

PROFIBUS-DP

SLAVE PROTOCOL MANUAL

FOR RAMSEY

MICRO-TECH 2000

BATCH WEIGHT INDICATOR

MODEL 2102

RAMSEY

A THERMO SENTRON COMPANY

501 90th Avenue N.W. • Minneapolis, MN 55433 • 612.783.2500

REC 4064 REV B 10/00
PART NO. 058214

REVISION HISTORY

REVISION A
REVISION B

JUNE 1998
OCTOBER 2000

COPYRIGHT 1998, 2000 BY RAMSEY TECHNOLOGY, INC.

THIS DOCUMENT IS CONFIDENTIAL AND IS THE PROPERTY OF RAMSEY TECHNOLOGY, INC. ("RAMSEY"). IT MAY NOT BE COPIED OR REPRODUCED IN ANY WAY WITHOUT THE EXPRESS WRITTEN CONSENT OF RAMSEY. THIS DOCUMENT ALSO IS AN UNPUBLISHED WORK OF RAMSEY. RAMSEY INTENDS TO AND IS MAINTAINING THE WORK AS CONFIDENTIAL INFORMATION. RAMSEY ALSO MAY SEEK TO PROTECT THIS WORK AS AN UNPUBLISHED COPYRIGHT. IN THE EVENT OF EITHER INADVERTENT OR DELIBERATE PUBLICATION, RAMSEY INTENDS TO ENFORCE ITS RIGHTS TO THIS WORK UNDER THE COPYRIGHT LAWS AS A PUBLISHED WORK. THOSE HAVING ACCESS TO THIS WORK MAY NOT COPY, USE OR DISCLOSE THE INFORMATION IN THIS WORK UNLESS EXPRESSLY AUTHORIZED BY RAMSEY.

**PLEASE READ AND OBSERVE THE FOLLOWING SAFETY
PRECAUTIONS FOUND THROUGHOUT THIS MANUAL.**

DANGER

FAILURE TO OBSERVE WILL CAUSE VERY SERIOUS PERSONAL INJURY OR DEATH.

WARNING

FAILURE TO OBSERVE COULD CAUSE SERIOUS PERSONAL INJURY.

CAUTION

FAILURE TO OBSERVE MAY CAUSE MINOR OR MODERATE PERSONAL INJURY
OR DAMAGE TO THE EQUIPMENT.

RAMSEY PRODUCTS
PROFIBUS-DP SLAVE PROTOCOL
TABLE OF CONTENTS

<u>CHAPTER</u>	<u>TITLE</u>	<u>PAGE</u>
1.0	INTRODUCTION	1-1
1.1	GENERAL	1-1
1.2	PROFIBUS-DP	1-1
1.3	DATA TRANSFER	1-1
1.4	PROTECTION MECHANISMS	1-1
1.5	WARRANTY	1-2
2.0	INSTALLATION	2-1
2.1	GENERAL	2-1
2.2	PROFIBUS-DP INTERFACE BOARD	2-1
2.2	CABLE TYPE AND LENGTH	2-1
	2.2.1 Application	2-1
	2.2.2 RS 485 Repeater Application	2-1
2.3	PROFIBUS-DP CONNECTOR	2-2
	2.3.1 Bus Connector Wiring	2-2
	2.3.2 Termination Power	2-3
2.4	PROFIBUS-DP BOARD INSTALLATION	2-3
3.0	OPERATION	3-1
3.1	GENERAL	3-1
3.2	MICRO-TECH 2000 CONFIGURATION	3-1
	3.2.1 Slave Address and Buffer Dimensions	3-1
	3.2.2 PROFIB Scroll	3-1
	3.2.3 Baud Rate	3-2
3.3	DATA TRANSFER	3-3
	3.3.1 Write Operations	3-3
	3.3.2 Read Operations	3-5
3.4	TIMINGS	3-6

TABLE OF CONTENTS
(continued)

