



راهنمای جامع راه اندازی ارتباط مدباس درایو PM300

MODBUS RTU • RS485



PM300-12k0 | PM300-2k5

از نخستین اتصال تا فرمان حرکت، پایش و تنظیم مد سوئیچینگ

برای توسعه دهندگان نرم افزار، میکروکنترلر و PLC

ویرایش ۲ • مهر ۱۴۰۵

شرکت ویرا صنعت الکترونیک سورین

hipowervses.ir

هدف راهنما و شروع سریع

این راهنما اتصال یک کنترل‌کننده به یک درایو PM300 را آموزش می‌دهد. کنترل‌کننده می‌تواند رایانه، PLC، HMI یا میکروکنترلر باشد. پس از راه‌اندازی ارتباط، کاربر فرکانس هدف و فرمان حرکت را می‌فرستد، وضعیت موتور را می‌خواند و مد سوئیچینگ و رندوم فرکانسی را تنظیم می‌کند.

روش ارتباط برای دو مدل PM300-12k0 و PM300-2k5 یکسان است. در هر اتصال، یکی از این دو مدل به کنترل‌کننده وصل می‌شود. ارتباط از نوع Modbus RTU و رابط الکتریکی آن RS485 است. این راهنما تمام اطلاعات لازم برای استفاده از رابط ارتباطی محصول را بیان می‌کند.

ابتدا اتصال و تنظیمات پورت را انجام دهید؛ سپس وضعیت درایو را بخوانید و فرمان توقف بفرستید. پس از تأیید سلامت و توقف، تنظیمات مورد نیاز را اعمال کنید و در پایان فرمان حرکت بدهید. همه آدرس‌ها، مگر آن‌که صریحاً شماره نمایشی نوشته شده باشد، آدرس واقعی پیام هستند.

گام	اقدام	نتیجه
۱	اتصال RS485 کنترل‌کننده به درایو	یک مسیر ارسال و دریافت
۲	تنظیم پورت مطابق جدول ارتباط	9600 و 8N1 و شناسه 1
۳	خواندن رجیستر 14 با FC03	دریافت وضعیت معتبر
۴	نوشتن مقدار 1 در رجیستر 2	ارسال فرمان STOP
۵	تنظیم قابلیت‌ها و فرکانس هدف	بازخوانی و تأیید تنظیمات
۶	فرمان RUN و پایش دوره‌ای	نمایش وضعیت و مقادیر تازه

پاسخ موفق، پذیرش پیام را تأیید می‌کند. رسیدن موتور به سرعت هدف و توقف واقعی محور از روی بازخورد بررسی می‌شود. قطع کابل ارتباط، فرمان توقف موتور ایجاد نمی‌کند. در ماشین‌هایی که قطع ارتباط باید باعث توقف شود، مسیر توقف مستقل ماشین را در طراحی لحاظ کنید.

فهرست مطالب

در Word می‌توانید با انتخاب عنوان، به بخش مربوط بروید. شماره صفحه‌ها با نسخه PDF همراه تطبیق داده شده‌اند.

صفحه	عنوان بخش
2	هدف راهنما و شروع سریع
4	آشنایی ساده با مدباس
5	اتصال کنترل‌کننده به یک درایو
6	تنظیمات ارتباط و زمان‌بندی
7	آدرس رجیستر و ترتیب بایت‌ها
8	انتخاب نوع درخواست
9	رجیسترهای فرمان و تنظیم
10	خواندن مقادیر و تبدیل واحد
11	وضعیت کار و کدهای خطا
12	آشنایی با سوئیچینگ و رندوم فرکانسی
13	تنظیم سوئیچینگ از طریق مدباس
14	ترتیب راه‌اندازی و توقف
15	نمونه پیام‌های خواندن و فرمان تکی
16	نوشتن چند مقدار در یک پیام
17	نگهداری تنظیمات پس از خاموشی
18	آزمون ارتباط و شمارنده‌های خطا
19	خواندن اطلاعات شناسایی دستگاه
20	راه‌اندازی با PLC و HMI
21	پیاده‌سازی مستر با میکروکنترلر
22	راه‌اندازی با نرم‌افزار رایانه
23	پاسخ رد درخواست و مهلت انتظار
24	رفع اشکال و آزمون نهایی
25	ضمیمه ساختار دقیق فریم‌ها
26	ضمیمه محاسبه CRC
27	واژه‌نامه و مراجع

آشنایی ساده با مدباس

مدباس روش مشخصی برای پرسیدن و پاسخ دادن میان دو دستگاه است. کنترل کننده درخواست را می فرستد و درایو پاسخ می دهد. برای مثال، کنترل کننده می پرسد «جریان چقدر است؟» یا درخواست می کند «فرکانس هدف را روی 50 هرتز قرار بده». درایو بدون درخواست کنترل کننده، داده پایش ارسال نمی کند.

اصطلاح	معنای ساده	مثال
Master	آغازکننده درخواست	رایانه، PLC یا میکروکنترلر
Slave	پاسخ دهنده به درخواست	درایو PM300 با شناسه 1
Register	محل داده با شماره مشخص	آدرس 5 برای خواندن جریان
Read	خواندن بدون تغییر تنظیم	دریافت وضعیت از آدرس 14
Write	فرستادن مقدار یا فرمان جدید	نوشتن فرکانس در آدرس 1
Function Code	کد مشخص کننده نوع کار	FC03 برای خواندن رجیستر
Frame	پیام کامل روی سیم	شناسه، تابع، داده و CRC

تفاوت رابط و پروتکل

RS485 روش انتقال الکتریکی روی کابل است؛ Modbus RTU ترتیب و معنی بایتهای پیام را تعیین می کند. داشتن پورت RS485 کافی نیست؛ کنترل کننده باید پیامهای Modbus RTU را نیز تولید و پاسخ آنها را تفسیر کند.

رجیستر چگونه عدد را منتقل می کند

هر رجیستر 16 بیت، یعنی دو بایت، دارد و عدد خامی از 0 تا 65535 منتقل می کند. واحد عدد را جدول رجیستر مشخص می کند. برای مثال، مقدار 235 در رجیستر جریان به معنی 23.5 آمپر است. عدد خام را پیش از نمایش طبق جدول تبدیل کنید.

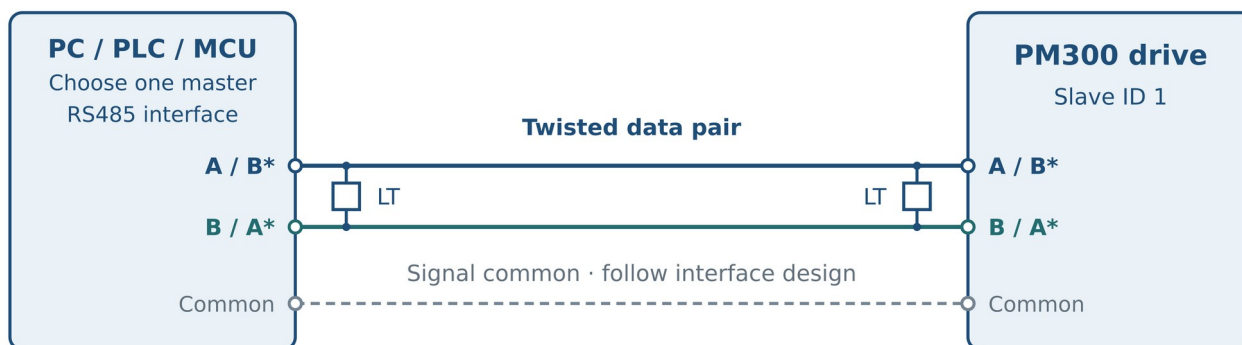
شناسه دستگاه با آدرس رجیستر فرق دارد

شناسه 1 مشخص می کند پیام برای درایو است. آدرس 5 مشخص می کند کدام داده همان درایو خوانده شود. برای نخستین ارتباط، شناسه را 1، تابع را FC03، آدرس را 14 و تعداد را 1 قرار دهید. درخواست بعدی را پس از دریافت پاسخ یا پایان مهلت انتظار بفرستید.

