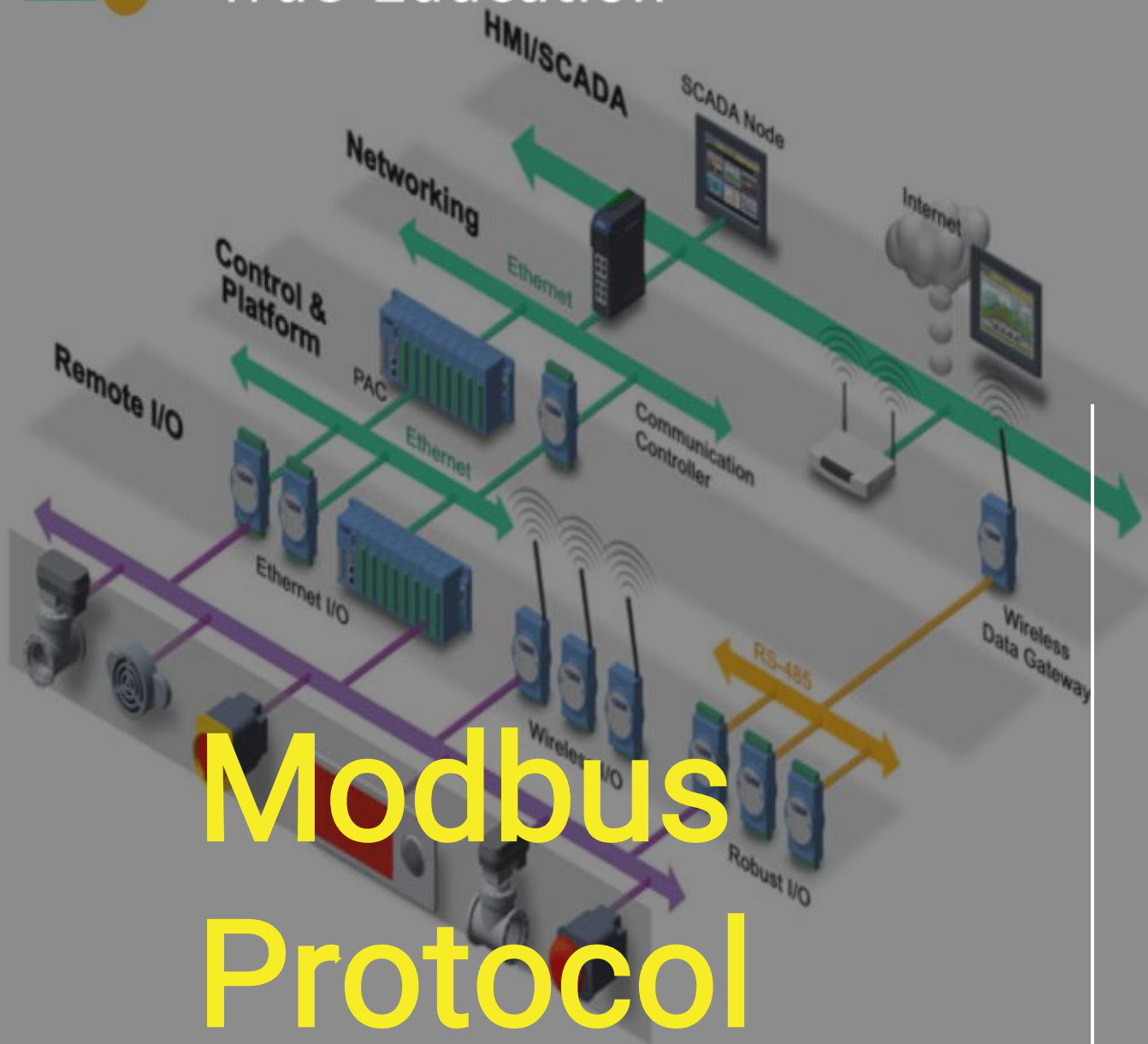




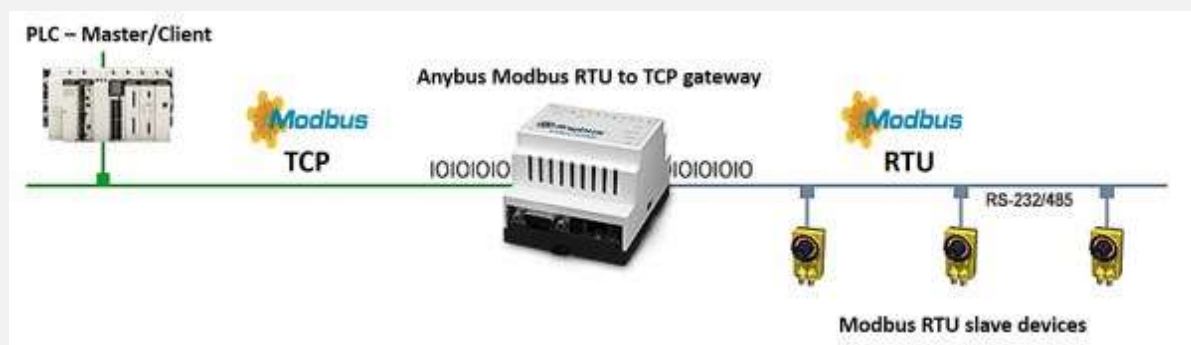
فهرست مطالب

۱. پروتکل مدباس یکی از پروتکل های باز (Open Protocol)
۲. پروتکل مدباس چیست؟
۳. انواع پروتکل مدباس
۴. پروتکل مدباس و رسانه های فیزیکی آن (Physical Media)
۵. ساختار پیام مدباس
۶. پردازش خطای پیام های پروتکل مدباس
۷. رایج ترین کدهای عملکرد مربوط به پروتکل مدباس RTU
۸. شبیه سازی پروتکل مدباس
۹. انواع دستگاه های Slave

برقراری ارتباط بین دو دستگاه یکی از مهمترین ارکان در صنایع مختلف است و چندین روش متفاوت دارد که یکی از پرکاربردترین روش ها پروتکل مدباس است.

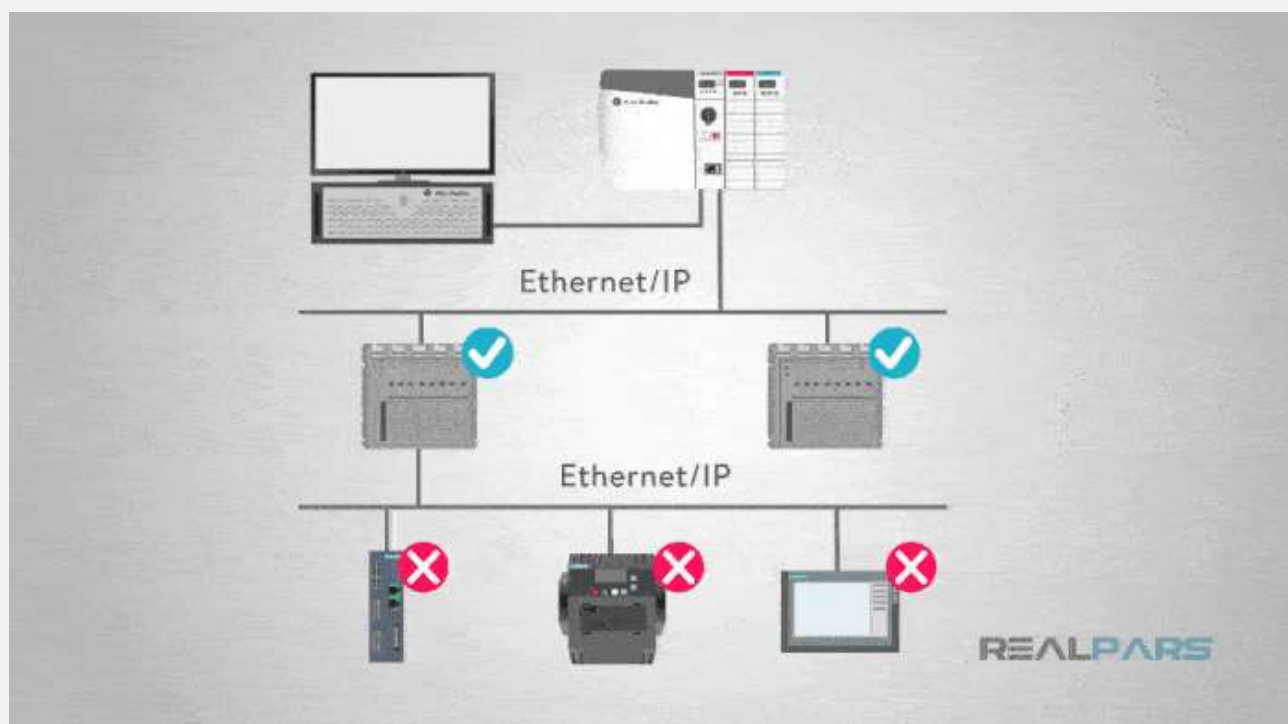
آشنایی با این پروتکل مهم و محبوب در میان صنعت گران برای شروع کار در صنعت لازم است. در این مقاله با پروتکل ارتباطی مدباس، انواع و نحوه کار آن آشنا می شویم.

در این مقاله خواهیم دید که مدباس به عنوان نوعی از پروتکل های باز در زمینه اتوماسیون صنعتی چطور کار می کند و مزایای استفاده از این نوع پروتکل چیست که آن را به محبوب ترین پروتکل تبدیل کرده است.



پروتکل مدباس یکی از پروتکل های باز (Open Protocol)

بسته به سازنده تجهیزات اتوماسیون صنعتی، زبان خاصی مورد استفاده قرار می گیرد که ممکن است این زبان، به صورت پروتکل های باز در دسترس باشد. این پروتکل های باز امکان سازگار کردن و یکپارچه سازی محصولات موجود در بازار را، به تولیدکنندگان می دهد.



پروتکل های باز در واقع مشخصات منتشر شده ای هستند که هر کس می تواند از آن ها آزادانه یا با مجوز استفاده کند. پروتکل های باز معمولاً توسط جمعی از شرکت های سازنده، گروه های کاربر، جوامع حرفه ای و دولت ها،

استفاده می شوند و به کاربران این امکان را می دهند که از بین دستگاه ها و سیستم های مورد استفاده، انتخاب گسترده تری داشته باشند.

