

Modbus Register Definitions

EM-346A • EM-347B • EM-379

Release revision 2.0 | General register map and integration guide | Released 1 September 2026

This document describes the common Modbus register structure for EM-346A, EM-347B and EM-379 motor drivers. It covers control registers, device identification, user memory, parameter transfer and monitoring data. Parameter meanings, limits and available functions remain product- and firmware-dependent.

This register map was originally based on “EM-347B Modbus Data Address 1v0”, revision 1, dated 23 August 2021. Revision 2.0 extends the coverage to EM-346A, EM-347B and EM-379 and incorporates register definitions verified against the EM-346A_347B_379 firmware source code.

Communication settings

Setting	Supported values	Source default
Protocol	Modbus RTU	Modbus RTU
Baud rate	9600 / 19200 bit/s	19200 bit/s
Data bits	8 (RTU framing)	8
Parity	None / odd / even	Even
Stop bits	1 / 2	1
Device address	1–247	1

The source-code defaults are 19200 bit/s, even parity, one stop bit and device address 1. Communication settings can be changed through the applicable device parameters.

Protocol and electrical interface

Modbus RTU is the protocol, not the electrical interface. EM-346A provides 5 V TTL serial communication; use the companion TTL-level Modbus guide for its pinout and wiring. RS-485 operation uses a suitable interface, such as EM-P189 where applicable. The source describes RS-485 as two-wire, half-duplex. Do not infer that all three models have an onboard RS-485 port.

Supported functions

Supported Modbus functions are 0x01 Read Coils, 0x02 Read Discrete Inputs, 0x03 Read Holding Registers, 0x04 Read Input Registers, 0x05 Write Single Coil, 0x06 Write Single Register, 0x0F Write Multiple Coils and 0x10 Write Multiple Registers. The firmware accepts holding-register writes with functions 0x06 and 0x10 across the complete holding-register area, including MB_VERSION and the Device ID registers.

Addressing and control registers

All data addresses in this guide are decimal, zero-based offsets used in Modbus requests. The 4xxxx and 3xxxx numbers are reference labels; they are not transmitted as the raw address. Holding and input registers are separate address spaces, selected by the function code.

Register area	Reference number	Data address
Holding register	40001 + address	Reference – 40001
Input register	30001 + address	Reference – 30001

Example: holding register 40002 is data address 1. Input register 30002 is also address 1, but is read using function 0x04 rather than 0x03. Check whether the host software expects a full reference, a one-based index or a zero-based address.

Motor control and memory command

Reference	Addr.	Name	Range / definition
40001	0	MB_LOCALREMOTE	0 = local control active; 1 = Modbus/remote control active. Status register maintained by the firmware; writing it does not select the control source.
40002	1	MB_SPEED	0-1023; speed command used when MB_LOCALREMOTE = 1. Values above 1023 are limited to 1023 by the application.
40003	2	MB_DISABLE	0 = no Modbus disable request; 1 = disable. The command affects operation only while remote control is active.
40004	3	MB_DIR	0-1; direction command used while remote control is active. The physical direction depends on wiring and device configuration.
40005–40006	4–5	N/A	Not assigned; do not write.
40007	6	MB_userMemControl	0 = no action; 1 = EEPROM → user memory; 2 = user memory → EEPROM.

Read holding registers with function 0x03. Functions 0x06 and 0x10 can write the complete holding-register area, including MB_VERSION and Device ID. Avoid write blocks that unintentionally include reserved addresses or command registers.

Control semantics

MB_LOCALREMOTE reports the active control source: 0 = local control and 1 = Modbus/remote control. It is maintained by the application and is not a command for selecting remote operation. Remote control is selected through the configured auxiliary function and the applicable physical selection input.

MB_SPEED is a raw 0-1023 speed command used only while MB_LOCALREMOTE = 1. MB_DIR is likewise used as the direction command only in remote operation. MB_DISABLE = 1 requests disable while remote operation is active.

Identification and user memory

MB_VERSION and the Device ID occupy writable holding registers. MB_VERSION is a numerical value: decimal 12 means v1.2 and decimal 29 means v2.9. The six Device ID registers are initialized at start-up with one character value per register, but the user may overwrite them during operation.

