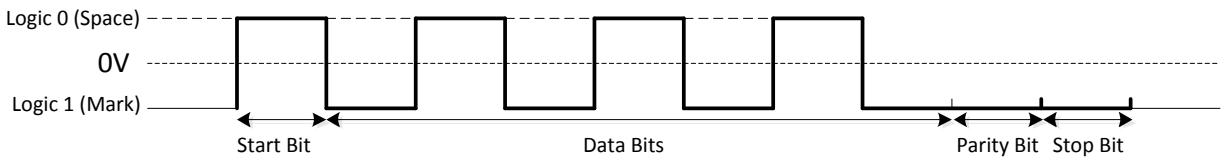
ارتباط سریال :

این ارتباط فقط از یک خط استفاده نموده و بنابراین در هر لحظه تنها یک بیت می تواند بین فرستنده و گیرنده مبادله شود . به همین دلیل ارتباط سریال نسبت به موازی سرعت کمتری دارد ولی ارزانتر است . پورت مربوط به ارتباطات سری با COM معرفی می شود.

دو موضوع اصلی که باید در ارتباطات داده در نظر گرفته شوند به شرح زیر می باشند:

- **استاندارد واسط (مانند سیم کشی فیزیکی و ملاحظات مربوط به ولتاژ)**
- **پروتکل نرم افزار (مانند نوع و ترتیب کاراکترهای مورد انتقال)**

شکل و قالب پیام های ارتباط اطلاعات در یک سیستم آسنکرون مانند RS-232 ، به صورت زیر است.



پیغام آسنکرون (ناهماهنگ و غیرهمزمان) سریال

لینک ارتباطات داده در ابتدا در سطح منطقی 1 می باشد.

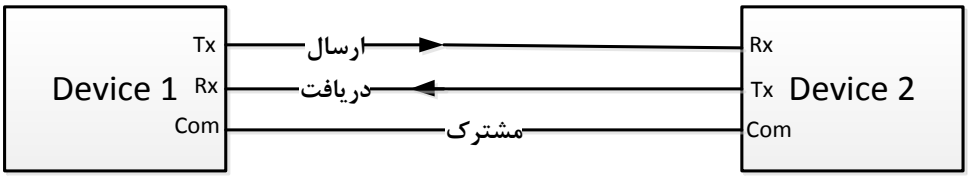
بیت توازن موجود در انتهای کاراکتر در حقیقت اثر انگشت کاراکتر می باشد تا گیرنده بتواند مشخص سازد که آیا خطایی در انتقال صورت گرفته است یا خیر. به عنوان مثال توازن زوج به این معنی است که عدد کل بیت های 1 منطقی در داده به همراه بیت توازن مربوطه می بایست یک عدد زوج باشد.

به طور خلاصه، تنظیم های انتخابی مربوط به انتقال و ارسال در ارتباط اطلاعات در یک سیستم آسنکرون به شرح زیر می باشند: بیت های داده Data Bit ، بیت های توازن Parity Bit ، بیت های توقف Stop Bit و سرعت انتقال داده Baude rate

سرعت انتقال داده به ازای تعداد بیت ارسالی یا دریافتی در یک ثانیه بیان می شود (bps) . سرعت در حقیقت به عدد و تعداد تغییرات سیگنال در ثانیه، بدون توجه و در نظر گرفتن وجود داده های مفید در جریان بیت مربوط می گردد.

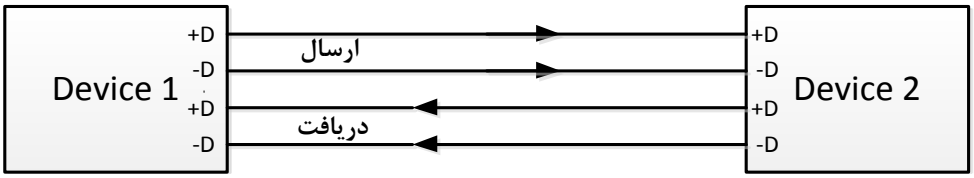
مفاهیم اولیه در ارتباط بین تجهیزات :

SIMPLEX : در این ارتباط، یک سیم برای ارسال و یک سیم برای دریافت، و یک سیم مشترک استفاده می شود. (مانند RS-232) در این نوع اتصال بدلیل ارسال بصورت تک سیم ، نویزپذیرند و بنابراین نمی توان در فواصل دور از این نوع استفاده کرد.