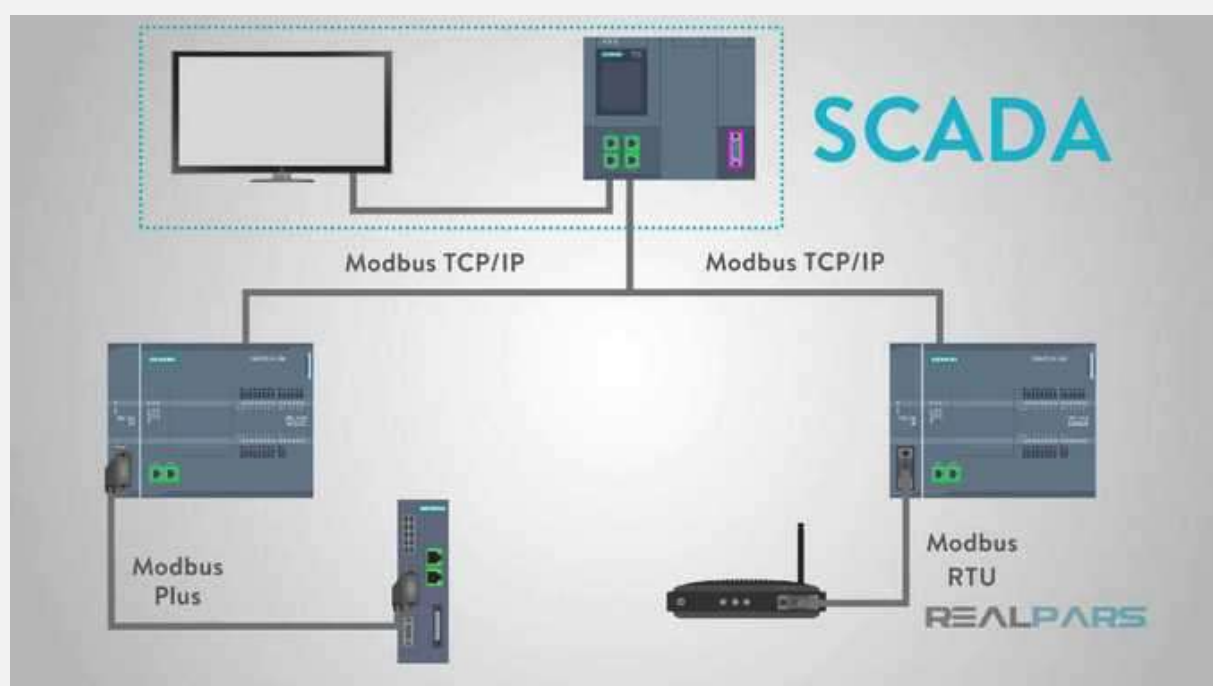
از مزایای پروتکل های باز می توان به این موارد اشاره کرد:

- پشتیبانی شدن توسط طیف وسیعی از تولید کننده ها، فروشندگان نرم افزار و مراکز نصب و سرویس
 - تایید و حمایت شدن از طرف گروه های فعال اجتماعی
 - محدود نبودن و امکان افزودن امکانات و توانایی های بیشتر در آینده
- همین مزایا مدباس را به رایج ترین پروتکل های اتوماسیون مورد استفاده در اتصال دستگاه های الکترونیکی، تبدیل کرده است.
- حال پروتکل ارتباطی مدباس و انواع مدباس مورد استفاده در صنعت را توضیح می دهیم.

پروتکل مدباس چیست؟

پروتکل ارتباطی مدباس، قدیمی ترین و محبوب ترین پروتکل اتوماسیون در زمینه اتوماسیون صنعتی و SCADA (نظارت بر کنترل و تهیه داده ها) است.

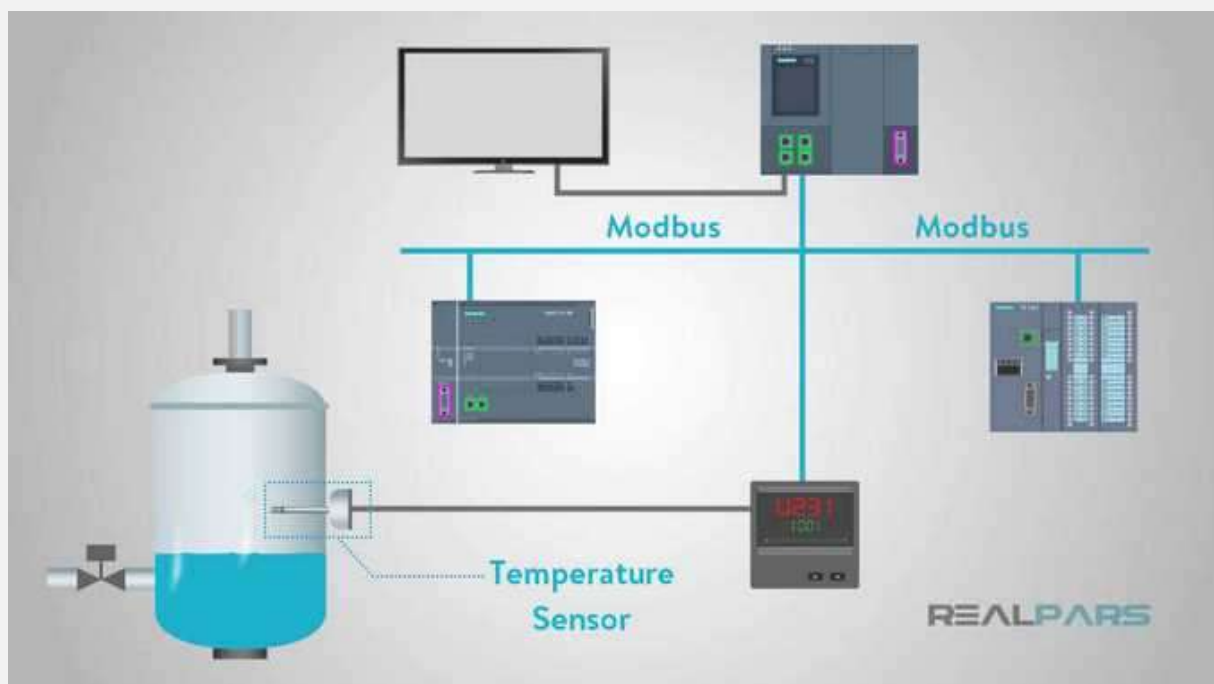
مدباس یک پروتکل ارتباط سریال است که در سال ۱۹۷۹ توسط مادیکن ([Modicon](#)) برای استفاده در PLC منتشر و در نهایت به طور غیر رسمی، تبدیل به پروتکل ارتباطی استاندارد برای اتصال دستگاه های الکترونیکی صنعتی شد. در حال حاضر شرکت مادیکن متعلق به اشنایدر الکتریک ([Schneider Electric](#)) است.



پروتکل ارتباطی مدباس همگام با کاربردهای صنعتی توسعه پیدا کرده و به صورت رایگان و با دسترسی آزاد عرضه شده است. مدباس راه اندازی و نگهداری ساده ای دارد و در کاربردهای متنوع مبتنی بر master-slave استفاده می شود. همچنین می تواند، دیتاهای خام را بدون اعمال محدودیت های زیادی بر تجهیزات، ارسال کند.

مدباس با استفاده از کامپیوترها و HMI ها دستگاه ها را مانیتور کرده و با برنامه ریزی، بین دستگاه های هوشمند و سنسورها یا ابزارها ارتباط برقرار می کند. دانستن نحوه ایجاد شبکه های مبتنی بر مدباس برای هر تکنسین و مهندس برقی که در این زمینه کار می کند ضروری است. توانایی یکپارچه سازی ابزارهای تولید شده توسط سازندگان مختلف، مهارتی است که بازار تقاضای بالایی دارد و بر ارزش و فروش کار شما اثرگذار است. در واقع پروتکل ارتباطی مدباس، یک زبان مشترک برای برقراری ارتباط بین دستگاه ها و تجهیزات فراهم می کند.

به عنوان مثال، با استفاده از مدباس، دستگاه ها در یک سیستم و تحت یک شبکه واحد می توانند، نتایج را به یک SCADA یا PLC انتقال داده و تحت همین شبکه دما و رطوبت را نیز اندازه گیری کنند.



توسعه و به روز رسانی پروتکل های مدباس توسط سازمان مدباس مدیریت می شود. این سازمان از کاربران و تامین کنندگان دستگاه های سازگار با مدباس، تشکیل شده است.

انواع پروتکل مدباس

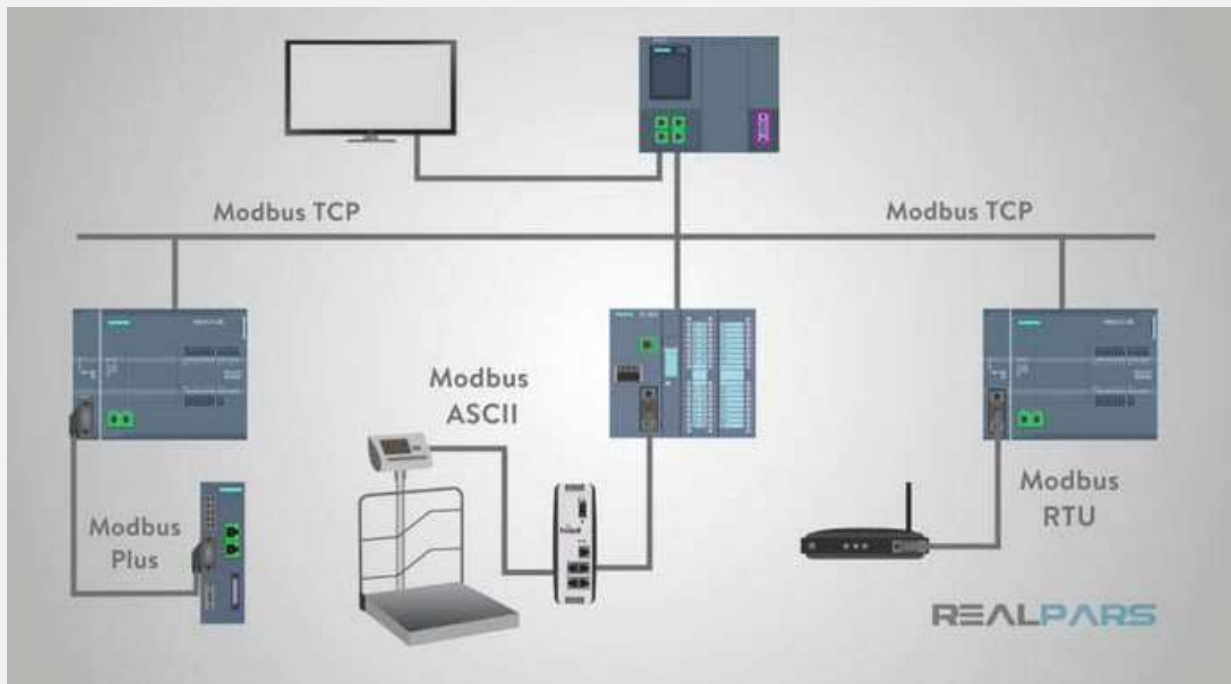
نسخه های متفاوتی از پروتکل ارتباطی مدباس برای پورت سریال (serial port) و اترنت (Ethernet) وجود دارد که رایج ترین آن ها عبارتند از:

• Modbus RTU

• Modbus ASCII

• Modbus TCP

• Modbus Plus



شرکت Modicon یک رابط مدباس (Modbus communication interface) را بر پایه (Master-Slave) منتشر کرد. در این مدل، ارتباط بین گره های مدباس (Modbus nodes) با ارسال درخواست و خواندن نوع پاسخ پیام ها به دست می آید.

