

# **DAM-E3062**

## **User's Manual**



**Beijing ART Technology Development Co., Ltd.**

## DAM-E3062 Module

### Introduction

#### Features

##### Two Channels Analog Output

- Output Type: V/mA
- Output Range: 0~20mA, 4~20mA or 0~5V, 0~10V
- 12-bit resolution
- Precision:  $\pm 0.1\%$  (full-scale)
- Read-back Precision:  $\pm 1\%$  (full-scale)
- Zero Drift:
  - Voltage Output:  $\pm 30\mu\text{V}/^\circ\text{C}$
  - Current Output:  $\pm 0.2\mu\text{A}/^\circ\text{C}$
- Full-scale Drift:  $\pm 25\text{ppm}/^\circ\text{C}$
- Programmable output conversion slope
- Isolated Voltage: 2000V
- Current Load Resistance
  - Internal Power:  $500\Omega$
  - External Power:  $1050\Omega$
- Support 10/100Mbps Ethernet
- Provide default web page or custom-built web page
- Support double watchdogs
- Power Supply: unregulated  $10\sim 30\text{V}_{\text{DC}}$
- Power Consumption:  $1.4\text{W}@24\text{V}_{\text{DC}}$

### Easy to operate

DAM-E3062 utility software can help you to select configuration, set the operating parameter for your process control needed.

### Industrial Design

DAM-E3062 was designed to use in industrial environment. It can be installed in standard DIN rail inside the cabinet. And it can be powered by unregulated  $10\sim 30\text{V}_{\text{DC}}$  to meet the various power supplied source in field. It also withstands ambient temperature up to  $60^\circ\text{C}$  and resists the effects of vibration and mechanical shock.

### Wiring & Installation

Power supply requirements: unregulated  $+10\text{V}_{\text{DC}} \sim +30\text{V}_{\text{DC}}$ . "+Vs" is a positive, and "GND" is ground. The Ethernet interface is connected to a computer switch directly via network cable. If the wiring and power is ok, then the power indicator should be stop flash and keep on. The module initialization is completed. The indicator flashes according to the transmitted data when the module is working.

## DAM-E3062

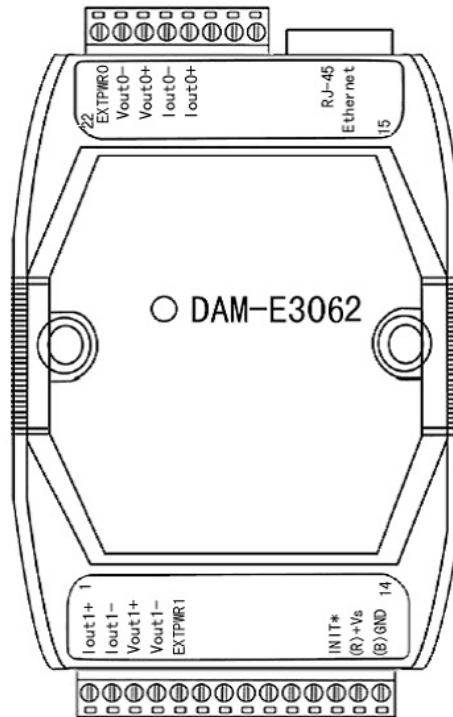


Fig.1 DAM-E3062 Drawing

DAM-E3062 can be installed in standard DIN rail inside the cabinet; it also can be installed by stacking mode.