DUPLEX : در این ارتباط از جفت سیم برای ارسال یا دریافت استفاده می شود و بین سیم های ارسال و دریافت پایه مشترک وجود ندارد. این ارتباط به دو شکل Full , Half می باشد. در این نوع سیم کشی بدلیل اتصال دیفرانسیلی نویزپذیری بسیار کمی دارند و در فواصل دور به خوبی کار می کنند.

FULL DUPLEX : در این ارتباط از چهار رشته سیم استفاده می شود، دو رشته سیم برای ارسال و دو رشته سیم برای دریافت اطلاعات استفاده می شود. (مانند Ethernet, RS-422)



HALF DUPLEX :

در این ارتباط از دو رشته سیم استفاده می شود، از این دو رشته سیم هم برای ارسال و هم برای دریافت اطلاعات استفاده می شود.

(مانند RS-485)

Modbus

این نوع شبکه ابتدا در سال 1979 توسط Modicon که امروزه Schneider Electric آنرا خریداری کرده است عرضه شد . کاربرد اولیه آن برای استفاده در PLC ها بود ولی بتدریج بعنوان یک استاندارد ارتباطی پذیرفته شد و بسیاری از سازندگان تجهیزات اتوماسیون آن را پشتیبان کردند بدین ترتیب MODBUS بصورت یک استاندارد باز در آمد به گونه ای که محصولات سازندگان مختلف به سهولت توسط این پروتکل با یکدیگر ارتباط برقرار کردند . سازندگان وسایل کوچک نیز ترجیح دادند این پروتکل را با ارتباط RS232 یا RS485 روی وسایل خود بکار ببرند تا استفاده از آنها درپروژه های بزرگ میسر گردد.

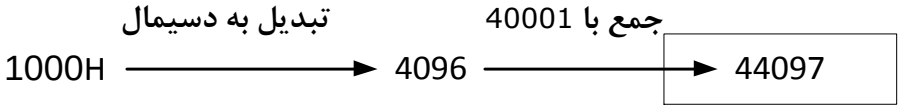
Modbus دارای سه نسخه اصلی زیر است که ویژگی های آنها با هم متفاوتند:

- **Modbus RTU/ASCII : ارتباط بصورت سریال از طریق پورتهای RS485 یا RS232 .**
- **Modbus TCP/IP : ارتباط از طریق پورت اترنت**
- **Modbus Plus : ارتباط بصورت Token Pass و سرعت بالا و باس انحصاری می باشد.**

آدرس دهی حافظه ها و رجیسترها در شبکه مودباس :

هر رجیستر در پروتوکل مودباس با یک کد مشخص می شود. برای مثال در DELTA PLC رجیستر D0 در پروتوکل مدباس با کد 1000H تعریف می شود (آدرس رجیسترهای هر دستگاه باید توسط شرکت سازنده ارائه شوند). بعضی از سازندگان آدرس های مودباس رجیسترهای خود را با اعداد هگز و بعضی دیگر بصورت عددی مثل 44097 ارائه می کنند.

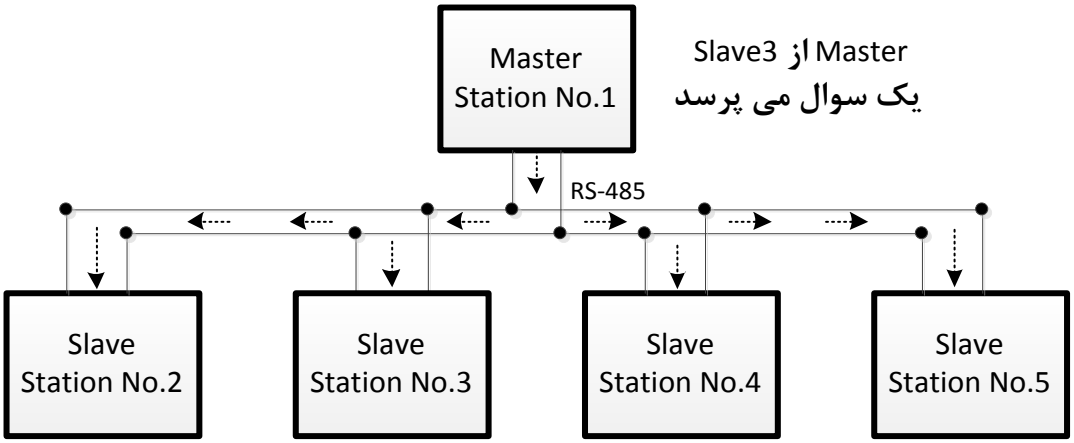
روش تبدیل این دو نوع فرمت به یکدیگر :

