

Application Note		Ref No: an2k-049	Last Modify 150610
テーマ	ILNumerics を使った加減速・軌跡チャートツール		
使用機器	MPC メイン CPU、Visual Studio VB.net、ILNumerics		

概要

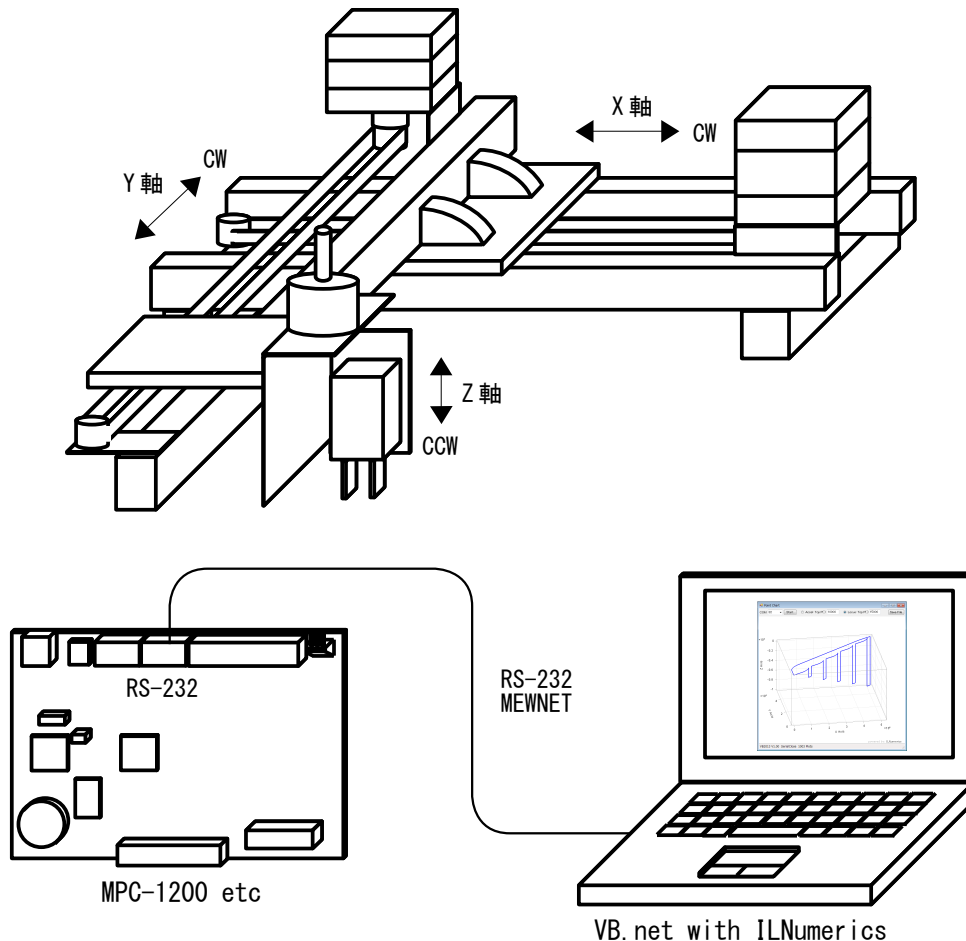
四次元配列の点データは測定値や生産情報などを記録するのにとても便利です。このアプリケーションは点データの記録をパソコンに吸い上げて分析をするツールの一例で、XYZ ロボット動作の加減速の様子と軌跡を ILNumerics でチャート化します。

ILNumerics は .net アプリケーション用の数値計算拡張ツールでチャート表示機能も備えています。基本的には配列にデータを格納して Panel に突っ込めば自動的にチャートを描いてくれます。VB.net の Chart や MS-Excel ではできない(?) 三次元散布図的な表示も簡単、マウスでグルグル回転もします。

今回は Visual Studio 2013 Pro の VB.net、ILNumerics は現在無償で提供されている 3.3.3 を nuget からインストールしました。ILNumerics については WEB 上に多くの情報・サンプルがありますのでそちらをご参照ください。

