

ADAM-4530

Addressable Radio/Leased Modem Interface Module

Introduction

ADAM-4530 is an extremely cost-effective RS-232/485 converter specifically designed to interface any bidirectional RS-485 device with a radio/leased modem. Radio/leased modems are used in data acquisition applications where the sensor data is inaccessible to the host computer due to large distances. The ADAM-4530 provides a low-cost intelligent interface between the radio modem and RS-485 data acquisition devices.

Built-in Intelligence

The ADAM-4530, equipped with an on-board microprocessor, can be used to interface a radio/leased modem to an RS-485 network with many devices. Most RS-485 devices have a quick response turn around time that is much faster than the transmitter turn-on delays required by many radio/leased modems. In addition, radio/leased modems also require an RS-232 signal to activate the transmitter. The ADAM-4530 provides user-programmable delay times between the modem's transmitter key signal (RTS) and the transmitted data. This cost-effectively replaces the need for a local computer at the remote site.

RS-485 Network with Automatic Data Flow Control

The RS-485 standard supports half-duplex communication, meaning a single pair of wires is used to both transmit and receive data. Handshaking signals such as RTS (Request To Send) are normally used to control the direction of the data flow. But with ADAM-4530, handshaking signals are not necessary because it contains a special I/O circuit that automatically senses the direction of the data flow and switches the transmission direction.

Features

- Built-in microprocessor
- Transmission speeds of up to 115.2 kbps
- Surge protection on RS-485 line
- RS-232 and RS-485 can be set to different baud rates
- Automatic RS-485 data flow control
- Watchdog timer function included
- Power and data flow indicator for troubleshooting
- Reserved space for termination resistor
- Software configurable to either addressable or non-addressable mode
- Easily mounted on DIN rail, panel or piggyback
- All communication setups stored in EEPROM

Specifications

- **Transmission speed (bps):** 00, 600, 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K, 115.2K (software configurable)
- **Power requirement:** Unregulated +10 to +30 V_{DC} with protection from power reversals
- **Case:** ABS with captive mounting hardware
- **Accessories (included):** Nylon DIN-rail mounting adapter, SECC panel mounting bracket
- **RS-232 interface connector:** Female DB-9
- **RS-422/RS-485 interface connector:** Plug-in screw terminal [accepts AWG 1-#12 or 2-#14-#22 (0.5 to 2.5mm²) wires]
- **Operating temperature:** -10 to 70°C
- **Dimension:** 2.36" x 4.41" (60mm x 120mm)
- **Power consumption:** 1 W

Installation

Initial Inspection

We carefully inspect the ADAM-4530 both mechanically and electrically before we ship it. It should be free of marks and scratches and in perfect order upon receipt.

As you unpack the module, check it for signs of shipping damage (damaged box, scratches, dents, etc.). If it is damaged or fails to meet our specifications, notify our service department or your local sales representative immediately. Also, call the carrier immediately and retain the shipping carton and packing material for inspection by the carrier. We will then make arrangements to repair or replace the unit.

Before you begin installation, please make sure you have the following items:

- 1 ADAM-4530 module
- 1 Bracket
- 1 Utility disk

Basic Configuration

Before installing the ADAM-4530 in an existing network, it should be configured. Though all modules are initially configured at the factory, it is recommended that you check the baud rate settings.

Factory Default Settings:

Protocol: RS-485
Baud Rate: 9600 bps
Delimiter: {
Mode: Addressable
Add cr: Yes
Address: 01

Power Supply

For the ease of use in industrial environments the ADAM modules are designed to accept industry standard +24 V_{DC} unregulated power. Operation is guaranteed when using any power supply between +10 and +30 V_{DC}. Power ripples must be limited to 5 V peak to peak while the voltage in all cases must be maintained between +10 and +30 V_{DC}. All power supply specifications are referenced at module connector.

The power cables should be selected according to the number of modules connected and the length of the power lines. When using a network with long cables, we advise the use of thicker wire to limit the line voltage drop. In addition to serious voltage drops, long voltage lines can also cause interference with communication wires.