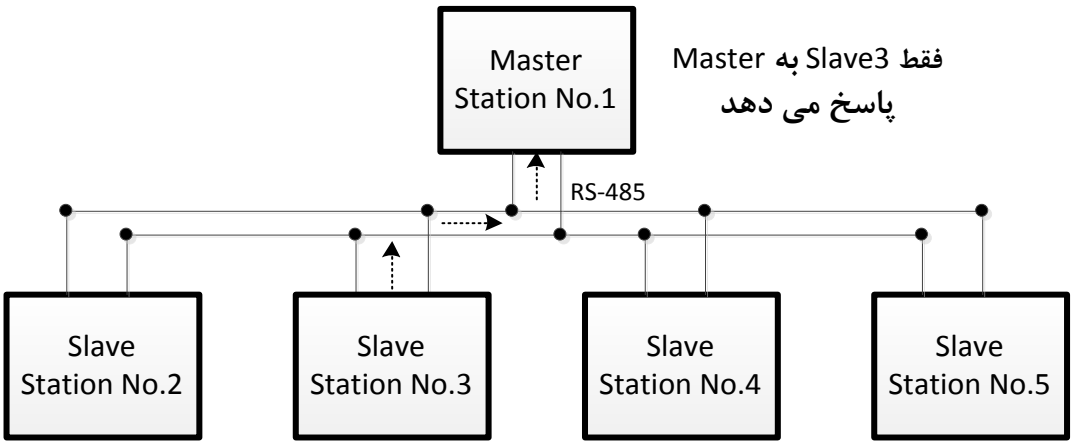


ارتباط بین تجهیزات در شبکه های مودباس :

ارتباط بین تجهیزات در پروتوکل Modbus RTU/ASCII بر روی پورتهای سریال RS-232, RS-485, RS-422 بطور MASTER/SLAVE می باشد .

چنانچه Master بخواهد با هر Slave ارتباط برقرار کند ، شماره آدرس آن Slave را در پیغام ارسالی قرار می دهد و پیغام را می فرستد. تمام Slave ها این پیغام را دریافت می کنند ولی آن Slave ، که شماره آدرس آن در پیغام نوشته شده است پاسخ می دهد بدین ترتیب در هر لحظه فقط یک پیغام بر روی خط وجود دارد. SLAVE می تواند ورودی و خروجی دیجیتال یا آنالوگ، شیرهای برقی، درایورهای موتورها، یا تجهیزات اندازه گیری مانند ترنسمیترهای دما ، لودسل و غیره باشد.





تبادل اطلاعات در پروتوکل مودباس در شبکه های سریال

پروتوکل مودباس بر روی پورتهای سریال RS-232, RS-485, RS-422 از دو مد انتقال ASCII و RTU استفاده می کند:

ASCII (American Standard for Information Interchange): انتقال اطلاعات با فرمت کدهای اسکی

RTU (Remote Terminal Unit): انتقال اطلاعات با فرمت هگزا دسیمال

شکل فریم اطلاعات در MODBUS RTU :

Addres	Functio	Data	CRC Check
8 bits	8 bits	N x 8 bits	16 bits

بوسیله بایت اول می توانیم تعیین کنیم که این پیغام برای کدام Slave است. توسط بایت دوم، نوع پیغام تعریف می شود، این عمل در نرم افزار انجام می شود. برای هر نوع پیغام، یک کد استاندارد در پروتوکل وجود دارد. بایتهای بعدی، اطلاعاتی است که متناسب با نوع پیغام نوشته می شود (شماره رجیستر، تعداد رجیستر، مقدار رجیستر و ...). دو بایت آخر نیز مربوط به CRC (بررسی خطا در اطلاعات) می باشد.