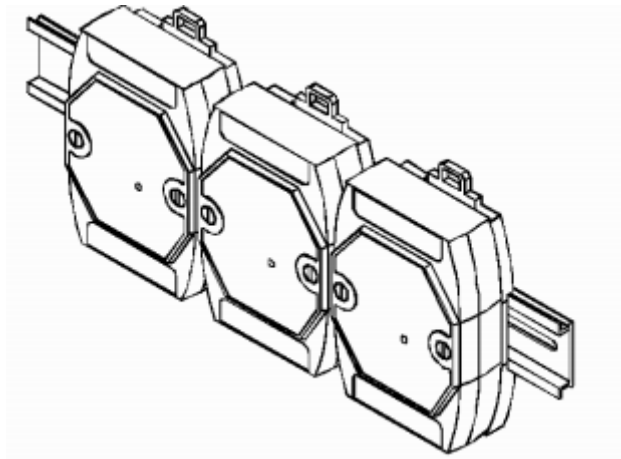


Fig.2 DAM-E3062 standard DIN installation

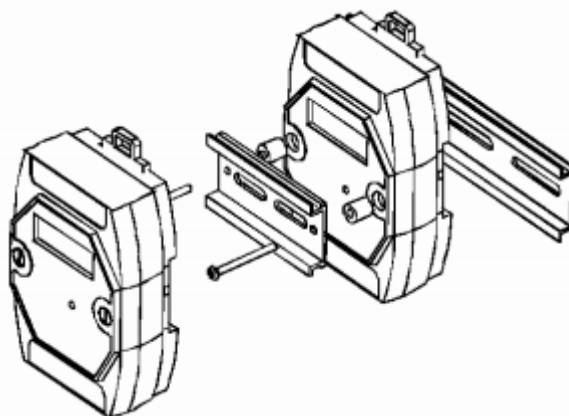


Fig.3 DAM-E3062 stack installation

## Code Configuration

### Analog Output Range Configuration Code Table

| Signal Type | Range  | Code |
|-------------|--------|------|
| mA          | 0~20mA | 0B   |
|             | 4~20mA | 0C   |
| V           | 0~5V   | 0D   |
|             | 0~10V  | 0E   |

### Analog Output Rate Configuration Code Table

| Output rate (20mA/S) | Output rate (10V/S) | Code (16Hex) |
|----------------------|---------------------|--------------|
| Immediate            | Immediate           | 00           |
| 0.125                | 0.0625              | 01           |
| 0.25                 | 0.125               | 02           |
| 0.5                  | 0.25                | 03           |
| 1                    | 0.5                 | 04           |
| 2                    | 1                   | 05           |
| 4                    | 2                   | 06           |
| 8                    | 4                   | 07           |
| 16                   | 8                   | 08           |
| 32                   | 16                  | 09           |
| 64                   | 32                  | 0A           |
| 128                  | 64                  | 0B           |
| 256                  | 128                 | 0C           |
| 512                  | 256                 | 0D           |
| 1024                 | 512                 | 0E           |
| 2048                 | 1024                | 0F           |

## Default Setting

If the module’s network configuration is wrong, or forget the last modified value, the module can be reverted to default settings. Steps: short-circuit the “INIT\*” and “GND” when there is no power; power-on for 3 seconds, power off, disconnect “INIT\*” and “GND”. The module is reverted to the default settings.

1. IP Address: 192.168.2.80
2. Subnet Mask: 255.255.255.0
3. Default Gateway: 192.168.2.1
4. TCP Port: 502
5. HTTP Port: 80

## Communication Protocol

At present, Ethernet Distributed Acquisition Modules of our company use MODBUS TCP mode. Supported function code includes the following categories:

- 01 READ COIL STATUS
- 02 READ INPUT STATUS
- 03 READ HOLDING REGISTERS
- 04 READ INPUT REGISTERS
- 05 FORCE SINGLE COIL
- 06 PRESET SINGLE REGISTER
- 15 FORCE MULTIPLE COILS
- 16 FORCE MULTIPLE REGISTERS
- 20 READ FILE RECORD
- 21 WRITE FILE RECORD

## Address Mapping Table

### Read Holding Register

Function Code: 03

Note: Read the holding register value

#### Data Description:

|             |                                |   |
|-------------|--------------------------------|---|
| 40513       | For Watchdog Control Register  | bit0: 0 disabled, 1 enabled<br>bit1: 0 normal, 1 overflow<br>bit2: 0 normal, 1reset |
| 40514       | For Watchdog Overflow Register | Watchdog Overtime Constant  |
| 40515       | For Watchdog Reset Register    | 0x55AA  |
| Reservation |                                |   |
| 401025      | Analog Output Range            | Corresponding current channel 0   |
| 401026      | Analog Output Range            | Corresponding voltage channel 0   |
| 401027      | Analog Output Range            | Corresponding current channel 1   |
| 401028      | Analog Output Range            | Corresponding voltage channel 1   |
| Reservation |                                |   |
| 401041      | Analog Output Slope            | Corresponding current channel 0   |
| 401042      | Analog Output Slope            | Corresponding voltage channel 0   |