پروتکل مدباس و رسانه های فیزیکی آن (Physical Media)

پروتکل ارتباطی مدباس یک استاندارد باز است که پیام موجود در هر گفتگو را توصیف می کند. مدباس بین انواع مختلف رسانه های فیزیکی ارتباط برقرار می کند.

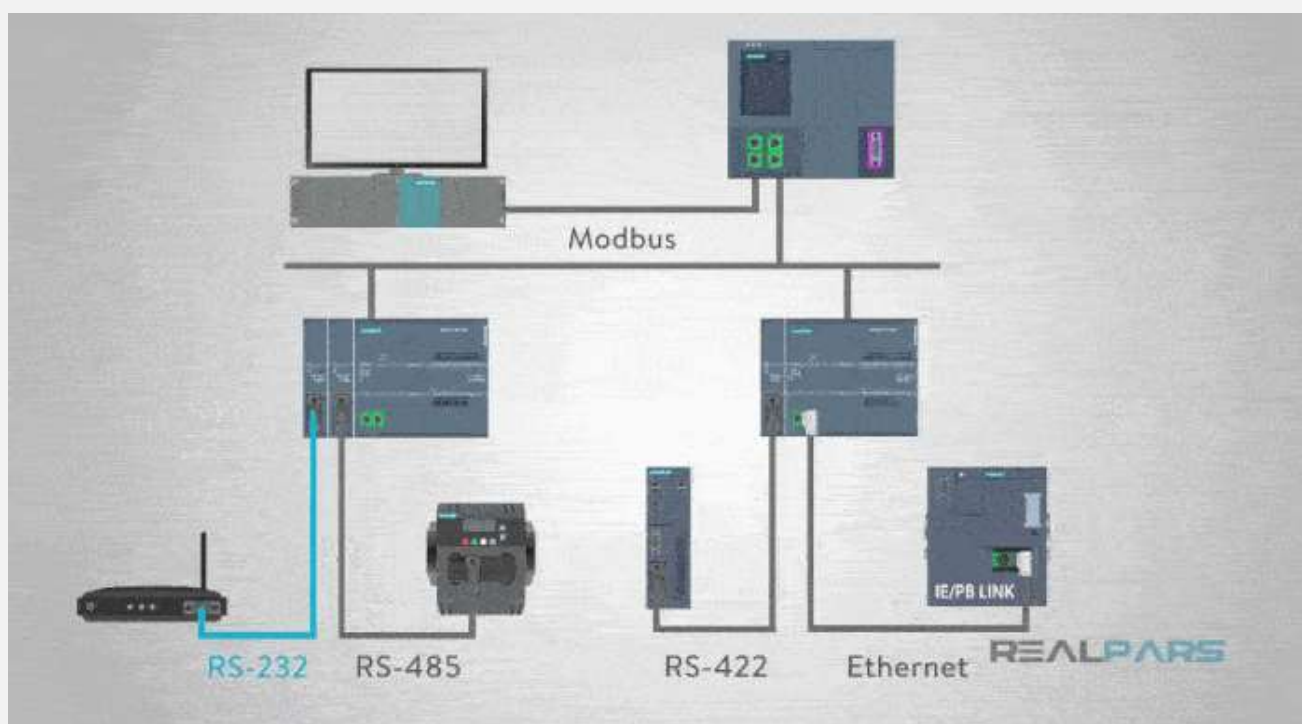
از انواع این رسانه های فیزیکی می توان موارد زیر را نام برد:

Serial RS-232 •

Serial RS-485 •

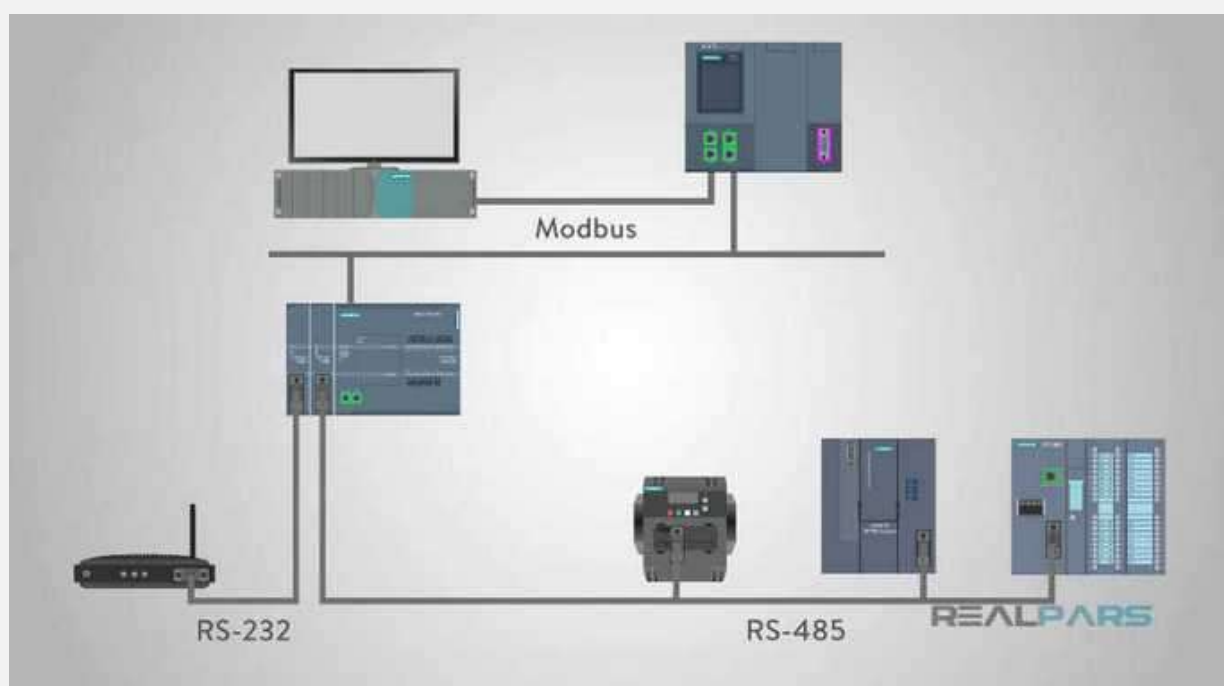
Serial RS-422 •

Ethernet •

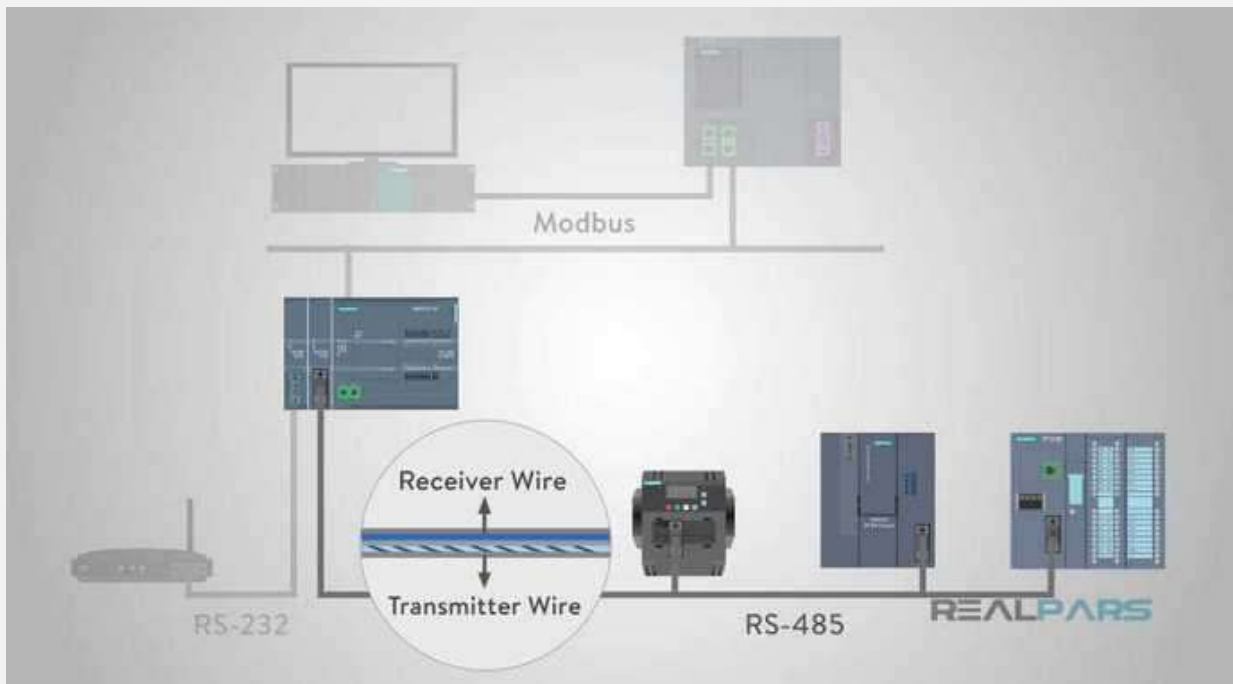


رابط اصلی مدباس بر روی ارتباطات سریال RS-232 اجرا می شود، اما بیشتر مدباس های اخیر از RS-485 استفاده می کنند زیرا این نوع، مزایای زیر را ایجاد می کند:

- فاصله های طولانی تر
- سرعت بالاتر
- امکان استفاده از دستگاه های چندگانه (multiple devices) در یک شبکه