در مد RTU هر بایت پیام شامل عددی بین 0 تا 255 می باشد و برای شناسایی شروع وپایان فریم از پارامتر زمان استفاده می شود.

شکل فریم اطلاعات در MODBUS ASCII :

Start	Address	Function	Data	LRC	End
1 char :	2 chars	2 chars	0 up to 2x252 char(s)	2 chars	2 chars CR,LF

در مد ASCII هر بایت اطلاعات توسط دو کاراکتر (بایت) که فقط می توانند شامل کدهای اسکی A~F , 0~9 باشند فرستاده می شود. برای شناسایی ابتدای فریم از کاراکتر ' : ' و برای شناسایی انتهای فریم از کاراکترهای CR,LF استفاده می شود.

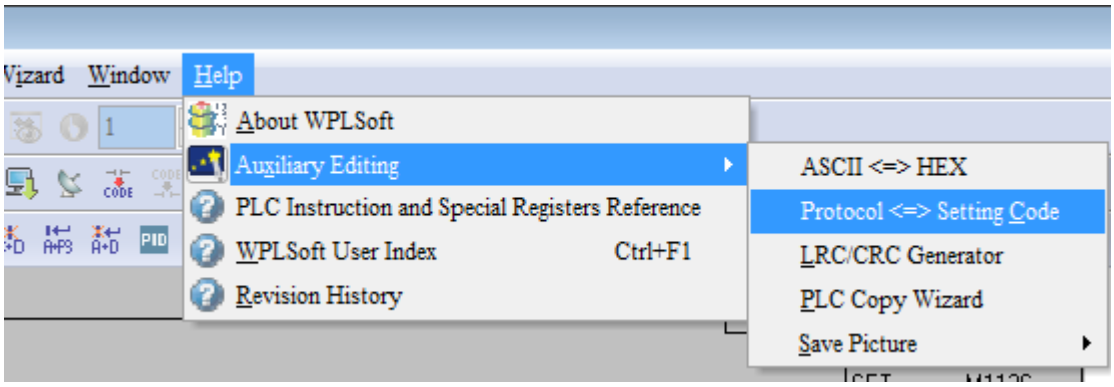
مد RTU بدلیل کوچکتر بودن فریم های مبادله شده ، نسبت به مد ASCII دارای سرعت تبادل اطلاعات بالاتری می باشد .

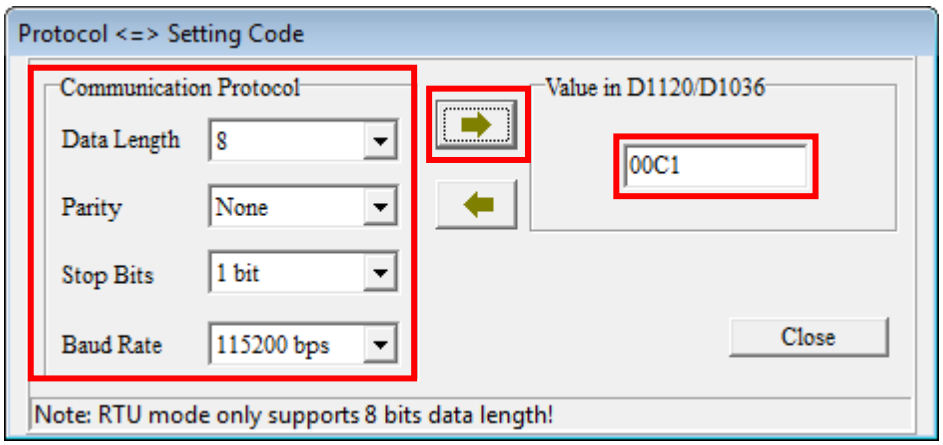
خطایابی فریمها در شبکه های مودباس :
در ارسال یا دریافت یک بایت، بیتی به نام PARITY نیز فرستاده می شود. فرستنده براساس زوج یا فرد بودن تعداد یک های بایت ارسالی ، این بیت را تنظیم می کند و گیرنده با چک کردن این بیت، بر وقوع یا عدم وقوع خطا مطلع می شود.