|             |                              |                                 |
|-------------|------------------------------|---------------------------------|
| 401043      | Analog Output Slope          | Corresponding current channel 1 |
| 401044      | Analog Output Slope          | Corresponding voltage channel 1 |
| Reservation |                              |                                 |
| 401057      | Analog Output Power-on Value | Corresponding current channel 0 |
| 401058      | Analog Output Power-on Value | Corresponding voltage channel 0 |
| 401059      | Analog Output Power-on Value | Corresponding current channel 1 |
| 401060      | Analog Output Power-on Value | Corresponding voltage channel 1 |
| Reservation |                              |                                 |
| 401073      | Analog Output Current Value  | Corresponding current channel 0 |
| 401074      | Analog Output Current Value  | Corresponding voltage channel 0 |
| 401075      | Analog Output Current Value  | Corresponding current channel 1 |
| 401076      | Analog Output Current Value  | Corresponding voltage channel 1 |
| Reservation |                              |                                 |
| 401089      | Analog Output Security Level | Corresponding current channel 0 |
| 401090      | Analog Output Security Level | Corresponding voltage channel 0 |
| 401091      | Analog Output Security Level | Corresponding current channel 1 |
| 401092      | Analog Output Security Level | Corresponding voltage channel 1 |
| Reservation |                              |                                 |

**Request**

| Domain Name   | Byte    | Value            |
|---------------|---------|------------------|
| Function Code | 1 byte  | 0×03             |
| Start Address | 2 bytes | 0×0000 to 0×FFFF |
| Read Amount   | 2 bytes | 1 to 125 (0×7D)  |

**Response**

| Domain Name   | Byte     | Value |
|---------------|----------|-------|
| Function Code | 1 byte   | 0×03  |
| Byte Count    | 1 byte   | 2n    |
| Input Status  | 2n bytes |       |

**Exception**

| Domain Name   | Byte   | Value      |
|---------------|--------|------------|
| Function Code | 1 byte | 0×03+0×80  |
| Error Code    | 1 byte | 0×1 or 0×2 |

**Example**

| Request                |            | Response            |            |
|------------------------|------------|---------------------|------------|
| Domain Name            | Data (hex) | Domain Name         | Data (hex) |
| Function Code          | 03         | Function Code       | 03         |
| Start Address H (byte) | 00         | Byte Count          | 02         |
| Start Address L (byte) | 08         | Input Register High | 00         |
| Read Amount H (byte)   | 00         | Input Register Low  | 0A         |
| Read Amount L (byte)   | 01         |                     |            |

Note 1: The unit of the pulse output level width: millisecond

The unit of the watchdog timer length: millisecond

Note 2: When power on, the highest bit of the watchdog control register is 1, can be done to do determine whether the module is rested or not.

## Read Input Register

Function Code: 04

Description : Read input data

### Data Description:

| Address     | Description                   | Note                            |
|-------------|-------------------------------|---------------------------------|
| 401073      | Analog Output Read-back Value | Corresponding current channel 0 |
| 401074      | Analog Output Read-back Value | Corresponding voltage channel 0 |
| 401075      | Analog Output Read-back Value | Corresponding current channel 1 |
| 401076      | Analog Output Read-back Value | Corresponding voltage channel 1 |
| Reservation |                               |                                 |

### Request

| Domain Name   | Byte    | Value            |
|---------------|---------|------------------|
| Function Code | 1 byte  | 0×04             |
| Start Address | 2 bytes | 0×0000 to 0×FFFF |
| Read Amount   | 2 bytes | 1 to 125 (0×7D)  |

### Response

| Domain Name   | Byte     | Value |
|---------------|----------|-------|
| Function Code | 1 byte   | 0×04  |
| Byte Count    | 1 byte   | 2n    |
| Input State   | 2n bytes |       |

### Exception

| Domain Name   | Byte   | Value      |
|---------------|--------|------------|
| Function Code | 1 byte | 0×04+0×80  |
| Error Code    | 1 byte | 0×1 or 0×2 |