<u>CHAPTER</u>	<u>TITLE</u>	<u>PAGE</u>
3.5.	COMMUNICATION ERRORS	3-6
3.5.1.	Diagnostics Data	3-6
3.6	DESCRIPTION OF THE VARIABLE	3-8
3.6.1	Legend	3-8
3.6.2	Calibration Commands	3-12
3.6.3	Batch Commands	3-12
3.6.4	Reset Commands	3-12
3.7	READ DATA GROUPS COMPOSITION	3-19
3.7.1	Read GROUP 0 - Dynamic Variables, Status and Sets ...	3-19
3.7.2	Read GROUP 1 - Sets and Thresholds	3-20
3.7.3	Read GROUP 2 - Display	3-21
3.7.4	Read GROUP 3 - Reading of the Component Total Registers	3-22
3.7.5	Read GROUP 4 - Reading of the Component Last Total Registers	3-23
3.7.6	Read GROUP 5 - Reading of the Recipe Total Registers	3-23
3.7.7	GROUP 6 - Reading of the Recipe Last Total Registers .	3-23
3.8	WRITE DATA GROUPS COMPOSITION	3-24
3.8.1	Write GROUP 100 - Commands	3-24
3.8.2	Write GROUP 101 - Dynamic Data and Sets	3-25
3.8.3	Write GROUP 102 - Writing a Recipe Line	3-25

LIST OF ILLUSTRATIONS

<u>FIGURE NO.</u>	<u>TITLE</u>	<u>PAGE</u>
2-1	PROFIBUS-DP Interface Board	2-1
2-2	Micro-Tech 2000 Mother Board	2-4

LIST OF TABLES

<u>TABLE NO.</u>	<u>TITLE</u>	<u>PAGE</u>
2-1	Maximum Cable Length of a Segment (Standard LAN Cable)	2-2
2-2	Maximum Cable Length Between Two Nodes (Standard LAN Cables)	2-2
2-3	Bus Connector	2-3
3-1	Composition of the Write Package	3-4
3-2	Composition of the Read Buffer After Power On	3-5
3-3	Composition of the Diagnostics Data	3-6
3-4	Variables and Their Characteristics	3-9
3-5	Key Codes	3-16
3-6	Read Group 0 [Vat1]	3-19
3-7	Read Group 1 [Vat2]	3-20
3-8	Read Group 2 [Vat3]	3-21
3-9	Read Group 3 [Vat4]	3-22
3-10	Read Group 4 [Vat4]	3-23
3-11	Read Group 5 [Vat4]	3-23
3-12	Read Group 6 [Vat4]	3-23
3-13	Write Group 100 [Vat5]	3-24
3-14	Write Group 101 [Vat6]	3-24
3-15	Write Group 102 [Vat7]	3-25

APPENDIX

<u>NO.</u>	<u>TITLE</u>
A/1	Floating Point Notation
A/2	Micro-Tech 2000 Configuration, The GSD File

CHAPTER 1.0 INTRODUCTION

1.1 GENERAL

PROFIBUS was developed by the need to have an open, vendor independent communication system that allows connection to any components used in the automation of technical processes.

The PROFIBUS is a standardized communication system, all its characteristics, rules and technical information are collected in the PROFIBUS DIN normative.

There are three main variations of PROFIBUS corresponding to the intended application: PROFIBUS-FMS, PROFIBUS-DP and PROFIBUS-PA. The Micro-Tech 2000 variation is PROFIBUS-DP.

1.2 PROFIBUS-DP

PROFIBUS-DP is the performance optimized version specifically dedicated to time-critical communication between automation systems and distributed peripherals.

It is typically used to transfer I/O images between a main PLC and remote devices (sensors, actuators, transmitters, etc.). In this case, it will be used to transfer (read and write) blocks of data.

PROFIBUS is a typical master/slave communication where the main PLC is the master or scanner, and the Micro-Tech 2000's are slaves. The connection is EIA RS 485 through a 2-wire twinax Siemens cable.

1.3 DATA TRANSFER

The interface between the master and slaves is structured in parameterization, configuration, and data transfer phase. In the parameterization and configuration phases, each slave compares its real configuration with the configuration data received from the master. When verifying the configuration, device type, format and length of information have to be identical. This guarantees a protection against a configuration fault. Maximum number of data that is possible to transfer in this phase at one time is limited to 246 bytes (123 words).

1.4 PROTECTION MECHANISMS

Besides the configuration, check to avoid erroneous configurations, the slave uses the watch dog control to detect failures on the bus. If a slave recognizes no successful data transfer with the master within the watch dog control interval, it generates an alarm condition.

1.5 WARRANTY

RAMSEY TECHNOLOGY, INC.