اتصال کنترل‌کننده به یک درایو

در این اتصال، یک کنترل‌کننده و یک درایو وجود دارد. رایانه به مبدل USB به RS485 نیاز دارد. PLC یا HMI باید پورت RS485 با قابلیت Modbus RTU Master داشته باشد. میکروکنترلر به یک ترنسیور RS485 متصل می‌شود؛ ترنسیور، سیگنال UART را به سطح الکتریکی RS485 تبدیل می‌کند.

Direct connection · RS485 / Modbus RTU



* A/B names vary: match signal polarity at both interfaces.
LT: line termination at the two physical cable ends.

شکل ۱ اتصال یک کنترل‌کننده به یک درایو PM300

بخش اتصال	روش اجرا
زوج داده	دو سیم داده را با کابل زوج‌تابیده وصل کنید. نام A و B میان سازندگان یکسان نیست؛ نام و قطبیت هر دو رابط را با نقشه همان تجهیزات تطبیق دهید.
مرجع سیگنال	سیم Common را مطابق نقشه رابط وصل کنید. مرجع سیگنال با شیلد، ارت حفاظتی و منفی مدار قدرت یکسان نیست.
مسیر کابل	کابل را مستقیم میان کنترل‌کننده و درایو قرار دهید و از سیم‌های خروجی موتور و مسیرهای قدرت دور نگه دارید.
پایان خط	مقاومت پایان خط متناسب با کابل در دو انتهای اتصال قرار می‌گیرد. مقاومت داخلی رابط را پیش از افزودن مقاومت بررسی کنید.

شکل، مسیر سیگنال را نشان می‌دهد. محل و شماره پایه‌های کانکتور را از نقشه اتصال مدل انتخاب‌شده بردارید. خروجی UART با سطح TTL یا کابل USB را مستقیم به زوج RS485 درایو وصل نکنید.

تنظیمات ارتباط و زمان بندی

دو طرف ارتباط باید سرعت و قالب سریال یکسان داشته باشند. تنظیمات جدول زیر را در نرم افزار، بلوک ارتباط PLC یا UART مستر وارد کنید. تغییر این موارد از طریق رجیسترهای کاربر انجام نمی شود.

گزینه	مقدار ارتباط	معنی
Protocol	Modbus RTU	پیام باینری روی پورت سریال
Baud rate	9600 bit/s	سرعت تبادل داده
Data bits	8	هشت بیت داده در هر کاراکتر
Parity	None	بدون بیت زوجیت
Stop bits	1	یک بیت توقف
Slave ID	1	شناسه درایو در تمام درخواست ها
Flow control	None	بدون کنترل جریان RTS/CTS یا XON/XOFF

عبارت 8N1 یعنی 8 بیت داده، بدون زوجیت و 1 بیت توقف. انتخاب سرعت یا زوجیت دیگری در تنظیمات رایانه، تنظیم درایو را تغییر نمی دهد. شماره COM به مبدل متصل به رایانه وابسته است؛ پورتی را انتخاب کنید که با اتصال مبدل ظاهر می شود.

سه زمان مستقل در برنامه مستر

تنظیم	مقدار راه اندازی	کاربرد
Response timeout	1000 ms	حداکثر انتظار مستر برای یک پاسخ معتبر
Polling interval	500 ms	فاصله برنامه ریزی شده میان خواندن های دوره ای
فاصله سکوت	10 ms	مکث پس از پایان پاسخ و پیش از درخواست بعدی

این سه زمان، تنظیم برنامه مستر هستند. ابتدا با مقادیر جدول کار کنید. برای کاهش دوره پایش تا 250 میلی ثانیه، تکمیل پاسخ قبلی و فاصله سکوت را حفظ کنید. زمان سکوت این اتصال را کمتر از 5 میلی ثانیه قرار ندهید.

هر فریم را پیوسته بفرستید. تا پاسخ درخواست قبلی نرسیده یا مهلت آن تمام نشده است، درخواست بعدی را ارسال نکنید. پایان ارسال یعنی خروج آخرین بیت از UART؛ خالی شدن صف نرم افزار به تنهایی پایان ارسال روی سیم نیست.

آدرس رجیستر و ترتیب بایت‌ها

آدرس خام همان عدد داخل پیام است. شماره نمایشی، نامی است که بعضی نرم‌افزارها و PLCها برای همان رجیستر به کار می‌برند. پیش از وارد کردن آدرس مشخص کنید ابزار شما آدرس خام می‌خواهد یا شماره نمایشی.

کاربرد	آدرس خام	آدرس Hex	شماره نمایشی
فرکانس هدف	1	0x0001	40002
فرمان حرکت	2	0x0002	40003
جریان با FC03	5	0x0005	40006
جریان با FC04	5	0x0005	30006
وضعیت	14	0x000E	40015
مد سوئیچینگ	258	0x0102	40259

شماره نمایشی متداول Holding برابر 40001 به علاوه آدرس خام است. برای وضعیت، آدرس خام 14 یا شماره نمایشی 40015 استفاده می‌شود. عدد 4 ابتدای شماره نمایشی داخل پیام ارسال نمی‌شود. رجیستر صفر در این نقشه معتبر نیست.

دهدهی و Hex

هر دو یک عدد را به دو روش می‌نویسند. عدد 14 دهدهی در مبنای شانزده برابر 0x0E است. پیشوند 0x علامت نمایش Hex است و خودش بایت ارسالی نیست. نمونه فریم‌های راهنما با عددهای Hex نوشته شده‌اند.

جزء فریم	اندازه	کاربرد
Slave address	1 byte	شناسه 1 برای درایو
Function code	1 byte	نوع درخواست
Data	وابسته به تابع	آدرس، تعداد، مقدار یا پاسخ
CRC	2 bytes	کنترل سالم‌رسیدن پیام

در آدرس و مقدار 16 بیتی، بایت پرارزش ابتدا ارسال می‌شود. مقدار 500 برابر 0x01F4 است؛ ابتدا بایت 0x01 و سپس بایت 0xF4 را بفرستید. فقط CRC ترتیب کم‌ارزش‌اول دارد. بایت باینری ارسال کنید، نه متن کاراکترهای هگز؛ شناسه 1 یک بایت با مقدار 0x01 است.

انتخاب نوع درخواست

کد تابع مشخص می‌کند درایو چه کاری انجام دهد. برای راه‌اندازی معمول، سه تابع کافی است: FC03 برای خواندن، FC06 برای نوشتن یک مقدار و FC10 برای نوشتن جفت تنظیم سوئیچینگ. سایر توابع جدول برای ساخت مستر کامل‌تر و عیب‌یابی به کار می‌روند.

کد Hex و دهدهی	کاربرد	روش استفاده
0x03 / 3	خواندن رجیسترها	خواندن فرمان، تنظیم یا داده پایش؛ حداکثر 125 رجیستر معتبر
0x04 / 4	خواندن داده پایش	برای مقادیر آدرس‌های 5 تا 14 این راهنما
0x06 / 6	نوشتن یک رجیستر	ارسال فرمان هدف، فرمان حرکت یا حافظه مرجع
0x08 / 8	آزمون ارتباط	بازگرداندن داده آزمون و خواندن شمارنده‌ها
0x10 / 16	نوشتن چند رجیستر	مقادیر پیوسته در یک پیام؛ حداکثر 123 رجیستر
0x17 / 23	نوشتن سپس خواندن	حداکثر 121 رجیستر نوشتن و 125 رجیستر خواندن
0x2B / 43	اطلاعات شناسایی	نوع شناسه MEI برابر 0x0E

تعداد مجاز با آدرس معتبر فرق دارد

حد 125 به معنی وجود 125 رجیستر پیوسته در محصول نیست. تمام آدرس‌های یک درخواست باید معتبر باشند. خواندن از آدرس 1 به تعداد 15 معتبر است؛ تعداد 16، آدرس نامعتبر 16 را هم وارد درخواست می‌کند و کل خواندن رد می‌شود. تنظیمات سوئیچینگ را جداگانه از آدرس 256 به تعداد 7 بخوانید.