### Example

| Request                |            | Response             |            |
|------------------------|------------|----------------------|------------|
| Domain Name            | Data (hex) | Domain Name          | Data (hex) |
| Function Code          | 04         | Function Code        | 04         |
| Start Address H (byte) | 00         | Byte Count           | 02         |
| Start Address L (byte) | 08         | Input Register H (9) | 00         |
| Read Amount H (byte)   | 00         | Input Register L (9) | 0A         |
| Read Amount L (byte)   | 01         |                      |            |

## Single Relay Setting

Function Code: 05

### Request

| Domain Name   | Byte   | Value |
|---------------|--------|-------|
| Function Code | 1 byte | 0×05  |

|                 |         |   |
|-----------------|---------|---|
| Address Setting | 2 bytes | 0×0000 to 0×FFFF  |
| Content Setting | 2 bytes | 0x0000 or 0xFF00<br>0x0000 release relay<br>0xFF00 engage relay |

**Response**

|                 |         |                  |
|-----------------|---------|------------------|
| Domain Name     | Byte    | Value            |
| Function Code   | 1 byte  | 0x05             |
| Address Setting | 2 bytes | 0x0000 to 0xFFFF |
| Content Setting | 2 bytes | 0x0000 or 0xFF00 |

**Exception**

|               |        |            |
|---------------|--------|------------|
| Domain Name   | Byte   | Value      |
| Function Code | 1 byte | 0x05+0x80  |
| Error Code    | 1 byte | 0x1 or 0x2 |

**Example**

| Request              |            | Response             |            |
|----------------------|------------|----------------------|------------|
| Domain Name          | Data (hex) | Domain Name          | Data (hex) |
| Function Code        | 05         | Function Code        | 05         |
| Set Address H (byte) | 00         | Set Address H (byte) | 00         |
| Set Address L (byte) | 05         | Set Address L (byte) | 05         |
| Set Content H (byte) | FF         | Set Content H (byte) | FF         |
| Set Content L (byte) | 00         | Set Content L (byte) | 00         |

**Single Holding Register Setting**

Function Code: 06

**MODBUS Request**

|                 |         |                  |
|-----------------|---------|------------------|
| Domain Name     | Byte    | Value            |
| Function Code   | 1 byte  | 0×06             |
| Address Setting | 2 bytes | 0×0000 to 0×FFFF |
| Content Setting | 2 bytes | 0×0000 to 0×FFFF |

**MODBUS Response**

|                 |         |                  |
|-----------------|---------|------------------|
| Domain Name     | Byte    | Value            |
| Function Code   | 1 byte  | 0×06             |
| Address Setting | 2 bytes | 0×0000 to 0×FFFF |
| Content Setting | 2 bytes | 0×0000 to 0×FFFF |

**Exception**

|               |        |            |
|---------------|--------|------------|
| Domain Name   | Byte   | Value      |
| Function Code | 1 byte | 0×06+0×80  |
| Error Code    | 1 byte | 0×1 or 0×2 |

**Example**

| Request       |           | Response      |           |
|---------------|-----------|---------------|-----------|
| Domain Name   | Data(hex) | Domain Name   | Data(hex) |
| Function Code | 06        | Function Code | 06        |



|                         |    |                         |    |
|-------------------------|----|-------------------------|----|
| Set Address High (byte) | 00 | Set Address High (byte) | 00 |
| Set Address Low (byte)  | 08 | Set Address Low (byte)  | 08 |
| Set Content High (byte) | 00 | Set Content High (byte) | 00 |
| Set Content Low (byte)  | 19 | Set Content Low (byte)  | 19 |

## Set Multiple Relay

Function Code: 0F

### Request

| Domain Name           | Byte    | Value            |
|-----------------------|---------|------------------|
| Function Code         | 1 byte  | 0×0F             |
| Start Address Setting | 2 bytes | 0×0000 to 0×FFFF |
| Length Setting        | 2 bytes | 0×0000 to 0×07B0 |
| Byte count            | 1 byte  | n                |
| Content Setting       | n bytes |                  |

### Response

| Domain Name           | Byte    | Value            |
|-----------------------|---------|------------------|
| Function Code         | 1 byte  | 0×0F             |
| Start Address Setting | 2 bytes | 0×0000 to 0×FFFF |
| Length Setting        | 2 bytes | 0×0000 to 0×07B0 |