MPC の点データ取り込み手順は技術情報「[MEWNET プロトコル機能拡張 \(タッチパネルモニターと VB 用通信モジュール\)](#)」をご参照ください。

機器構成



単位時間 (サンプルは 5msec) ごとの各軸パルス数を P(10000) 以降に記録します。
同時に各軸座標を P(15000) 以降に記録します。

通信についてはプログラムに MEWNET 宣言をするだけです。

VB.net with ILNumerics

点データの記録を吸い上げて ILNumerics でチャートを描きます。
CSV 保存も可能。

使用例 1 : XY 水平移動 + Z 上下移動

原点復帰後 XY 水平移動と Z 軸上下を 5 回繰り返して元に戻ります。
点データ配列へのインプットは *PgRecord タスクで行います。記録範囲は原点復帰後の移動です。
MOVS コマンドは直線補間をしません。

MPC プログラム

```
MEWNET 38400 2          /* 通信宣言
pgnum=17
QUIT_FORK 1 *main
END

*main
FILL P(10000) 10000 0  /* 点データ初期化
PG pgnum

GOSUB *HOME

ACCEL X_A 100000 6000 500
ACCEL Y_A 80000 6000 500
ACCEL Z_A 50000 3000 100
FEED X_A|Y_A|Z_A 100

QUIT_FORK 2 *PgRecord  /* 記録開始

span=10000
FOR pls=span TO 50000 STEP span
  MOVS pls pls/2      /* 水平移動
  WAIT RR(X_A|Y_A)==0
  MOVS Z_A pls/5*-1  /* Z 下降
  WAIT RR(Z_A)==0
  RMVL 2000 2000     /* ちょっと水平移動
  WAIT RR(X_A|Y_A)==0
  MOVS Z_A 0         /* Z 上昇
  WAIT RR(Z_A)==0
  TIME 10
NEXT
MOVS 0 0
WAIT RR(X_A|Y_A)==0
TIME 10
QUIT 2              /* 記録終了
PR "owari" SYSCLK pointnum
END

*HOME              /* 原点復帰
SHOM &H55         /* SD 無効
ACCEL X_A|Y_A 10000 1000 500
ACCEL Z_A 8000 500 100
IF SW(203)==1 THEN
  RMVS Z_A -1000
  WAIT RR(Z_A)==0
END_IF
HOME Z_A POS_L    /* Z 原点復帰
WAIT RR(ALL_A)==0
IF SW(200)==1 THEN
  RMVS X_A 5000
END_IF
IF SW(201)==1 THEN
  RMVS Y_A 5000
END_IF
WAIT RR(ALL_A)==0
TIME 500
HOME X_A|Y_A NEG_L /* XY 原点復帰
```

```

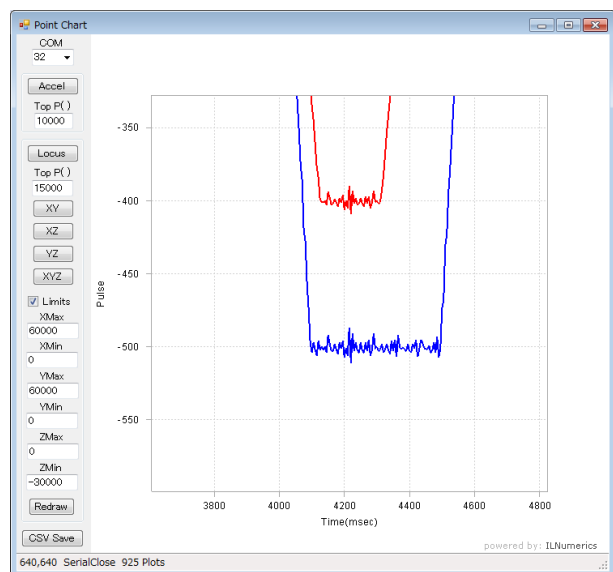
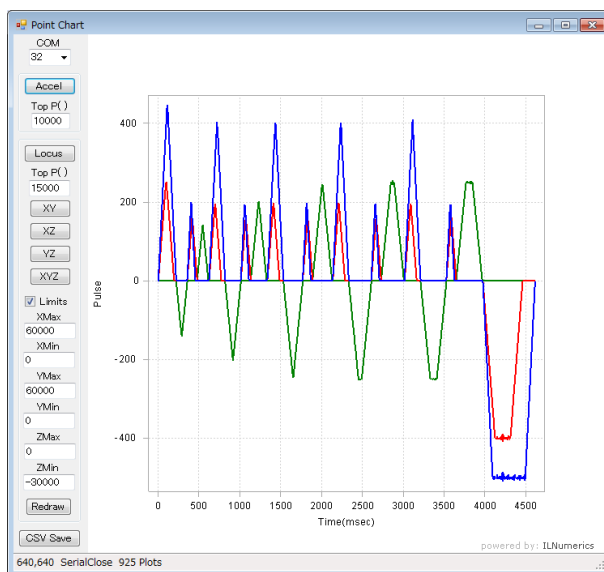
WAIT RR(ALL_A)==0
RMVS X_A|Y_A 1000 /* OFF SET
RMVS Z_A -1000 /* OFF SET
WAIT RR(ALL_A)==0
CLRPOS
TIME 500
RETURN

*PgRecord /* 記録タスク
PG pgnum
FILL P(10000) 10000 0
oldposX_=X(0)
oldposY_=Y(0)
oldposZ_=Z(0)
pointnum_=10000 /* 加減速記録先頭点番号
locusnum_=15000 /* 軌跡記録先頭点番号
SYSCLK=0
interval_=5 /* 記録間隔 msec
DO
WAIT SYSCLK%interval_==0
nowclk_=SYSCLK
nowposX_=X(0) /* X 現在位置
nowposY_=Y(0) /* Y 現在位置
nowposZ_=Z(0) /* Z 現在位置
difposX_=nowposX_-oldposX_ /* X 差分
oldposX_=nowposX_
difposY_=nowposY_-oldposY_ /* Y 差分
oldposY_=nowposY_
difposZ_=nowposZ_-oldposZ_ /* Z 差分
oldposZ_=nowposZ_
/* 点番号 時間 X差分 Y差分 Z差分
SETP pointnum_ nowclk_ difposX_ difposY_ difposZ_
pointnum_=pointnum_+1
/* 点番号 時間 X現位置 Y現位置 Z現位置
SETP locusnum_ nowclk_ nowposX_ nowposY_ nowposZ_
locusnum_=locusnum_+1
WAIT SYSCLK%interval_<>0
LOOP