برای بررسی کل فریم، فرستنده دوبایت آخر فریم را با مقداری پر می کند، و گیرنده با چک کردن این دو بایت بر صحت فریم دریافت شده اطمینان پیدا می کند. روش خطایابی CRC (Cyclical Redundancy Check) در مد MODBUS RTU و روش خطایابی LRC (Longitudinal Redundancy Check) در مد MODBUS ASCII مورد استفاده قرار می گیرد.

تنظیم پورت :

با استفاده از منوی، Help >> Auxiliary Editing >> Protocol ⇔ Setting Code می توان عدد مربوط به نوع تنظیمات هر پورت را معلوم کرد.





- **عدد بدست آمده را برای COM1 باید در رجیستر D1036 قرار داد و با یک کردن بیت M1138 این مقدار را در CPU ذخیره کرد.**
- **رجیسترهای مربوط به COM2 ، D1120 , M1120 و برای COM3 ، D1109 , M1136 می باشند.**
- **با روشن کردن بیتهای M1139, M1143, M1320 برای COM1, COM2, COM3 می توان پروتوکل پورت مورد نظر را از ASCII به RTU تبدیل کرد.**
- **برای ارسال دستور، با روشن کردن بیتهای M1312, M1122, M1316 برای COM1, COM2, COM3 تعیین می کنیم که دستور از کدام COM ارسال شود.**

تابع **MODRW** :

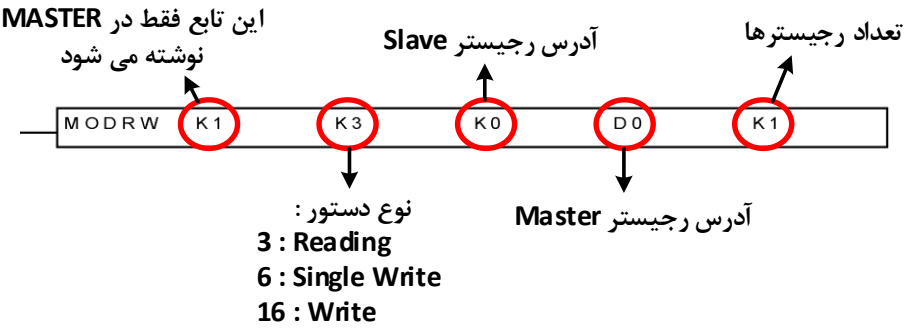
سه نوع فرمان در شبکه می توان داشت :

- دستور کد **03** : خواندن تعدادی رجیستر از **Slave** **(Reading)**
- دستور کد **06** : نوشتن بر روی فقط و فقط یک رجیستر از **Slave** **(Single Write)**
- دستور کد **16** : نوشتن بر روی تعدادی رجیستر از **Slave** **(Writing)**

اطلاعات ارسالی/دریافتی بین **Master** و **Slave** بصورت زیر می باشد :

خواندن مقادیر رجیسترهای **16** بیتی از **Slave** :

تابع **150** **"MODRW"** برای ارسال دستور توسط **MASTER** می باشد.