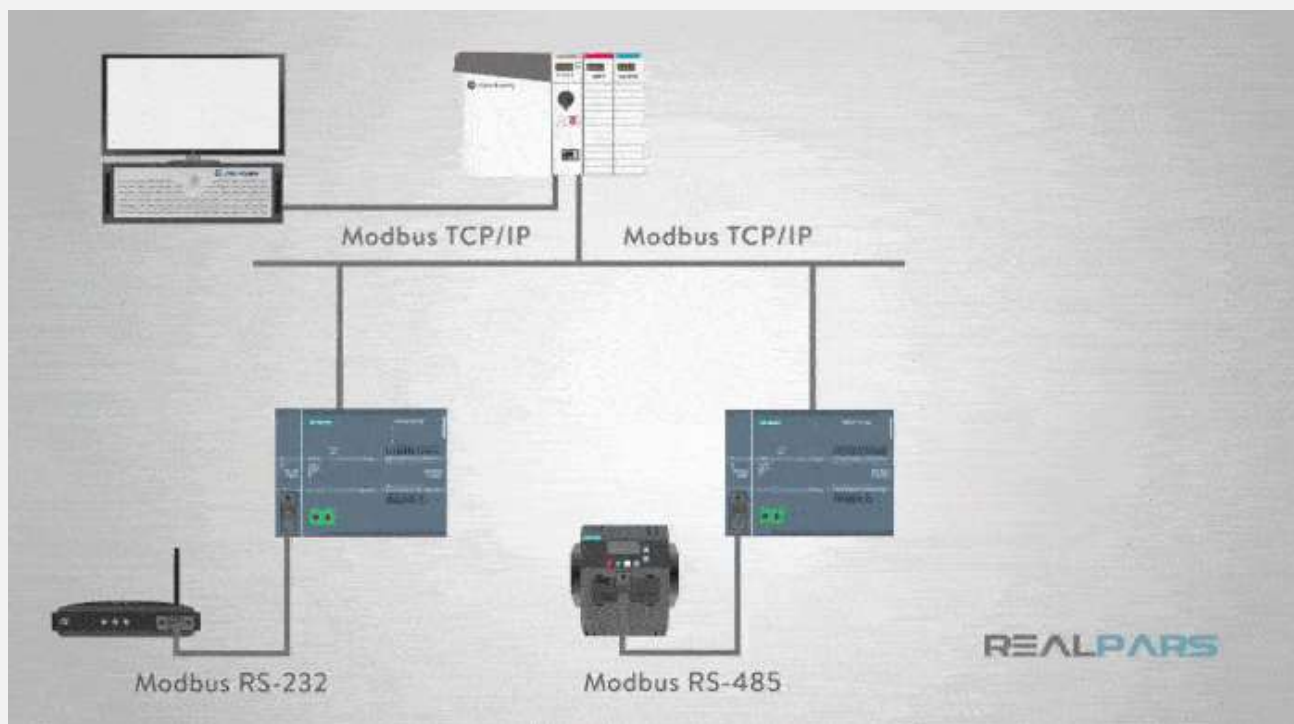


مدباس Master-Slave مبتنی بر RS-485 از اتصالات فرستنده و گیرنده دو سیمه (two-wire) استفاده می کند.



در رابط های ساده ای مانند RS232 یا RS485، پیام های مدباس با فرم ساده در شبکه ارسال می شوند و شبکه تنها به اتصال مدباس اختصاص داده می شود. با این حال، اگر شبکه شما به چندین دستگاه ناهمگن (multiple heterogeneous devices) نیاز داشته باشد که این دستگاه ها از یک شبکه چند منظوره مانند TCP / IP مبتنی بر اترنت استفاده می کنند، پیام های مدباس در بسته های اترنت با فرمت مورد نظر برای این رابط فیزیکی تعبیه شده است. بنابراین در این مورد، مدباس و سایر پروتکل های ترکیبی (mixed protocols)، می توانند هم زمان در یک رابط فیزیکی هماهنگ وجود داشته باشند. یعنی می توان گفت، به این دلیل که مدباس

از پروتکل های استاندارد TCP / IP استفاده می کند، پس با همه دستگاه های مجهز به استاندارد اترنت سازگار است.



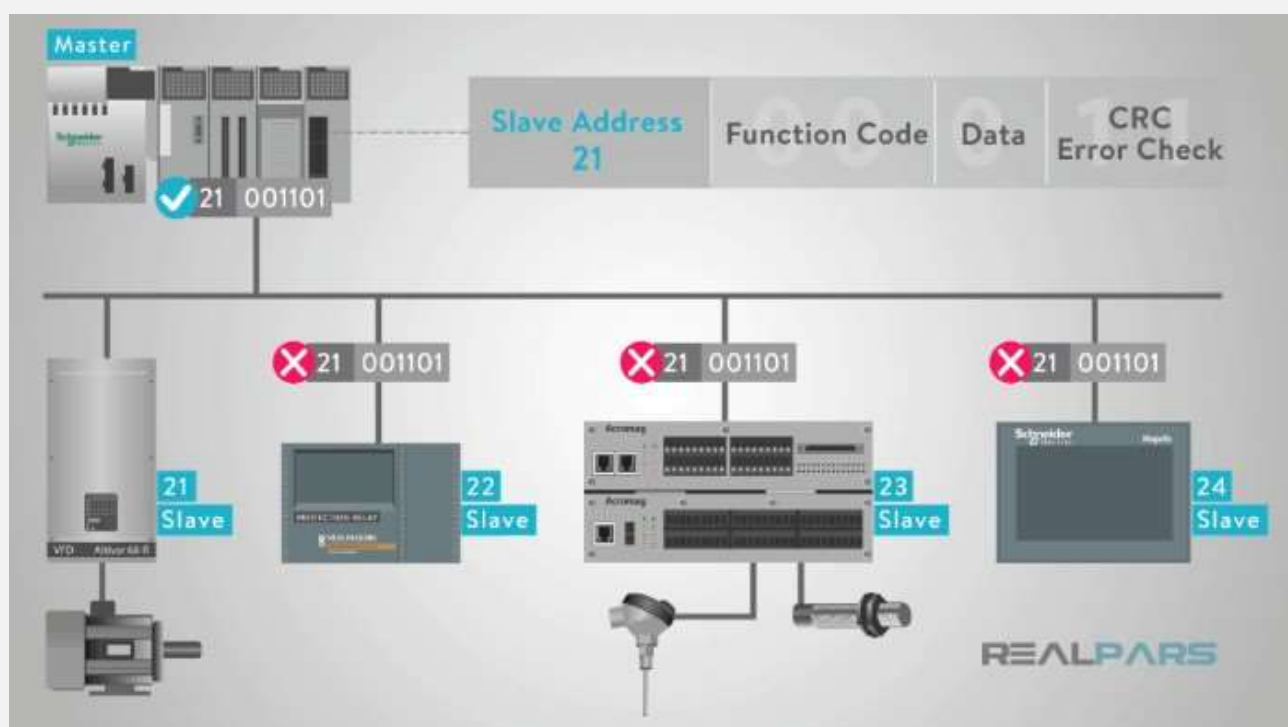
ساختار پیام مدباس

پیام های مدباس ساختار مشابهی دارند که از ۴ فاکتور اصلی تشکیل شده اند. نظم و ترتیب این فاکتورها در همه پیام ها یکسان است. همین مسئله موجب تجزیه آسان محتوای پیام های مدباس می شود.

در شبکه مدباس ، مکالمه همواره توسط یک master آغاز می شود. Master پیامی را ارسال می کند. با توجه به محتویات آن، slave پیام را تفسیر کرده و به آن پاسخ می دهد.

آدرس دهی فیزیکی slave در سرتیتر هر پیام، مشخص می کند که کدام دستگاه slave باید به پیام پاسخ دهد.

نودهای (nodes) دیگر موجود در شبکه مدباس در صورتی که فیلد آدرس با آدرس خودشان مطابقت نداشته باشد، این پیام را نادیده می گیرند.



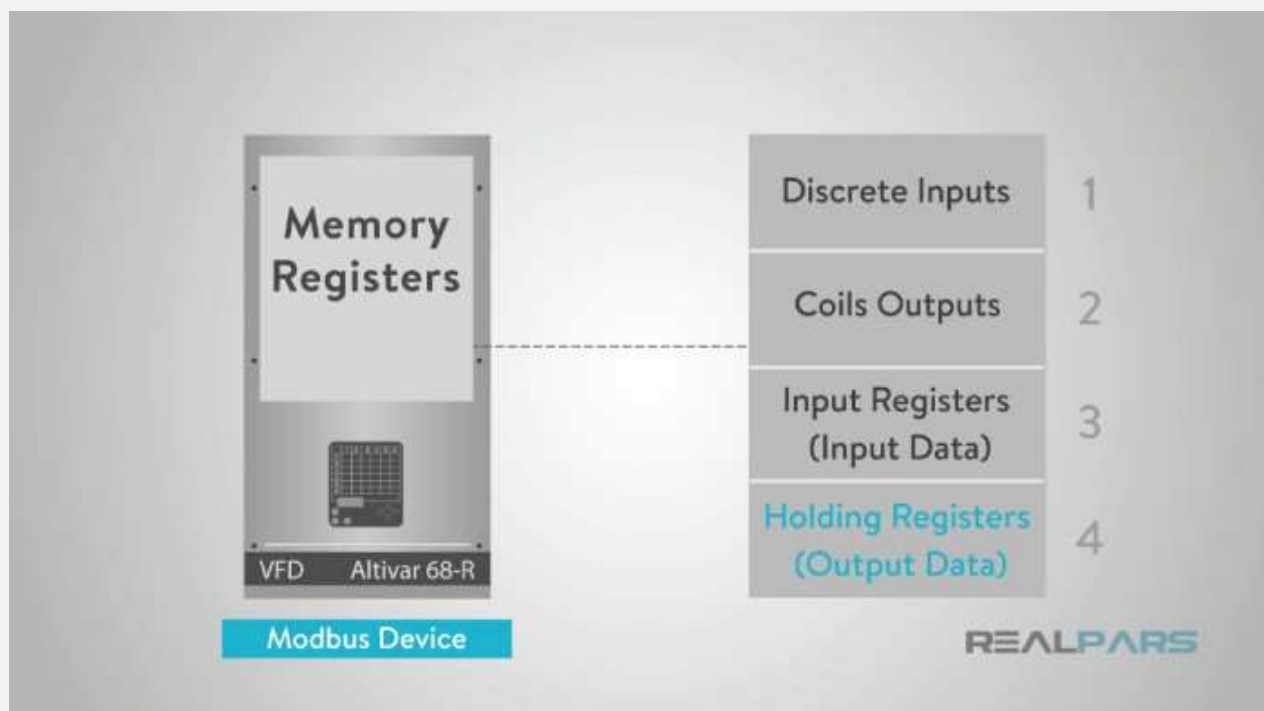
توابع پروتکل مدباس دستورات خواندن و نوشتن را برای حافظه داخلی slave اجرا می کنند تا پیکربندی، نظارت و کنترل ورودی ها و خروجی های slave را ثبت کند.

دستگاه های مبتنی بر پروتکل مدباس

دستگاه های مبتنی بر پروتکل مدباس به طور معمول، حاوی یک نقشه رجیستر کلی هستند. با استفاده از این نقشه، پیکربندی، ورودی و خروجی داده ها خوانده و نوشته می شوند. برای درک بهتر عملیات کلی باید به نقشه رجیستر slave مربوط به دستگاه مراجعه کرد.