We advise that the following standard colors (as indicated on the modules) be used for power lines:

+Vs → (R) Red
GND → (B) Black

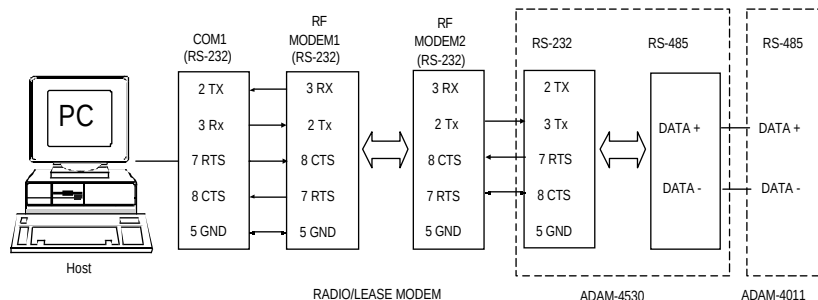
Communication Wiring

We recommend that shielded-twisted-pair cables that comply with the EIA RS-485 standard be used with the ADAM network to reduce interference. Only one set of twisted-pair cables is required to transmit both Data and RTS signals. We advise that the following standard colors (as indicated on the modules) be used for the communication lines:

DATA + (Y) → Yellow
DATA - (G) → Green

Basic Connection Diagram: Addressable Mode

Note: For Non-Addressable mode wiring, please refer to p.7.



Setup

When you select Setup from the menu, a selection bar will appear in the status field. Highlight the module you wish to configure and press <Enter>. A new screen displaying the module's current configuration will offer six menu options: Configuration, RS-232 Setup, RS-422/RS-485 Setup, Digital Output, Timer and Identification.



Figure 3 Setup Options

1. To change the basic settings, select Configuration and press <Enter>. Highlight the parameter you wish to change, and press <Enter>. A window will appear with the configuration options for that parameter. Highlight the proper value, and press <Enter>. To return to the previous screen without making any changes, press <Esc>. This works for most screens in the program.

NOTE: When you choose Alternate RTS as DO, the "ARTS/DO" signal can be regarded as a normal Digital output signal (open collector.) Refer to the Digital Output Command listed on p. 5. When you choose Alternate RTS as RTS, the ARTS/DO simply works in sync with RTS as a normal RTS.

2. To change the RS-232/RS-485(RS-422) baud rate settings, select RS-232 Setup or RS-422/485 Setup and press <Enter>. Highlight the parameter you wish to change and press <Enter>. A window will appear with the configuration options for that parameter. Highlight the proper value and press <Enter>.

NOTE: This module allows different baud rates for the RS-232 and the RS-485. However, if there is a difference of 3 times (3x) or more, the data command must be limited to 16 bytes.

3. The Mode (addressable or non-addressable) and Baud Rate options need special attention because they can only be changed when ADAM-4530 is in the INIT* state. To alter the Mode and Baud Rate, do the following:
 - Power all components on except ADAM-4530.
 - Power ADAM-4530 on while shorting the INIT* and GND terminals.
 - Configure the Mode and/or Baud Rate.
 - Power ADAM-4530 off.
 - Remove the grounding of the INIT* terminal, and power ADAM-4530 on.
 - Check the settings.

Init* state defaults

Baud rate: 9600 bps
Data bit: 8
Stop bit: 1
Parity check: None
Protocol: None

4. If you wish to set the values of the module's output, select Digital Output and press <Enter>. Then toggle the output value with the spacebar and press <Enter> when finished.
5. To configure programmable delay time, select Timer and press <Enter>. Next, type a specific value after selecting the appropriate parameter.
6. In addition, you can assign the module with proper ID through the Identification option.
7. After you have made the changes for a block of parameters, press <ESC>. You will be asked if you are satisfied with the changes you have made or not. Press Y to keep the changes you have made, N to escape without changing the values.

NOTE: When changing configuration parameters, always make sure a window appears notifying you that the target module has confirmed the changes.

Terminal

This selection allows you to directly send and receive commands on the RS-485 line. It has two options: Single Line and Full Screen.

