

關於 modbus RTU comm..and 0x03 的解說如下面的連結：

<https://blog.csdn.net/Andy2019/article/details/63676075>

文件中有範例如下圖：

想要讀記憶體位置 108~110 一共 3 個暫存器，

實際上送出來的信號是 0x01, 0x03, 0x00, 0x6b, 0x0, 0x03, 0x74, 0x17

其中 0x6b（十進位數值為 107）就是用來讀 108 暫存器，

Request

Function code	1 Byte	0x03
Starting Address	2 Bytes	0x0000 to 0xFFFF
Quantity of Registers	2 Bytes	1 to 125 (0x7D)

Response

Function code	1 Byte	0x03
Byte count	1 Byte	2 x N*
Register value	N* x 2 Bytes	

*N = Quantity of Registers

Error

Error code	1 Byte	0x83
Exception code	1 Byte	01 or 02 or 03 or 04

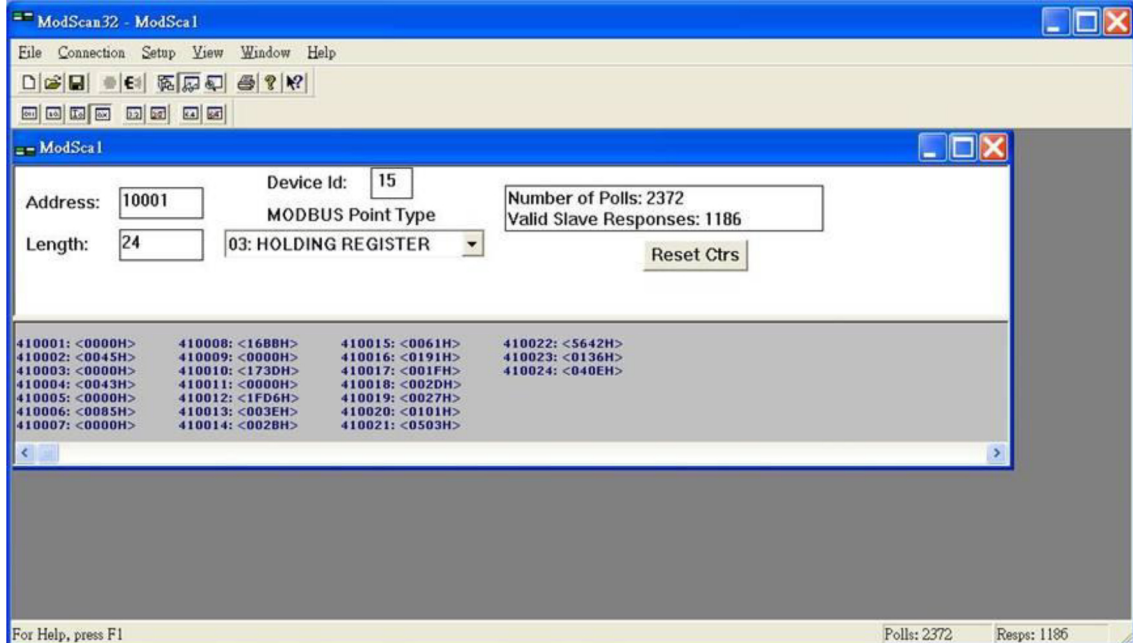
<http://blog.csdn.net/Andy2019>

Here is an example of a request to read registers 108 – 110:

Request		Response	
Field Name	(Hex)	Field Name	(Hex)
Function	03	Function	03
Starting Address Hi	00	Byte Count	06
Starting Address Lo	6B	Register value Hi (108)	02
No. of Registers Hi	00	Register value Lo (108)	2B
No. of Registers Lo	03	Register value Hi (109)	00
		Register value Lo (109)	00
		Register value Hi (110)	00
		Register value Lo (110)	64