مدل داده ها در پروتکل مدباس دارای یک ساختار ساده شامل چهار نوع داده اساسی است:

- ورودی های مجزا (Discrete Inputs)
- خروجی های کویل (سیم پیچ) (Coils Outputs)
- ورودی رجیستری یا داده های ورودی (Input Registers or Input Data)
- نگهدارنده رجیستری یا داده های خروجی (Holding Registers or Output Data)

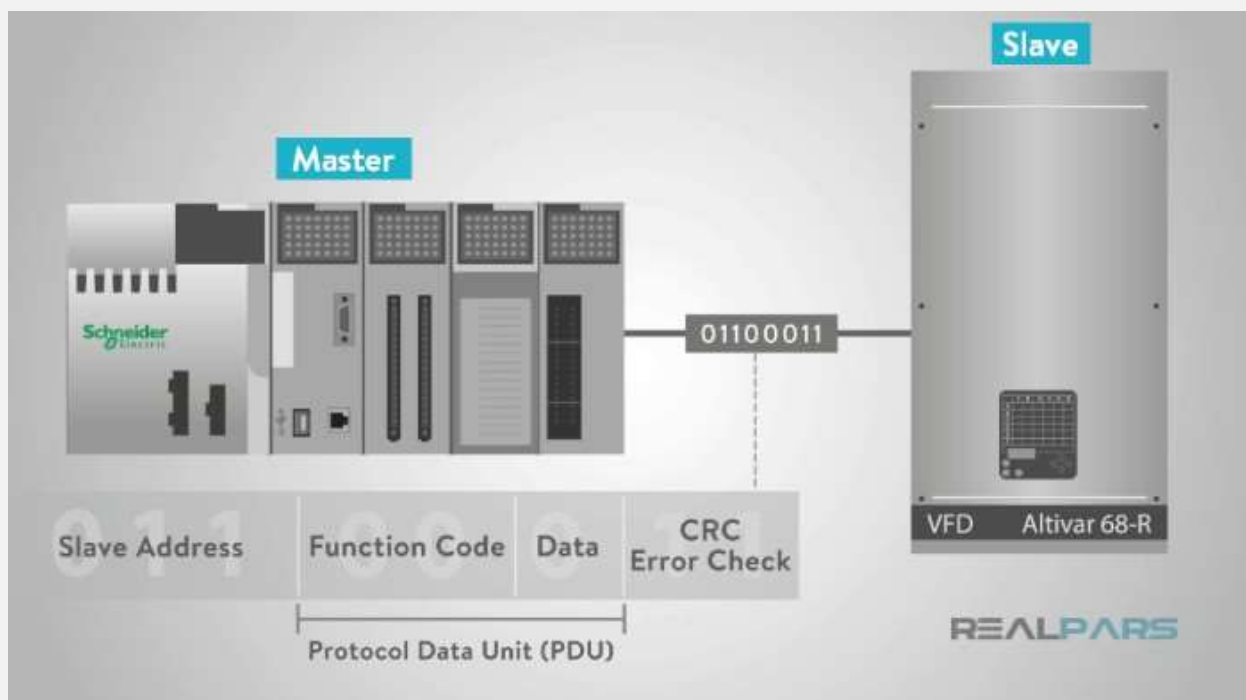


واحد داده در پروتکل مدباس (PDU)

محل درخواست خدمات در پیام یا واحد داده پروتکل مدباس یا PDU

متشکل است از:

- کد عملکرد یا کد تابع (Function code)
- تعداد داده های (data bytes) درخواست شده توسط master



حافظه رجیستری مدباس (memory registers) در هر دستگاه حول چهار نوع مرجع داده سازماندهی می شوند. این نوع داده ها بیشتر توسط عدد آغازین مورد استفاده در حافظه آدرس دستگاه (memory address)، شناخته می شوند. مانند:

• صفر (۰) موجود در رجیستر به معنی: خواندن و یا نوشتن «خروجی های مجزا یا کویل». «

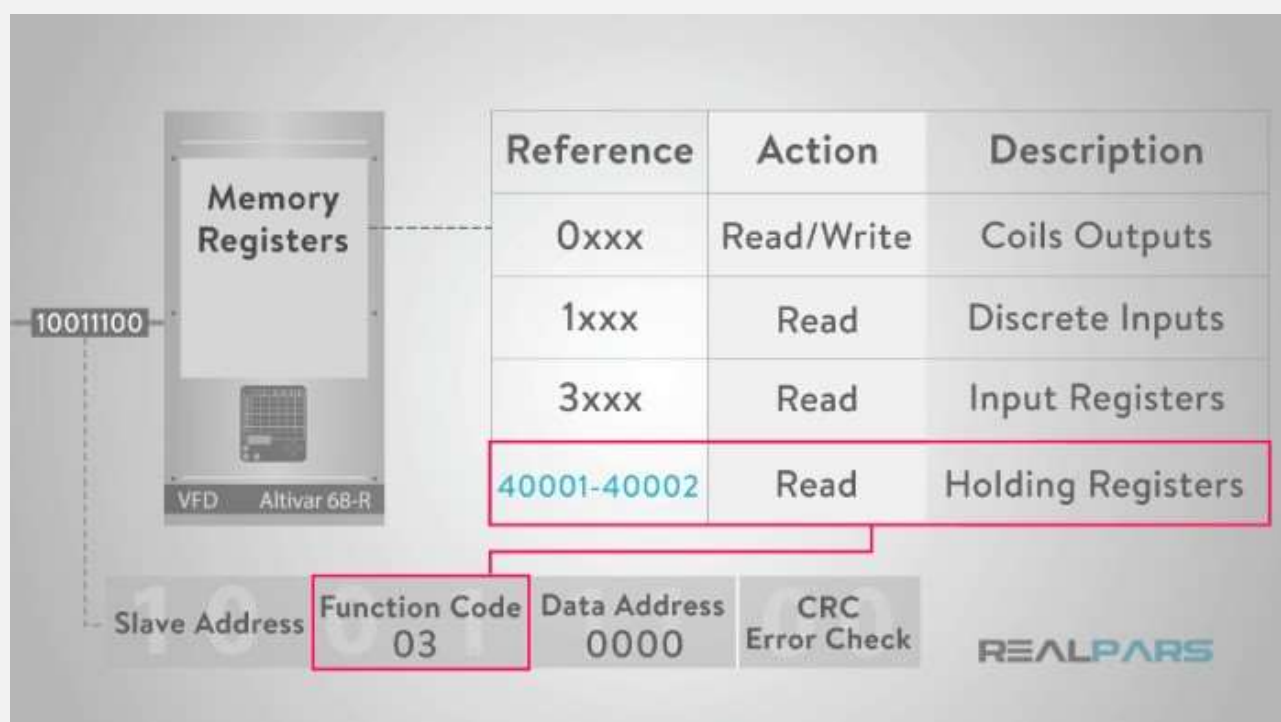
• یک (۱) موجود در رجیستر به معنی: خواندن «ورودی های مجزا». «

• سه (۳) موجود در رجیستر به معنی: خواندن «ورودی رجیستری». «

• چهار (۴) موجود در رجیستر به معنی: خواندن یا نوشتن بر روی « داده خروجی و نگهدارنده رجیستری». «

فیلد « کد عملکرد پیام مدباس » (Function Code Field)

«فیلد کد عملکرد» مشخص میکند که با استفاده از کدام گروه داده های رجیستری از slave، آن را می خواند یا می نویسد. به عنوان مثال، یک کد عملکرد ۰۳، که نگهدارنده رجیستری ۴۰۰۰۱-۴۰۰۰۲ را می خواند، به عنوان داده های رجیستری ۰۰۰۰ آدرس دهی می شود که در فیلد آدرس داده ها در پیام ارسالی به slave قرار می گیرد. کد عملکرد ۰۳ بر روی موجودی رجیستری نوع (4xxxx) در نقشه داده های slave کار می کند، زیرا درخواست نوع عملیات داده ها نگهدارنده رجیستری مورد استفاده را مشخص می کند و بر آدرس دهی (4xxxx) دلالت دارد.



فیلدهای PDU به بایت ها تقسیم شده و براساس عنوان فیلد گروه بندی می شوند.

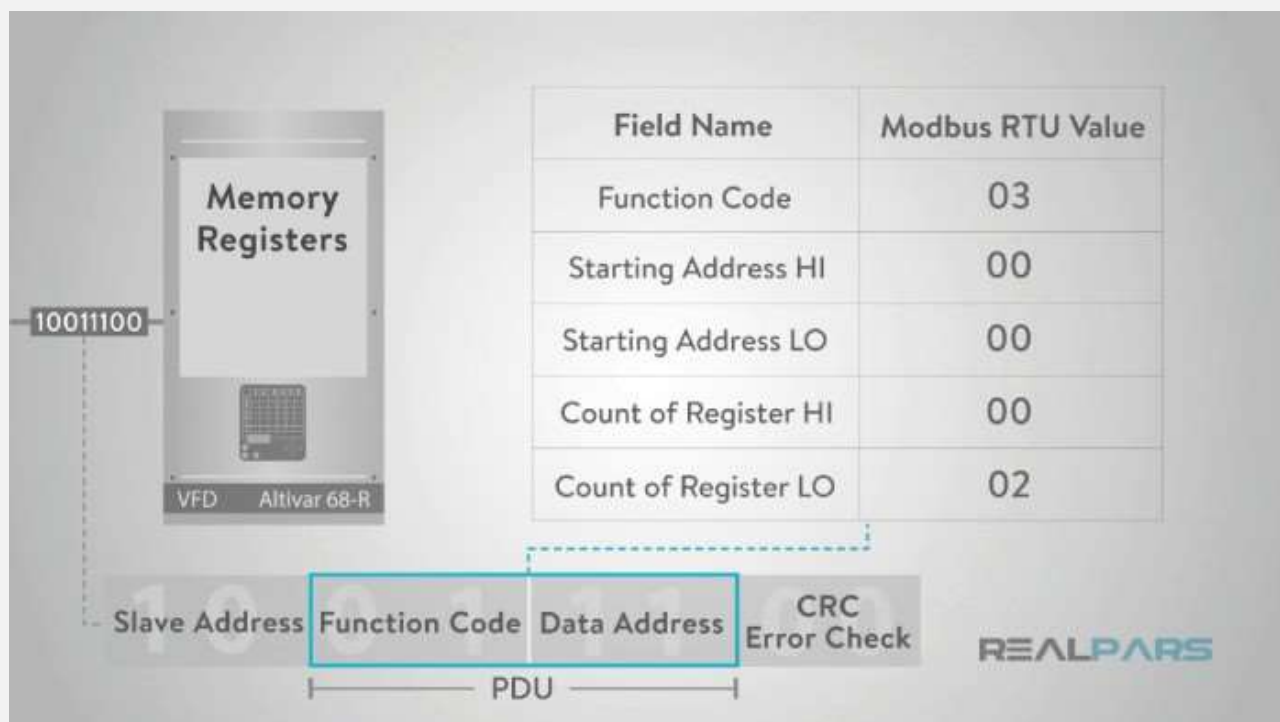
پیام درخواست شامل:

• «آدرس ۰۱» slave

• «کد عملکرد ۰۳» (Function code) «خواندن از نگهدارنده رجیستری های ۴xxxx»

• «دستور شروع (starting address)» همان بایت های HI و LO یا (۰۰۰۰) هستند که شروع رجیستر را مشخص می کنند.

