

*DATA ACQUISITION MODULES*

# 产品说明



R4018

## 模块用户手册

2005年8月

<http://www.sa68.com>  
info@sa68.com

北京捷麦通信器材有限公司

# 目录

1. 概述	2
1.1 端子分布	2
1.2 特性	3
1.3 结构图	3
1.4 默认设置	4
1.5 跳线设置	4
1.6 接线说明	4
1.7 校准	5
1.8 设置列表	6
2. 指令	8
2.1 %AANNTCCFF	9
2.2 \$AA2	9
2.3 \$AAF	10
2.4 \$AAM	11
2.5 ~AAO(数据)	11
2.6 ~AAEV	12
2.7 \$AA0	12
2.8 \$AA1	13
2.9 #AA	13
2.10 \$AA5VV	14
2.11 \$AA6	14
2.12 #AAN	15
2.13 \$AA3	15
2.14 \$AA9(数据)	16
2.15 ~**	16
2.16 ~AA0	17
2.17 ~AA1	17
2.18 ~AA2	18
2.19 ~AA3EVV	18
3. 应用注意	19
3.1 INIT* 端子操作	19
3.2 双重看门狗操作	20
3.3 热电偶测量	20

# 1. 概述

R4000是具有网络数据采集和控制功能的一系列模块。它们提供模拟到数字,数字到模拟,数字输入/输出,定时器/计数器和其他的功能。这些模块由一系列指令远程控制完成。R4018是含有8通道的模拟量输入模块,特征如下:

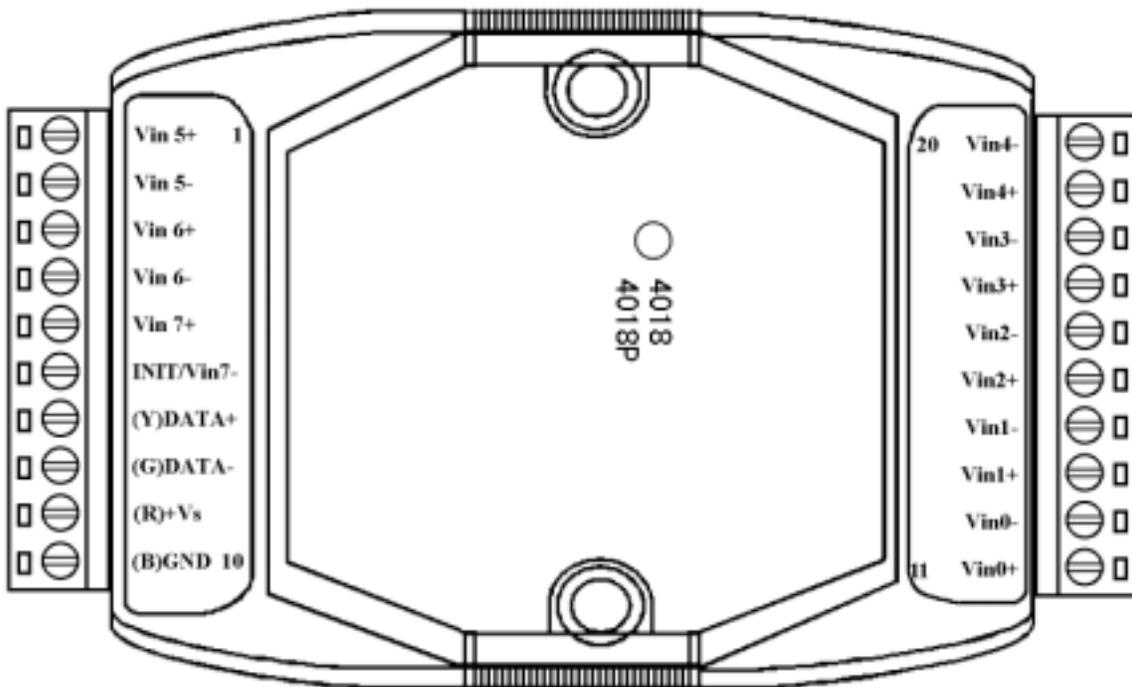
3000VDC 隔离模拟量输入

24位sigma-delta ADC来提供较高的精度

内置CJC,与热电偶直接连接

软件校准

## 1.1 端子分布



## 1.2 特性

R4018

### 模拟输入

输入通道 :8 路或 6 路差分, 2 路单端.跳线选择.