VB310 文件寫的暫存器位置就是實際送出信號的位置，因此要用 Modscan32 這個軟體讀取 VB-310 文件中的 10000~10023，

在 modscan32 要輸入起始位址要輸入 10001(實際送出信號的位置 10000)，長度要輸入 24，如下圖的作法，

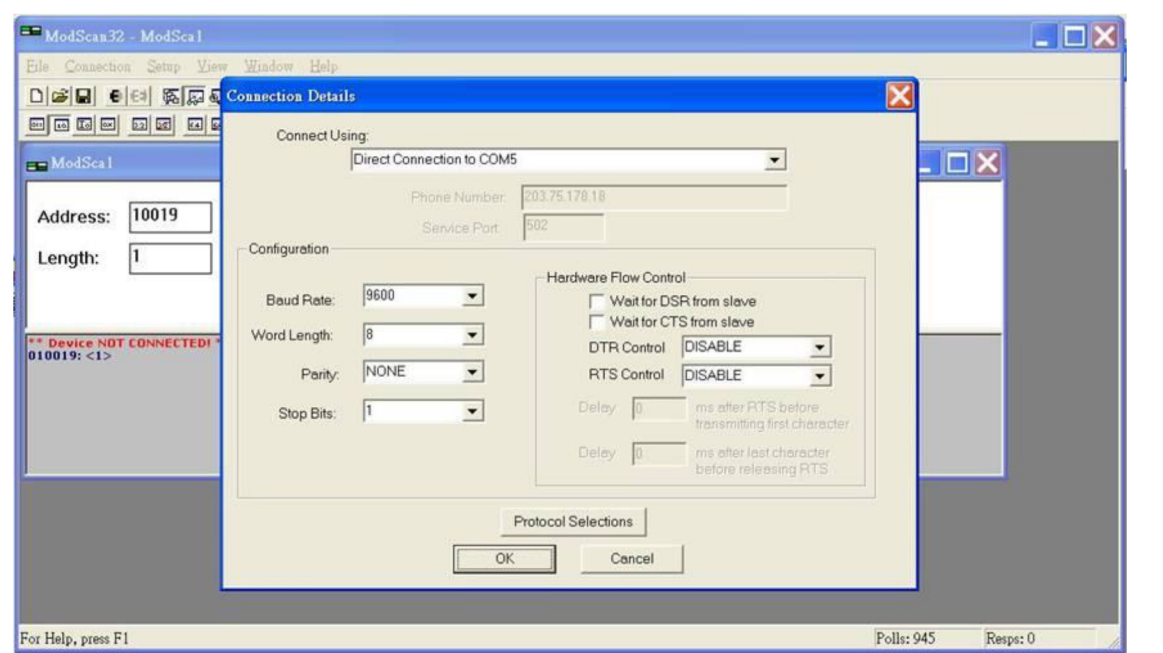


如此可以把下面的資訊都讀取進來，

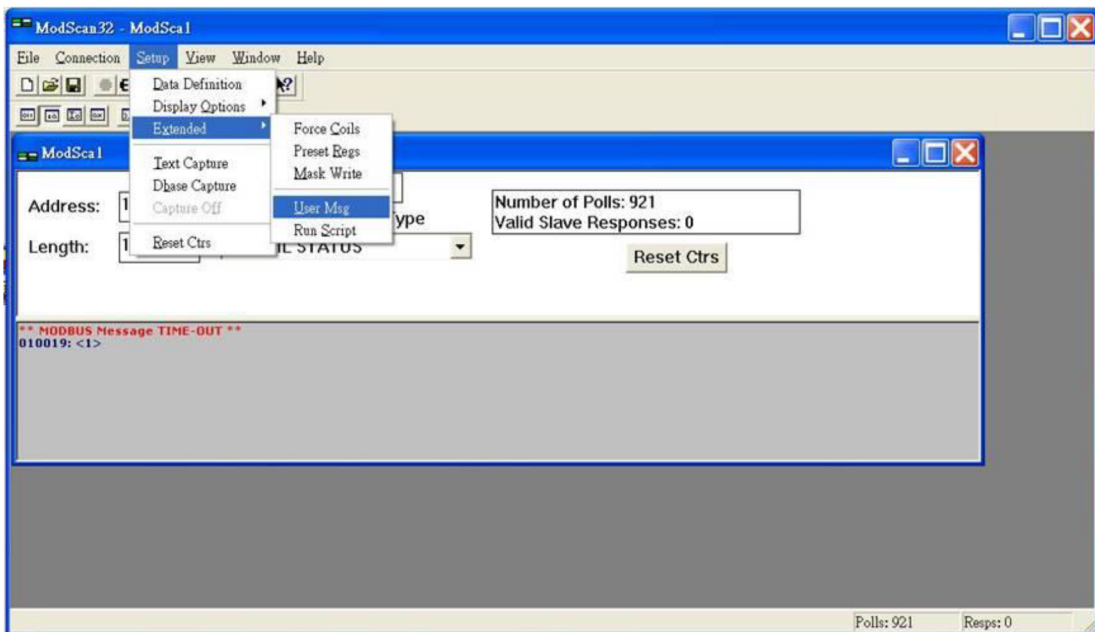
位址(十進位)	名稱	描述
10000	X 軸速度的高 16 位元	32 位元正整數(包含零)
10001	X 軸速度的低 16 位元	1 單位= 0.001 mm/s(r.m.s)
10002	Y 軸速度的高 16 位元	32 位元正整數(包含零)
10003	Y 軸速度的低 16 位元	1 單位= 0.001 mm/s(r.m.s)
10004	Z 軸速度的高 16 位元	32 位元正整數(包含零)
10005	Z 軸速度的低 16 位元	1 單位= 0.001 mm/s(r.m.s)
10006	X 軸加速度的高 16 位元	32 位元正整數(包含零)
10007	X 軸加速度的低 16 位元	1 單位= 0.01 mm/s/s(r.m.s)
10008	Y 軸加速度的高 16 位元	32 位元正整數(包含零)
10009	Y 軸加速度的低 16 位元	1 單位= 0.01 mm/s/s(r.m.s)
10010	Z 軸加速度的高 16 位元	32 位元正整數(包含零)
10011	Z 軸加速度的低 16 位元	1 單位= 0.01 mm/s/s(r.m.s)
10012	X 軸位移	16 位元正整數(包含零)
		1 單位= 0.1 微米(峰到峰)