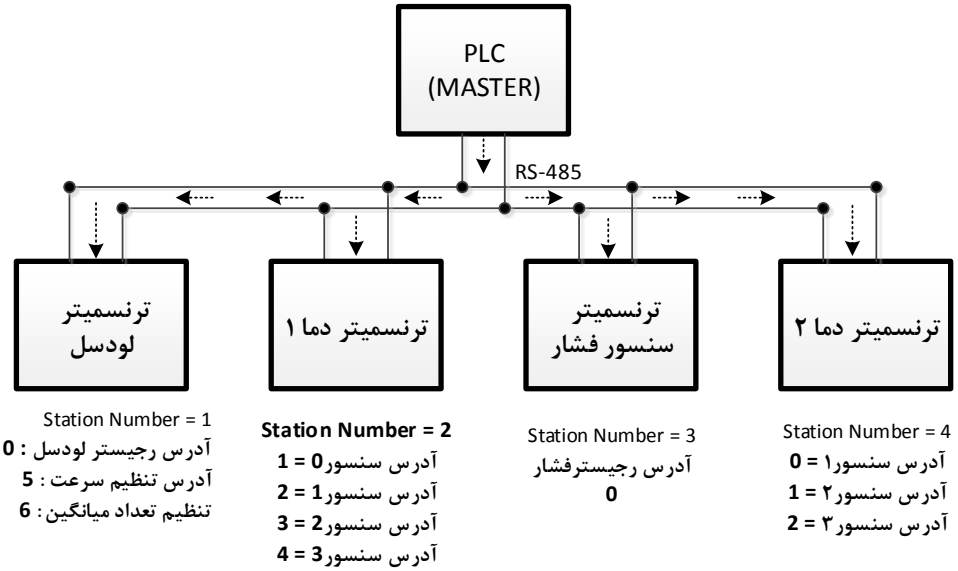


پس از این دستور مقادیر رجیسترهای خوانده شده در رجیسترهای **D1296 ~ D1311** نمایش داده می شوند و باید این مقادیر را به رجیسترهای دلخواه انتقال داد.

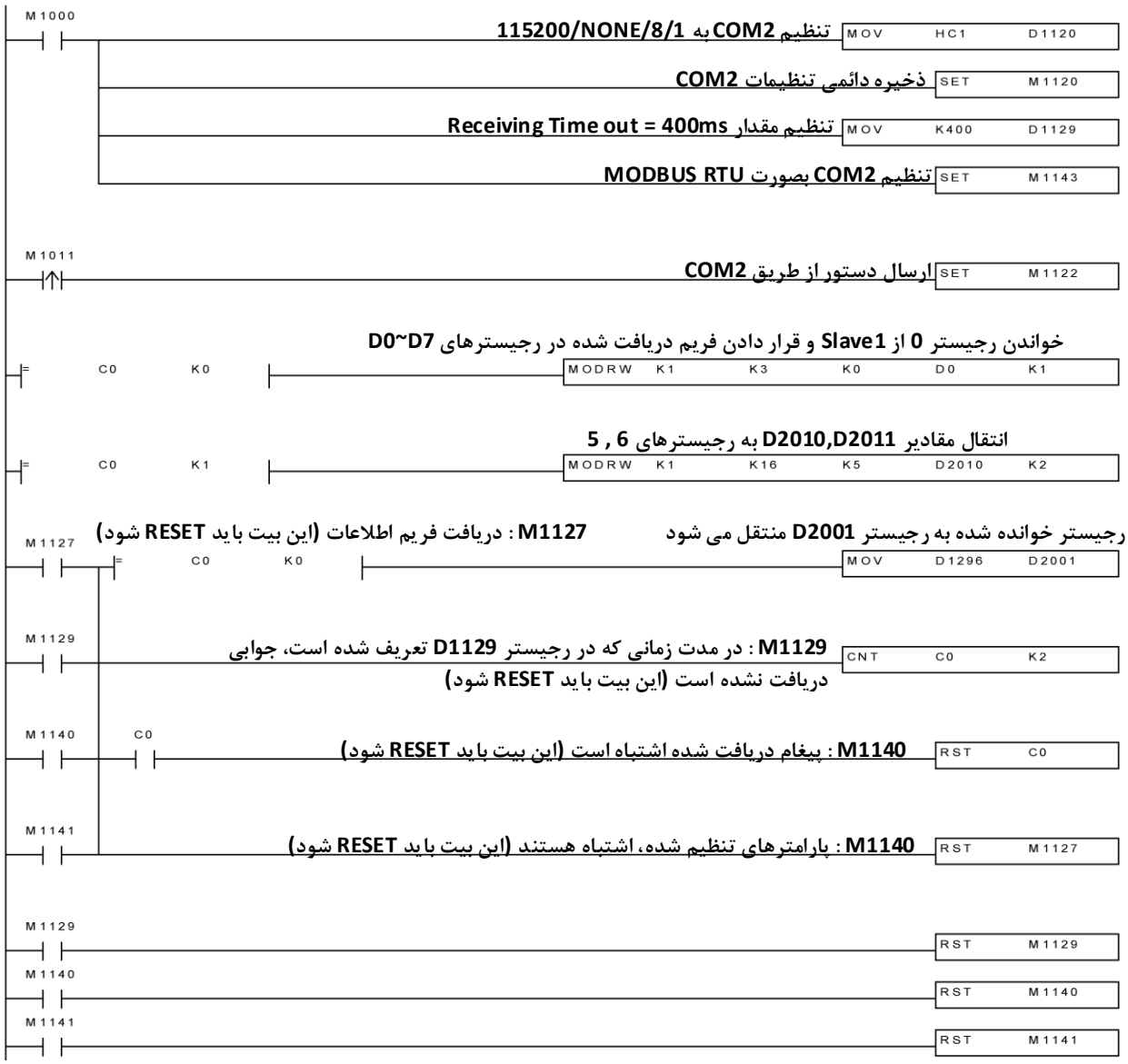
در مثال زیر، **4** ماژول ترنسمیتر مودباس به باس **RS-485** متصل هستند، آدرس رجیسترهای هر ماژول نیز در زیر آن نوشته شده است.