输入类型 :mV, V, mA(外带 125 ohms 电阻)

热电偶: 类型 J, K,T, E, R, S, B, N, C

采样速率 :10 样本/秒

带宽 :15.7 Hz

精度 : $\pm 0.1\%$

零漂移 : $0.5\mu\text{V}/^\circ\text{C}$

量程漂移 :25ppm/ $^\circ\text{C}$

CMR@50/60Hz :150dB

NMR@50/60Hz :100dB

输入阻抗 :20M Ohms

过载电压保护 : $\pm 35\text{V}$

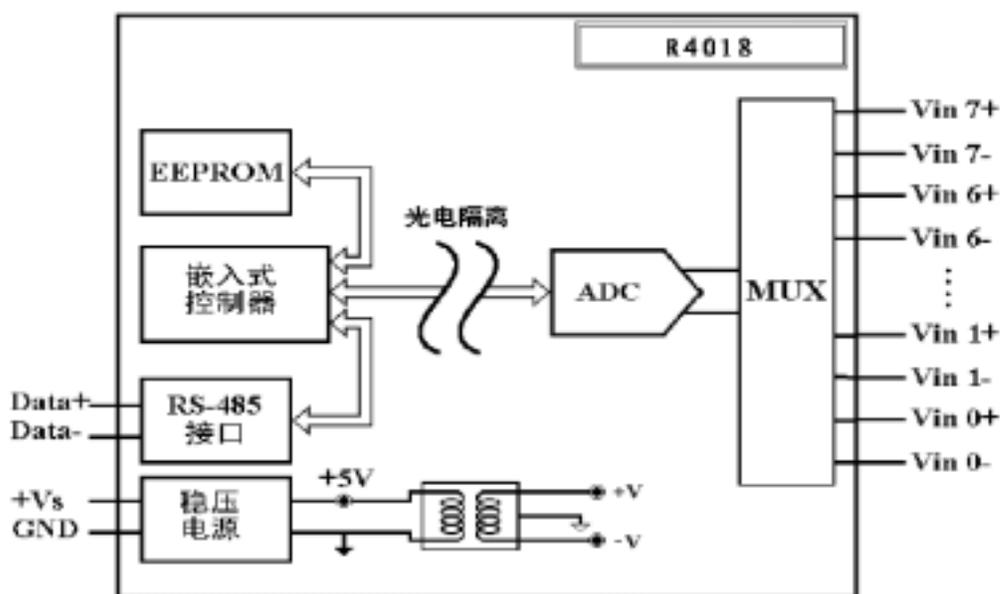
隔离 :3000VDC

### 供应功率

输入: +10 到+30 VDC

功耗 :1.0W

## 1.3 结构图



## 1.4 默认设置

R4018 的默认设置：

地址：01

模拟输入类型：类型 05, -2.5 到+2.5 V

波特率: 9600 bps

总效验禁止, 抑制 60Hz 干扰, 工程量单位格式

R4018 设置在 INIT\* 模式, 且模拟量输入是 6 路差分, 2 路单端模式

## 1.5 跳线设置

R4018：跳线 JP1 选择端子 INIT\*/Vin 7-

选择 8 路差分模式, 端子 INIT\*/Vin7- 设置成 Vin7-