Choosing Single Line will place the ADAM-4530 in command test mode, where you send commands one at a time by typing them into the top blank and pressing <Enter>. The response appears in the bottom blank. To send the command again, simply press <Enter>.

Terminal emulation mode is a full-screen version of command test. Previous commands and their responses stay on the screen for your reference. If you want to repeatedly send a command, press <F10>. A dialogue box will appear into which you can enter the command. Press <Enter> to send the command. To stop the repeating command, press any key.



Figure 4 Command Test Mode

A box on the right hand side of the screen shows the communication parameters for the serial line such as the baud rate and number of stop bits.

Quit

Choosing Quit ends the ADAM utility program.

Command Set

To avoid communication conflicts when several devices try to send data at the same time, all actions are instigated by the host computer. The basic form is a command/response protocol with the host initiating the sequence.

When modules are not transmitting, they are in listen mode. The host issues a command to a module with a specified address and waits a certain amount of time for the module to respond. If no response arrives, a timeout aborts the sequence and returns control to the host.

Syntax

[delimiter character] [address] [command] [data]
[checksum] [carriage return]

Every command begins with a delimiter character. Most of the commands can use a dollar sign \$ or a percentage sign %, but the *Data Pass* command uses one of eight special characters, as described on the following page.

The delimiter character is followed by a two-character address (hexadecimal) that specifies the target module. The actual two character command follows the address. Depending on the command, an optional data segment follows the command string. An optional two character checksum may be appended to the total string. Every command is terminated by a carriage return (cr).

NOTE: All commands should be issued in uppercase characters.

Command Syntax	Command Name	Description
\$AA1(D)	Digital Output Command	Requests to set digital output from module at address AA.
\$AA3(DDDD) (DDDD) (DDDD)	Set T1, T2, T3	Sets T1, T2, T3 timer to the module at address AA.
\$AA4	Read T1, T2, T3	Reads T1, T2, T3 timer from the module at address AA.
\$AA6(ID)	Write Identification	Sets the identification data (0 to 24 bytes) to the module at address AA.
\$AA7	Read Identification	Reads identification data from the module at address AA.
\$AACD	Set Delimiter	Sets delimiter character (can be ~, !, ", #, \$, %, &, ' , (,), *, +, -, ., /, : , ; , < , = , > , ? , [, \ , ^ , _ , ` , { , , } , ~) for data pass command.
\$AAD	Read Delimiter	Reads delimiter character from the module at address AA.
&AA8(D1D2)	Set Data Gate Command	Requests the module at address to set data gate to open(1) or close(0).
&AA9	Read Data Gate Status	Requests the module at address AA to return data gate status.
{AADATA	Data Pass from RS-232 to RS-485/422	Allows module at address AA pass data to RS-485.

Command Descriptions

Digital Output Command

Description Requests to set digital output from the module at address AA.

Syntax \$AA1(D) (cr)

Response !AA (cr) if the command is valid.
?AA (cr) if an invalid operation is entered.

Example command: \$2410 (cr)
response: !24 (cr)

The command asks the module at address 24h to set digital output to 0 (off).

Set T1, T2, T3

Description Sets T1, T2, T3 timer to the module at address AA.

Syntax \$AA3(DDDD) (DDDD) (DDDD)(cr)

Response !AA (cr) if the command is valid.

Example command: \$2430001 0001 0010 (cr)
response: !24 (cr)

The command asks the module at address 24h to write T1=0001ms, T2=0001ms, T3=0010ms to EEPROM.

Read T1, T2, T3

Description Reads T1, T2, T3 timer from the module at address AA.

Syntax \$AA4 (cr)

Response !AAT1 T2 T3 (cr) if the command is valid

Example command: \$244 (cr)
response: !240001 0001 0010 (cr)

The command asks the module at address 24h to return the T1, T2, T3 timer data from EEPROM.

Write Identification

Description Assigns identification data to the module at address AA. Identification lengths can range from 0 to 24 bytes.

