

Modbus

維基百科，自由的百科全書

Modbus是一種串行通訊協定，是Modicon於1979年，為使用可編程邏輯控制器（PLC）通訊而發表的。Modbus是工業領域通訊協定的業界標準（*De facto*），並且現在是工業電子裝置之間相當常用的連線方式。^[1] Modbus比其他通訊協定使用的更廣泛的主要原因有：

1. 公開發表並且無版稅要求
2. 相對容易的工業網路部署
3. 對供應商來說，修改移動原生的位元或位元組沒有很多限制

Modbus允許多個（大約240個）裝置連線在同一個網路上進行通訊，舉個例子，一個由測量溫度和濕度的裝置，並且將結果傳送給電腦。在資料採集與監視控制系統（SCADA）中，Modbus通常用來連線監控電腦和遠端終端控制系統（RTU）。

目錄

- 1 協定版本
- 2 通訊和裝置
- 3 實作
- 4 限制
- 5 參考文獻
- 6 外部連結

協定版本

Modbus協定目前存在用於串列埠、乙太網以及其他支援網際網路協定的網路的版本。

大多數Modbus裝置通訊通過串列埠EIA-485實體層進行^[2]。

對於串行連線，存在兩個變種，它們在數值資料表示不同和協定細節上略有不同。Modbus RTU是一種緊湊的，採用二進位表示資料的方式，Modbus ASCII是一種人類可讀的，冗長的表示方式。這兩個變種都使用串行通訊（**serial communication**）方式。RTU格式後續的命令／資料帶有迴圈冗餘校驗的校驗和，而ASCII格式採用縱向冗餘校驗的校驗和。被配置為RTU變種的節點不會和設定為ASCII變種的節點通訊，反之亦然。

對於通過TCP/IP（例如乙太網）的連線，存在多個Modbus/TCP變種，這種方式不需要校驗和的計算。

對於所有的這三種通訊協定在資料模型和功能呼叫上都是相同的，只有封裝方式是不同的。

Modbus有一個擴充功能版本Modbus Plus（Modbus+或者MB+），不過此協定是Modicon專有的，和Modbus不同。它需要一個專門的協處理器來處理類似HDLC的高速令牌旋轉。它使用1Mbit/s的雙絞線，並且每個節點都有轉換隔離裝置，是一種採用轉換／邊緣觸發而不是電壓／水平觸發的裝置。連線Modbus Plus到電腦需要特別的介面，通常是支援ISA（SA85），PCI或者PCMCIA匯流排的板卡。

通訊和裝置

Modbus協定是一個master/slave架構的協議。有一個節點是master節點，其他使用Modbus協定參與通訊的節點是slave節點。每一個slave裝置都有一個唯一的位址。在串行和MB+網路中，只有被指定為主節點的節點可以啟動一個命令（在乙太網上，任何一個裝置都能傳送一個Modbus命令，但是通常也只有一個主節點裝置啟動指令）。

一個**ModBus**命令包含了打算執行的裝置的**Modbus**位址。所有裝置都會收到命令，但只有指定位置的裝置會執行及回應指令（位址**o**例外，指定位址**o**的指令是廣播指令，所有收到指令的裝置都會執行，不過不回應指令）。所有的**Modbus**命令包含了檢查碼，以確定到達的命令沒有被破壞。基本的**ModBus**命令能指令一個**RTU**改變它的暫存器的某個值，控制或者讀取一個**I/O**埠，以及指揮裝置回送一個或者多個其暫存器中的資料。

有許多**modems**和開道器支援**Modbus**協定，因為**Modbus**協定很簡單而且容易複製。它們當中一些為這個協定特別設計的。有使用有線、無線通訊甚至簡訊和**GPRS**的不同實作。不過設計者需要克服一些包括高延遲和時序的問題。

實作

幾乎所有的實作都是官方標準的某種變體。不同的供應商裝置之間可能無法正確的通訊。一些主要的變化有：

- 資料類型
 - **IEEE**標準的浮點數
 - **32**位元整型數
 - **8**位元資料
 - 混合資料類型
 - 整數中的位域
 - multipliers to change data to/from integer. 10, 100, 1000, 256 ...
- 協定擴充功能
 - **16**位元的從站位址
 - **32**位元的資料大小（**1**個位址 = 返回**32**位元資料）
 - 字交換資料