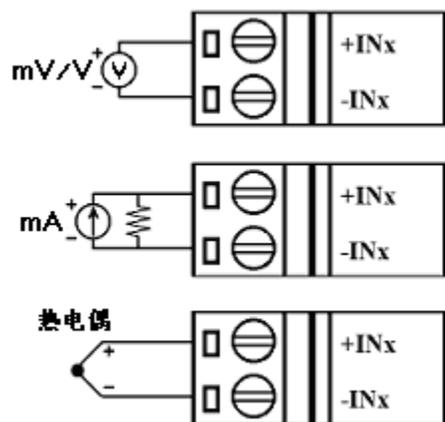
选择 INIT\*模式,端子 INIT\*/Vin7-设置成 INIT\*



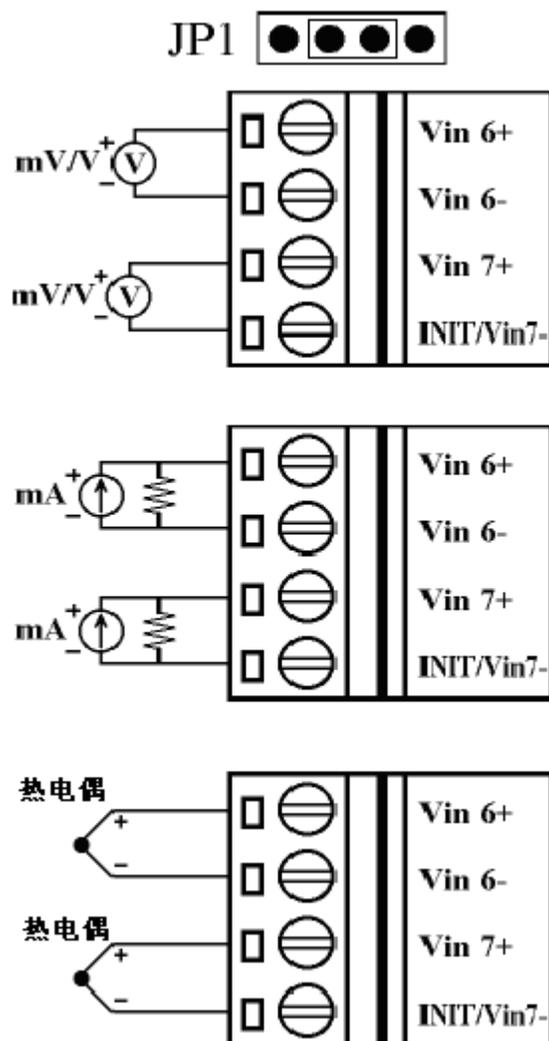
## 1.6 接线说明

R4018

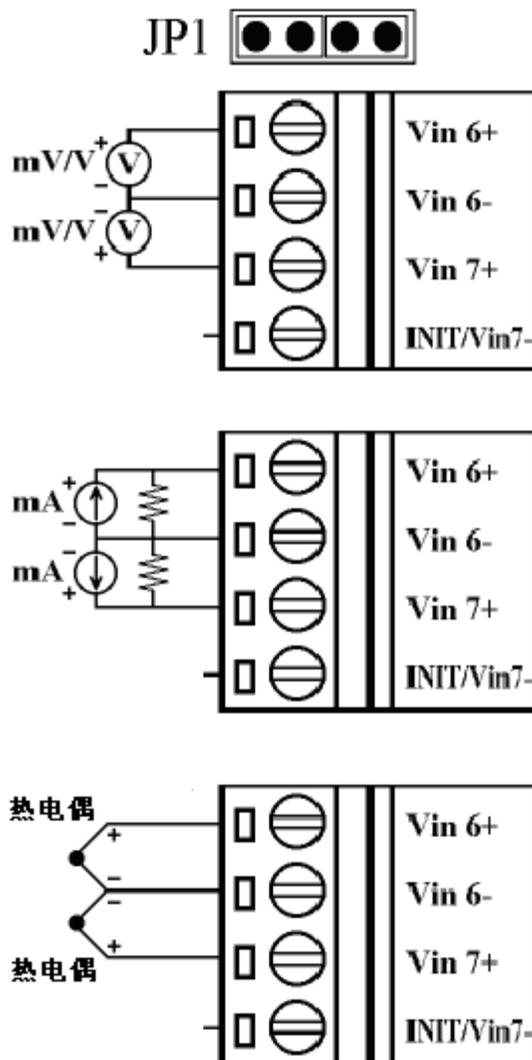
模拟量输入通道 0 到 5 的接线说明



R4018 模拟输入通道 6 和 7 的  
接线说明，跳线 JP1 设置在 8 路  
差分模式



R4018 模拟输入通道 6 和 7 的  
接线说明，跳线 JP1 设置在 INIT\*  
模式



## 1.7 校准

如果您还没有真正理解校准含义，请不要执行校准  
R4018 的校准需求

代码类型	00	01	02	03	04	05	06
0输入	0mV	0mV	0mV	0mV	0V	0V	0mA
满量程输入	+15mV	+50mV	+100mV	+500mV	+1V	+2.5V	+20mA

注意:

1 校准类型是 06 时，需要连接外部的电阻，125 ohms, 0.1%

- 2 连接校准电压(或电流)信号到模块的输入端。对 R4018,连接到通道 0.
- 3 校准前,先要通电预热模块大约 30 分钟,以确保更好的精度。

以类型 00 的校准为例,步骤如下:

- 1 设置类型为 00 ->%AANNCCFFTT
- 2 启动校准 ->~AAEV
- 3 给定零校准电压 (0mV)
- 4 执行零校准指令 ->\$AA1
- 5 给定满量程校准电压(15mV)
- 6 执行满量程校准指令 ->\$AA0
- 7 重复步骤 1 到 6 三次。

其他类型的校准步骤大体相同,只是步骤一-的设置类型有所不同。

## 1.8 设置列表

### R4018 的设置列表 波特率设置(CC)

代码	03	04	05	06	07	08	09	0A
波特率	1200	2400	4800	9600	19200	38400	57600	115200

### 模拟输入类型设置(TT)

类型代码	00	01	02	03	04	05	06
最小输入	-15mV	-50mV	-100mV	-500mV	-1V	-2.5V	-20mA
最大输入	+15mV	+50mV	+100mV	+500mV	+1V	+2.5V	+20mA

类型代码	0E	0F	10	11	12	13	14	15	16
T.C 类型	J	K	T	E	R	S	B	N	C
最小温度	-210	-270	-270	-270	0	0	0	-270	0
最大温度	760	1372	400	1000	1768	1768	1820	1300	2320
温度以摄氏度为单位									

### 数据格式设置 (FF)

7	6	5	4	3	2	1	0
*1	*2	0	0	0	0	*3	

\*1 :滤波器选择: 0 = 60Hz 抑制接收  
1 = 50Hz 抑制接收

\*2 :校验位: 0=禁止, 1=允许

\*3 : 00 =工程量单位格式

01 =百分比格式

10 = 二进制补码采用 16 进制格式

## 模拟量输入类型及数据格式表

类型代码	输入范围	数据格式	正幅值	零点	正幅值
00	-15 到+15mV	工程单位	+15.000	+00.000	-15.000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
01	-50 到+50mV	工程单位	+50.000	+00.000	-50.000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
02	-100 到+100mV	工程单位	+100.00	+000.00	-100.000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
03	-500 到+500mV	工程单位	+500.00	+000.00	-500.000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
04	-1 到+1V	工程单位	+1.0000	+0.0000	-1.0000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
05	-2.5 到+2.5V	工程单位	+2.5000	+0.0000	-2.5000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000
06	-20 到+20mA	工程单位	+20.000	+00.000	-20.000
		百分比格式	+100.00	+000.00	-100.00
		16 进制补码	7FFF	0000	8000

类型代码	输入范围	数据格式	正幅值	零点	负幅值
0E	<b>类型 J</b> -210 到 760 摄氏度	工程单位	+760.00	+000.00	-210.00
		百分比格式	+100.00	+000.00	-027.63
		16 进制补码	7FFF	0000	DCA2
0F	<b>类型 K</b> -270 到 1320 摄氏度	工程单位	+1320.0	+0000.0	-0270.0
		百分比格式	+100.00	+000.00	-019.68
		16 进制补码	7FFF	0000	E6D0
10	<b>类型 T</b> -270 到 400 摄氏度	工程单位	+400.00	+000.00	-270.00
		百分比格式	+100.00	+000.00	-067.50
		16 进制补码	7FFF	0000	A99A
11	<b>类型 E</b> -270 到 1000 摄氏度	工程单位	+1000.0	+0000.0	-0270.0
		百分比格式	+100.00	+000.00	-027.00
		16 进制补码	7FFF	0000	DD71

12	<b>类型 R</b> 0 到 1768 摄氏度	工程单位	+1768.0	+0000.0	+0000.0
		百分比格式	+100.00	+000.00	+000.00
		16 进制补码	7FFF	0000	0000
13	<b>类型 S</b> 0 到 1768 摄氏度	工程单位	+1768.0	+0000.0	+0000.0
		百分比格式	+100.00	+000.00	+000.00
		16 进制补码	7FFF	0000	0000
14	<b>类型 B</b> 0 到 1820 摄氏度	工程单位	+1820.0	+0000.0	+0000.0
		百分比格式	+100.00	+000.00	+000.00
		16 进制补码	7FFF	0000	0000
15	<b>类型 N</b> -270 到 1300 摄氏度	工程单位	+1300.0	+0000.0	-0270.0
		百分比格式	+100.00	+000.00	-020.77
		16 进制补码	7FFF	0000	E56B
16	<b>类型 C</b> 0 到 2320 摄氏度	工程单位	+2320.0	+0000.0	+0000.0
		百分比格式	+100.00	+000.00	+000.00
		16 进制补码	7FFF	0000	0000