الف) برنامه ایی بنویسید که با استفاده از **COM2** مربوط به **PLC** فقط از رجیسترشماره **0** مقدار لودسل را بخوانیم و مقادیر سرعت و تعداد میانگین گیری آنرا تنظیم نماییم.

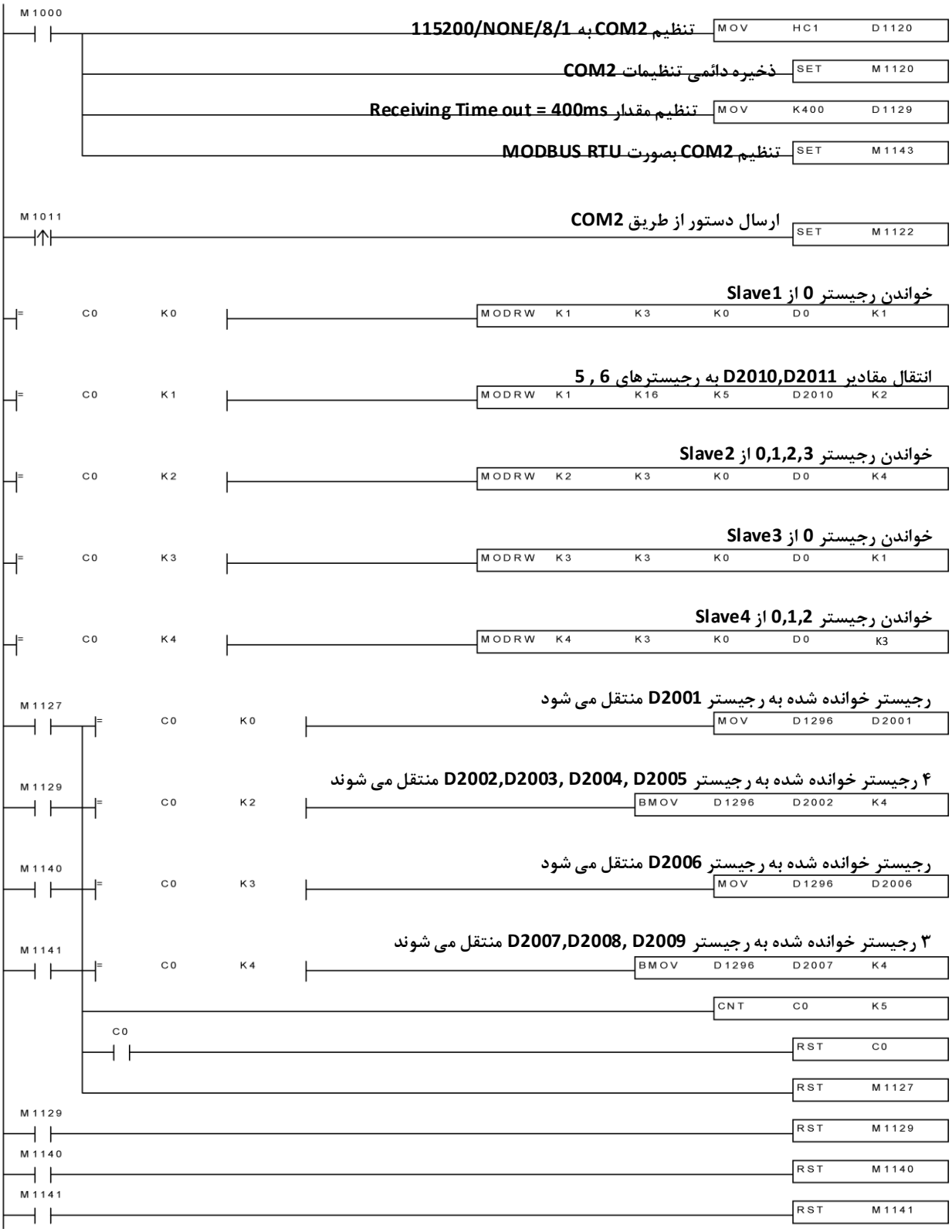
ب) برنامه ایی بنویسید که با استفاده از **COM2** مربوط به **PLC** ، از رجیسترشماره **0** مقدار لودسل را بخوانیم و مقادیر سرعت و تعداد میانگین گیری آنرا تنظیم نماییم و **4** رجیستر **Slave2** و یک رجیستر از **Slave2** و سه رجیستر از **Slave3** را بخوانیم.