限制

- **Modbus**是在**1970**年末為可編程邏輯控制器通訊開發的，這些有限的資料類型在那個時代是可以被**PLC**理解的，大型二進位物件資料是不支援的。
- 對節點而言，沒有一個標準的方法找到資料物件的描述資訊，舉個例子，確定一個暫存器資料是否表示一個介於**30-175**度之間的溫度。
- 由於**Modbus**是一個主／從協定，沒有辦法要求裝置「報告異常」（構建在乙太網的**TCP/IP**協定之上，被稱為**open-mpbus**除外）- 主節點必須迴圈的詢問每個節點裝置，並尋找資料中的變化。在頻寬可能比較寶貴的應用中，這種方式在應用中消耗頻寬和網路時間，例如在低速率的無線鏈路上。
- **Modbus**在一個資料鏈路上只能處理**247**個位址，這種情況限制了可以連線到主控站點的裝置數量（再一次指出乙太網**TCP/IP**除外）
- **Modbus**傳輸在遠端通訊裝置之間緩衝資料的方式進行，有對通訊一定是連續的限制，避免了傳輸中的緩衝區漏洞的問題
- **Modbus**協議自身提供針對未經授權的命令或截取資料沒有安全性。^[3]

參考文獻

- ↑ Drury, Bill. Control Techniques Drives and Controls Handbook (PDF) 2nd. Institution of Engineering and Technology. 2009: 508–. （需要訂閱（說明））.
- ↑ [1] (http://www.obvius.com/documentation/faq/modbus.html)
- ↑ Palmer; Shenoj, Sujeet (編). Critical Infrastructure Protection III. Third IFIP WG 11. 10 International Conference. Hanover, New Hampshire: Springer: 87. 23–25 March 2009. ISBN 3-642-04797-1.

外部連結

- 詳細協定說明 (<http://www.modbus.org/specs.php>)
- Modbus-IDA站點 (<http://www.modbus.org>)
- 你應該知道的Modbus (<http://www.modbusdriver.com/doc/libmbusmaster/modbus.html>)
- 從Java開發人員角度談協定解釋 (<http://jamod.sourceforge.net/kbase/protocol.html>)
- 從SCADA系統角度比較Modbus和DNP3 (http://www.trianglemicroworks.com/documents/Modbus_and_DNP_Comparison.pdf)
- Modbus到WebServices/SOAP整合 (<http://www.gridlogix.com>)
- Modbus到Profibus-DP (<http://www.sibotech.net>)

開源軟體

- 一個基於GNU/Linux C開發的開源Modbus庫 (<http://libmodbus.org/>)
- 為微控制器開發的免費的Modbus ASCII/RTU和TCP軟體 (<http://www.freemodbus.org/index.php?lang=cn>)。
- C開發。新站點位置<http://www.freemodbus.org> (<http://www.freemodbus.org/>)。商業支援版本在<http://www.embedded-solutions.at> (<http://www.embedded-solutions.at/>)。
- NModbus (<http://NModbus.com>)是一個有例子的C#實作版本
- Perl語言開發的Protocol::Modbus (<http://search.cpan.org/~cosimo/Protocol-Modbus-0.04/lib/Protocol/Modbus.pm>)
- Perl語言開發的Modbus::Client (<http://search.cpan.org/~dvklein/Modbus-Client-1.03/lib/Modbus/Client.pm>)
- Ruby開發的Modbus master (<http://www.messen-und-deuten.de/modbus.html>)。公共領域的範例代碼，可以很容易的用其他手稿語言重新開發。

取自 "https://zh.wikipedia.org/w/index.php?title=Modbus&oldid=37372777"

-
- 本頁面最後修訂於2015年9月28日 (週一) 05:50。
 - 本站的全部文字在創用CC 姓名標示-相同方式分享 3.0 協議之條款下提供，附加條款亦可能應用（請參閱使用條款）。Wikipedia®和維基百科標誌是維基媒體基金會的註冊商標；維基™是維基媒體基金會的商標。維基媒體基金會是在美國佛羅里達州登記的501(c)(3)免稅、非營利、慈善機構。