```

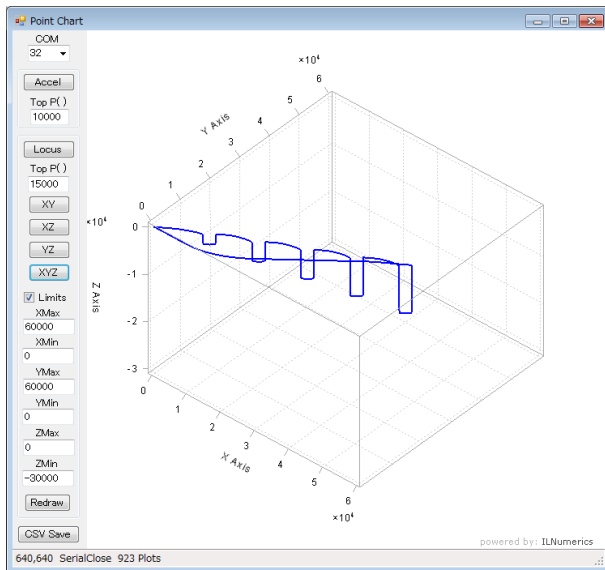
加減速チャート

青：X軸、赤：Y軸、緑：Z軸です。移動距離が短いと最高速まで達せずに三角波になります。部分拡大もできます(右図)。インタプリタの影響でギザギザが出ますが加減速の状態を把握できます。