WARRANTY

The seller agrees, represents, and warrants that the equipment delivered hereunder shall be free from defects in material and workmanship. Such warranty shall not apply to accessories, parts, or material purchased by the seller unless they are manufactured pursuant to seller's design, but shall apply to the workmanship incorporated in the installation of such items in the complete equipment. To the extent purchased parts or accessories are covered by the manufacturer's warranty, seller shall extend such warranty to buyer.

Seller's obligation under said warranty is conditioned upon the return of the defective equipment, transportation charges prepaid, to the seller's factory in Minneapolis, Minnesota, and the submission of reasonable proof to seller prior to return of the equipment that the defect is due to a matter embraced within seller's warranty hereunder. Any such defect in material and workmanship shall be presented to seller as soon as such alleged errors or defects are discovered by purchaser and seller is given opportunity to investigate and correct alleged errors or defects and in all cases, buyer must have notified seller thereof within one (1) year after delivery, or one (1) year after installation if the installation was accomplished by the seller.

Said warranty shall not apply if the equipment shall not have been operated and maintained in accordance with seller's written instructions applicable to such equipment, or if such equipment shall have been repaired or altered or modified without seller's approval; provided, however, that the foregoing limitation of warranty insofar as it relates to repairs, alterations, or modifications, shall not be applicable to routine preventive and corrective maintenance which normally occur in the operation of the equipment.

"EXCEPT FOR THOSE WARRANTIES SPECIFICALLY CONTAINED HEREIN, SELLER DISCLAIMS ANY AND ALL WARRANTIES WITH RESPECT TO THE EQUIPMENT DELIVERED HEREUNDER, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE. THE SOLE LIABILITY OF SELLER ARISING OUT OF THE WARRANTY CONTAINED HEREIN SHALL BE EXCLUSIVELY LIMITED TO BREACH OF THOSE WARRANTIES. THE SOLE AND EXCLUSIVE REMEDY FOR BREACH OF THE WARRANTIES SET OUT ABOVE SHALL BE LIMITED TO THE REPAIR OR REPLACEMENT OF ANY DEFECTIVE ACCESSORY, PART OR MATERIAL WITH A SIMILAR ITEM FREE FROM DEFECT, AND THE CORRECTION OF ANY DEFECT IN WORKMANSHIP. IN NO EVENT SHALL SELLER BE LIABLE FOR ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES."

FIELD SERVICE

Purchaser agrees to underwrite the cost of any labor required for replacement; including time, travel, and living expenses of Ramsey Field Service Engineer at closest factory base.

RAMSEY TECHNOLOGY, INC.
501 90th Avenue N.W.
Minneapolis, MN 55433
Phone: (612) 783-2500
Fax: (612) 783-2525

CHAPTER 2.0 INSTALLATION

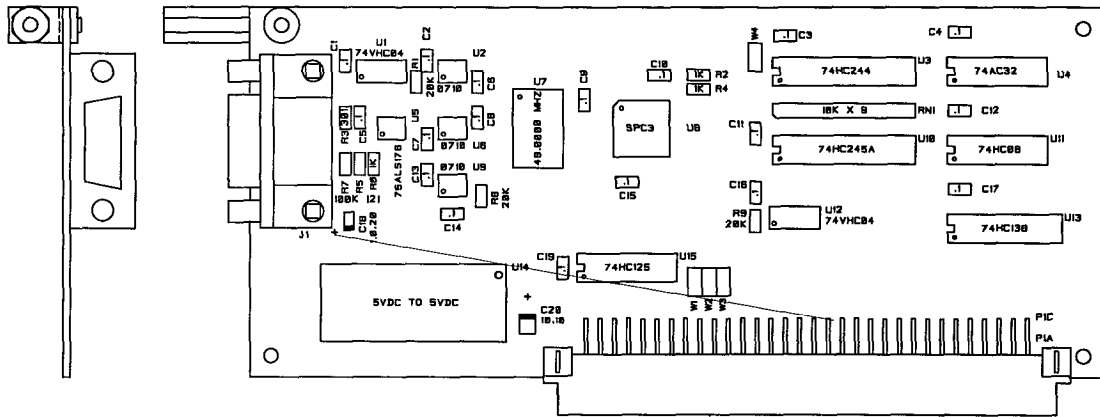
2.1 GENERAL

This chapter describes the installation procedure and hardware configuration for the PROFIBUS-DP interface option. If the PROFIBUS-DP option was installed at the factory, skip to Section 2.3.