10013	Y 軸位移	16 位元正整數(包含零) 1 單位= 0.1 微米(峰到峰)
10014	Z 軸位移	16 位元正整數(包含零) 1 單位= 0.1 微米(峰到峰)
10015	中央處理器內部溫度	16 位元整數. 1 單位= 0.1°C, 範圍=[-32768, 32767] == [-3276.8°C, 3276.7°C]
10016	X 軸峭度	16 位元正整數(包含零) 1 單位= 0.1 峭度
10017	Y 軸峭度	16 位元正整數(包含零) 1 單位= 0.1 峭度
10018	Z 軸峭度	16 位元正整數(包含零) 1 單位= 0.1 峭度
10019	軟體版別	軟體版別= 高位元組 . 低位元組
10020	硬體版別	硬體版別= 高位元組 . 低位元組
10021	振動規類型	0x5642(==“VB”, 表示 VB-310)
10022	振動規編號	310 (表示 VB-310)
10023	系統時間	16 位元正整數(包含零); 每秒加一.

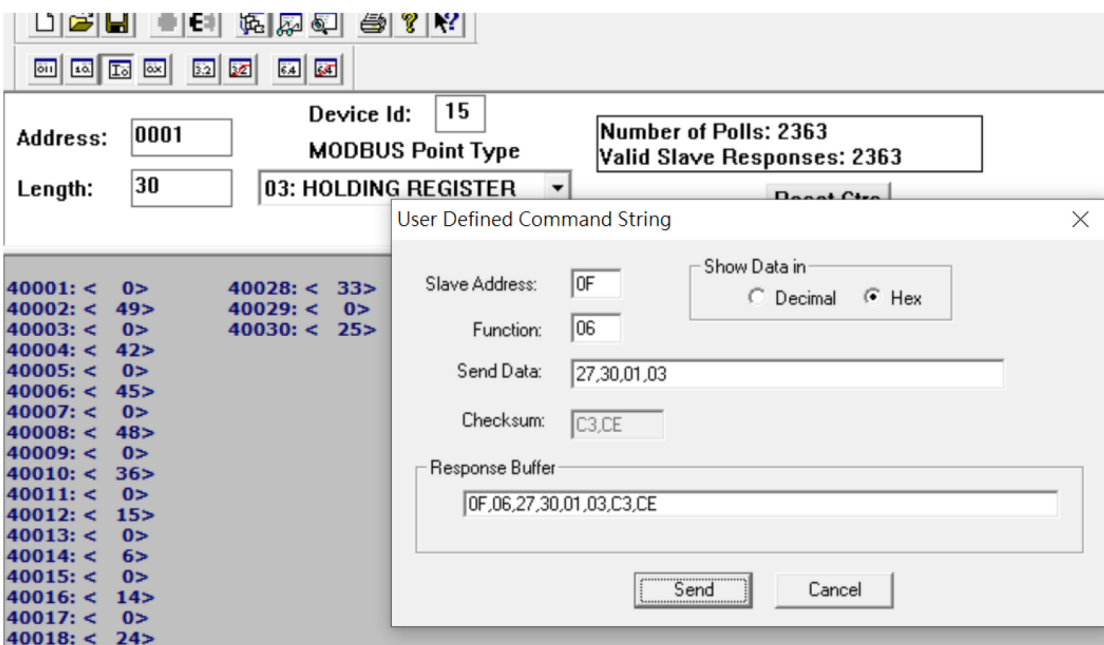
讀取 Z 軸速度的 FFT，方法如下：



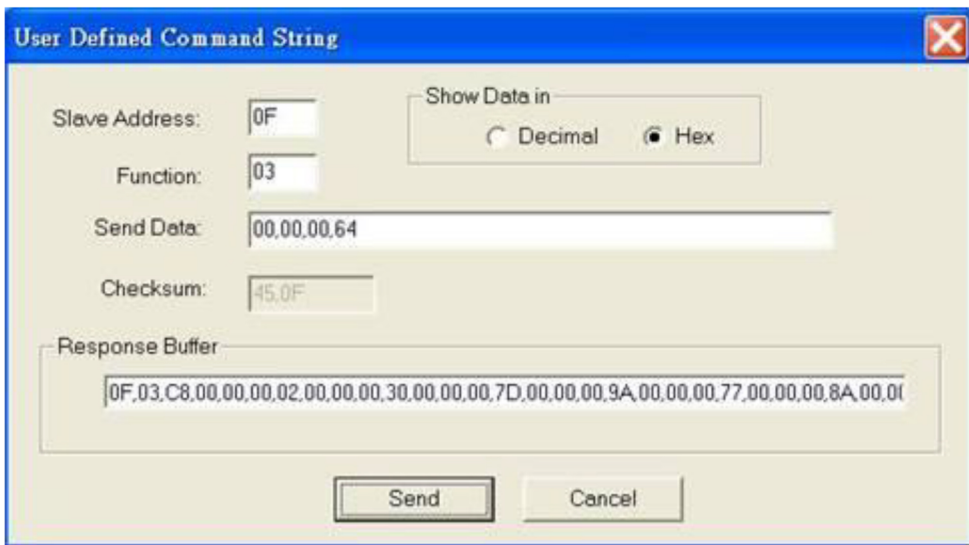
1. 使用 User Msg，定義命令內容。



2. 利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x3, 0xC3, 0xCE，啟動 Z 軸速度 FFT。



如果 SW version >= 1.1 每次讀取 100 個暫存器(50 Hz 也就是 200 個 Byte)



The image shows a Windows-style dialog box titled "User Defined Command String". It contains several input fields and a "Show Data in" section. The "Slave Address" field is set to "0F". The "Function" field is set to "03". The "Send Data" field contains the hex string "00,00,00,64". The "Checksum" field contains "45,0F". The "Show Data in" section