## 2. 指令

指令格式：(第一位)(地址)(指令)[CHK](cr)

响应格式：(第一位)(地址)(数据)[CHK](cr)

[CHK] 2 字符效验

(cr) 指令结束符，返回字符(0x0D)

通用指令设置		
指令	响应	描述
%AANN TTCCFF	!AA	设置模块
\$AA2	!AATTCFF	读配置信息
\$AAF	!AA(数据)	读版本信息
\$AAM	!AA(数据)	读模块名
~AAO(数据)	!AA	设置模块名

模拟量测量指令设置		
指令	响应	描述
~AAEV	!AA	校准允许/禁止
\$AA0	!AA	执行量程校准
\$AA1	!AA	执行零校准
#AA	>(数据)	读模拟量输入
\$AA5VV	!AA	设置通道工作
\$AA6	!AAVV	读通道状态
#AAN	>(数据)	从通道 N 读模拟输入
\$AA3	>(数据)	读 CJC 温度
\$AA9(数据)	!AA	设置 CJC 偏移值

主机看门狗指令设置		
指令	响应	描述
~**	无响应	主机正常
~AA0	!AASS	读主看门狗状态
~AA1	!AA	复位主看门狗状态
~AA2	!AAVV	读主机看门狗溢出时间
~AA3Evv	!AA	设置主机看门狗溢出时间

## 2.1 %AANNTTCCFF

**描述** : 设置模块配置

**语法** : %AANNTTCCFF[CHK](cr)

% 字符分隔符

AA 设置模块地址(00 到 FF)

NN 设置模块新地址(00 到 FF)

TT 设置模块新类型

CC 设置模块新波特率

FF 设置模块新的数据格式

当转换波特率或效验和时，将 INIT\*端子接地。

**响应**: 有效指令: !AA[CHK](cr)

无效指令: ?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符

? 无效指令分隔符。当转换波特率或效验和时没有将 INIT\*接地的话，模块将会返回无效指令。

AA 响应模块地址(00 to FF)

**例如** :

指令 : %0102050600                      接收: !02

转换地址从 01 到 02, 返回成功.

## 2.2 \$AA2

**描述**: 读配置信息

**语法**: \$AA2[CHK](cr)

\$ 字符分隔符

AA 读模块地址 (00 到 FF)

2 读配置指令

**响应**: 有效指令: !AATTCCFF[CHK](cr)

	无效指令: ?AA[CHK](cr)
	语法错误或通信错误会导致无响应。
!	有效指令分隔符
?	无效指令分隔符
AA	响应模块地址(00 到 FF)
TT	模块类型代码
CC	模块波特率代码
FF	模块数据格式

例如：

指令: \$012            接收: !01080600

读地址为01的配置信息, 返回模式为08, 波特率9600, 无校验, 工程量单位。

指令: \$002            接收: !020A0702

INIT 模式下, 读模块配置信息, 返回地址为 02, 模式为 0A, 波特率 9600, 无校验, 16 进制单位。

## 2.3 \$AAF

**描述** : 读版本信息

**语法** : \$AAF[CHK](cr)

\$	字符分隔符
AA	读模块地址 (00 到 FF)
F	读版本指令

**响应** : 有效指令: !AA(数据)[CHK](cr)

无效指令: ?AA[CHK](cr)

语法错误或通信错误会导致无响应。

!	有效指令分隔符
?	有效指令分隔符
AA	响应模块地址(00 到 FF)
(数据)	模块的版本

例如:

指令 : \$01F            接收 : !01A2.0

读地址为 01 的版本数据, 返回版本 BBA1

## 2.4 \$AAM

**描述**：读模块名称

**语法**：\$AAM[CHK](cr)

\$ 字符分隔符

AA 读模块地址 (00 到 FF)

M 读模块名称指令

**响应**：有效指令: !AA(数据)[CHK](cr)

无效指令: ?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符

? 无效指令分隔符

AA 响应模块地址(00 到 FF)

(数据) 模块名称

**例如**：

指令：\$01M 接收：!014018

读地址为 01 的模块名，返回名称 4018.

指令：\$03M 接收：!034011D

读地址为 03 的模块名，返回名称 4011D.