نام تابع در نرم‌افزارها

بعضی ابزارها کد را دهدهی نمایش می‌دهند. گزینه 16 دهدهی همان 0x10 و گزینه 23 دهدهی همان 0x17 است. تابع 0x16 هگز با نوشتن چند رجیستر فرق دارد و در این رابط پشتیبانی نمی‌شود. توابع Coil برای فرمان حرکت به کار نمی‌روند؛ RUN و STOP با رجیستر 2 انجام می‌شوند.

رجیسترهای فرمان و تنظیم

خواندن، مقدار ثبت شده را دریافت می‌کند؛ نوشتن، تنظیم یا فرمان جدید می‌فرستد. در جدول، «فقط خواندن» روش استفاده برنامه کاربر است. آدرس‌های فرمان را با آدرس‌های پایش اشتباه نگیرید.

آدرس	کاربرد	استفاده کاربر	مقدار و واحد
1	فرکانس هدف	خواندن و نوشتن	واحد 0.1 Hz؛ مقدار 500 یعنی 50.0 Hz
2	فرمان حرکت	نوشتن	مقدار 1: توقف؛ مقدار 2: مستقیم؛ مقدار 4: معکوس
2	روش دریافت فرمان	خواندن	مقدار ثابت 3 به معنی فرمان‌گیری از مدباس
3	زمان افزایش سرعت ACC	فقط خواندن	واحد 0.1 s؛ مقدار 200 یعنی 20.0 ثانیه
4	زمان کاهش سرعت DEC	فقط خواندن	واحد 0.1 s؛ مقدار 10 یعنی 1.0 ثانیه
15	حافظه فرکانس هدف	خواندن و نوشتن	مقدار 0: غیرفعال؛ مقدار 1: فعال

ACC و DEC برای نمایش هستند

ACC زمان افزایش سرعت و DEC زمان کاهش سرعت را بیان می‌کنند. در برنامه کاربر این دو آدرس را فقط بخوانید و قابل ویرایش نمایش ندهید. درایو زمان حرکت را هنگام راه‌اندازی و تغییر وضعیت کنترل خودش تعیین می‌کند. عدد خوانده شده، اندازه‌گیری زمان واقعی حرکت محور نیست. پذیرش یک نوشتن در این دو آدرس، تنظیم پایدار زمان حرکت نیست.

خواندن رجیستر فرمان، وضعیت حرکت نیست

نوشتن مقدار 2 در آدرس 2 فرمان حرکت مستقیم است. پاسخ موفق نوشتن همان مقدار 2 را برمی‌گرداند، اما خواندن آدرس 2 مقدار 3 و روش فرمان‌گیری را نشان می‌دهد. برای تشخیص RUN یا STOP، آدرس 14 را بخوانید. مقدار 3، فرمان حرکت مجاز نیست. جهت واقعی محور نیز با خواندن آدرس 2 اندازه‌گیری نمی‌شود.

فرکانس مجاز

فرکانس صفر پذیرفته می‌شود. فرکانس غیرصفر باید در محدوده مجاز موتور و تنظیمات محصول باشد؛ مقدار خارج از محدوده با خطای پروتکل 03 رد می‌شود. انتخاب نمایش RPM، واحد رجیستر 1 را عوض نمی‌کند. برای لغو فرمان حرکت، STOP بفرستید؛ صفرکردن فرکانس جای STOP نیست.

خواندن مقادیر و تبدیل واحد

داده دریافتی مدباس عدد خام است. در جدول، همان عدد رجیستر، V_{dc} ولتاژ باس بر حسب ولت و f فرکانس الکتریکی بر حسب هرتز است. همه رجیسترهای این صفحه فقط خواندنی‌اند. برای خواندن یک‌جای آن‌ها، FC04 یا FC03 را از آدرس 5 به تعداد 10 اجرا کنید.

آدرس	کمیت	تبدیل مقدار خام
5	جریان نمایشگر	$I = raw / 10 \text{ A}$
6	فرکانس الکتریکی	$f = raw / 10 \text{ Hz}$
7	ولتاژ خروجی محاسبه‌شده	$V_{out} = raw / 10 \text{ V}$
8	ولتاژ باس DC	$V_{dc} = raw / 10 \text{ V}$
9	دمای درایو	$T = raw / 10 \text{ }^{\circ}\text{C}$
10	جریان محور گشتاور q	$I_q = (raw - 100) / 10 \text{ A}$
11	جریان محور شار d	$I_d = (raw - 100) / 10 \text{ A}$
12	ولتاژ محور q	$V_q = ((raw \times 10 - 262144) / 262144) \times V_{dc}$
13	ولتاژ محور d	$V_d = ((raw \times 10 - 262144) / 262144) \times V_{dc}$
14	وضعیت اجرا یا خطا	کد 16 بیتی مطابق جدول وضعیت

برای نمایش معمول، جریان، فرکانس، ولتاژ و وضعیت کافی‌اند. محورهای q و d ، دو مؤلفه محاسباتی کنترل موتور هستند؛ جریان آن‌ها با جریان فاز موتور یکسان نیست. در تبدیل این دو جریان، ابتدا عدد خام را به نوع علامت‌دار یا اعشاری تبدیل و سپس 100 را کم کنید. مقدار خام 75 برابر منفی 2.5 آمپر است. حد پایین نمایش این دو کانال منفی 10 آمپر است.

جریان نمایشگر فیلترشده است؛ فرکانس نمایشگر نیز اندازه‌گیری مستقل سرعت محور نیست. تازه‌سازی فرکانس در کار عادی حدود یک ثانیه است. ولتاژ خروجی محاسبه می‌شود و ولتاژ باس فیلترشده است. دمای منفی به‌صورت صفر گزارش می‌شود. در فرمول ولتاژهای q و d از محاسبات اعشاری یا عدد علامت‌دار با ظرفیت کافی استفاده کنید.

تبدیل فرکانس به دور موتور

برای نمایش RPM، تعداد کل قطب‌های موتور را وارد کنید. رابطه دور مکانیکی برابر $RPM = 120 \times f / P$ است؛ P تعداد کل قطب‌هاست. مثال: فرکانس 60 هرتز و موتور 12 قطب، 600 دور در دقیقه می‌دهد. تعداد جفت‌قطب را به‌جای تعداد کل قطب وارد نکنید. این تبدیل فقط واحد نمایش را عوض می‌کند و فرمان فرکانس همچنان بر حسب هرتز است.

وضعیت کار و کدهای خطا

رجیستر 14 یک کد وضعیت است؛ غیرصفر بودن آن به معنی خطا نیست. 0xA000 حالت START و 0xA001 حالت STOP بدون خطای فعال را نشان می‌دهند. برای پاسخ نرم‌افزار به هر وضعیت از جدول زیر استفاده کنید؛ تشخیص علت فیزیکی و رفع حفاظت با راهنمای فنی موتور و درایو انجام می‌شود.

کد Hex	نام	برداشت برای مستر
0xA000	START	درایو در حالت کار بدون خطای فعال
0xA001	STOP	درایو در حالت توقف بدون خطای فعال
0xA002	OCt	اضافه‌جریان سخت‌افزاری
0xA003	OLt	اضافه‌بار
0xA004	LVt	افت ولتاژ
0xA005	OHt	دمای بیش از حد
0xA006	HWE	خطای سخت‌افزار
0xA007	OVt	اضافه‌ولتاژ
0xA008	POF	قطع فاز خروجی
0xA009	EST	توقف اضطراری
0xA00A	OC2	اضافه‌جریان نرم‌افزاری
0xA00B	IOL	اضافه‌بار وابسته به زمان
0xA00C	ntC	خطای حسگر دما
0xA00D	CSE	خطای حسگر جریان
0xA00E	EEP	خطای حافظه تنظیمات
0xA00F	PVt	حفاظت ولتاژ PVt
0xA010	OAs	خطای هم‌زمانی حلقه باز
0xA011	CAs	خطای استال حلقه بسته

کد ناشناخته را به صورت Hex دقیق نمایش دهید و آن را به START تبدیل نکنید. رجیستر 14 فقط خواندنی است. فرمان پاک‌کردن خطای موتور از طریق این رابط ارائه نشده است. نوشتن صفر به رجیستر وضعیت و آزمون ارتباط، خطای موتور را پاک نمی‌کند. فرمان STOP برای توقف است؛ رفع خطای حفاظتی باید طبق دستورالعمل همان خطا انجام شود.