has two radio buttons: "Decimal" (unselected) and "Hex" (selected). Below these is a "Response Buffer" section with a text area containing the hex string "0F,03,C8,00,00,00,02,00,00,00,30,00,00,00,7D,00,00,00,9A,00,00,00,77,00,00,00,8A,00,00". At the bottom are "Send" and "Cancel" buttons.

User Defined Command String

Slave Address: 0F

Function: 03

Send Data: 00,00,00,64

Checksum: 45,0F

Show Data in
☐ Decimal ☒ Hex

Response Buffer

0F,03,C8,00,00,00,02,00,00,00,30,00,00,00,7D,00,00,00,9A,00,00,00,77,00,00,00,8A,00,00

Send Cancel

使用上來看，VB-310 搭配 MODSCAN32 可行。

可利用下面的命令正確啟動 FFT：

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x01, 0x42, 0x0f，啟動 X 軸速度 FFT。

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x02, 0x2, 0x0e，啟動 Y 軸速度 FFT。

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x03, 0xc3, 0xce，啟動 Z 軸速度 FFT。

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x05, 0x43, 0xcc，啟動 X 軸加速度 FFT。

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x06, 0x03, 0xcd，啟動 Y 軸加速度 FFT。

利用 0x0f, 0x06, 0x27, 0x30, 0x01, 0x07, 0xc2, 0x0d，啟動 Z 軸加速度 FFT。