## 2.5 ~AAO(数据)

**描述**：设置模块名称

**语法**：~AAO(数据)[CHK](cr)

~ 字符分隔符

AA 设置模块地址 (00 到 FF)

O 设置模块名称指令

(数据) 模块的新名称，最多 6 字符

**响应**：有效指令: !AA[CHK](cr)

无效指令: ?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符

? 无效指令分隔符

AA 响应模块地址(00 到 FF)

**例如**：

指令：~01O4018 接收: !01

设置地址为 01 的模块名为 4018, 返回成功。

指令 : \$01M            接收: !014018

读地址为 01 的模块名, 返回 4018.

## 2.6 ~AAEV

**描述:**校准允许/禁止

**语法:** ~AAEV[CHK](cr)

~        字符分隔符

AA       设置模块地址 (00 到 FF)

E        校准允许/禁止指令

V        1=校准允许, 0=校准禁止

**响应 :**    有效指令: !AA[CHK](cr)

          无效指令: ?AA[CHK](cr)

          语法错误或通信错误会导致无响应。

!        有效指令分隔符

?        有效指令分隔符

AA       响应模块地址(00 到 FF)

**例如 :**

指令: \$010            接收: ?01

执行地址为 01 的范围校准, 返回没有准备好校准。

指令 : ~01E1          接收: !01

设置地址为 01 的校准允许, 返回成功.

指令: \$010            接收 : !01

执行地址为 01 的范围校准, 返回成功.

## 2.7 \$AA0

**描述 :** 执行范围校准

**语法 :** \$AA0[CHK](cr)

\$        字符分隔符

AA       设置模块地址(00 到 FF)

0        执行范围校准指令

**响应 :**    有效指令 : !AA[CHK](cr)

          无效指令 : ?AA[CHK](cr)

          语法错误或通信错误会导致无响应。

!        有效指令分隔符

? 无效指令分隔符或校准没有启动

AA 响应模块地址(00 到 FF)

例如:

指令 : \$010            接收 : !01

执行地址为 01 范围校准, 返回成功。

指令 : \$020            接收 : ?02

执行地址为 02 范围校准指令, 返回执行校准指令前, 不能执行校准功能。

## 2.8 \$AA1

**描述** : 执行零点校准

**语法** : \$AA1[CHK](cr)

\$        字符分隔符

AA       设置模块地址 (00 到 FF)

1        执行零点校准指令

**响应** : 有效指令: !AA[CHK](cr)

          无效指令: ?AA[CHK](cr)

          语法错误或通信错误会导致无响应。

!        有效指令分隔符

?        无效指令分隔符或校准没有启动

AA       响应模块地址(00 到 FF)

例如 :

指令 : \$011            接收: !01

执行地址为 01 的零点校准, 返回成功。

指令: \$021            接收: ?02

执行地址为 02 的零点校准, 返回执行校准指令前不能执行校准功能。

## 2.9 #AA

**描述**: 读模拟输入

**语法**: #AA[CHK](cr)

#        字符分隔符

AA       读模块指令(00 到 FF)

**响应** : 有效指令 : >(数据)[CHK](cr)

语法错误或通信错误会导致无响应。

> 有效指令分隔符  
(数据) 模拟输入值，格式为各通道数据依次首尾项链通道值的混合。

例如：

指令：#04

接收：>+05.123+04.153+07.234-02.356+10.000-05.133+02.345+08.234

模块地址为 04，得到 8 通道的模拟输入数据。

## 2.10 \$AA5VV

**描述：**设置单通道测量允许

**语法：**\$AA5VV[CHK](cr)

\$ 字符分隔符

AA 设置模块地址 (00 到 FF)

5 设置单通道测量允许指令

VV 通道的允许/禁止, 00=禁止所有通道, FF=开启所有通道。

**响应：**有效指令：!AA[CHK](cr)

无效指令：?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符

? 无效指令分隔符

AA 响应模块地址(00 到 FF)

例如：

指令：\$0155A 接收：!01

设置地址 01 通道 1,3,4,6 允许，通道 0,2,5,7 禁止,返回成功。

指令：\$016 接收：!015A

读地址 01 通道状态, 返回通道 1,3,4,6 允许，通道 0,2,5,7 禁止。

## 2.11 \$AA6

**描述：**读单通道测量状态

**语法：**\$AA6[CHK](cr)

\$ 字符分隔符

AA 读模块地址 (00 到 FF)

6 读读单通道测量状态指令

**响应：**有效指令：!AAVV[CHK](cr)

无效指令：?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符  
 ? 无效指令分隔符  
 AA 响应模块地址(00 到 FF)  
 VV 通道允许/禁止, 00=禁止所有通道, FF=开启所有通道。