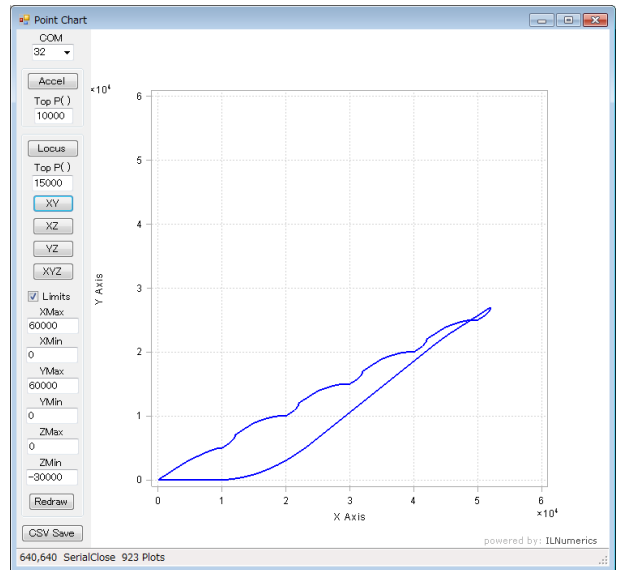


軌跡チャート

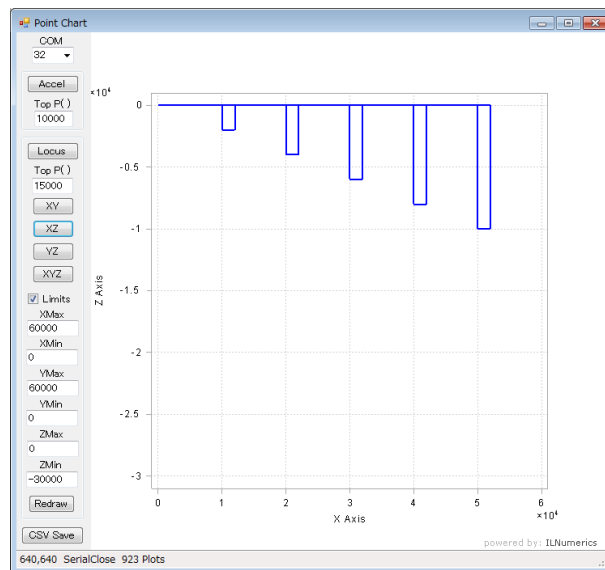
マウスドラッグ、ボタン操作で視点を変えることができます。



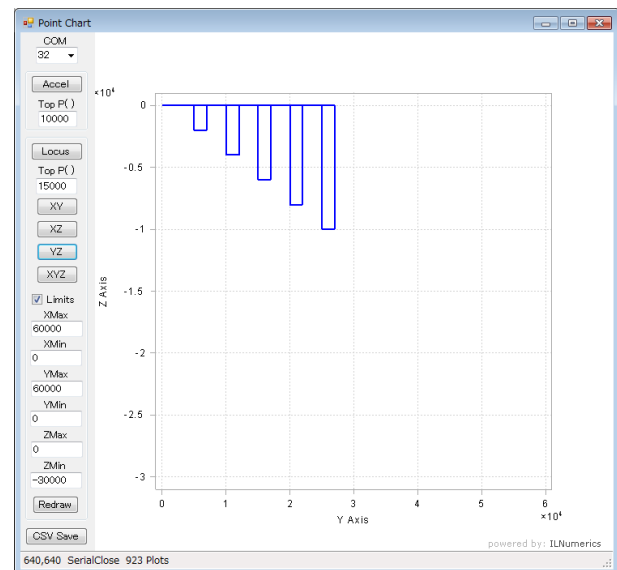
XYZ View



XY View



XZ View



YZ View

使用例 2 : 速度途中変更

X 単軸の移動途中で FEED を変えて速度を下げます。

MPC プログラム

```
ACCEL X_A 100000 6000 500
FEED X_A 100

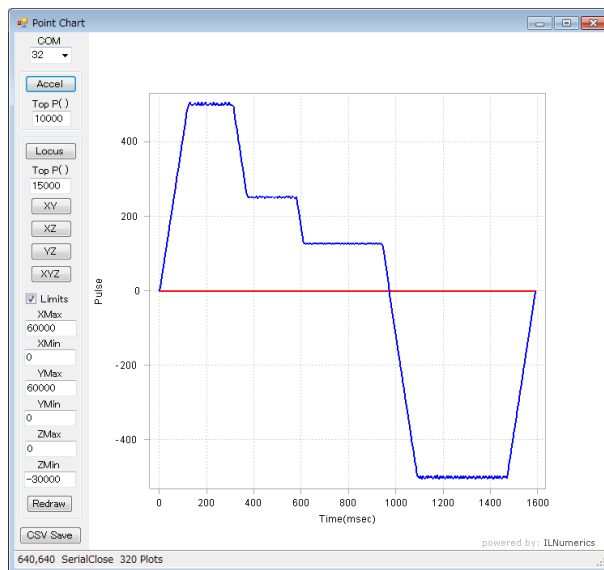
QUIT_FORK 2 *PgRecord /* 記録開始

MOVS X_A 50000
WAIT X(O)>25000
FEED X_A 50
WAIT X(O)>40000
FEED X_A 25
WAIT RR(X_A)==0

FEED X_A 100
MOVS X_A 0
WAIT RR(X_A)==0
TIME 10

QUIT 2 /* 記録終了
```

