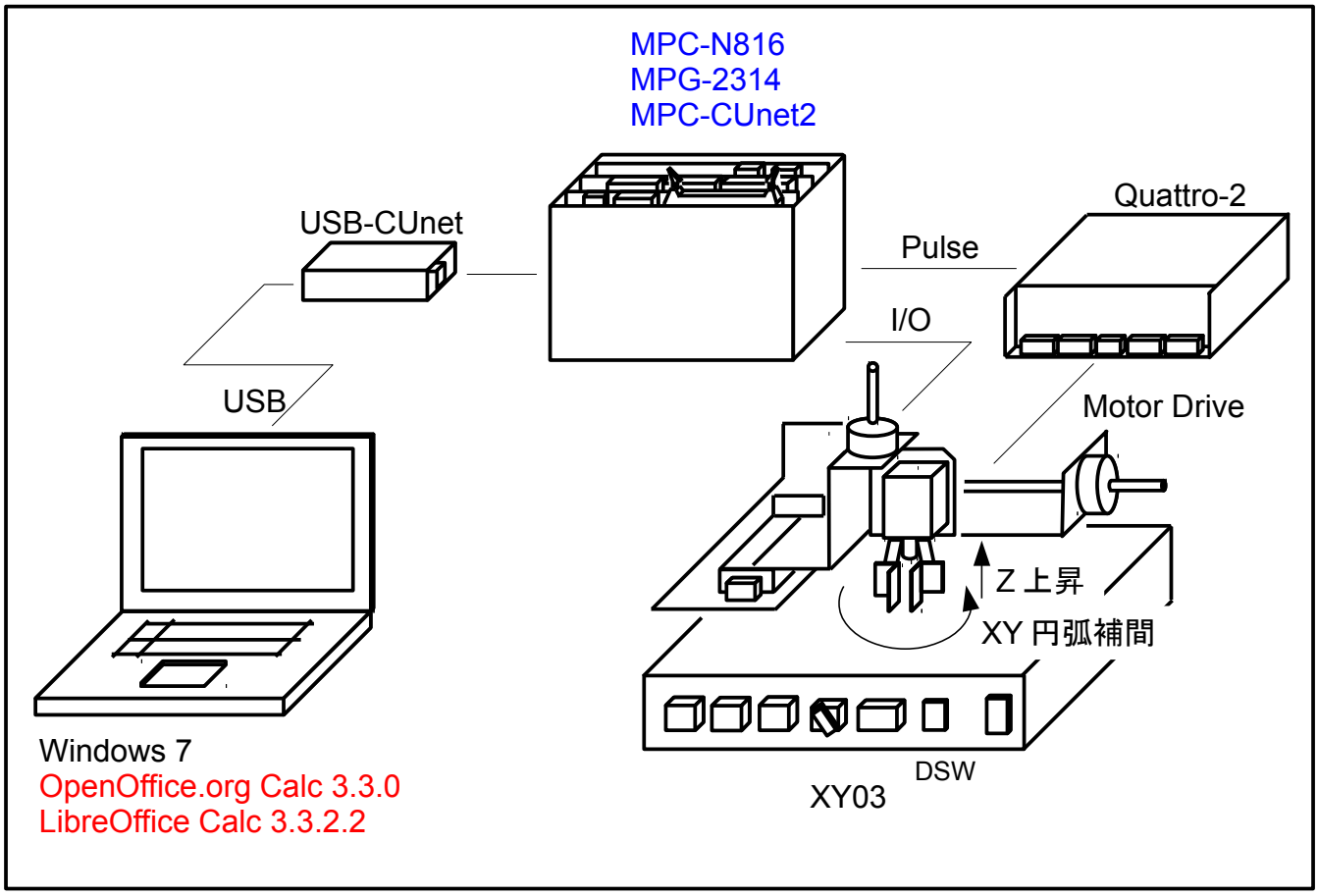
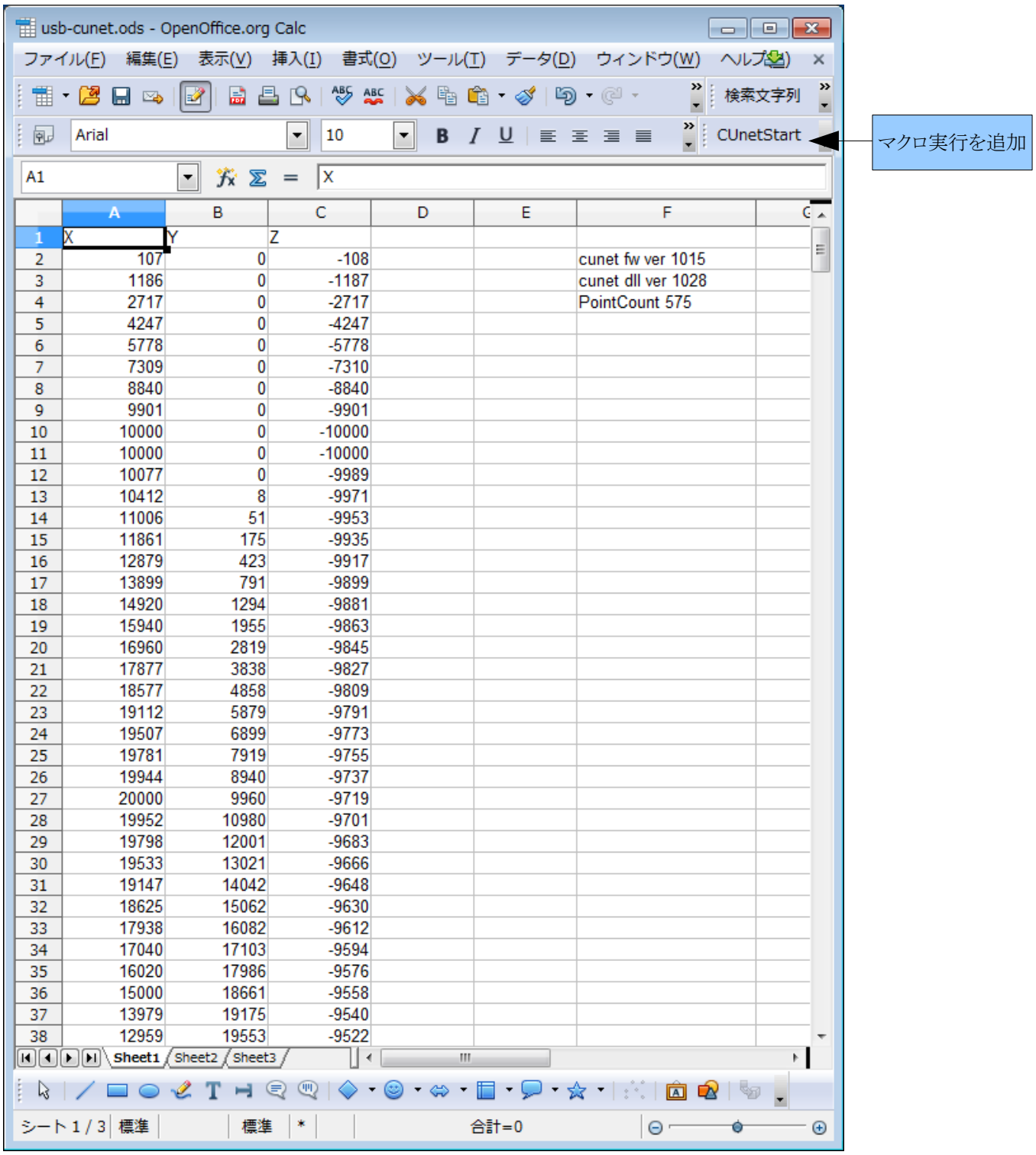
Application Note		資料作成 110429	資料番号 an2k-031
テーマ	Cunet メールを使った XYZ ロボットの 3 次元トレース　OpenOffice.org、LibreOffice 編		
使用機器	MPC-N816(MPC-2000 シリーズどれでも可)、MPG-2314、MPC-CUnet2、USB-CUnet、OpenOffice 他		