例如：

指令：\$015A5                   接收：!01  
 设置地址 01 通道 0,2,5,7 允许，通道 1,3,4,6 禁止,返回成功。  
 指令：\$016                   接收：!01A5  
 读地址 01 通道状态, 返回通道 0,2,5,7 允许，通道 1,3,4,6 禁止。

## 2.12 #AAN

**描述**：由通道 N 读模拟量输入

**语法**：#AAN[CHK](cr)

# 字符分隔符  
 AA 读模块地址 (00 到 FF)  
 N 读通道, 从 0 到 7

**响应**：有效指令：>(数据)[CHK](cr)  
 无效指令：?AA[CHK](cr)  
 语法错误或通信错误会导致无响应。

> 有效指令分隔符  
 ? 无效指令分隔符  
 AA 响应模块地址(00 到 FF)  
 (数据) 模拟输入值

例如：

指令：#032                   接收：>+02.513  
 读地址为 03，通道 2, 成功得到数据。  
 指令：#029                   接收：?02  
 读地址为 02，通道 9, 返回错误通道名

## 2.13 \$AA3

**描述**：读 CJC 温度

**语法**：\$AA3[CHK](cr)

\$ 字符分隔符

AA 读模块地址 (00 到 FF)  
 3 读 CJC 温度指令  
**响应:** 有效指令: >(数据)[CHK](cr)  
 无效指令: ?AA[CHK](cr)  
 语法错误或通信错误会导致无响应。  
 > 有效指令分隔符  
 ? 无效指令分隔符  
 AA 响应模块地址 (00 到 FF)  
 (数据) CJC 温度用摄氏度表示。  
**例如:**  
 指令: \$033 接收: >+0025.4  
 读地址为 03 的 CJC 温度, 返回 25.4°C.

## 2.14 \$AA9(数据)

**描述:** 设置 CJC 偏移量

**语法:** \$AA9(数据)[CHK](cr)

\$ 字符分隔符  
 AA 设置模块地址 (00 到 FF)  
 9 设置 CJC 偏移量指令  
 (数据) CJC 偏移量包含 1 个标志位和 4 位 16 进制数, 从 -1000 到 +1000, 每个计数 0.01°C.  
**响应:** 有效指令: !AA[CHK](cr)  
 无效指令: ?AA[CHK](cr)  
 语法错误或通信错误会导致无响应。  
 ! 有效指令分隔符  
 ? 有效指令分隔符  
 AA 响应模块地址 (00 到 FF)

**例如:**

指令: \$019+0010 接收: !01  
 设置地址 01 CJC 偏移量增加 16 个单位(+0.16°C), 返回成功。

## 2.15 ~\*\*

**描述:** 主机 OK.

主机发送信息“Host OK”给所有模块

**语法:** ~\*\*[CHK](cr)

~ 字符分隔符  
 \*\* 给所有模块的指令

响应：无。

例如：

指令：~\*\* 无响应

发送 Host OK 给所有模块。

## 2.16 ~AA0

描述：读主看门狗状态

语法：~AA0[CHK](cr)

~ 字符分隔符

AA 读模块地址 (00 到 FF)

0 读模块状态指令

响应：有效指令：!AASS[CHK](cr)

无效指令：?AA[CHK](cr)

语法错误或通信错误会导致无响应。

! 有效指令分隔符

? 无效指令分隔符

AA 响应模块地址(00 到 FF)

SS 状态将存储到 EEPROM 中，且只有指令~AA1 可以复位。

7	6	5	4	3	2	1	0
*1	保留				*2	保留	

\*1：主机看门狗状态, 0=禁止, 1=允许

\*2：主机看门狗超时溢出标志，0=清除，1=设置

例如：

指令：~010 接收：!0104

读地址为 02 的模块状态，返回 04，主看门狗溢出标志被设置。

## 2.17 ~AA1

描述：复位主看门狗状态

语法：~AA1[CHK](cr)

~ 字符分隔符

AA 设置模块地址 (00 到 FF)

1            复位模块状态指令  
**响应 :**    有效指令: !AA[CHK](cr)  
             无效指令: ?AA[CHK](cr)  
             语法错误或通信错误会导致无响应。  
 !            有效指令分隔符  
 ?            无效指令分隔符  
 AA          响应模块地址(00 到 FF)