Reference	Address	Name	Definition / use
40008	7	MB_VERSION	Writable 16-bit numerical firmware version. Decimal 12 means firmware v1.2; decimal 29 means v2.9. The value is initialized by the firmware at start-up.
40009-40014	8-13	MB_Device ID	Six writable registers initialized with the device name at start-up, one character value in each register. The user may replace the name while the device is operating.
40010-40016	9-15	MB_userMem_1-7	Seven writable user-memory registers. Addresses 9-13 intentionally overlap the Device ID field, allowing the same area to be used as additional memory or to replace the displayed device name. EEPROM transfer uses all seven registers.

16-bit data and ASCII representation

A Modbus register contains 16 bits and is transmitted most-significant byte first. Device ID addresses 9-13 overlap user-memory registers 1-5 by design. This provides additional general-purpose memory and allows the user to replace the device name. A reset initializes the Device ID again; reload user memory from EEPROM after start-up if stored replacement values are required.

The current firmware stores one Device ID character value in the low byte of each 16-bit register. The high byte is zero when initialized by the firmware. Modbus still transmits every register most-significant byte first.

User memory transfer

MB_userMemControl at holding reference 40007 / data address 6 transfers seven user-memory words at addresses 9-15 between the working register area and EEPROM. Addresses 9-13 overlap the writable Device ID field; loading user memory can therefore replace five characters of the current Device ID.

Value	Action	Effect
0	No action	No memory transfer requested.
1	Read from EEPROM	Load stored user memory into the working user memory registers.
2	Write to EEPROM	Store the working user memory values in non-volatile memory.

To inspect stored user memory, request value 1 and then read addresses 9-15 after the command returns to zero. To save changed user memory, write addresses 9-15 first and then request value 2. Verify the data by reloading and reading it back.

MB_userMemControl returns to zero after the EEPROM transfer has been processed by the application. Avoid repetitive EEPROM writes; use command 2 only when retained data must change.

Device parameter register map

Parameter registers 1-21 form a working copy of device settings. Parameter 22 / address 42 is N/A in the current application. Parameter Update at 40021 / data address 20 controls transfer between the supported parameter block and the active parameters.

Parameter	Holding reference	Data address
Parameter_1	40022	21
Parameter_2	40023	22
Parameter_3	40024	23
Parameter_4	40025	24
Parameter_5	40026	25
Parameter_6	40027	26
Parameter_7	40028	27
Parameter_8	40029	28
Parameter_9	40030	29
Parameter_10	40031	30
Parameter_11	40032	31
Parameter_12	40033	32
Parameter_13	40034	33
Parameter_14	40035	34
Parameter_15	40036	35
Parameter_16	40037	36
Parameter_17	40038	37
Parameter_18	40039	38
Parameter_19	40040	39
Parameter_20	40041	40
Parameter_21	40042	41
N/A	40043	42

Mapping rule: Parameter_n -> data address 20 + n -> holding reference 40021 + n, for n = 1-21. Data address 42 / holding reference 40043 is N/A.

Model-dependent definitions

For each parameter, obtain its name, allowed values, units, scaling, default and operating restrictions from the manual for EM-346A, EM-347B or EM-379 and the actual firmware version. The same parameter number need not represent the same function in different models. Do not copy a complete parameter block between models without checking every entry.

Numbering correction: the original source lists inconsistent holding reference numbers for Parameter_9–Parameter_22. This edition corrects them to 40030–40043 using the unchanged data addresses 29–42. No data addresses have been shifted.

Input registers: monitoring and status

Input-register meanings are listed in the table above. Boolean status values use 0 = inactive and 1 = active. Analogue current and ILIM/AUX scaling remains model-dependent; voltage is reported in 0.1 V units.

Reference	Addr.	Source name	Meaning
30001	0	MB_STATUS	N/A; reserved in the current application and not updated by the firmware.
30002	1	MB_Current	Motor-current monitor value, raw range 0-250. Physical-unit scaling is model-dependent.
30003	2	MB_BrakeCurrent	Braking-current monitor value, raw range 0-250. Physical-unit scaling is model-dependent.
30004	3	MB_Voltage	Supply-voltage monitor in 0.1 V units; for example, value 550 represents approximately 55.0 V.
30005	4	MB_Freq	Calculated motor feedback frequency/speed value. Conversion to mechanical speed depends on the motor pulse basis and installation.
30006	5	MB_IO_SPEED2	Physical SPEED2 selection input: 0 = inactive, 1 = active.
30007	6	MB_IO_DIR	Physical direction input: 0 = inactive, 1 = active.
30008	7	MB_IO_SPEED	Filtered analogue speed-input value, normally 0-1023.
30009	8	MB_IO_ILIM	Scaled analogue ILIM/AUX input value. Scaling is model-dependent.
30010	9	MB_IO_DISABLE	Physical disable input: 0 = inactive, 1 = active.
30011	10	MB_PWM	Motor-driving PWM command/monitor value used internally by the firmware.
30012	11	MB_Speed2Enable	Filtered second-speed selection status: 0 = inactive, 1 = active.
30013	12	MB_Fault	General fault output state: 0 = inactive, 1 = active.
30014	13	MB_Fail_I	Current-related fault state: 0 = inactive, 1 = active.
30015	14	MB_FAIL_LAMPO	N/A; retained in the register map but set to 0 by the current firmware.
30016	15	MB_FAIL_VS	Supply-voltage fault state: 0 = inactive, 1 = active.
30017	16	MB_Overvoltage	Overvoltage state: 0 = inactive, 1 = active.

