

R□M シリーズ用 Modbus Driver (形式:MODOCX) 取扱説明書

(本取扱説明書は OCX アプリケーションのバージョン Ver.1.0.0 に対応しています。)

NM-9226-A 改1

<目次>

1. 技術的特徴	3
1.1. はじめに.....	3
1.2. ActiveX コントロール通信ライブラリ	3
1.3. レジスタ登録.....	3
1.4. 動作環境.....	4
1.5. 免責事項.....	4
1.6. 通信データのタグ名設定.....	4
1.7. データログ機能	5
2. Msystem MD Modbus メソッド説明	6
2.1. 通信	6
2.1.1. Modbus_Serial_Connect()	6
2.1.2. Modbus_Serial_Disconnect()	6
2.1.3. Modbus_TCP_Connect()	7
2.1.4. Modbus_TCP_Disconnect()	7
2.1.5. Modbus_ReadVal()	8
2.1.6. Modbus_ReadVals()	8
2.1.7. Modbus_WriteVal()	9
2.1.8. Modbus_WriteVals()	9
2.2. タグ情報.....	10
2.2.1. Modbus_SetTag()	10
2.3. 通信ログ.....	10
2.3.1. Modbus_StartLog()	10
2.3.2. Modbus_StopLog()	10
2.4. ステータスコード	11

1. 技術的特徴

1.1. はじめに

本書は、「R□M シリーズ用 Modbus ActiveX Control : MsysMDModbus.ocx」の取り扱い方法、操作手順、注意事項などを説明したものです。Windows 上の OLE コンテナアプリケーションである Visual Basic, Excel などの操作や用語を理解している方を前提にしています。アプリケーション Visual Basic, Excel などの操作や用語については、それぞれのマニュアルを参照してください。

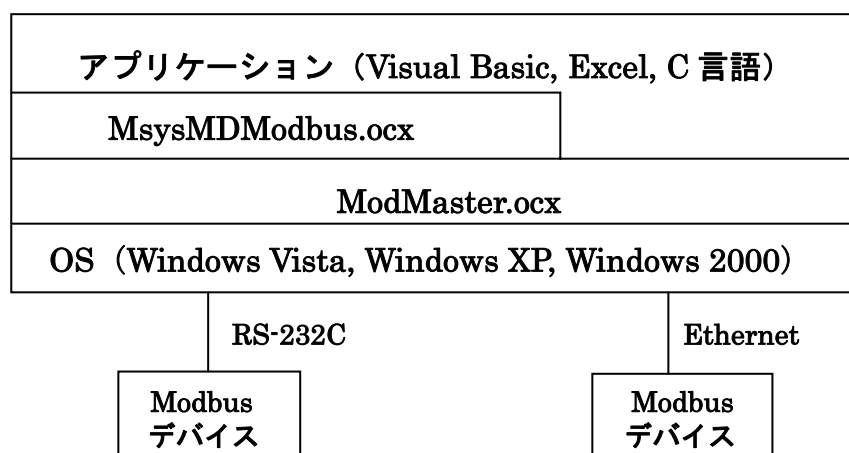
1.2. ActiveX コントロール通信ライブラリ

一般に、ActiveX コントロールとは、Windowsにおいてアプリケーションソフト間のデータ連携を実現する仕組みであるOLEに準拠して作成されたソフトウェア部品です。単独で実行することはできず、アプリケーションソフトに組み込んで使用します。

R□M シリーズ用 ActiveX Control (MsysMDModbus.ocx) は、Visual Basic や Excel などの OLE コンテナアプリケーション上で、PC に接続された Modbus デバイスのデータに迅速にかつ容易にアクセスすることを可能にします。接続は、複数の Modbus デバイスにアクセスでき、非同期動作が可能です。

MsysMDModbus.ocx は、Modbus 通信のデータ構造や通信の仕組み、通信エラーの処理方法を知る必要なく、Modbus 通信の Open/Close やデータの Read/Write を容易に行なえるようにします。

また、MsysMDModbus.ocx は、アドレス指定で 1 byte データにアクセスできる ModMaster.ocx を含む形式で作成されています。それゆえ、MsysMDModbus.ocx を組み込んだアプリケーション上では、ModMaster.ocx のコントロールも直接利用することができます。ModMaster.ocx のコントロールに関しては、Modbus Master ActiveX Control 取扱説明書に記述してありますので、ご覧下さい。