**例如 :**

指令 : ~010            接收: !0104  
 读地址为 01 的模块状态,返回 04,主看门狗溢出标志被设置。  
 指令 : ~011            接收: !01  
 复位地址为 01 的模块状态, 返回成功。  
 指令 : ~010            接收: !0100  
 读地址为 01 的模块状态, 返回 00, 模块状态被清除。

## 2.18 ~AA2

**描述:**读主看门狗超时间隔

**语法:** ~AA2[CHK](cr)

~            字符分隔符  
 AA          读模块地址 (00 到 FF)  
 2            读主看门狗超时间隔指令  
**响应 :**    有效指令: !AAVV[CHK](cr)  
             无效指令: ?AA[CHK](cr)  
             语法错误或通信错误会导致无响应。  
 !            有效指令分隔符  
 ?            无效指令分隔符  
 AA          响应模块地址(00 到 FF)  
 VV          超时间隔用 16 进制表示, 每个计数 0.1 秒。01=0.1 秒 ,  
             FF=25.5 秒。

**例如 :**

指令 : ~012    接收: !01FF  
 读地址为 01 的主看门狗超时间隔, 返回 FF,主看门狗超时间隔为 25.5 秒。

## 2.19 ~AA3E VV

**描述 :**设置主看门狗溢出时间

**语法** : ~AA3EVV[CHK](cr)

~            字符分隔符  
 AA          设置模块地址 (00 到 FF)  
 3            设置主看门狗溢出时间指令  
 E            主机看门狗 1=开启/0=关闭  
 VV          溢出时间, 从 01 到 FF, 每个数字代表 0.1 秒

**响应** :    有效指令: !AA[CHK](cr)  
           无效指令: ?AA[CHK](cr)  
           语法错误或通信错误会导致无响应。

!            有效指令分隔符  
 ?            无效指令分隔符  
 AA          响应模块地址(00 到 FF)

**例如:**

指令: ~013164            接收 : !01  
 设置地址为 01 的主看门狗启动, 且溢出时间为 64 (10.0 秒), 返回成功。  
 指令 : ~012            接收 : !0164  
 读地址为 01 的主看门狗溢出时间, 返回溢出时间为 64 (10.0 秒).

## 3. 应用注意

### 3.1 INIT\* 端子操作

每个 R4000 模块 都内嵌了一块 EEPROM 来存储配置信息如地址, 信号类型, 波特率以及其他参数。有时, 用户可能会忘记模块的配置信息。因此, R4000 有一个特殊的模式叫做“INIT 模式”, 可以用来帮助用户解决这样的问题。在“INIT 模式”下, 模块被强行设置为 地址=00, 波特率=9600bps, 无校验位

为了激活 INIT 模式, 请参考如下步骤:

- 步骤 1. 关闭电源
- 步骤 2. 将 INIT\* 端子接地。
- 步骤 3. 打开电源
- 步骤 4. 以 9600bps 发送指令 \$002(cr) 来读取存储在模块 EEPROM 中的配置信息。

## 3.2 双重看门狗操作

**双重看门狗 = 模块看门狗+主看门狗**

模块看门狗是模块的硬件复位电路，可用来监控模块的操作状态。当工作在恶劣或噪声严重的不良环境中，模块将会被外部信号干扰。该电路将会使模块及时复位以继续工作。

主看门狗是模块内软件实现的看门狗，用来监控主机操作状态。它的目的是预防网络上的通信故障或主机死机。这可以预防控制对象免受不可预料的情况影响。

拥有双重看门狗的 R4000 模块可以使控制系统变得更加稳定可靠。

## 3.3 热电偶测量

当两种不同材料的金属丝放到一起加热时，这两个金属丝温度变化将会引起开路电压的变化。所有不同材料连接都会有这样的影响。该电压叫做“指示电压”。温度上的微小变化都会引起指示电压呈线性变化并显示出来。

直接测量指示电压是不可行的，因为我们必须首先把电压计连接到热电偶上，这样，电压表可以使它们形成一个新的热电偶回路。因此，我们需要排除连接的热电偶来测量指示电压，这叫做“冷连接补偿”。

对于大多数的热电偶，温度 0°C 时，指示电压为 0V。排除热电偶连接电压的一个简单方法是将连接部分放到 0°C 的环境中，这时连接部分电压为 0V。一般来说，这对大多数应用情况并不适用，典型的方法是通过电热调节器来测量连接部分温度以测出连接部分电压。然后我们就可以从热电偶电压和连接部分电压测出指示电压，最后，我们就可以从指示电压算出温度了。