■機器構成



- ハードウェア構成、MPC のプログラムはアプリケーションノート an2k-030 と同じですが、パソコンのソフトに表計算ソフト「OpenOffice.org Calc」を用いました。
  ↓アプリケーションノート an2k-030
  http://departonline.jp/mpc2000/ref/headline/appendix/pdf/an2k-030.pdf
- OpenOffice.org Basic(MS-Excel の VBA 相当)のマクロで USB-CUnet から直接データをワークシートに入力します。
- 表計算ソフトに直接入力することはデータ収集・解析・保存に有用と思います。
- 「OpenOffice.org」で作成後に「LibreOffice」をインストールしたところ「 LibreOffice Calc」でもそのまま動きました。マクロボタンの設定も反映されました。根っ子が同じだから当然か...

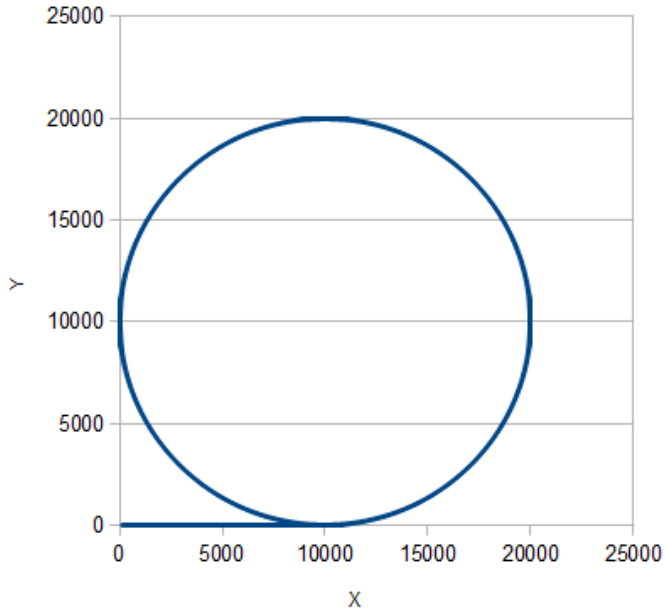
■マクロ実行後の OpenOffice.org Calc 画面



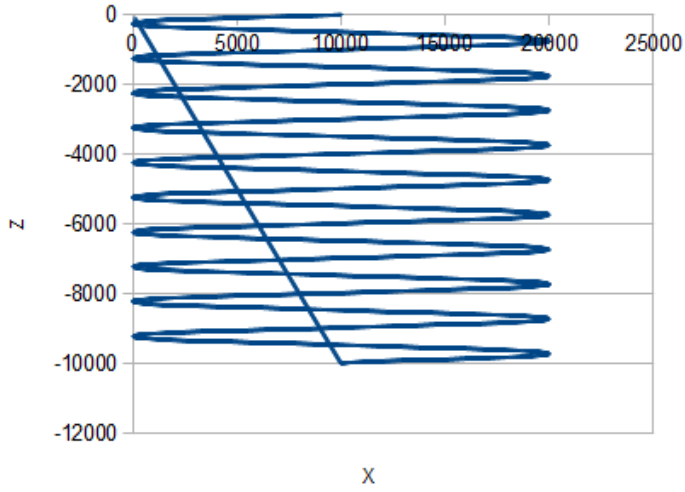
	A	B	C	D	E	F
1	X	Y	Z			
2	107	0	-108			cunet fw ver 1015
3	1186	0	-1187			cunet dll ver 1028
4	2717	0	-2717			PointCount 575
5	4247	0	-4247			
6	5778	0	-5778			
7	7309	0	-7310			
8	8840	0	-8840			
9	9901	0	-9901			
10	10000	0	-10000			
11	10000	0	-10000			
12	10077	0	-9989			
13	10412	8	-9971			
14	11006	51	-9953			
15	11861	175	-9935			
16	12879	423	-9917			
17	13899	791	-9899			
18	14920	1294	-9881			
19	15940	1955	-9863			
20	16960	2819	-9845			
21	17877	3838	-9827			
22	18577	4858	-9809			
23	19112	5879	-9791			
24	19507	6899	-9773			
25	19781	7919	-9755			
26	19944	8940	-9737			
27	20000	9960	-9719			
28	19952	10980	-9701			
29	19798	12001	-9683			
30	19533	13021	-9666			
31	19147	14042	-9648			
32	18625	15062	-9630			
33	17938	16082	-9612			
34	17040	17103	-9594			
35	16020	17986	-9576			
36	15000	18661	-9558			
37	13979	19175	-9540			
38	12959	19553	-9522			