2.2 PROFIBUS-DP INTERFACE BOARD

The PROFIBUS-DP interface board (Figure 2-1) contains the Siemens SPC3 PROFIBUS-DP controller ASIC. The SPC3 controller is an integrated circuit provided by Siemens that handles the interface between the Micro-Tech 2000 (slave) and the master.

No hardware configuration jumpers or switches are present on the PROFIBUS-DP board.



A01761

PROFIBUS-DP INTERFACE BOARD
FIGURE 2-1

2.2 CABLE TYPE AND LENGTH

2.2.1 Application

Different types of cables are available from Siemens for various applications when assembling PROFIBUS networks. Standard Siemens PROFIBUS LAN cable number 6XV1 830-0AH10 is available from Ramsey, part number 057415.

Consult the Siemens PROFIBUS network manual 6GK1 970-5CA10-0AA1 for additional information.

2.2.2 RS 485 Repeater Application

An RS 485 repeater amplifies data signals on LAN cables and connects bus segments. An RS 485 repeater is required in the following situations:

1. When there are more than 32 stations connected to the bus.
2. When electrically isolated bus segments are required, or
3. When the maximum cable length of a segment (standard LAN cable) is exceeded (see Table 2-1).

**TABLE 2-1
MAXIMUM CABLE LENGTH OF A SEGMENT (STANDARD LAN CABLE)**

Transmission Rate	Maximum Cable Length of a Segment (in m)
9.6 to 93.75 Kbps	1000
187.50 Kbps	800
500.00 Kbps	400
1.50 Mbps	200
3.0 to 12.00 Mbps	100

RULES

If you want to install the bus with RS 485 repeaters, the following rules apply:

1. A maximum of 9 RS 485 repeaters can be connected in series.
2. The maximum cable length between two nodes must not exceed the values in Table 2-2.

**TABLE 2-2
MAXIMUM CABLE LENGTH BETWEEN TWO NODES (STANDARD LAN CABLES)**

Transmission Rate	Maximum Cable Length Between Two Nodes (in m) With RS 485 Repeater
9.6 to 93.75 Kbps	10000
187.50 Kbps	8000
500.00 Kbps	4000
1.50 Mbps	2000
3.0 to 12.00 Mbps	1000

2.3 PROFIBUS-DP CONNECTOR

The bus connector should be rated for 12 Mbaud and contain a termination resistor that can be switched on or off. Siemens bus connector 6ES7 972-OBA20-OXAO is available from Ramsey, part number 057416.

2.3.1 Bus Connector Wiring

1. Refer to the information sheet supplied with the bus connector and Table 2-3.

**TABLE 2-3
BUS CONNECTOR**

Bus Connector 9 Pin Sub "D"	SIEMENS LAN CABLE
1 Chassis Ground	Braided Shield
2 ---	
3 RxD/TxD-P	Data Line B (Red)
4 Control-P (Receive/(not) Transmit)	
5 Data Ground	
6 Power (+5 VDC @ 100 mA max)	Supplied by Micro-Tech 2000
7 ---	
8 RxD/TxD-N	Data Line A (Green)
9 ---	

2. Connect the green and red cable wires to the screw terminal block located in the connector. The two pairs of terminals for signal wires A and B are identical.

The following scheme is recommended for a PROFIBUS LAN:

Terminal A: green wire

Terminal B: red wire

NOTE: The same wires (green or red) must always be connected to the same terminal A or B in all bus terminals and with all bus connections, and be uniform throughout the segment.