• «تعداد آدرس هایی (count number of addresses)» که از slave خوانده می شود، معادل است با داده های رجیستری HI و LO از ارزش تعدادی (۰۰۰۲) و مشخص کننده «تعداد رجیسترهایی» است که از slave خوانده خواهند شد.



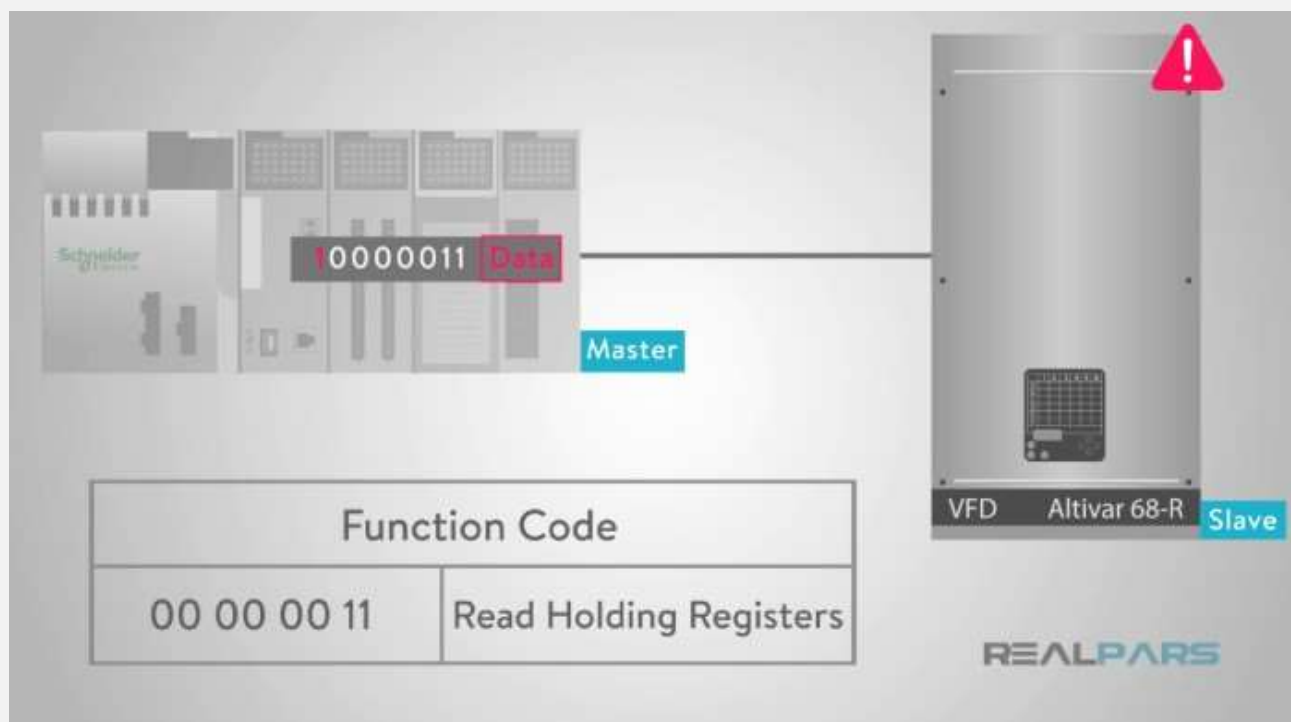
فیلد «داده» در پیام مدباس (Data Field)

در فیلد درخواست «داده» master، اطلاعات اضافی مورد نیاز slave برای تکمیل هر عملیاتی که با «کد عملکرد» مشخص شده، ارائه می شود. درخواست های master معمولاً شامل موارد زیر است:

- آدرس نقشه رجیستری slave. (map register address)
- تعداد رجیسترها برای ارائه در درخواست
- هر دستور نوشتن «داده» از master. (write data)

پردازش خطای پیام های پروتکل مدباس

پاسخ عادی slave، به سادگی کد عملکرد اصلی درخواست را بازتاب می کند. اما پاسخ خطای slave، کد معادل کد عملکرد اصلی را با استفاده از مهمترین بیت متعلق به (logic 1)، باز می گرداند. به عنوان مثال، دستور «خواندن موجودی رجیستری» (کد عملکرد ۰۳) دارای کد عملکرد ۱ بایستی حاوی ۸ بیت دوتایی (03 Hex) 0011 0000 است. اگر دستگاه slave درخواست را بدون خطا قبول کند، همان کد را در پاسخ خود باز می گرداند. با این حال، اگر خطایی رخ دهد، slave یک بایت حاوی ۸ بیت دوتایی ۰۰۱۱ ۱۰۰۰ (مهمترین بیت متعلق به منطق ۱) در فیلد «کد عملکرد» باز می گرداند. همچنین یک کد منحصر به فرد (کد حذف) را در فیلد داده ها در پاسخ اضافه می کند که به دستگاه slave نوع خطای رخ داده و یا علت خطا را نشان می دهد.



رایج ترین کدهای عملکرد مربوط به پروتکل مدباس RTU

کد عملکرد ۰۱، «کد خواندن کوئل» (Read Coils)

برای خواندن رجیستری های پیوسته ۱ تا ۲۰۰۰ مربوط به وضعیت کوئل ها در یک دستگاه slave استفاده می شود. درخواست PDU اولین آدرس کوئل مربوط به حافظه رجیستری های slave و تعداد کوئل هایی که باید از دستگاه slave خوانده شوند را، مشخص می کند.

کد عملکرد ۰۲، «کد خواندن داده های گسسته» (Read Discrete Input)

برای خواندن حالت های متناوب ۱ تا ۲۰۰۰ از خروجی های گسسته مربوط به یک slave راه دور استفاده می شود. درخواست PDU اولین آدرس ورودی مربوط به حافظه رجیستری های slave و تعداد ورودی هایی که باید از دستگاه slave خوانده شوند را، مشخص می کند.

کد عملکرد ۰۳، «کد خواندن نگهدارنده رجیستری» (Read Holding Registers)

برای خواندن محتویات یک بلوک متوالی از نگهدارنده رجیستری مربوط به یک slave راه دور استفاده می شود. درخواست PDU شروع آدرس رجیستری و تعداد رجیسترهایی که باید از دستگاه slave خوانده شوند را، مشخص می کند.

کد عملکرد ۰۴، «کد خواندن رجیستری های ورودی» (Read Input Registers)

برای خواندن ۱ تا ۱۲۵ از رجیستری های ورودی پیوسته در یک دستگاه راه دور استفاده می شود. درخواست PDU شروع آدرس رجیستری و تعداد رجیسترها را، مشخص می کند.

کد عملکرد ۰۵، «کد خواندن تک کوئل» (Read Input Registers)

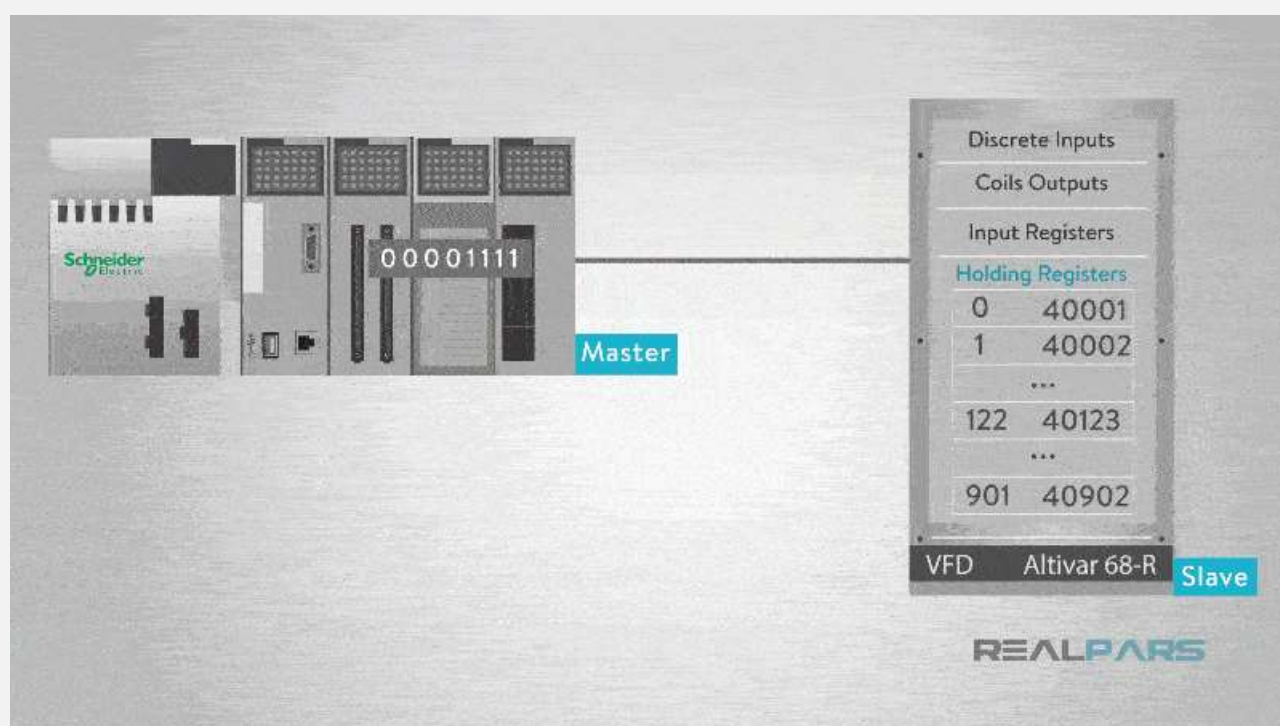
برای نوشتن یک خروجی یکتا به صورت ON یا OFF در یک دستگاه slave راه دور استفاده می شود.

کد عملکرد ۰۶، «کد خواندن تک رجیستر» (Write Single Register)

برای نوشتن یک نگهدارنده رجیستری یکتا در یک دستگاه slave راه دور استفاده می شود. درخواست PDU آدرس حافظه رجیستری مربوط به slave را که باید نوشته شود، تعیین می کند.