加減速チャート



使用例 3 : S 字加減速

X 単軸の S 字加減速の様子です。
PG ボードは MPG-2314 です。

MPC プログラム

ACCEL コマンド [1]は S 字加減速無し、[2]は S 字加減速有りです。

(このプログラムは部分です。記録タスクは使用例 1 と同じです。)

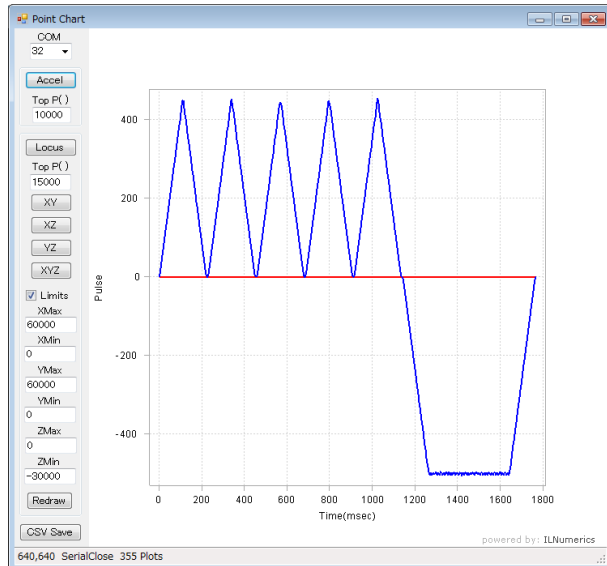
```
/* ACCEL X_A 100000 6000 500      /* [1]
ACCEL X_A|SACL 100000 6000 500  /* [2]
FEED X_A 100

QUIT_FORK 2 *PgRecord          /* 記録開始

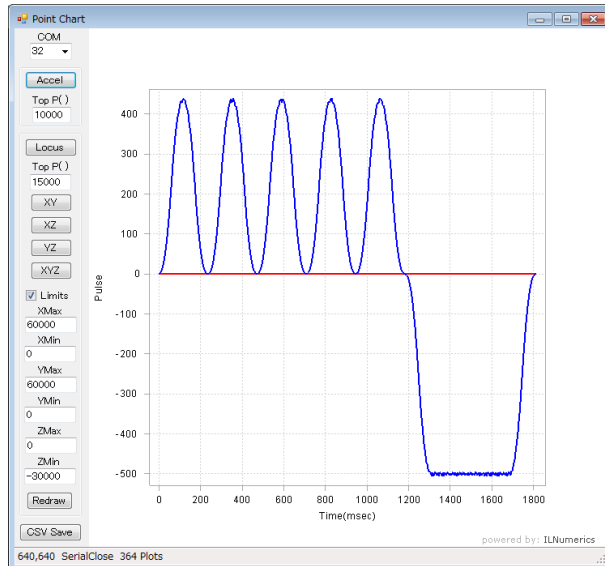
span=10000
FOR pls=span TO 50000 STEP span
  MOVS X_A pls
  WAIT RR(X_A)==0
  TIME 10
NEXT
MOVS X_A 0
WAIT RR(X_A)==0
TIME 10

QUIT 2                          /* 記録終了
```