آشنایی با سوئیچینگ و رندوم فرکانسی

درایو برای ساخت ولتاژ مناسب موتور، کلیدهای قدرت را با سرعت زیاد روشن و خاموش می‌کند. به این عمل سوئیچینگ می‌گویند. روش تنظیم طول و ترتیب این پالس‌ها PWM نام دارد. با تغییر پالس‌ها، ولتاژ لازم برای حرکت موتور ساخته می‌شود.

دو فرکانس با دو کاربرد متفاوت

مثال	کاربرد	کمیت
مرجع 60 Hz در رجیستر 1	فرمان سرعت و محاسبه RPM	فرکانس الکتریکی موتور
12000 تا 14000 Hz در حالت رندوم	سرعت تکرار پالس‌های کلیدهای قدرت	فرکانس سوئیچینگ

فرکانس سوئیچینگ بسیار بالاتر از فرکانس موتور است. مقدار 13000 هرتز در رجیستر حامل به معنی سرعت 13000 هرتز موتور نیست. برای فرمان سرعت از رجیستر 1 و برای مشاهده حامل از رجیستر 262 استفاده کنید.

SVPWM و DPWM چه هستند

مقدار انتخاب	توضیح ساده	مد
1	روش بردار فضایی؛ پالس‌های سه خروجی با الگویی هماهنگ برای ساخت ولتاژ موتور تولید می‌شوند.	SVPWM
2	روش ناپیوسته؛ در بخش‌هایی از چرخه، وضعیت یک شاخه ثابت نگه داشته می‌شود تا تعداد کلیدزنی کاهش یابد.	DPWM

هر دو گزینه روش تولید پالس‌های خروجی‌اند. انتخاب آن‌ها، نوع فرمان RUN یا واحد سرعت را تغییر نمی‌دهد. تغییر مد فقط در توقف انجام می‌شود. مقدار ذخیره‌شده دستگاه را ابتدا بخوانید؛ برای انتخاب مد مناسب، دستور راه‌اندازی همان مجموعه موتور و درایو را مبنا قرار دهید.

رندوم فرکانسی چیست

در حالت رندوم، فرکانس سوئیچینگ در بازه 12 تا 14 کیلوهرتز تغییر می‌کند. هدف، پخش کردن انرژی کلیدزنی در یک بازه و کاهش تمرکز صدای تونال است. فرکانس هدف موتور با فعال کردن این گزینه تغییر نمی‌کند. خاموش یعنی مقدار 0 و روشن یعنی مقدار 1 در رجیستر 259.

روشن بودن گزینه، اجازه استفاده از رندوم را فعال می‌کند. درایو اجرای آن را متناسب با مرحله کار مدیریت می‌کند. برای مشاهده فرکانس سوئیچینگ همان لحظه، مقدار رجیستر 262 را بخوانید. حدود 12 و 14 کیلوهرتز فقط خواندنی هستند.

تنظیم سوئیچینگ از طریق مدباس

این بخش دو انتخاب کاربر را به داده‌های مدباس تبدیل می‌کند: نوع سوئیچینگ و روشن یا خاموش بودن رندوم. برای خواندن تمام اطلاعات جدول، FC03 را از آدرس 256 به تعداد 7 اجرا کنید. دسترسی «خواندن و نوشتن» فقط برای آدرس‌های 258 و 259 است.

آدرس	دسترسی	معنی و مقدار
256	فقط خواندن	کد شناسایی قابلیت: 0x504D
257	فقط خواندن	نسخه قالب این قابلیت: 1
258	خواندن و نوشتن	مقدار 1: SVPWM؛ مقدار 2: DPWM
259	خواندن و نوشتن	مقدار 0: رندوم خاموش؛ مقدار 1: رندوم روشن
260	فقط خواندن	حد پایین رندوم: Hz 12000
261	فقط خواندن	حد بالای رندوم: Hz 14000
262	فقط خواندن	فرکانس سوئیچینگ جاری بر حسب Hz

چرا ابتدا کد شناسایی خوانده می‌شود

کد 0x504D و نسخه 1 تأیید می‌کنند که پاسخ مربوط به همین قالب تنظیمات است. تنها پس از تطبیق این دو مقدار، دکمه اعمال تنظیمات را فعال کنید. در پاسخ نامعتبر، پیغام ناسازگاری نشان دهید و از نوشتن تنظیمات جلوگیری کنید.

اعمال دو انتخاب در یک پیام

گام	عملیات
۱	فرمان STOP بفرستید و وضعیت توقف تازه را بخوانید.
۲	نوع سوئیچینگ و وضعیت رندوم را از کاربر بگیرید.
۳	با FC10 از آدرس 258 به تعداد 2، هر دو مقدار را بنویسید.
۴	پاسخ را بررسی کنید؛ سپس آدرس‌های 258 و 259 را دوباره بخوانید.
۵	فقط پس از تطبیق دو مقدار، پیام تأیید تنظیم را نمایش دهید.

اعمال تنظیمات به توقف کامل خروجی، نبود فرمان حرکت یا عملیات شناسایی موتور و سلامت حافظه نیاز دارد. کد خطای پروتکل 06 یعنی شرایط اجرا آماده نیست؛ وضعیت را دوباره بخوانید و پس از آماده شدن اقدام کنید. این جفت تنظیم با کنترل ذخیره در حافظه ثبت می‌شود. در قطع پاسخ، ابتدا بازخوانی کنید و نتیجه را از مقدار واقعی تشخیص دهید.

ترتیب راه اندازی و توقف

نوشتن فرکانس و ارسال فرمان حرکت دو کار جدا هستند. ابتدا فرکانس هدف را تنظیم کنید و سپس RUN بدهید. پذیرش پیام RUN، رسیدن موتور به سرعت هدف را اعلام نمی‌کند؛ وضعیت و فرکانس را پس از فرمان پایش کنید.

گام	عملیات	شرط عبور
۱	بازکردن پورت و خواندن وضعیت	شناسه، CRC و طول پاسخ درست باشند
۲	نوشتن مقدار 1 در رجیستر 2	دریافت پاسخ و وضعیت STOP
۳	بررسی سلامت و تنظیمات سوئیچینگ	خطای فعال وجود نداشته باشد
۴	اعمال تغییر سوئیچینگ در صورت نیاز	نوشتن و بازخوانی تأیید شود
۵	نوشتن فرکانس هدف در رجیستر 1	مقدار هدف بازخوانی شود
۶	نوشتن 2 برای مستقیم یا 4 برای معکوس	فرمان آگاهانه کاربر یا منطق ماشین
۷	خواندن دوره‌ای آدرس‌های 5 تا 14	وضعیت و زمان دریافت تازه باشد
۸	نوشتن مقدار 1 برای توقف	وضعیت STOP و توقف مورد نیاز ماشین

صفرکردن فرکانس جای توقف نیست

این محصول فرمان حرکت و فرکانس هدف را از مدباس دریافت می‌کند و برای فعال کردن ارتباط به تنظیم دیگری نیاز ندارد. صفرکردن فرکانس، فرمان RUN قبلی را لغو نمی‌کند؛ با فرکانس غیرصفر بعدی، فرمان حرکت همچنان فعال است. برای لغو حرکت، مقدار 1 را در رجیستر 2 بنویسید.

تغییر جهت و وصل مجدد ارتباط

برای تغییر جهت، نخست STOP بفرستید، شرایط توقف ماشین را بررسی کنید و سپس RUN جهت دیگر را بدهید. وضعیت STOP، سنجش صفرشدن واقعی سرعت محور نیست. توقف نرم‌افزاری نیز جای توقف اضطراری ماشین را نمی‌گیرد.

پس از وصل مجدد کابل یا بازکردن نرم‌افزار، ابتدا وضعیت را بخوانید. برگشت ارتباط نباید خودکار RUN صادر کند. اگر پاسخ یک فرمان قطع شده است، دریافت‌نشده پاسخ را شکست قطعی آن فرمان فرض نکنید؛ وضعیت و تنظیم مربوط را بازخوانی کنید.

