motion Library (RTPL-M3/EC)

MECHATROLINK / EtherCAT対応のモーション開発用ソフトです。(モーション機能のソフト関数群)
多軸モーションシステムをC言語で自由に自社開発できます。サンプルソースを用意しているため
短期間に開発できます。さらにPCを利用することで最大64軸を制御できます。
INtime とFAPC(Factory Automation PC)によりシステムの信頼性を向上できます。

System Configuration



Feature

- モーションコントローラをC言語で自社開発(Visual Studioで開発可能)
- サンプルソースがあるため短期間で開発
- 高性能CPUによる高速動作 (0.25msec/8軸 0.5msec/16軸 1msec/32軸 2msec/64軸)
- INtimeによりWindowsに依存しない高信頼性 / FAPC使用(FANレス・SSD)による高信頼性
- Windows上のアプリソフトから簡単に使える
- 画像処理などのソフトとも容易に連係

Specification

関数呼び出しフロー例

ライブラリ初期化
- Rtpl[ML3 | ECT]InitializeLib()
- Rtpl[ML3 | ECT]InitializeAPIThread()
- Rtpl[ML3 | ECT]InitializeMaster()

アラームクリア
- Rtpl[ML3 | ECT]ClearAlarm()

サーボON
- Rtpl[ML3 | ECT]ServoON()

直線補間指令
- Rtpl[ML3 | ECT]LinInterpolate()
- Rtpl[ML3 | ECT]CmdActive()

直線補間完了待機
- Rtpl[ML3 | ECT]ResWait()

サーボOFF
- Rtpl[ML3 | ECT]ServoOFF()

ライブラリ終了
- Rtpl[ML3 | ECT]ReleaseAPIThread()
- Rtpl[ML3 | ECT]CloseLib()

ライブラリ関数

ライブラリ初期化	Rtpl[ML3 | ECT]InitializeLib();	サーボOFF	Rtpl[ML3 | ECT]ServoOFF();
ライブラリリセット	Rtpl[ML3 | ECT]ResetLib();	サーボアラームクリア	Rtpl[ML3 | ECT]ClearAlarm();
ライブラリ終了	Rtpl[ML3 | ECT]CloseLib();	移動中断	Rtpl[ML3 | ECT]StopPositioning();
APIリソースの初期化	Rtpl[ML3 | ECT]InitializeAPIThread();	一時停止	Rtpl[ML3 | ECT]HoldAxis();
APIリソースの破棄	Rtpl[ML3 | ECT]ReleaseAPIThread();	一時停止再開	Rtpl[ML3 | ECT]ResumeAxis();
マスター初期化	Rtpl[ML3 | ECT]InitializeMaster();	同一指令2軸制御設定	Rtpl[ML3 | ECT]SetGantryAxis();
コマンド実行	Rtpl[ML3 | ECT]CmdActive();	同一指令2軸制御解除	Rtpl[ML3 | ECT]ResetGantryAxis();
応答待機	Rtpl[ML3 | ECT]ResWait();	速度オーバライド設定	Rtpl[ML3 | ECT]SetOverride();
位置モニタ	Rtpl[ML3 | ECT]GetPosition();	指令バッファ クリア	Rtpl[ML3 | ECT]ClearCmdBuff();
速度モニタ	Rtpl[ML3 | ECT]GetVelocity();	システムエラークリア	Rtpl[ML3 | ECT]ClearSysError();
ステータスモニタ	Rtpl[ML3 | ECT]GetStatus();	原点復帰	Rtpl[ML3 | ECT]HomePosition();
TPCロギング開始	Rtpl[ML3 | ECT]StartTPCLogging();	位置決め	Rtpl[ML3 | ECT]Positioning();
TPCロギング停止	Rtpl[ML3 | ECT]StopTPCLogging();	ラッチ位置決め	Rtpl[ML3 | ECT]LatchPositioning();
バージョン取得	Rtpl[ML3 | ECT]GetVersion();	直線補間	Rtpl[ML3 | ECT]LinInterpolate();
SDO書込み	RtplECTSetSDO();	円弧補間	Rtpl[ML3 | ECT]CirInterpolate();
SDO読込み	RtplECTGetSDO();	JOG送り	Rtpl[ML3 | ECT]JOGStart();
PDO書込み	RtplECTSetPDO();	JOG停止	Rtpl[ML3 | ECT]JOGStop();
ポジションプリセット	Rtpl[ML3 | ECT]DefinePosition();	トルク制御開始	Rtpl[ML3 | ECT]Torqu[ML3 | ECT]rlStart();
サーボON	Rtpl[ML3 | ECT]ServoON();	トルク制御停止	Rtpl[ML3 | ECT]Torqu[ML3 | ECT]rlStop();

Application Sample

◆半導体製造装置　◆印刷システム　◆電子機器製造ライン　◆その他の多軸制御装置

For more information

PC Based Motion Library : http://www.open-mc.com/products/pdt06.html
INtime : http://www.mnc.co.jp/INtime/