1.3. レジスタ登録

MsysMDModbus.ocx のコントロールをアプリケーションで使う前に、Windows レジストリーに MsysMDModbus.ocx と MbMaster.ocx を登録する必要があります。以下の例のように、DOS コマンドで RegSvr32.exe の引数にフルパスのコントロール名を与えることで、登録することができます。

```
Regsvr32 C:\¥xxxxxx¥xxxxxx¥MsysMDModbus.ocx
```

```
Regsvr32 C:\¥xxxxxx¥xxxxxx¥ModMaster.ocx
```

簡単な登録方法として、SampleProj フォルダ内にバッチファイル MsysReg.bat が用意してあります。MsysMDModbus.ocx と ModMaster.ocx の両方のレジスタ登録を一緒に行うことができます。

レジスタ登録“succeeded”を示すメッセージがそれぞれ出力されますので、OK をクリックして下さい。

1.4. 動作環境

MsysMDModbus.ocx をお使いいただくためには、以下のハードウェアとソフトウェアが必要です。

- ・ Windows2000,XP,Vista が正しくインストールされた DOS/V 互換パーソナルコンピュータ。
- ・ Visual Basic や Excel などの Windows 上で動作するソフトウェア

1.5. 免責事項

サンプルソースファイルのコーディング、その他の内容に関するご質問等のお問い合わせには対応いたしませんので、ご了承ください。また、本ソフトウェアは無償で提供されますが、本製品の保守、保証は行わないものとします。

1.6. 通信データのタグ名設定

Modbus デバイスとやりとりする全てのデータは、タグ名を付けて管理します。プログラム内では、そのタグ名をもとに簡単にデータの通信が行えます。このタグは、下図の専用アプリケーション (TagEditor.exe) で作成することができます。また、作成したタグ名の情報は CSV ファイルとして管理しますので、Excel を利用して作成&変更することもできます。

各部の説明は次ページをご参照ください。

ID	Tag Name	Comment	Address	Type	Length
1	D01	D0 Ch1	1	unsigned int	1
1	CJCSW	冷接点温度補償SW Ch1	33	unsigned int	1
1	InputReal1	アナログ入力工業単位値 Ch1	30017	float	2
1	InputReal2	アナログ入力工業単位値 Ch2	30019	float	2
1	InputReal3	アナログ入力工業単位値 Ch3	30021	float	2
1	InputReal4	アナログ入力工業単位値 Ch4	30023	float	2
1	CJCData	冷接点温度値	30049	float	2
1	InputType1	入出力タイプ番号 Ch1	40145	int	1
1	InputType2	入出力タイプ番号 Ch2	40146	int	1
1	InputType3	入出力タイプ番号 Ch3	40147	int	1
1	InputType4	入出力タイプ番号 Ch4	40148	int	1
1	Burnout	バーンアウトタイプ	40161	unsigned int	1
1	DI	トリガーSW入力	10001	int	1
1	AISatus1	チャンネル・ステータス	30081	int	1
1	ModelName	型式	30514	string	8
1	SerialName	機番	30522	string	8
1	HardVer	ハードウェアバージョン番号	30530	string	8
1	SoftVer	ファームウェアバージョン番号	30538	string	8

Tag Editor

ファイル(E)

Node ID (1~247)

Tag Name Prev. Next

Comment

Address

Type

Length

各データの情報を設定します。直接入力または値を選択します。

Copy Paste OK

ID	Tag Name	Comment	Address	Type	Length
1	D01	D0 Ch1	1	unsigned int	1
1	CJCSW	冷接点温度補償SW Ch1	33	unsigned int	1
1	InputReal1	アナログ入力工業単位値 Ch1	30017	float	2
1	InputReal2	アナログ入力工業単位値 Ch2	30019	float	2
1	InputReal3	アナログ入力工業単位値 Ch3	30021	float	2
1	InputReal4	アナログ入力工業単位値 Ch4	30023	float	2
1	CJCData	冷接点温度値	30049	float	2
1	InputType1	入出力タイプ番号 Ch1	40145	int	1
1	InputType2	入出力タイプ番号 Ch2	40146	int	1
1	InputType3	入出力タイプ番号 Ch3	40147	int	1
1	InputType4	入出力タイプ番号 Ch4	40148	int	1
1	Burnout	バーニアウトタイプ	40161	unsigned int	1
1	DI	トリガー-SW入力	10001	int	1
1	AISatus1	チャンネル・ステータス	30081	int	1
1	ModelName	型式	30514	string	8
1	SerialName	機番	30522	string	8
1	HardVer	ハードウェアバージョン番号	30530	string	8
1	SoftVer	ファームウェアバージョン番号	30538	string	8