برنامه الف)


M1000
تنظیم COM2 به 115200/NONE/8/1
MOV HC1 D1120
ذخیره دائمی تنظیمات COM2
SET M1120
تنظیم مقدار Receiving Time out = 400ms
MOV K400 D1129
تنظیم COM2 بصورت MODBUS RTU
SET M1143
M1011
ارسال دستور از طریق COM2
SET M1122
خواندن رجیستر 0 از Slave1 و قرار دادن فریم دریافت شده در رجیسترهای D0~D7
= C0 K0
MODRW K1 K3 K0 D0 K1
انتقال مقادیر D2010,D2011 به رجیسترهای 6 , 5
= C0 K1
MODRW K1 K16 K5 D2010 K2
M1127
M1127 : دریافت فریم اطلاعات (این بیت باید RESET شود)
رجیستر خوانده شده به رجیستر D2001 منتقل می شود
= C0 K0
MOV D1296 D2001
M1129
M1129 : در مدت زمانی که در رجیستر D1129 تعریف شده است، جوابی دریافت نشده است (این بیت باید RESET شود)
CNT C0 K2
M1140
C0
M1140 : پیغام دریافت شده اشتباه است (این بیت باید RESET شود)
RST C0
M1141
M1140 : پارامترهای تنظیم شده، اشتباه هستند (این بیت باید RESET شود)
RST M1127
M1129
RST M1129
M1140
RST M1140
M1141
RST M1141

برنامه ب)


M1000
تنظیم COM2 به 115200/NONE/8/1
MOV HC1 D1120
ذخیره دائمی تنظیمات COM2
SET M1120
تنظیم مقدار Receiving Time out = 400ms
MOV K400 D1129
تنظیم COM2 بصورت MODBUS RTU
SET M1143
M1011
ارسال دستور از طریق COM2
SET M1122
خواندن رجیستر 0 از Slave1
C0 K0
MODRW K1 K3 K0 D0 K1
انتقال مقادیر D2010,D2011 به رجیسترهای 6 , 5
C0 K1
MODRW K1 K16 K5 D2010 K2
خواندن رجیستر 0,1,2,3 از Slave2
C0 K2
MODRW K2 K3 K0 D0 K4
خواندن رجیستر 0 از Slave3
C0 K3
MODRW K3 K3 K0 D0 K1
خواندن رجیستر 0,1,2 از Slave4
C0 K4
MODRW K4 K3 K0 D0 K3
M1127
رجیستر خوانده شده به رجیستر D2001 منتقل می شود
C0 K0
MOV D1296 D2001
M1129
۴ رجیستر خوانده شده به رجیستر D2002,D2003, D2004, D2005 منتقل می شوند
C0 K2
BMOV D1296 D2002 K4
M1140
رجیستر خوانده شده به رجیستر D2006 منتقل می شود
C0 K3
MOV D1296 D2006
M1141
۳ رجیستر خوانده شده به رجیستر D2007,D2008, D2009 منتقل می شوند
C0 K4
BMOV D1296 D2007 K4
CNT C0 K5
C0
RST C0
RST M1127
M1129
RST M1129
M1140
RST M1140
M1141
RST M1141