نمونه پیام‌های خواندن و فرمان تکی

همه نمونه‌ها برای شناسه دستگاه 1 هستند. فاصله‌ها فقط برای خوانایی‌اند. خطوط RX پاسخ موفق را نشان می‌دهند. رد درخواست با پاسخ Exception گزارش می‌شود. CRCها برای همین بایت‌ها محاسبه شده‌اند؛ با تغییر آدرس یا مقدار، CRC را دوباره بسازید.

خواندن وضعیت توقف

```
TX 01 03 00 0E 00 01 E5 C9
RX 01 03 02 A0 01 01 84
```

در پاسخ، مقدار 0x02 تعداد بایت‌های داده است. مقدار رجیستر در توقف 0xA001 و در کار 0xA000 است؛ با تغییر داده پاسخ، CRC نیز تغییر می‌کند.

خواندن یکپارچه آدرس‌های 1 تا 14

```
TX 01 03 00 01 00 0E 95 CE
```

پاسخ FC03 دارای Byte Count = 28 یا 0x1C است و 14 مقدار به ترتیب آدرس 1 تا 14 دارد. طول کامل پاسخ با CRC برابر 33 بایت است.

خواندن پایش با FC04 از آدرس 5 به تعداد 10

```
TX 01 04 00 05 00 0A 60 0C
```

ثبت مرجع 50.0 هرتز

```
TX 01 06 00 01 01 F4 D8 1D
RX 01 06 00 01 01 F4 D8 1D
```

فرمان RUN مستقیم

```
TX 01 06 00 02 00 02 A9 CB
RX 01 06 00 02 00 02 A9 CB
```

فرمان RUN معکوس

```
TX 01 06 00 02 00 04 29 C9
RX 01 06 00 02 00 04 29 C9
```

فرمان STOP

```
TX 01 06 00 02 00 01 E9 CA
RX 01 06 00 02 00 01 E9 CA
```

نوشتن چند مقدار در یک پیام

وقتی دو تنظیم به یکدیگر مربوطاند، می‌توان آن‌ها را در یک درخواست نوشت. تابع FC10 برای چند رجیستر پیوسته است؛ «چند رجیستر» در اینجا به چند داده‌همین یک درایو اشاره دارد.

مثال انتخاب SVPWM و روشن کردن رندوم

```
TX 01 10 01 02 00 02 04 00 01 00 01 EF E6
RX 01 10 01 02 00 02 E1 F4
```

آدرس شروع 0x0102 یعنی 258 است. تعداد رجیسترها 2 و تعداد بایت‌های داده 4 است. مقدار آدرس 258 برابر 1 برای SVPWM و مقدار آدرس 259 برابر 1 برای رندوم روشن است. پاسخ موفق، آدرس و تعداد را تأیید می‌کند؛ برای تأیید مقادیر، هر دو رجیستر را دوباره بخوانید.

منظور از اعمال یک‌جای تنظیمات

در این جفت تنظیم، هر دو مقدار باید معتبر باشند و شرایط توقف برقرار باشد. سپس تنظیمات به صورت یک مجموعه ذخیره و کنترل می‌شوند. به چنین رفتاری اعمال یک‌جا یا اتمیک می‌گویند. برای آدرس‌های 258 و 259 از یک درخواست FC10 استفاده کنید.

این رفتار را به هر ترکیب دلخواه از فرمان‌ها تعمیم ندهید. در راه‌اندازی موتور، ابتدا فرکانس را بنویسید و تأیید کنید؛ سپس فرمان RUN را در پیام جداگانه بفرستید. ACC و DEC در توالی تنظیم کاربر قرار ندارند.

مثال ثبت مرجع و سپس خواندن با FC17

```
TX 01 17 00 01 00 0E 00 01 00 01 02 01 F4 44 ED
```

FC17 ابتدا مقدار جدید را می‌نویسد و سپس داده‌ها را می‌خواند. در این مثال، مقدار 500 یعنی 50.0 هرتز در آدرس 1 نوشته می‌شود و آدرس‌های 1 تا 14 خوانده می‌شوند. پاسخ، داده‌های خوانده‌شده را با کد تابع 0x17 برمی‌گرداند.

این تابع دو کار ارتباطی را در یک درخواست جمع می‌کند. مقادیر پاسخ، نمونه‌برداری هم‌زمان فیزیکی از همه کمیت‌های موتور نیستند. برای شروع کار، پیام‌های جداگانه FC06 و FC03 ساده‌تر قابل بررسی‌اند.

نگهداری تنظیمات پس از خاموشی

بعضی تنظیمات فقط هنگام روشن بودن دستگاه استفاده می‌شوند و بعضی در حافظه می‌مانند. پیش از طراحی برنامه مشخص کنید آیا فرکانس هدف باید پس از خاموش و روشن شدن حفظ شود. این انتخاب از طریق رجیستر 15 انجام می‌شود.

داده	رفتار پس از خاموشی
فرکانس هدف در آدرس 1	با فعال بودن حافظه مرجع ذخیره می‌شود.
حافظه مرجع در آدرس 15	مقدار 1: حفظ مرجع؛ مقدار 0: ذخیره صفر برای راه اندازی بعدی
ACC و DEC در آدرس های 3 و 4	تنظیم قابل ذخیره کاربر نیستند؛ فقط برای خواندن استفاده شوند.
سوئیچینگ و رندوم در 258 و 259	هر دو تنظیم در حافظه نگهداری می‌شوند.
فرمان RUN	ذخیره فرکانس، فرمان شروع خودکار صادر نمی‌کند.

مثال فعال کردن حافظه فرکانس هدف

```
TX 01 06 00 0F 00 01 78 09
RX 01 06 00 0F 00 01 78 09
```

پس از این فرمان، مرجع مورد نیاز را در رجیستر 1 بنویسید و آن را بازخوانی کنید. غیرفعال کردن حافظه، مرجع فعلی هنگام کار را صفر نمی‌کند؛ مقدار نگهداری شده برای روشن شدن بعدی صفر می‌شود.

پاسخ نوشتن با پایان ذخیره فرق دارد

ذخیره مرجع فرکانس و گزینه حافظه با زمان بندی دستگاه انجام می‌شود. پاسخ موفق به این دو نوشتن، پایان فیزیکی ذخیره را اعلام نمی‌کند. بلافاصله پس از پاسخ، برق دستگاه را قطع نکنید. رجیستر جداگانه ای برای اعلام پایان ذخیره ارائه نشده است.

تنظیم حافظه را فقط هنگام تغییر انتخاب کاربر بنویسید. آن را در هر دور پایش تکرار نکنید. در آزمون راه اندازی، پس از ثبت تنظیمات و توقف کامل، یک خاموش و روشن کردن کنترل شده انجام دهید و مقدار مرجع و تنظیم سوئیچینگ را دوباره بخوانید.

تأیید نوشتن با پاسخ و بازخوانی

تمام درخواست های این راهنما را با شناسه دستگاه 1 ارسال کنید. آدرس دستگاه صفر پاسخ تأیید ندارد و در این روش راه اندازی استفاده نمی‌شود. نبود پاسخ به یک نوشتن، مجوز تکرار خودکار فرمان حرکت نیست؛ ابتدا وضعیت را بخوانید.

آزمون ارتباط و شمارنده‌های خطا

پیش از فرمان حرکت، می‌توان فقط مسیر ارسال و دریافت را آزمایش کرد. در آزمون Echo، یک داده آزمایشی فرستاده می‌شود و درایو همان را برمی‌گرداند. این آزمون فرمان حرکت صادر نمی‌کند. برابر بودن درخواست و پاسخ همراه با CRC درست، دریافت سالم همان پیام را تأیید می‌کند.

نمونه Echo با تابع FC08

```
TX 01 08 00 00 12 34 ED 7C
RX 01 08 00 00 12 34 ED 7C
```

زیرتابع چیست

FC08 چند کار مرتبط با عیب‌یابی دارد. عدد دیگری به نام زیرتابع، کار دقیق را انتخاب می‌کند. جدول زیر زیرتابع‌ها را نشان می‌دهد. شمارنده‌ها تعداد رخدادهای ارتباطی را گزارش می‌کنند و با خطاهای حفاظتی موتور فرق دارند.