データ情報をコピー
／ペーストします。

設定内容を登録
します。

「ファイル」メニューの説明

項目	内容
新規作成	新規登録画面を表示します。
開く	指定した Tag Editor ファイルを読み出し、内容を表示します。
上書き保存	登録した内容を現在編集中の CSV 形式ファイルに上書き保存します。
名前を付けて保存	登録した内容を CSV 形式ファイルでファイル名を指定して保存します。
閉じる	登録した全ての内容を削除します。
終了	Tag Editor を終了します。

1.7. データログ機能

プログラム内にログ開始メソッドと終了メソッドを記述するだけで、その期間の通信データの記録をファイルに管理します。通信時間に関するデータ（msec 単位）とエラーチェックによる要因も記録に残します。

2. Msystem MD Modbus メソッド説明

2.1. 通信

Modbus デバイスと通信を行うメソッドです。

2.1.1. Modbus_Serial_Connect()

Integer Modbus_Serial_Connect(Long MyHandle, Integer CommPort,
Integer BaudRate, Integer Parity, Integer Timeout)

戻り値 通信接続時に得たステータスコード（別表参照）を返します。

引数 MyHandle : 接続時のハンドル番号を受け取ります。
 CommPort : Comm ポート番号 (1,2,3,..) を指定します。
 BaudRate : ボーレート (9600,19200,38400) を指定します。
 Parity : パリティ (0:None、1:Odd、2:Even) を指定します。
 Timeout : Modbus ドライバーがデータを送った後にデバイスからレスポンスを待つ時間 (単位: ミリ秒) を指定します。

解説 Modbus Serial 通信を接続します。

例 Dim MyHandle As Long MsysMDModbus1.Modbus_Serial_Connect(MyHandle, 1, 38400, 1, 2000)
--

2.1.2. Modbus_Serial_Disconnect()

Integer Modbus_Serial_Disconnect(Long MyHandle)

戻り値 通信切断時に得たステータスコードを返します。

引数 MyHandle : 切断する通信のハンドル番号を指定します。

解説 Modbus Serial 通信を切断する

例 MsysMDModbus1.Modbus_Serial_Disconnect(MyHandle,)
--

2.1.3. Modbus_TCP_Connect()

Long Modbus_TCP_Connect(Long MyHandle, String IPAddress, Integer Timeout)

戻り値 接続時のハンドル番号を返します。接続エラー時は -1 を返します。

引数 IPAddress : IP アドレス (“xxx.xxx.xxx.xxx”) を指定します。
 Timeout : Modbus ドライバーがデータを送った後にデバイスからレスポンスを待つ時間（単位：ミリ秒）を指定します。

解説 Modbus TCP/IP 通信を接続します。

例 Dim MyHandle As Long MyHandle = MsysMDModbus1.Modbus_TCP_Connect(“192.168.10.2”, 2000)

2.1.4. Modbus_TCP_Disconnect()

Long Modbus_TCP_Disconnect(MyHandle As Long) As

戻り値 通信切断時に得たステータスコードを返します。

引数 MyHandle : 切断する通信のハンドル番号を指定します。

解説 Modbus TCP/IP 通信を切断する

例 MsysMDModbus1.Modbus_TCP_Disconnect(MyHandle)
--

2.1.5. Modbus_ReadVal()

Integer Modbus_ReadVal (Long MyHandle, Integer Slave, String Tag, Variant GetDat)

戻り値 データ読み出し時に得たステータスコードを返します。

引数 MyHandle : 通信のハンドル番号を指定します。
 Slave : スレーブアドレス (1～247 の整数) を指定します。
 Tag : タグ名を指定します。
 GetDat : 指定したタグ名のデータを受け取ります。

解説 指定したタグ名のデータを読み出します。

```
例      Dim    GetDat As Variant  
         MsysMDModbus1. Modbus_ReadVal (MyHandle, 1, "InputType1", GetDat)  
         Label1.Caption = CStr(GetDat)
```

2.1.6. Modbus_ReadVals()

Integer Modbus_ReadVal (Long MyHandle, Integer Slave, String Tag, Integer Count, Variant GetDat())