Syntax \$AA6(identification) (cr)

Response !AA (cr) if the command is valid.

Example command: \$24ADAM NETWORK 1 (cr)
response: !24 (cr)

The command asks the module at address 24h to write "ADAM NETWORK 1" to EEPROM.

Read Identification

Description Reads identification data from the module at address AA.

Syntax \$AA7 (cr)

Response !AA(identifier) (cr)

Example command: \$246 (cr)
response: !24ADAM NETWORK 1 (cr)

The command asks the module at address 24h to return the identification data from EEPROM.

Set Delimiter

Description Sets the delimiter character for the Data Pass command. There are 8 special characters to choose from:

: [] ^ { | } ~

Syntax \$AAC(delimiter) (cr)

Response !AA (cr) if the command is valid

Example command: \$24C{ (cr)
response: !24 (cr)

The command asks the module at address 24h to write delimiter character { to EEPROM.

Read Delimiter

Description Reads the delimiter character from the module at address AA.

Syntax \$AAD (cr)

Response !AA(delimiter) (cr) if the command is valid. (delimiter) is a character for the Data Pass command.

Example command: \$24D (cr)
response: !24{ (cr)

The command asks the module at address 24h to return the delimiter character { from EEPROM.

Set Data Gate

Description Requests the module at address AA to set data gate to open (1) or close (0).

Syntax &AA8(D1D2) (cr)

Response !AA (cr) if the command is valid.
D1=1 Open data gate
D1=0 Close data gate
D2=1 Write to EEPROM
D2=0 Do not write to EEPROM

Example command: \$24A00 (cr)
response: !24 (cr)

The command asks the module at address 24h to close the data gate and not to write to EEPROM.

command: \$24A11 (cr)
response: !24 (cr)

The command asks the module at address 24h to open the data gate and write to EEPROM.

- Note:**
1. If the command asks the module to write to EPROM, the status will remain at **Open Gate/Write to EPROM** when the power is turned off and on again.
 2. If the command asks the module to open the data gate, the module will automatically assume non-addressable mode. This enables the module to issue command without adding {AA until the user issue another command to close the data gate.

Read Data Gate Status

Description Requests the module at address AA to return data gate status.

Syntax &AA9 (cr)

Response !AAS (cr)
S=0 Gate closed
S=1 Gate opened

Example command: \$24B (cr)
response: !240 (cr)

The command asks the module at address 24h to return the data gate status.

Data Pass

NOTE: This module allows different baud rates for the RS-232 and the RS-485. However, if there is a difference of three times (3x) or more, the data command must be limited to 16 bytes.

Description Allows the module at address AA to pass data from RS-232 to RS-485/422.

Syntax (delimiter)AADATA (cr)

Response Depends on the device.

Example command: {01#02 (cr)
response: (RS-485 response data) (cr)

The command asks the module at address 01h to send #02 to RS-485 communication port. (cr) automatically adds to #02 if you set "YES" in the "ADD <CR>" row. If ADAM-4530 receives any data from RS-485 port, it will pass the data to RS-232.

Example Programs

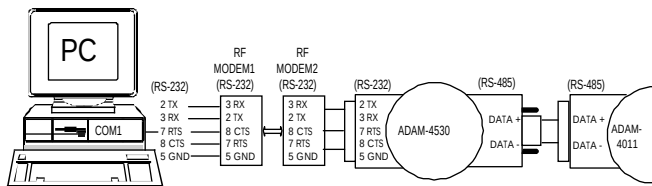
DEMO02.C for Address Mode

This demo program demonstrates how you might send a string of data to the ADAM-4011(RS-485) via the ADAM-4530.