زیرتابع Hex	کاربرد
0x0000	بازگرداندن داده آزمون Echo
0x0002	خواندن پرچم‌های خطای ارتباط
0x000A	پاک‌کردن شمارنده‌ها و پرچم‌های ارتباط
0x000B	تعداد پیام‌های معتبر دیده‌شده روی خط
0x000C	تعداد خطاهای ارتباط
0x000D	تعداد پاسخ‌های رد درخواست
0x000E	تعداد پیام‌های مربوط به این دستگاه
0x000F	تعداد درخواست‌های بدون پاسخ
0x0010	تعداد پاسخ‌های عدم پذیرش
0x0011	تعداد پاسخ‌های مشغول
0x0012	تعداد سرریز دریافت کاراکتر
0x0014	پاک‌کردن شمارنده و پرچم سرریز دریافت

به‌جز Echo، مقدار داده درخواست زیرتابع‌های جدول باید صفر و طول کامل فریم 8 بایت باشد. Echo داده‌ای با تعداد بایت زوج را بازمی‌گرداند؛ سقف کامل فریم 256 بایت است. در رجیستر پرچم، بیت 0 نشان‌دهنده خطای ارتباط و بیت 1 نشان‌دهنده سرریز دریافت است.

شمارنده‌ها تا 65535 افزایش می‌یابند و در همین مقدار می‌مانند. پس از خاموشی حفظ نمی‌شوند. پاک‌کردن این شمارنده‌ها، خطای حفاظتی موتور را پاک نمی‌کند. برای دیدن خطای موتور، رجیستر 14 را بخوانید.

خواندن اطلاعات شناسایی دستگاه

تابع شناسایی، نام سازنده و شناسه‌های متنی دستگاه را می‌خواند. پاسخ این تابع با پاسخ رجیستر فرق دارد: هر داده یک شماره، طول و متن دارد. این داده‌ها «شیء شناسایی» نامیده می‌شوند. برای راه‌اندازی معمول، همین اطلاعات خوانده می‌شوند و تنظیمی نوشته نمی‌شود.

درخواست اطلاعات شناسایی

TX 01 2B 0E 01 00 70 77

شماره شیء	عنوان	متن گزارش شده
0	نام سازنده	HiPowerVSES
1	شناسه محصول در پروتکل	PMSM-55
2	شناسه نسخه در پروتکل	V1.0

این متن‌ها شناسه‌های ارتباطی هستند. برای انتخاب مدل PM300-12k0 یا PM300-2k5، نام روی محصول را مبنا قرار دهید. برای تنظیم سوئیچینگ، کد شناسایی و نسخه قالب در آدرس‌های 256 و 257 نیز باید تطبیق داده شوند.

ترتیب خواندن پاسخ

S | 2B | 0E | ReadCode | Conformity | MoreFollows
 | NextObject | ObjectCount
 | ObjectID | Length | Text ... | CRC

S شناسه دستگاه و ReadCode نوع درخواست است. ObjectCount تعداد اشیای همین پاسخ است. برای هر شیء، ابتدا شماره و طول را بخوانید و سپس دقیقاً به همان تعداد بایت متن بردارید. متن را مانند عدد 16 بیتی رجیستر تبدیل نکنید.

فیلد	رفتار
ReadCode از 1 تا 3	خواندن اشیای موجود از شماره شروع
ReadCode برابر 4	خواندن یک شیء با شماره مشخص
Conformity	مقدار 0x81
MoreFollows	مقدار 0: بخش دیگری برای دنبال کردن وجود ندارد

برای خواندن معمول، درخواست نمونه را ارسال کنید. برای خواندن تکی با کد 4، فقط شماره‌های 0، 1 و 2 را به کار ببرید. شناسه نامعتبر، طول ناقص یا CRC نادرست را به عنوان پاسخ شناسایی معتبر ثبت نکنید.

راه‌اندازی با PLC و HMI

در PLC، یک بلوک ارتباطی وظیفه ساخت درخواست و دریافت پاسخ را انجام می‌دهد. در HMI نیز درایور Modbus RTU Master همین نقش را دارد. جدول زیر مقدارهای لازم برای نخستین خواندن را نشان می‌دهد.

بلوک ارتباط PLC را در حالت RTU Master تنظیم کنید. نام پارامترهای بلوک در برندها متفاوت است، ولی اطلاعات زیر مشترک‌اند. انتخاب Modbus TCP یا اتصال Ethernet به‌تنهایی با پورت RS485 درایو ارتباط برقرار نمی‌کند؛ برای این حالت به Gateway تبدیل TCP به RTU با تنظیم سمت سریال نیاز دارید.

ورودی بلوک مستر	مثال خواندن وضعیت	نکته
Port / Channel	پورت RS485 متصل به درایو	8N1 و 9600
Slave / Unit ID	1	شناسه ثابت این اتصال
Function / Mode	Read Holding Registers	FC03
Start address	14 خام یا 40015 نمایشی	روش آدرس‌دهی همان بلوک را بررسی کنید
Quantity	1 register	طول بر حسب واژه، نه بایت
Data buffer	WORD[1]	مقدار دریافت‌شده 16 بیتی است
Trigger	لبه درخواست طبق بلوک	تا BUSY پایان نیافته درخواست بعدی ندهید

توالی برنامه چرخه‌ای

در حالت IDLE درخواست را آماده کنید؛ پس از Trigger وارد WAIT شوید. تا BUSY فعال است همان تراکنش را نگه دارید. با DONE، نتیجه و وضعیت را تحلیل کنید و پس از فاصله مجاز به درخواست بعدی بروید. با ERROR، کد خطای بلوک، Timeout یا Exception را تفکیک کنید و با شمارنده تلاش محدود وارد بازیابی شوید. START برنامه یا برگشت ارتباط نباید خودکار فرمان RUN تولید کند.

برای فرمان 50 Hz یک WORD با مقدار 500 و برای RUN مستقیم یک WORD با مقدار 2 بنویسید. داده پایش را ابتدا در WORD خام نگه دارید و سپس به REAL تبدیل کنید؛ مثلاً جریان رجیستر 5 برابر WORD_TO_REAL(raw) 10 / است. برای جریان محور q، پیش از کم‌کردن 100 عدد را به نوع علامت‌دار یا REAL تبدیل کنید تا مقادیر منفی درست ساخته شوند.

اگر بلوک PLC عدد 40015 می‌خواهد، همان را وارد کنید؛ اگر Offset می‌خواهد، 14 بدهید. شماره D یا DB محل ذخیره داده در PLC است و آدرس رجیستر درایو نیست. صحت این انتخاب را با پاسخ شناخته‌شده STOP و سپس رجیسترهای دیگر بررسی کنید.

پیاده‌سازی مستر با میکروکنترلر

ماشین حالت یعنی برنامه، کار ارتباط را به گام‌های کوتاه تقسیم کند: آماده‌سازی، ارسال، انتظار پاسخ و بررسی نتیجه. در زمان انتظار، سایر کارهای میکروکنترلر ادامه پیدا می‌کنند.

پروتکل را به شکل ماشین حالت پیاده کنید تا انتظار UART اجرای سایر وظایف را متوقف نکند. UART داده را با قالب درایو ارسال می‌کند؛ ترنسیور RS485 تبدیل سطح و انتخاب جهت را انجام می‌دهد. در مبدل خودکار، جهت توسط سخت‌افزار مدیریت می‌شود؛ در ترنسیور معمولی، پایه DE و در صورت نیاز RE باید کنترل شوند.