戻り値 データ読み出し時に得たステータスコードを返します。

引数 MyHandle : 通信のハンドル番号を指定します。
 Slave : スレーブアドレス (1～247 の整数) を指定します。
 Tag : タグ名を指定します。
 Count : 読み出すデータの数を指定します。(2～32 の整数)
 GetDat() : 指定したタグ名から Count で指定した数のタグ名分のデータを受け取ります。

解説 指定したタグ名の Modbus アドレスから指定した数のタグ名分のデータを配列で読み出します。

注 読み出す全ての情報は、指定したタグ名のデータタイプと同じデータタイプのものにして下さい。

```
例      Dim    GetDat(16) As Variant  
         MsysMDModbus1. Modbus_ReadVals (MyHandle, 1, "InputType1", 16, GetDat())
```


2.1.7. Modbus_WriteVal0

Integer Modbus_WriteVal (Long MyHandle, Integer Slave, String Tag, Variant SetDat)

戻り値	データ書き込み時に得たステータスコードを返します。		
引数	MyHandle :	通信のハンドル番号を指定します。	
	Slave :	スレーブアドレス（1～247 の整数）を指定します。	
	Tag :	タグ名を指定します。	
	SetDat :	指定したタグ名のデータを指定します。	
解説	指定したタグ名のデータを書き込みます。		

例 MsysMDModbus1. Modbus_WriteVal (MyHandle, 1, "InputType1", 5)

2.1.8. Modbus_WriteVals0

Integer Modbus_WriteVals (Long MyHandle, Integer Slave, String Tag, Integer Count, Variant SetDat)

戻り値	データ書き込み時に得たステータスコードを返します。		
引数	MyHandle :	通信のハンドル番号を指定します。	
	Slave :	スレーブアドレス（1～247 の整数）を指定します。	
	Tag :	タグ名を指定します。	
	Count :	書き込むデータの数を指定します。（2～32 の整数）	
	SetDat0 :	指定したタグ名から Count で指定した数分のデータを指定します。	
解説	指定したタグ名から Modbus アドレス続きで、指定した数分の情報を書き込みます。		
	<div>注</div>	書き込む全ての情報は、指定したタグ名のデータタイプと同じデータタイプのものにして下さい。	

例 Dim SetDat(4) As Variant SetDat(0) = 5 SetDat(1) = 8 SetDat(2) = 2 SetDat(3) = 7 MsysMDModbus1. Modbus_WriteVals (MyHandle, 1, 4, "InputType1", SetDat)
--

2.2. タグ情報

タグ名の情報を設定するメソッドです。

2.2.1. Modbus_SetTag()

Integer Modbus_SetTag(Long MyHandle, String FileName)

戻り値 正常なら 0、NG なら-1

引数 MyHandle : 通信のハンドル番号を指定します。
 FileName : タグ名の情報ファイル名を指定します。

解説 タグ名の情報ファイル名を格納します。
 ファイル名にパス名を付けることが出来ます。
 パス名が無いときは、カレントになります。
 通信接続直後、必ず実行するメソッドです。

例 MsysMDModbus1. Modbus_SetTag (MyHandle,"Tag_R1M5.csv")

2.3. 通信ログ

通信ログを開始／終了を設定するメソッドです。

2.3.1. Modbus_StartLog()

Integer Modbus_StartLog(Long MyHandle, String FileName)

戻り値 正常なら 0、NG なら-1

引数 MyHandle : 通信のハンドル番号を指定します。
 FileName : ログファイル名を指定します。

解説 ログを開始します。指定したファイル名にログを残します。
 ファイル名にパス名を付けることが出来ます。
 パス名が無いときは、カレントに出力します。
 出力されるファイル名は、FileName.csv になります。

例 MsysMDModbus1. Modbus_StartLog(MyHandle,"Log_R1M5")

2.3.2. Modbus_StopLog()

Integer Modbus_StopLog(Long MyHandle)

戻り値 正常なら 0、NG なら-1

引数 MyHandle : 通信のハンドル番号を指定します。

解説 ログを終了します。

例 MsysMDModbus1. Modbus_StopLog(MyHandle)

2.4. ステータスコード

エラーコード	内容
0	正常
N < 255	Slave Device Exception Response N = Exception Value
256	Invalid Connection Handle
257	Message Overrun
258	Invalid Point Address
259	Invalid Slave Node Address
260	Invalid Length
261	Unsupported Modbus Command
263	Slave Device Time-Out
264	Invalid Transmission Mode
265	Invalid CRC In Slave Response
266	Connection Not Established
267	Invalid Slave Response
271	Demo Time Expired
272	Invalid Modbus/TCP Command