Interpreting monitoring values

Values without a specified physical conversion shall be handled as raw register values. SPEED and ILIM are analogue values and shall not be interpreted as Boolean status signals.

Register 30006 reports the physical SPEED2 selection input as MB_IO_SPEED2. Fault registers 30013, 30014, 30016 and 30017 report Boolean states, while MB_FAIL_LAMPO at 30015 is reserved and remains zero in the current firmware.

Parameter update and commissioning

Value	Parameter Update action (40021 / address 20)
0	No change to parameters.
1	Copy Modbus parameter registers to active device parameters.
2	Copy Modbus parameter registers to active parameters and store them in EEPROM.
3	Copy device parameters to the Modbus parameter registers.

The device resets Parameter Update to 0 after the read/write operation. No maximum processing time is defined by the firmware interface; the host shall use a bounded timeout. A return to 0 shall be followed by value verification.

Recommended parameter-change sequence

Read the current settings. With movement prevented, write 3 to address 20. Wait for the command to return to 0, then read addresses 21–42, limited to parameters supported by the device.

Modify the working copy. Change only the intended parameter registers. Keep valid current values in all other supported entries, because the update command transfers the parameter block.

Apply and verify. Write 1 to address 20 for an update without an EEPROM storage request, or 2 to apply and store. Wait for 0, reload with command 3 and compare the relevant values.

Check operation safely. Confirm operating limits and local/bus behaviour before allowing motion. If persistent settings were requested, verify retention after a controlled restart when safe.

Address examples for host software

Read motor current: function 0x04, start address 1, quantity 1. The response is a raw value with model-dependent physical scaling. Read Device ID: function 0x03, start address 8, quantity 6. Reload parameter registers: function 0x06, address 20, value 3. These are request fields, not complete RTU frames; the host stack adds device address, framing and CRC.

Application-specific commissioning checks

Before commissioning, verify model-specific current and ILIM/AUX scaling, frequency-to-mechanical-speed conversion, communication-loss behavior and permitted parameter values. MB_VERSION, Device ID and the complete holding-register area are writable in the current firmware.

Source and revision notes

Register source: EM-347B Modbus Data Address 1v0.pdf, revision 1, 23 August 2021/Jam. Revision 2.0 extends the model coverage, adds integration guidance and corrects the register definitions against the reviewed EM-346A_347B_379 firmware. This document defines the Modbus interface for the firmware versions identified in its scope.

Protocol reference: Modbus Application Protocol Specification V1.1b3, Modbus.org:
<https://www.modbus.org/file/secure/modbusprotocolspecification.pdf>. See also the companion EM-346A TTL-level Modbus Customer Guide for electrical connection details.

Revision History

Document release information and changes from the preceding document.

Revision	Date	Status	Description
2.0	1 September 2026	Released	Register map verified and corrected against firmware EM-346A_347B_379.c.

Changes from the previous document

- Register 40001 renamed to MB_LOCALREMOTE and documented as firmware-maintained control-source status.
- MB_SPEED command range corrected to 0-1023.
- Firmware version documented as a writable decimal value; for example, 12 means v1.2 and 29 means v2.9.
- Device ID and user-memory overlap documented, including use of the area as additional memory or a user-defined device name.
- User-memory area corrected to seven registers at addresses 9-15, including EEPROM transfer behavior.
- Parameter 22 marked N/A because it is not used by the current application.
- Input-register names, states, value ranges and descriptions corrected to match the firmware.
- Write access to MB_VERSION, Device ID and the complete holding-register area documented.
- Communication, parameter-update and commissioning descriptions aligned with the reviewed implementation.