حالت	کار لازم	خروج از حالت
IDLE	رعایت سکوت باس و انتخاب یک درخواست	موعد Poll یا فرمان کاربر
TX	فعال‌کردن ارسال، فرستادن فریم کامل	پرچم TC پس از آخرین Stop bit
RX	آزادکردن فرستنده و دریافت پاسخ	فریم کامل، خطا یا Timeout
VALIDATE	بررسی Slave، تابع، طول، CRC و داده	پاسخ معتبر یا خطای مشخص
RECOVERY	پاک‌سازی کنترل‌شده دریافت و مکث	آماده‌شدن باس برای تلاش محدود بعدی

ترتیب بررسی پاسخ

۱. طول حداقل و سقف بافر را کنترل کنید؛ بیشینه کامل فریم RTU برابر 256 بایت است. ۲. CRC کل پیام بدون دو بایت پایانی را با CRC دریافتی مقایسه کنید. ۳. Slave ID باید با درخواست یکسان باشد. ۴. اگر تابع برابر تابع درخواست OR 0x80 است، پاسخ Exception را تفسیر کنید. ۵. در پاسخ عادی، Byte Count یا Echo آدرس و مقدار را تطبیق دهید. ۶. فقط پس از این بررسی‌ها داده کاربر را به‌روز کنید.

در FC03 و FC04 پس از دریافت Byte Count، طول مورد انتظار برابر 5 به‌علاوه Byte Count است. پاسخ موفق FC06 و FC10 هشت بایت است؛ پاسخ Exception پنج بایت است. Device Identification طول متغیر دارد. از ثبت «داده صفر» به جای یک پاسخ ناقص خودداری کنید؛ پاسخ ناقص باید به‌عنوان داده نامعتبر باقی بماند.

وقفه RX فقط بایت و زمان یا وضعیت خطا را ثبت کند؛ تحلیل پروتکل و پردازش طولانی را در وظیفه اصلی انجام دهید. اگر DMA دارید، پایان DMA به‌تنهایی تضمین پایان آخرین بیت UART نیست. کنترل DE را با وضعیت TC هماهنگ کنید تا CRC انتهای پیام قطع نشود. کد مستقل CRC در ضمیمه آمده است.

راه اندازی با نرم افزار رایانه

نرم افزار عمومی مدباس

مبدل را متصل و پورت COM آن را انتخاب کنید. تنظیمات را روی 9600 و 8N1، شناسه را 1 و نوع ارتباط را Modbus RTU قرار دهید. نخست با FC03 از آدرس خام 14 یک رجیستر بخوانید. در ابزارهایی که شماره نمایشی می‌خواهند، همین رجیستر 40015 است.

هنگام استفاده از کتابخانه یا نرم افزاری که فریم مدباس را می‌سازد، CRC را خودتان به مقدار رجیستر اضافه نکنید. در ابزار ارسال خام بایت، تمام فریم شامل CRC ارسال می‌شود. این دو روش ورود داده را با یکدیگر ترکیب نکنید.

نرم افزار Drive Studio

مدل درج شده روی محصول را انتخاب کنید و اتصال را برقرار سازید. گزینه‌های سوئیچینگ و رندوم در DRIVE SETPOINTS قرار دارند. پس از انتخاب، فرمان اعمال را اجرا کنید و تأیید بازخوانی را ببینید. تغییر یک گزینه روی صفحه، بدون اعمال آن، تنظیم دستگاه را تغییر نمی‌دهد.

برای نمایش RPM، تعداد کل قطب‌های موتور را وارد کنید. تب CSV نمونه‌های ذخیره شده را به صورت زنده نشان می‌دهد. مکث نمودار فقط نمایش نمودار را متوقف می‌کند؛ پایان ثبت فایل با فرمان توقف ثبت CSV انجام می‌شود.

خواندن دوره‌ای بدون فشار اضافی

Polling یعنی نرم افزار در فاصله‌های مشخص داده را دوباره بخواند. برای پایش اصلی، آدرس‌های 1 تا 14 را در یک FC03 بخوانید. این کار یک درخواست 8 بایتی و پاسخ 33 بایتی دارد. در تنظیم این اتصال، زمان عبور همین بایت‌ها حدود 42.7 میلی ثانیه است؛ زمان پاسخ و سکوت نیز به آن افزوده می‌شود.

دوره پایش را ابتدا 500 میلی ثانیه قرار دهید. درخواست جدید را پیش از پایان قبلی صادر نکنید. تنظیمات ثابت را هنگام اتصال و پس از تغییر کاربر بخوانید. نوشتن مکرر تنظیمات ذخیره شونده در حلقه پایش انجام نشود. فرمان توقف کاربر باید پیش از خواندن‌های عادی منتظر در صف اجرا شود.

زمان دریافت آخرین پاسخ معتبر را کنار داده نگه دارید. تکرار یک عدد با قطع ارتباط یکسان نیست؛ مقدار فرکانس با نرخ پایین‌تری تازه می‌شود. در قطع پاسخ، مقدار قبلی را همراه برچسب داده قدیمی نمایش دهید. صفر ساختگی به جای پاسخ دریافت نشده ذخیره نکنید.

پاسخ رد درخواست و مهلت انتظار

Exception پاسخ مشخص درایو به یک درخواست رد شده است. یعنی پیام دریافت شده، اما عملیات پذیرفته نشده است. در مقابل، Timeout یعنی تا پایان مهلت تعیین شده پاسخ معتبر دریافت نشده است. خطای حفاظتی موتور نیز موضوع جداگانه‌ای است و از رجیستر 14 خوانده می‌شود.

کد	معنی	اقدام
01	تابع نامعتبر	نوع تابع یا نوع درخواست شناسایی را اصلاح کنید.
02	آدرس نامعتبر	آدرس شروع، تعداد و روش آدرس دهی را کنترل کنید.
03	مقدار یا قالب نامعتبر	واحد، مقدار، تعداد و طول داده را کنترل کنید.
04	انجام نشدن عملیات	وضعیت دستگاه و سلامت ذخیره تنظیمات را بررسی کنید.
06	دستگاه آماده اجرا نیست	وضعیت تازه بخوانید و پس از آماده شدن اقدام کنید.

پاسخ Exception پنج بایت دارد. کد تابع پاسخ با افزودن بیت 0x80 به تابع درخواست ساخته می‌شود؛ سپس کد علت و CRC می‌آیند. برای مثال کد تابع در پاسخ خطای FC06 برابر 0x86 است. کد ناشناخته را با عدد دقیق ثبت کنید.

مثال خواندن از آدرس نامعتبر صفر

```
TX 01 03 00 00 00 01 84 0A
RX 01 83 02 C0 F1
```

مثال فرمان حرکت نامعتبر با مقدار 3

```
TX 01 06 00 02 00 03 68 0B
RX 01 86 03 02 61
```

مثال رد تغییر سوئیچینگ هنگام کار

```
RX 01 90 06 CC 02
```

در Timeout، شناسه، قالب سریال، CRC، کامل بودن فریم و کنترل جهت را بررسی کنید. پیام دارای CRC خراب بدون پاسخ کنار گذاشته می‌شود. برای خواندن، تلاش مجدد محدود انجام دهید. برای نوشتن با پاسخ از دست رفته، ابتدا مقدار یا وضعیت را بخوانید؛ فرمان RUN را خودکار تکرار نکنید.

رفع اشکال و آزمون نهایی

نشانه	بررسی هدفمند
TX مستر فعالیت ندارد	بازبودن COM، شروع وظیفه Poll، صف درخواست و اشغال پورت را بررسی کنید. بدون ارسال درخواست، انتظار پاسخ از Slave درست نیست.
TX فعال و پاسخ وجود ندارد	Slave ID، قالب سریال، جهت ترنسیور، قطبیت زوج داده و CRC را با یک Echo یا خواندن کوتاه آزمون کنید.
Exception 02	مبنای خام یا 4xxxx، آدرس شروع و انتهای بازه را بررسی کنید؛ خواندن 1 تا 16 نامعتبر است.
Exception 03	مقیاس 10x، مقدار فرمان 1/2/4 و مقدار مجاز تنظیم را کنترل کنید.
ACK هست ولی موتور راه نمی‌افتد	وضعیت رجیستر 14، مرجع، آمادگی و شرایط حفاظت را بررسی کنید. ACK جای بازخورد حرکت نیست.
فرکانس ثابت ولی جریان تازه است	زمان آخرین پاسخ معتبر را بررسی کنید؛ فرکانس با نرخ پایین‌تری تازه می‌شود.
جدول و CSV تازه‌اند ولی نمودار ثابت است	حالت Pause یا مرور تاریخچه و Live Follow رابط را بررسی کنید؛ مسیر رسم را از دریافت داده جدا کنید.
وقفه‌های حدود یک ثانیه در ثبت	Timeout و زمان بازیابی را در لاگ ارتباط با زمان CSV تطبیق دهید؛ فاصله ثبت را با توقف واقعی موتور یکسان بگیرید.