加減速チャート



[1] S 字無し



[2] S 字有り

使用例 4 : 連続移動

MOV T コマンドによる連続移動です(コマンドリファレンス MOV T 参照)。
PG ボードは MPG-2314 です。

MPC プログラム

```
GOSUB *SET_POINT          /* 作業点と移動形態の作成

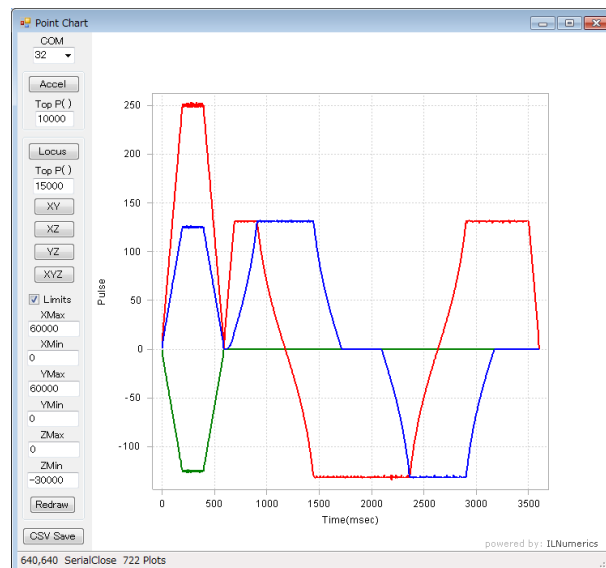
QUIT_FORK 2 *PgRecord     /* 記録開始
axis=X_A|Y_A
CLRPOS
ACCEL ALL_A 50000
MOVL P(1000)              /* スタートポイントへ移動 (3軸移動)
WAIT RR(ALL_A)==0
TIME 10
FEED axis Y(2000)        /* スピード変更
DS_DACL
FOR pnt=1001 TO X(2000)  /* P(1001)~P(1004)連続移動
  MOV T axis P(pnt) X(1000+pnt) /* P(pnt)は移動先、X(1000+pnt)は円弧または直線指定
NEXT
EN_DACL
WAIT RR(axis)==0

QUIT 2                    /* 記録終了
PR "owari" SYSCLK pointnum

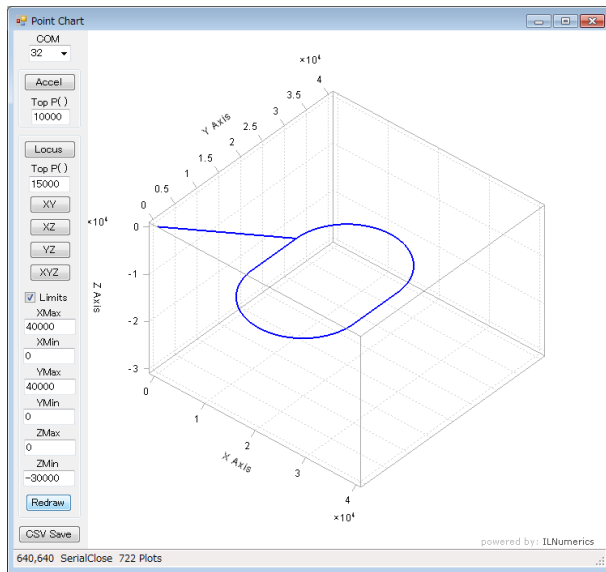
END

*SET_POINT
SETP 1000 10000 20000 0 -10000 /* スタートポイント
SETP 1001 30000 20000 20000 20000 /* XY=円弧終点座標、UZ=円弧中心座標
SETP 1002 30000 10000 0 0 /* XY=移動先座標
SETP 1003 10000 10000 20000 10000 /* XY=円弧終点座標、UZ=円弧中心座標
SETP 1004 10000 20000 0 0 /* XY=移動先座標
SETP 2000 1004 50 0 0 /* X=最終点番号、Y=FEED
SETP 2001 CW 0 0 0 /* P(1000)からP(1001)はCW
SETP 2002 0 0 0 0 /* P(1001)からP(1002)は直線
SETP 2003 CW 0 0 0 /* P(1002)からP(1003)はCW
SETP 2004 0 0 0 0 /* P(1003)からP(1004)は直線
RETURN
```