ADAM-4530 Configuration Status:

```
ADDRESS:02
RS-232 baud  9600, N, 8, 1
RS-485 baud 19200, N, 8, 1
MODE: ADDRESSABLE
T1=100ms T2=200ms T3=100ms
RTS, CTS Active Low
ADD (CR) : Yes

PC RS-232      baud(9600), N, 8, 1
RF-MODEM1      baud(9600), N, 8, 1 (EFJohnson 3472 in handshake mode, RTS, CTS Active Low)
RF-MODEM2      baud(9600), N, 8, 1 (EFJohnson 3472 in handshake mode, RTS, CTS Active Low)
ADAM-4011      baud(19200), N, 8, 1
ADDRESS:01
```



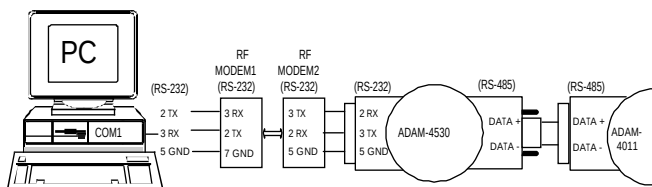
DEMO01.C for Non-Addressable Mode

This demo program demonstrates how you might send a string of data to the ADAM-4011 (RS-485) via the ADAM-4530.

ADAM-4530 Configuration Status:

```
RS-232 baud  9600, N, 8, 1
RS-485 baud 19200, N, 8, 1
MODE: NON-ADDRESSABLE

PC RS-232      baud(9600), N, 8, 1
RF-MODEM1      baud(9600), N, 8, 1 (Repcos ALR-96 in no handshake mode)
RF-MODEM2      baud(9600), N, 8, 1 (Repcos ALR-96 in no handshake mode)
ADAM-4011      baud(19200), N, 8, 1
ADDRESS:01
```



Program: DEMO02. C and DEMO01.C

```
#include <dos.h>
#include <io.h>
#include <stdio.h>
#include <conio.h>
#define TIME_OUT 500000
static int base0=0x3F8;
static char rec[36];
static char cmd[36];

void main()
{
    long int i, j, k, timeout;
    char flag;

    outp(base0+3,0x80); /* set DLAB=1 */
    outp(base0,0x0C); outp(base0+1,0x00); /* set baud=9600 */
    outp(base0+3,0x07); /* set data=8 stop=2 no parity */
    outp(base0+1,0x00); /* disable COM1 interrupt */

    printf("\nInput string : ");
    gets(cmd);
    while (cmd[0] != 'q' && cmd[0] != 'Q')
    {
        for (j=0;j<1;j++)
        {
            i=0;
            k=strlen(cmd);
            cmd[k]=0x0a;
            cmd[k]=0x0d;
            flag=1;
            outportb(base0+4,0); /* set RTS to low; (DEMO02.C only) */
            while((inportb(base0+6)&0x10)==0x10; /* waiting for RF modem ack; (DEMO02.C only)*/
            while (flag)
            k=strlen(cmd);
            cmd[k]=0x0a;
            cmd[k]=0x0d;
            flag=1;
            while (flag)
            {
                outportb(base0,cmd[i]); /* send data */
                while ((inportb(base0+5) & 0x60) != 0x60);
                if (cmd[i] == 0x0d)
                    flag=0;

                i++;
            }
            printf("i(send"="%d",i);
            while( (inportb(base0+5)&0x60) !=0x60);

            i=0;
            flag=1;
            timeout=TIME_OUT;
            while (flag)
            {
                /* Check receiver data ready? */
                if ((inportb(base0+5) & 1) !=0)
                {
                    rec[i]=inportb(base0); /* Receive data */
                    if (rec[i] == 0x0d)
                    {
                        rec[i+1]='\0';
                        flag=0;
                        printf("\nReceived data : %s\n",rec);
                    }
                }
            }
        }
    }
}
```

```

        i++;
    }
    else
    {
        /* Check timeout */
        timeout--;
        if (timeout == 0)
        {
            flag = 0;
            printf("\nTimeout error\n");
        }
    }
}
}
}
printf("\nInput command : ");
gets(cmd);
}
}

```

DEMO02.C Output (addressable mode)

```

Input string:  {02#01
Received data:  >+20,000
Input string:  {02$01M
Received data:  !014011

```

DEMO01.C Output (non-addressable mode)

```

Input string:  #01
Received data:  >+20.000
Input string:  $01M
Received data:  !014011

```

Dimensions are shown in mm