### Exception

| Domain Name   | Byte   | Value      |
|---------------|--------|------------|
| Function Code | 1 byte | 0×0F+0×80  |
| Error Code    | 1 byte | 0×1 or 0×2 |

### Example

| Request                 |           | Response                |           |
|-------------------------|-----------|-------------------------|-----------|
| Domain Name             | Data(hex) | Domain Name             | Data(hex) |
| Function Code           | 0F        | Function Code           | 0F        |
| Set Address High (byte) | 00        | Set Address High (byte) | 00        |
| Set Address Low (byte)  | 13        | Set Address Low (byte)  | 13        |
| Set Amount High (byte)  | 00        | Set Amount High (byte)  | 00        |
| Set Amount Low (byte)   | 0A        | Set Amount Low (byte)   | 0A        |
| Byte count              | 02        |                         |           |
| Set Content High (byte) | CD        |                         |           |
| Set Content Low (byte)  | 01        |                         |           |

## Set Multiple Holding Registers

Function Code: 10

### Request

| Domain Name           | Byte     | Value            |
|-----------------------|----------|------------------|
| Function Code         | 1 byte   | 0×10             |
| Start Address Setting | 2 bytes  | 0×0000 to 0×FFFF |
| Length Setting        | 2 bytes  | 0×0000 to 0×07B0 |
| Byte count            | 1 byte   | 2n               |
| Content Setting       | 2n bytes |                  |

**Response**

| Domain Name           | Byte    | Value            |
|-----------------------|---------|------------------|
| Function Code         | 1 byte  | 0×10             |
| Start Address Setting | 2 bytes | 0×0000 to 0×FFFF |
| Length Setting        | 2 bytes | 0×0000 to 0×07B0 |

**Exception**

| Domain Name   | Byte   | Value      |
|---------------|--------|------------|
| Function Code | 1 byte | 0×10+ 0×80 |
| Error Code    | 1 byte | 0×1 or 0×2 |

**Example**

| Request                 |           | Response                |           |
|-------------------------|-----------|-------------------------|-----------|
| Domain Name             | Data(hex) | Domain Name             | Data(hex) |
| Function Code           | 10        | Function Code           | 10        |
| Set Address High (byte) | 00        | Set Address High (byte) | 00        |
| Set Address Low (byte)  | 01        | Set Address Low (byte)  | 01        |
| Set Amount High(byte)   | 00        | Set Amount High(byte)   | 00        |
| Set Amount Low (byte)   | 02        | Set Amount Low (byte)   | 02        |
| Byte count              | 04        |                         |           |
| Set Content High (byte) | 00        |                         |           |
| Set Content Low (byte)  | 0A        |                         |           |
| Set Content High (byte) | 01        |                         |           |
| Set Content Low (byte)  | 02        |                         |           |

**Read File Record**

Function Code: 14/06

Read file record: in MODBUS, file is considered to be 16-bit array, addressing according to address. Read file: set the start address and read length and change the Start Address and Read Length can traverse the entire file. The file does not have a name, it has number only. You can only read and write one file at one time.

**Request**

| Domain Name       | Byte    | Value            |
|-------------------|---------|------------------|
| Function Code     | 1 byte  | 0×14             |
| Byte Count        | 1 byte  | 0×07 to 0×F5     |
| Sub Function Code | 1 byte  | 0×06             |
| File Number       | 2 byte  | 0x0000 to 0xFFFF |
| Record Number     | 2 bytes | 0x0000 to 0x270F |
| Read Length       | 2 bytes | n                |
| Sub Function Code | 1 byte  | 0x06             |
| .....             | .....   |                  |

**Response**

| Domain Name   | Byte   | Value |
|---------------|--------|-------|
| Function Code | 1 byte | 0×14  |

|                         |          |              |
|-------------------------|----------|--------------|
| Byte Count              | 1 byte   | 0x07 to 0xF5 |
| Sub Function Byte Count | 1 byte   | 0x07 to 0xF5 |
| Sub Function Code       | 1 byte   | 0x06         |
| Data                    | 2n bytes |              |

**Exception**

|               |        |            |
|---------------|--------|------------|
| Domain Name   | Byte   | Value      |
| Function Code | 1 byte | 0×14+ 0×80 |
| Error Code    | 1 byte | 0×1 or 0×2 |