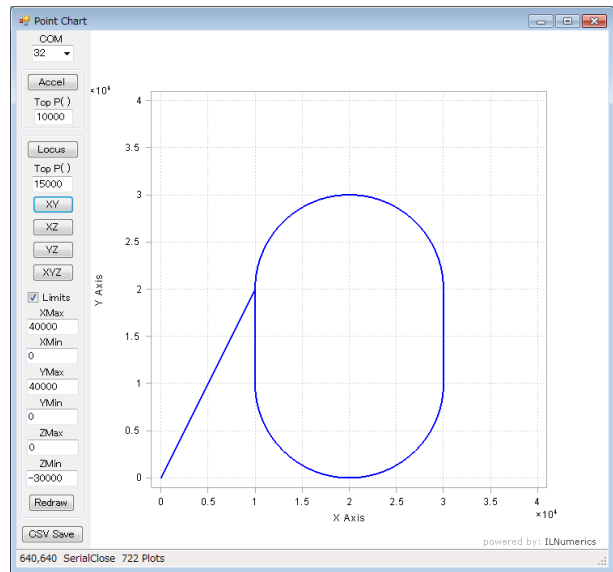
加減速チャート



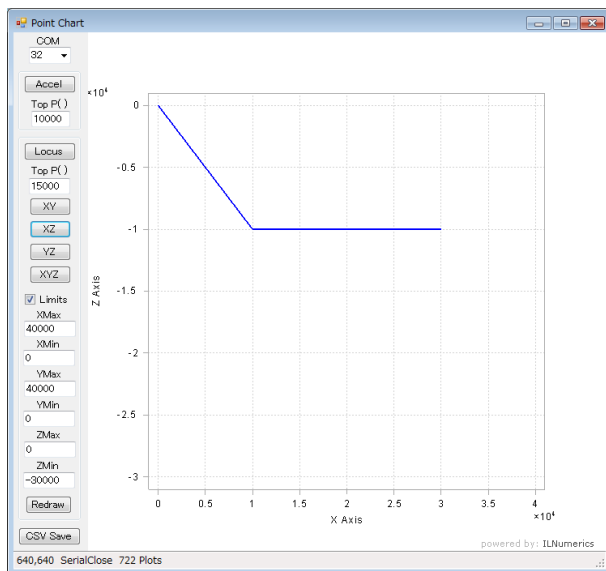
軌跡チャート



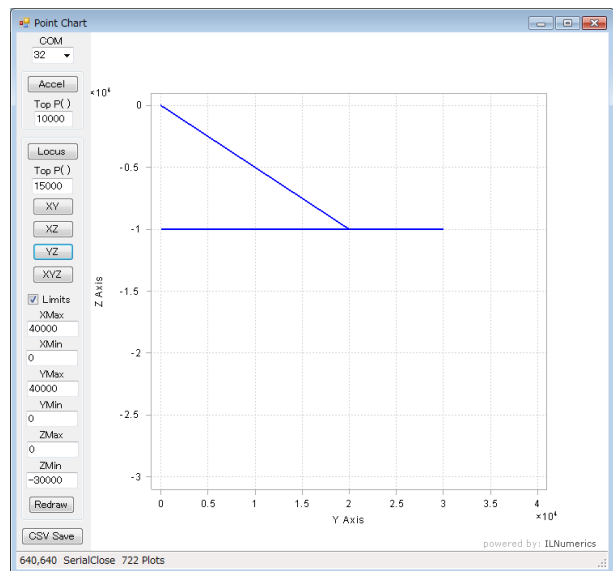
XYZ View



XY View



XZ View



YZ View

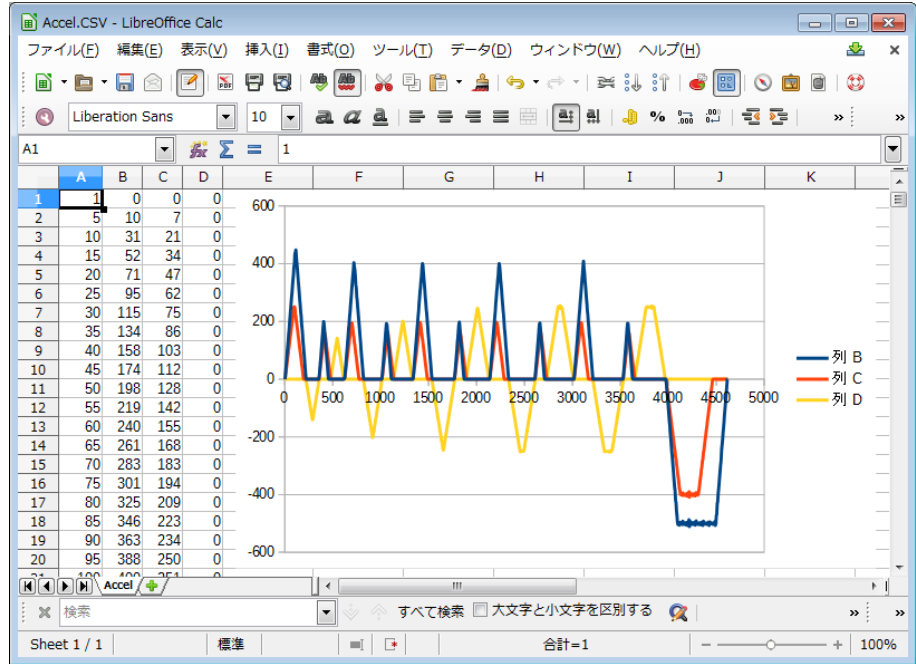
使用例 5 : CSV 保存

MS-Excel 等で扱いやすいようにカンマ区切りで保存します。右は LibreOffice Calc に読み込みウィザードで描画したものです。
データは使用例 1 のものです。

加減速データの CSV

左から 時間、Xパルス数、Yパルス数、Zパルス数

```
0, 0, 0, 0
5, 10, 7, 0
10, 30, 20, 0
15, 53, 35, 0
20, 73, 48, 0
25, 93, 60, 0
30, 113, 74, 0
35, 136, 88, 0
40, 156, 101, 0
45, 177, 115, 0
50, 198, 128, 0
55, 220, 142, 0
60, 241, 155, 0
65, 258, 167, 0
70, 284, 183, 0
75, 304, 196, 0
80, 321, 207, 0
85, 345, 223, 0
90, 361, 232, 0
(以下略)
```



軌跡データの CSV

散布図の 3D 表示の方法がわかりません(出来ない?)

左から X座標、Y座標、Z座標

```
0, 0, 0
10, 7, 0
40, 27, 0
93, 62, 0
166, 110, 0
259, 170, 0
372, 244, 0
508, 332, 0
664, 433, 0
841, 548, 0
1039, 676, 0
1259, 818, 0
1500, 973, 0
1758, 1140, 0
2042, 1323, 0
2346, 1519, 0
2667, 1726, 0
3012, 1949, 0
3373, 2181, 0
(以下略)
```