MPC の動作完了後、CUnet メールで点データを読み込みます。

■グラフウィザードで描いたグラフ

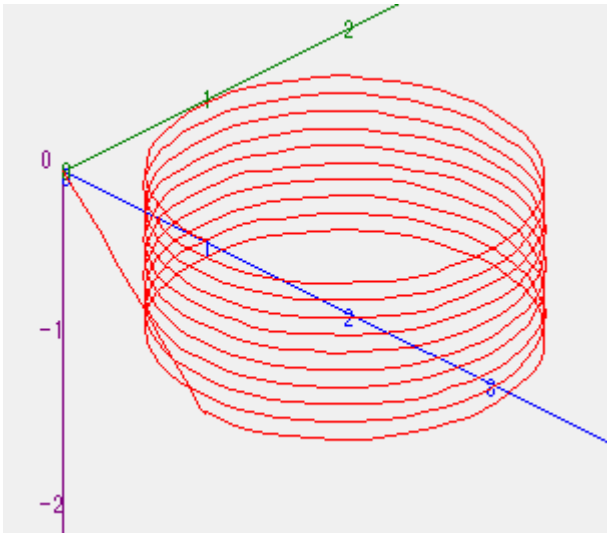


XY View



XZ View

本当は... 次の an2k-030 のように XYZ の 3 次元で描きたかったのですが OpenOffice.org Calc は?でした。



■OpenOffice.org Basic コード

MPC のプログラムは アプリケーションノート an2k-030 をご覧ください。

```
REM  *****  BASIC  *****
Option Explicit

'usbcunet.dll関数宣言
Declare Function cunet_usb_open Lib "usbcunet.dll" () As Long
Declare Function cunet_fw_ver Lib "usbcunet.dll" () As Long
Declare Function cunet_dll_ver Lib "usbcunet.dll" () As Long
Declare Sub cunet_init Lib "usbcunet.dll" _
       (ByVal sa As Long, ByVal ow As Long, ByVal en As Long)
Declare Function cunet_sw Lib "usbcunet.dll" _
       Alias "cunet_sw_s" (ByVal adr As Long) As Long
Declare Sub cunet_on Lib "usbcunet.dll" _
```

```
	Alias "cunet_on_s" (ByVal adr As Long)
Declare Sub cunet_off Lib "usbcunet.dll" _
	Alias "cunet_off_s" (ByVal adr As Long)
Declare Function cunet_in Lib "usbcunet.dll" _
	Alias "cunet_in_s" (ByVal adr As Long, ByVal siz As Long) As Long
Declare Sub cunet_out Lib "usbcunet.dll" _
	Alias "cunet_out_s" (ByVal dat As Long, ByVal adr As Long, ByVal siz As Long)
Declare Function cunet_req_pnt Lib "usbcunet.dll" _
	Alias "cunet_req_pnt_s" (ByVal req_sa As Long, ByVal ar_top As Long, ByRef rcv_ar as Variant)
As Long

Public Const CuChr = 1		'1byte符号無
Public Const CuInt = 2		'2Byte符号付
Public Const CuWrd = 4		'2Byte符号無
Public Const CuLng = 8		'4Byte符号付

sub Main

	call all_clear
	call CunetOpen
	call XyzStart

end sub


sub CunetOpen
	If cunet_usb_open <> 1 Then
		ThisComponent.Sheets(0).getCellByPosition(5,1).String="CUnet Open Error"
		Exit Sub
	End If

	ThisComponent.Sheets(0).getCellByPosition(5,1).String="cunet fw ver " + Cstr(cunet_fw_ver)
	ThisComponent.Sheets(0).getCellByPosition(5,2).String="cunet dll ver " + Cstr(cunet_dll_ver)

	cunet_init(255, 0, 0)		'initialize usbcunet
	Wait 500
	cunet_init(0,12,23)	'sets usbcunet SA , Global memory area
	Wait 500

end sub

sub XyzStart

	Dim sa_adr , p_adr , ar(60) as long

	Dim ln , rw , j
	Dim PointCount, BlockCount

	cunet_on 2000						' start MPC action
	Do While cunet_sw(2832)<>1			' wait for finish
		Wait 1
	Loop

	cunet_off 2000
	Do While cunet_sw(2832)<>0
		Wait 1
	Loop

	PointCount=cunet_in(2096,CuLng)
	ThisComponent.Sheets(0).getCellByPosition(5,3).String="PointCount " + Cstr(PointCount)
	BlockCount=PointCount/15+1

	ThisComponent.Sheets(0).getCellByPosition(0,0).String="X"
	ThisComponent.Sheets(0).getCellByPosition(1,0).String="Y"
	ThisComponent.Sheets(0).getCellByPosition(2,0).String="Z"

	sa_adr=12						' the number of reqest SA
	p_adr=1000						' reading the point number

	ln=1							' line number
	For j=1 To BlockCount
		cunet_req_pnt(sa_adr, p_adr, ar)
		For rw=0 to 59 Step 4
			IF ln <= PointCount Then
				ThisComponent.Sheets(0).getCellByPosition(0,ln).value=ar(rw)
```

```
                        ThisComponent.Sheets(0).getCellByPosition(1,ln).value=ar(rw+1)
                        ThisComponent.Sheets(0).getCellByPosition(2,ln).value=ar(rw+2)
                    End IF
                    ln=ln+1
                Next rw
                p_adr=p_adr+15
        Next j

end sub

sub all_clear

    rem ----------------------------------------------------------------------
    rem define variables
    dim document   as object
    dim dispatcher as object
    rem ----------------------------------------------------------------------
    rem get access to the document
    document   = ThisComponent.CurrentController.Frame
    dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

    rem ----------------------------------------------------------------------
    dispatcher.executeDispatch(document, ".uno:SelectAll", "", 0, Array())

    rem ----------------------------------------------------------------------
    dim args2(0) as new com.sun.star.beans.PropertyValue
    args2(0).Name = "Flags"
    args2(0).Value = "SVDFN"

    dispatcher.executeDispatch(document, ".uno:Delete", "", 0, args2())

    rem ----------------------------------------------------------------------
    dim args3(0) as new com.sun.star.beans.PropertyValue
    args3(0).Name = "ToPoint"
    args3(0).Value = "$A$1"

    dispatcher.executeDispatch(document, ".uno:GoToCell", "", 0, args3())

end sub
```

--End Of File--