**Example**

| Request                  |           | Response               |           |
|--------------------------|-----------|------------------------|-----------|
| Domain Name              | Data(hex) | Domain Name            | Data(hex) |
| Function Code            | 14        | Function Code          | 14        |
| Byte Count               | 07        | Byte Count             | 06        |
| Sub Function Code        | 06        | Respond Count          | 05        |
| File Number High (byte)  | 00        | Sub Function Code      | 06        |
| File Number Low(byte)    | 04        | Record Data High(byte) | 0D        |
| Record Number High(byte) | 00        | Record Date Low(byte)  | FE        |
| Record Number Low(byte)  | 01        | Record Data High(byte) | 00        |
| Read Length High(byte)   | 00        | Record Date Low(byte)  | 20        |
| Read Length Low(byte)    | 02        |                        |           |

**Write File Record**

Function Code: 15/06

**Request**

|                   |          |                  |
|-------------------|----------|------------------|
| Domain Name       | Byte     | Value            |
| Function Code     | 1 byte   | 0×15             |
| Byte Count        | 1 byte   | 0×07 to 0×F5     |
| Sub Function Code | 1 byte   | 0×06             |
| File Number       | 2 bytes  | 0x0000 to 0xFFFF |
| Record Number     | 2 bytes  | 0x0000 to 0x270F |
| Read Length       | 2 byte s | n                |
| Data              | 2n bytes |                  |
| .....             | .....    |                  |

**Response**

|                   |          |                  |
|-------------------|----------|------------------|
| Domain Name       | Byte     | Value            |
| Function Code     | 1 byte   | 0×15             |
| Byte Count        | 1 byte   | 0x07 to 0xF5     |
| Sub Function Code | 1 byte   | 0x06             |
| File Number       | 2 bytes  | 0x0000 to 0xFFFF |
| Record Number     | 2 bytes  | 0x0000 to 0x270F |
| Write Length      | 2 bytes  | n                |
| Data              | 2n bytes |                  |

**Exception**

| Domain Name   | Byte   | Value                        |
|---------------|--------|------------------------------|
| Function Code | 1 byte | $0 \times 15 + 0 \times 80$  |
| Error Code    | 1 byte | $0 \times 1$ or $0 \times 2$ |

**Example**

| Request                  |           | Response                 |           |
|--------------------------|-----------|--------------------------|-----------|
| Domain Name              | Data(hex) | Domain Name              | Data(hex) |
| Function Code            | 15        | Function Code            | 15        |
| Byte Count               | 0B        | Byte Count               | 0B        |
| Sub Function Code        | 06        | Sub Function Code        | 06        |
| File Number High (byte)  | 00        | File Number High (byte)  | 00        |
| File Number Low(byte)    | 04        | File Number Low (byte)   | 04        |
| Record Number High(byte) | 00        | Record Number High(byte) | 00        |
| Record Number Low(byte)  | 01        | Record Number Low(byte)  | 01        |
| Read Length High(byte)   | 00        | Write Length High(byte)  | 00        |
| Read Length Low(byte)    | 02        | Write Length Low(byte)   | 02        |
| Write Data               | 4 bytes   | Write Data               | 4 bytes   |

**EEPROM assignment**

```
//file 0
#define MODULE_NET_ADDR          0x0000
#define MODULE_VER_ADDR         0x0020
//file 1
#define NET_CONFIG_ADDR         0x0000
//file 2 to file 7
.....
```

**Note:**

EEPROM is 8kbyte capacity, divide to 8 file blocks, every block 1kbyte.

## 1. MODULE\_NET\_ADDR

Restore to factory defaults network configuration parameter.

| Byte    | 4          | 4               | 4           | 6           |
|---------|------------|-----------------|-------------|-------------|
| Content | IP Address | Default Gateway | Subnet Mask | MAC Address |

## 2. MODULE\_VER\_ADDR

Module Version Information:

| Byte    | 42byte                                   |
|---------|--|
| Content | DAM-E3012 V6.20 2006.09.01 ID:DAME123456 |

## 3. NET\_CONFIG\_ADDR

Network configuration parameter, call table 1 for structures.