آزمون تحویل یک مستر جدید

آدرس‌دهی را با STOP و یک رجیستر شناخته‌شده تأیید کنید؛ مرجع 50 Hz را بنویسید و مقدار 500 را بازخوانی کنید؛ خواندن مقدار 3 از رجیستر 2 را با معنای فرمان‌گیری از مدباس بررسی کنید؛ حالت RUN و STOP را از رجیستر 14 پایش کنید؛ نوشتن تنظیم PMSM هنگام RUN را ردشده ببینید؛ بازخوانی پس از ذخیره و ماندگاری مورد نیاز را آزمون کنید؛ قطع و وصل کابل را بدون RUN ناخواسته بررسی کنید. در پایان مدت اجرای پیوسته، شمارنده‌های خطای ارتباط و سن داده‌ها را کنترل کنید.

ضمیمه ساختار دقیق فریم‌ها

در این بخش ترتیب همه اجزا از چپ به راست، مطابق ترتیب ارسال روی سیم است. S یک بایت شناسه درایو، U16 یک عدد 16 بیتی با بایت پرارزش‌اول، N تعداد رجیستر، و CRC دو بایت کم‌ارزش‌اول است. تعداد رجیستر با تعداد بایت یکی نیست.

خواندن FC03 یا FC04

TX: S | FC | Start(U16) | N(U16) | CRC
RX: S | FC | ByteCount=2N | Values(U16 × N) | CRC

طول درخواست 8 بایت و طول پاسخ 5 + 2N است. Byte Count باید دقیقاً دو برابر تعداد درخواستی باشد؛ مقدار فرد یا طول ناسازگار را نپذیرید.

نوشتن یک رجیستر FC06

TX: S | 06 | Address(U16) | Value(U16) | CRC
RX: exact echo of TX

طول هر دو 8 بایت است. آدرس و مقدار Echo باید با همان درخواست یکسان باشند.

نوشتن چند رجیستر FC10

TX: S | 10 | Start(U16) | N(U16) | ByteCount=2N
| Values(U16 × N) | CRC
RX: S | 10 | Start(U16) | N(U16) | CRC

طول درخواست 9 + 2N و طول پاسخ 8 بایت است. پاسخ مقادیر نوشته‌شده را برنمی‌گرداند؛ برای آن‌ها Readback انجام دهید.

نوشتن سپس خواندن FC17

TX: S | 17 | ReadStart(U16) | ReadN(U16)
| WriteStart(U16) | WriteN(U16) | ByteCount=2WriteN
| WriteValues(U16 × WriteN) | CRC
RX: S | 17 | ByteCount=2ReadN | ReadValues | CRC

طول درخواست 13 + 2WriteN و طول پاسخ 5 + 2ReadN است. داده خواندن بعد از انجام نوشتن ساخته می‌شود؛ این موضوع تضمین نمونه‌برداری هم‌زمان همه کمیت‌های موتور نیست.

Exception و Diagnostics

TX08: S | 08 | Subfunction(U16) | 0000(U16) | CRC
RX08: S | 08 | Subfunction(U16) | Result(U16) | CRC
Error: S | (FC 0R 80) | ExceptionCode | CRC

فرم عادی Diagnostics هشت بایت و Exception پنج بایت است. Echo زیرتابع 0000 همان داده زوج ارسالی را بازمی‌گرداند و طول آن به تعداد داده ارسالی بستگی دارد. برای Device Identification ساختار اشیای طول‌دار را طبق بخش مربوط بخوانید.

ضمیمه محاسبه CRC

این تابع CRC16 Modbus را روی تمام بایت‌های پیام به جز دو بایت CRC محاسبه می‌کند. مقدار اولیه 0xFFFF و چندجمله‌ای بازتاب‌یافته 0xA001 است. دو بایت حاصل به ترتیب کم‌ارزش و پرارزش به پیام افزوده می‌شوند.

```
#include <stdint.h>
#include <stddef.h>

uint16_t modbus_crc16(const uint8_t *data, size_t len)
{
    uint16_t crc = 0xFFFFu;
    for (size_t i = 0; i < len; ++i) {
        crc ^= data[i];
        for (unsigned bit = 0; bit < 8; ++bit) {
            if (crc & 1u)
                crc = (uint16_t)((crc >> 1) ^ 0xA001u);
            else
                crc = (uint16_t)(crc >> 1);
        }
    }
    return crc;
}

/* Caller ensures room for payload_len + 2 bytes. */
void append_crc(uint8_t *tx, size_t payload_len)
{
    uint16_t crc = modbus_crc16(tx, payload_len);
    tx[payload_len] = (uint8_t)(crc & 0xFFu);
    tx[payload_len + 1] = (uint8_t)(crc >> 8);
}
```

دو آزمون برای پیاده‌سازی

بایت‌های افزوده‌شده	CRC عددی	ورودی بدون CRC
E5 C9	0xC9E5	01 03 00 0E 00 01
37 4B	0x4B37	متن ASCII برابر 123456789

آزمون دوم فقط بردار آزمون الگوریتم است و پیام معتبر فرمان درایو نیست. برای بررسی پاسخ، CRC محاسبه‌شده را با ترکیب بایت پایانی کم‌ارزش و پرارزش مقایسه کنید. اگر کتابخانه Modbus ساخت فریم را انجام می‌دهد، این CRC را دوباره به درخواست کتابخانه اضافه نکنید.

کد بالا فقط بخش CRC را نشان می‌دهد. ارسال UART، زمان‌بندی، کنترل جهت RS485، کنترل طول بافر و اعتبارسنجی پاسخ باید مطابق بخش میکروکنترلر پیاده شوند. کتابخانه مستر معتبر نیز قابل استفاده است، به شرط تطبیق همین نقشه و همین قالب سریال.

واژه‌نامه و مراجع

معنای کاربردی	اصطلاح
کنترل‌کننده‌ای که درخواست را آغاز می‌کند	Master
درایوی که به درخواست پاسخ می‌دهد	Slave
داده 16 بیتی با آدرس مشخص	Register
نام دو روش دسترسی به رجیسترها با FC03 و FC04	Holding / Input
تأیید پذیرش پیام؛ رسیدن موتور به هدف جداگانه بررسی می‌شود	ACK یا پاسخ موفق
پایان مهلت دریافت پاسخ معتبر	Timeout
پاسخ رد درخواست همراه کد علت	Exception
عدد کنترل خطای انتقال پیام	CRC
خواندن دوره‌ای داده توسط کنترل‌کننده	Polling
دوباره خواندن مقدار پس از نوشتن برای تأیید آن	Readback
ساخت ولتاژ خروجی با تنظیم پالس‌های کلیدهای قدرت	PWM
تعداد دور مکانیکی محور در یک دقیقه	RPM

دامنه نسخه حاضر

این ویرایش، راه‌اندازی یک درایو PM300 را با کنترل‌کننده مستقل پوشش می‌دهد. قالب تنظیم سوئیچینگ با کد 0x504D و نسخه 1 شناسایی می‌شود. آدرس‌ها و مثال‌ها برای همین قالب هستند. برای مشخصات توان، اتصال قدرت و شرایط موتور به راهنمای مدل درج‌شده روی محصول مراجعه کنید.

مرجع استاندارد ارتباط

برای تعریف عمومی فریم RTU، زمان‌بندی و CRC، مرجع Modbus Organization را ببینید. در استفاده از این محصول، نوع تابع، آدرس و مقدار را مطابق جدول‌های همین راهنما انتخاب کنید.

<https://www.modbus.org/modbus-specifications>
<https://www.modbus.org/file/secure/modbusoverserial.pdf>

شرکت ویرا صنعت الکترونیک سورین | HiPowerVSES

<https://hipowervses.ir>