کد عملکرد ۱۵، «کد خواندن کوپل های چندگانه» (Write Multiple Coils)

برای اعمال بر روی هر یک از کوپل های متوالی در یک دستگاه slave راه دور استفاده می شود. درخواست PDU آدرس حافظه کوپل های ON یا OFF شده رامشخص می کند.



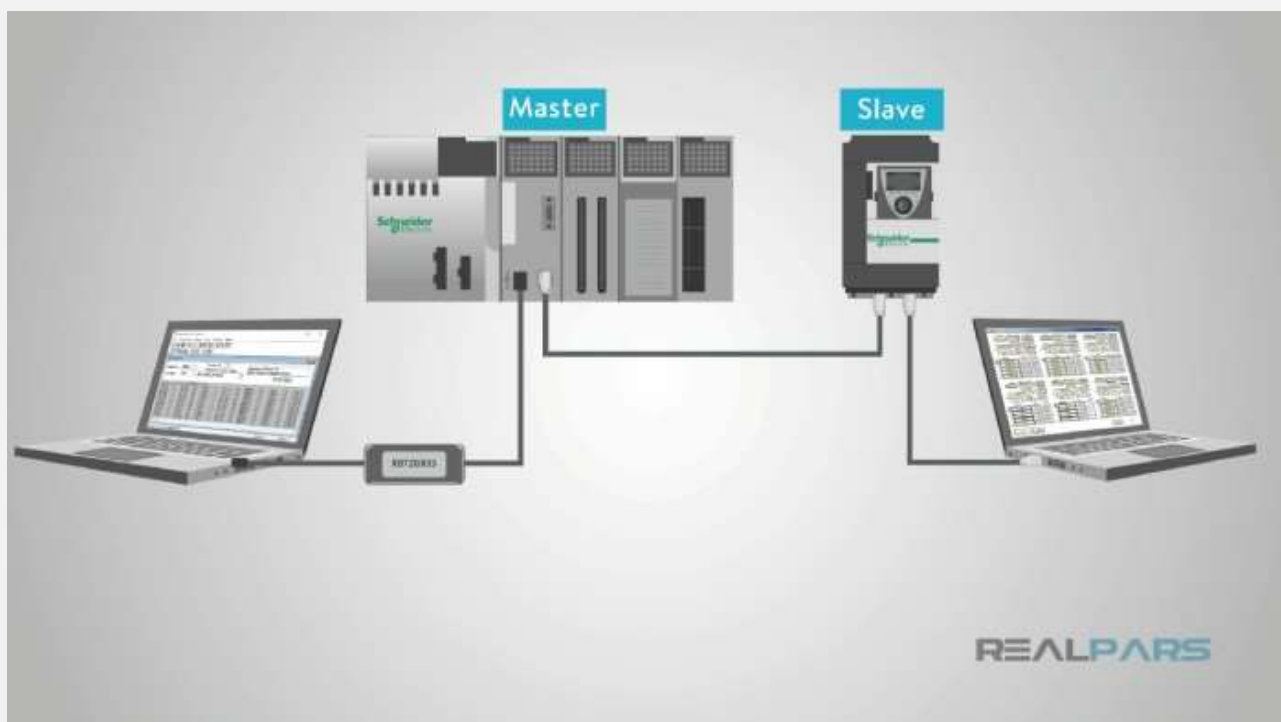
کد عملکرد ۱۶، «کد نوشتن رجیستری های چندگانه» (Write Multiple Registers)

برای نوشتن یک بلوک از رجیستری های پیوسته از ۱ تا ۱۲۳ رجیستر در یک دستگاه slave راه دور استفاده می شود.

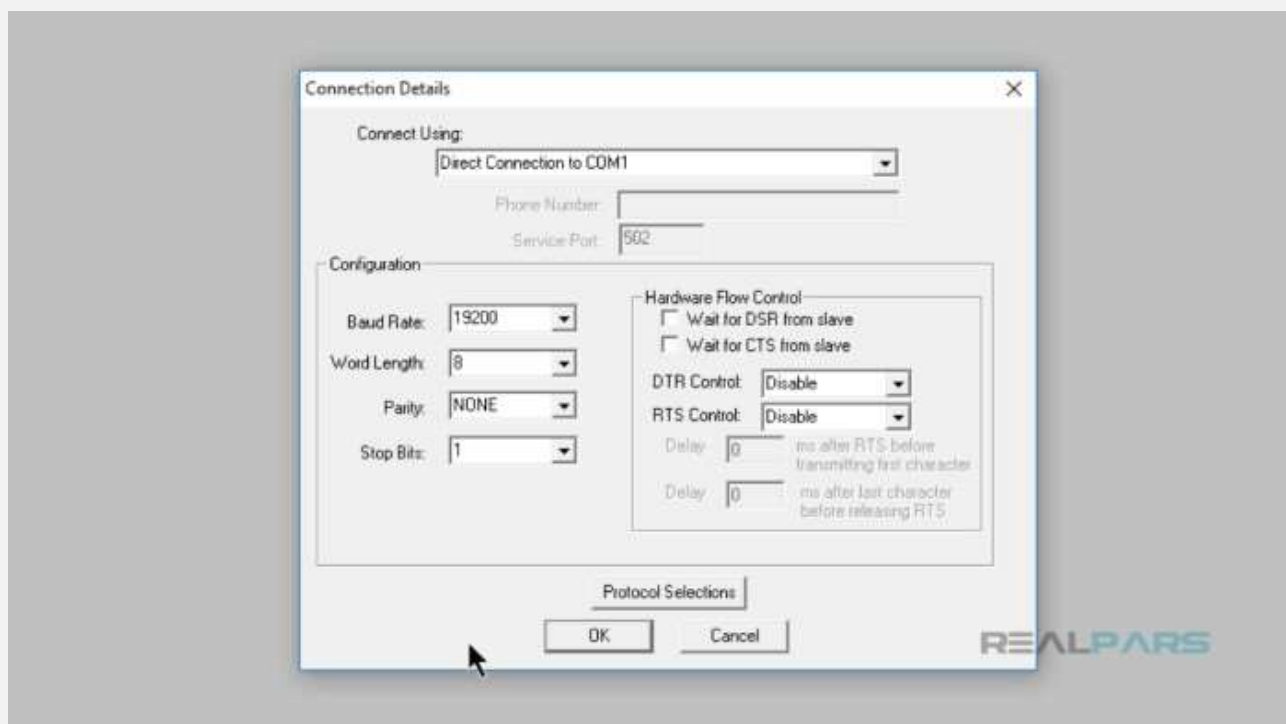
این کدهای عملکرد در مدباس، نمایانگر رایجترین کدهای دستوری خواندن و نوشتن هستند. و می توانید برای آشنایی بیشتر، مشخصات پروتکل مدباس ([Modbus protocol specification](#)) را دوباره بررسی کنید.

شبیه سازی پروتکل مدباس

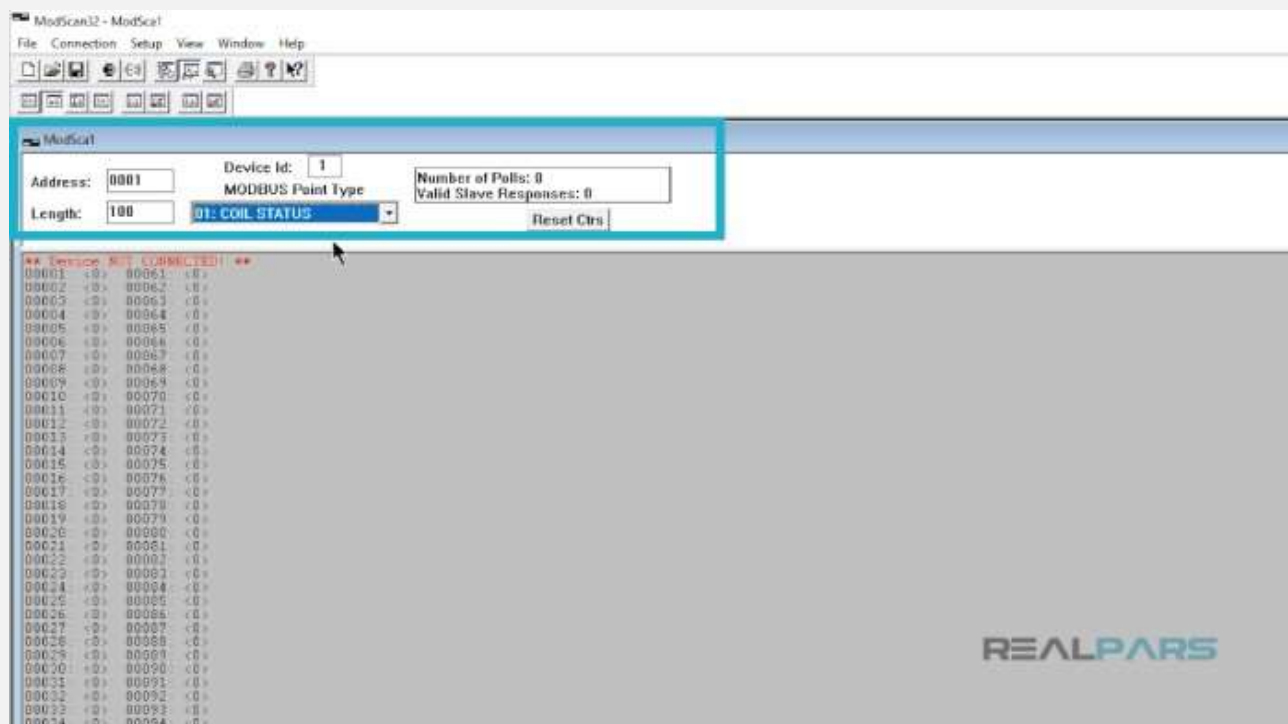
ارتباط دستگاه های slave یا PLC های کنترل کننده و یا کامپیوترها را می توان با استفاده از برنامه شبیه ساز پروتکل مدباس بر روی کامپیوتر شخصی اجرا کرد. اتصال می تواند سریال یا اترنت و به صورت یک master یا slave باشد. این نرم افزار امکان استفاده از تمام کدهای عملکرد (کدهای دستوری) مربوط به ارتباطات پروتکل مدباس را می دهد. با استفاده از این کدها به راحتی می توان خواندن و نوشتن را با یک slave انجام داد. می توان از یک کامپیوتر برای اجرای برنامه شبیه سازی slave استفاده کرد و کامپیوتر دیگری را برای اجرای برنامه شبیه سازی master به کار برد.



اتصال به یک slave ابتدا با تنظیم پارامترهای ارتباط برای سریال پورت COM انجام می شود.



سپس با وارد کردن آدرس slave در فیلد شناسه دستگاه، (به عنوان مثال ۱)، انتخاب کد دستوری ۰۱ برای خواندن موقعیت کویل های slave در آدرس ۰۰۰۱ و خواندن طول ۱۰۰ کویل، انجام می شود.

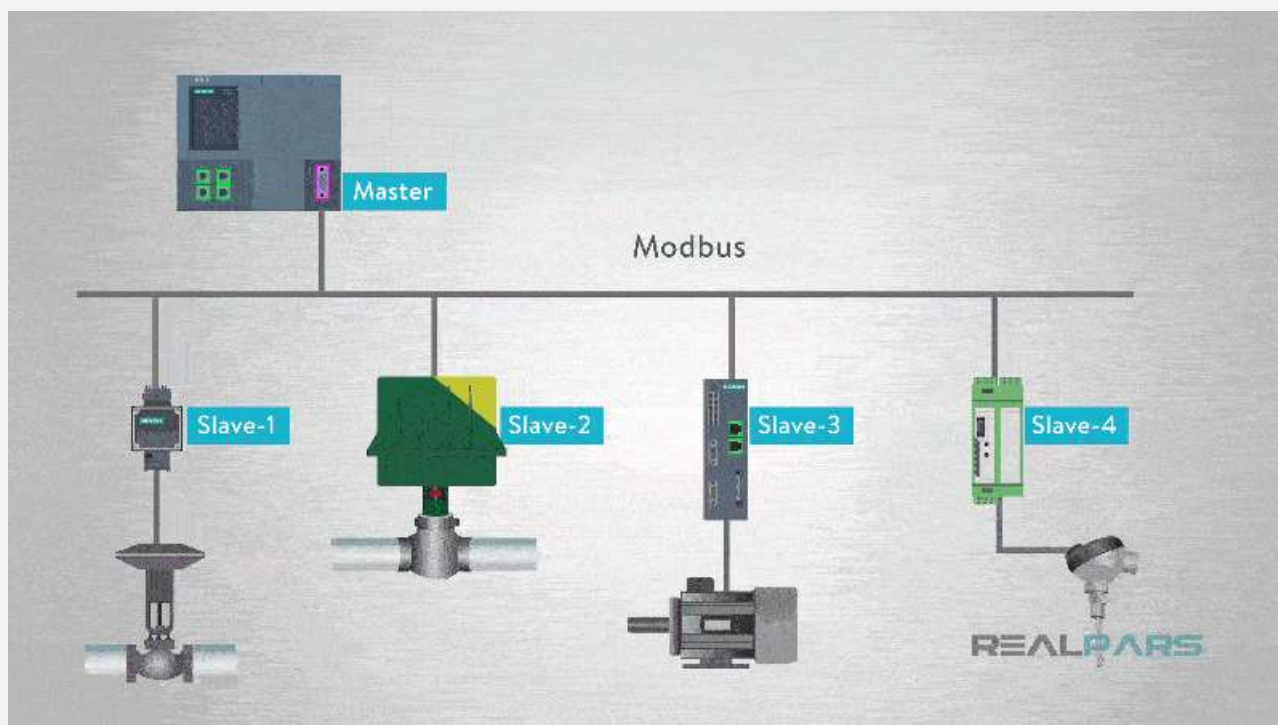


همچنین، چندین شرکت ارائه دهنده محصولات و پشتیبانی برای کمک به شما در استفاده از ارتباط پروتکل وجود دارد. ProSoft Technologies، HMS Industrial Communication و MOXA از جمله این شرکت ها هستند.

انواع دستگاه های Slave

• مبدل های ۰ و ۱ (I/O transducer)

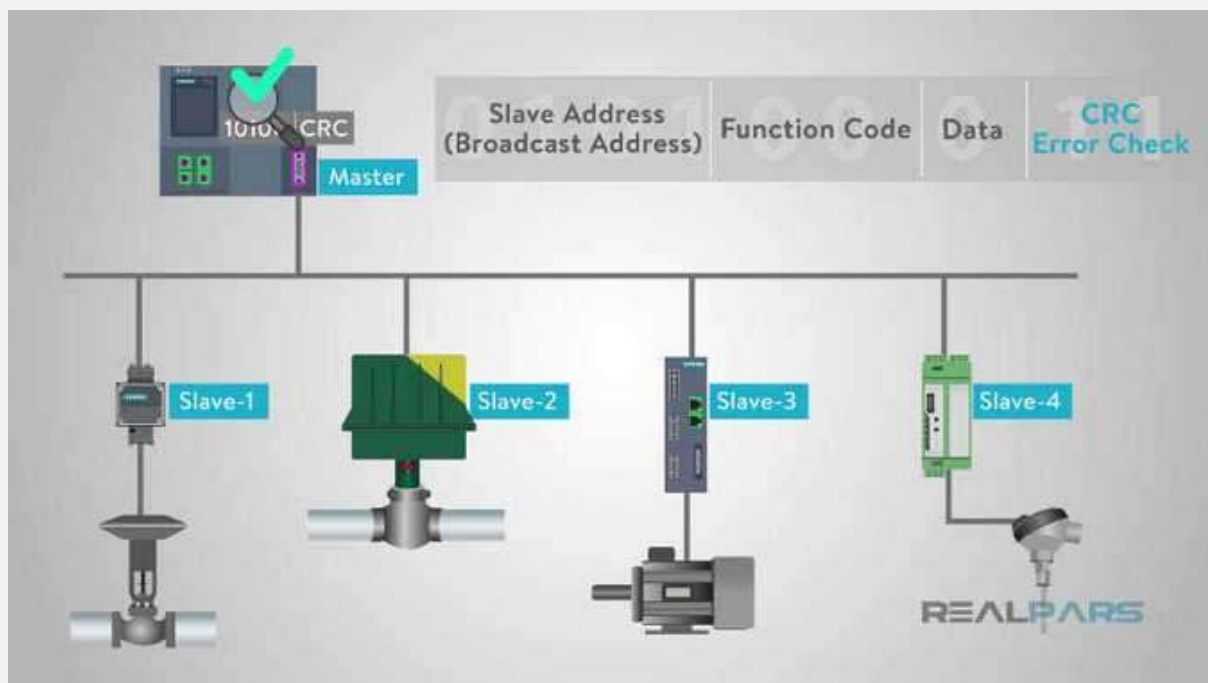
- شیر (valve)
 - درایو شبکه (network drive)
 - سایر دستگاه های اندازه گیری
- Master می تواند دستوری را فقط برای یک slave منحصر به فرد صادر کند و یا یک پیام broadcast را برای همه slave ارسال کند.
- Slave ها به تمام درخواست هایی که به طور اختصاصی به آن ها داده شده پاسخ می دهند، اما به پیام های بردکست پاسخ نمی دهند.
- Slave ها به اختیار خود پیامی را ارسال نمی کنند بلکه تنها به پیام های ارسال شده از master پاسخ می دهند.



دستورات master

- دستورات مربوط به slave ها (دستورات بردکست)
- کد عملکرد با فرمان خواندن یا نوشتن به هر slave
- فرمان نوشتن "داده" در صورتی که دستور از طرف master صادر شده باشد
- خطا در چک کردن فیلد

بررسی خطا (error checking) مقداری است که master یا slave، در ابتدای انتقال یا پاسخ ایجاد می کند و از بررسی آن برای تأیید صحت محتویات پیام دریافت شده، استفاده می کند.



پاسخ های slave

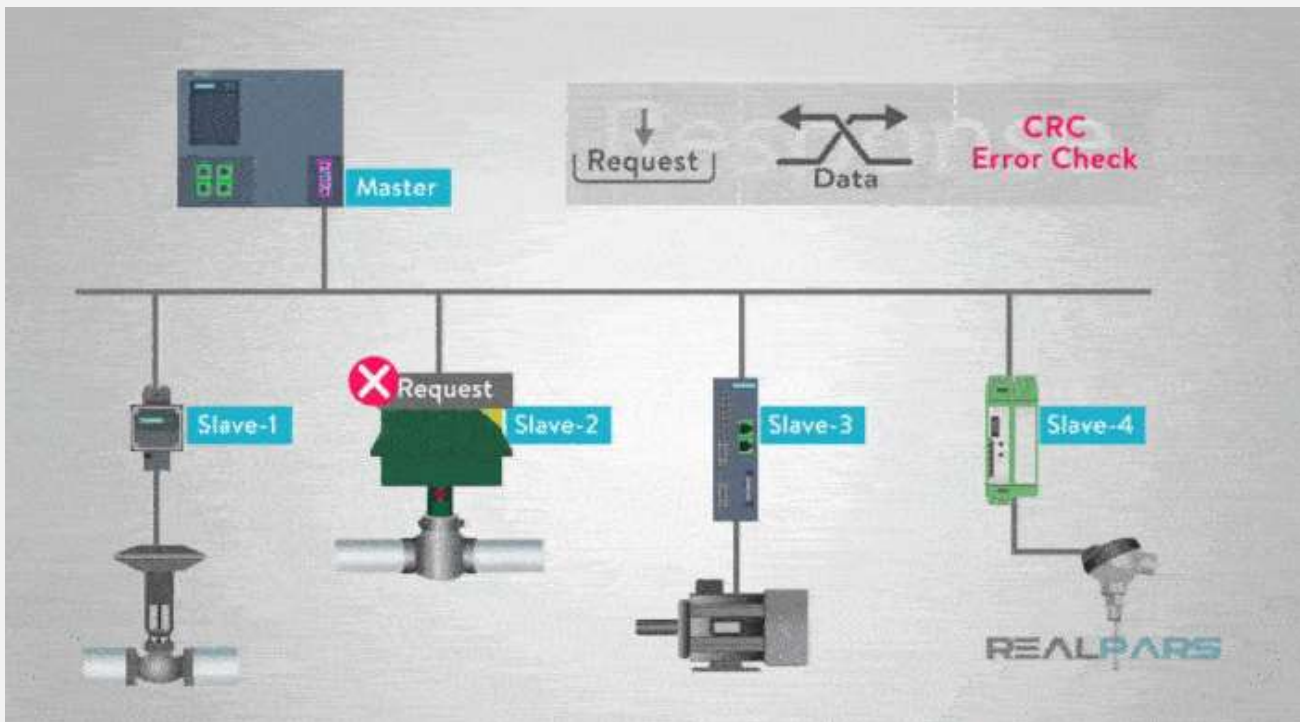
- تأیید درخواست دریافت شده

- بازگرداندن داده ها

- بررسی خطای داده ها

اگر هیچ خطایی رخ نداده باشد، پاسخ slave حاوی داده های درخواست شده خواهد بود.

اگر خطایی در پیام دریافت شده رخ دهد و یا اگر slave قادر به انجام اقدام مورد درخواست نباشد، slave پاسخی مبنی بر رد آن پیام ارسال می کند.



در واقع فیلد بررسی خطا (error check field) در کادر پیام های slave، به master اجازه میدهد که اعتبار محتویات پیام را تأیید کند.