3. Insure the braided shield is making metal-to-metal contact with the connector guide.

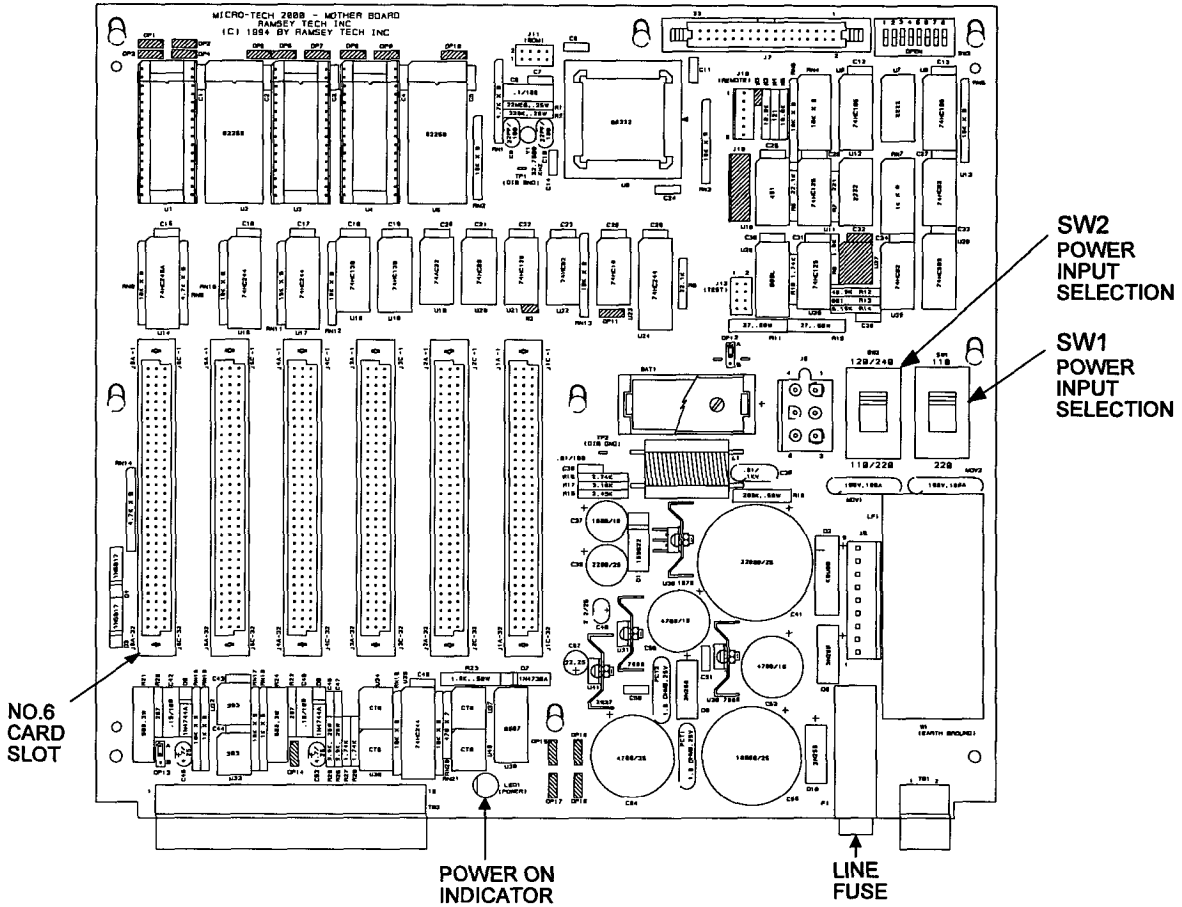
2.3.2 Termination Power

Up to 100 mA of isolated 5 VDC is available on J1, pin 6 to power bus termination or PROFIBUS optical link plugs.

2.4 PROFIBUS-DP BOARD INSTALLATION

To install the PROFIBUS-DP board, proceed as follows:

1. Open the Micro-Tech wall mount enclosure and turn off power at the mains, or remove panel mount enclosure from the panel and remove top cover, allowing access to the mother board.
2. Remove the hex head mounting screw from the connector end of the PROFIBUS-DP board (Figure 2-1).
3. Insert the PROFIBUS board in any available expansion slot on the mother board (see Figure 2-2).
4. Insert the hex head screw through the chassis and tighten.



A00926

MODEL 2000 MOTHER BOARD
FIGURE 2-2

5. Replace the panel mount top cover and reinstall the panel mount enclosure in the control panel.
6. Install the wired mating connector on the PROFIBUS board.
7. Turn power back on at the mains. The following screen appears when power is restored and remains on for 10 seconds.

```

-SLOT # N CHANGED
Acquire new
configuration?
YES    NO

```

The screen disappears after ten seconds. If the question is not answered, the Micro-Tech 2000 assumes the answer is NO. HW (hardware) alarm is on and cannot be reset. The above screen appears each time power is cycled if the question is not answered.

Answer YES because this is a hardware configuration change.

Set-up data for the newly installed PROFIBUS-DP interface board must now be entered. Proceed to Chapter 3.0, Operation.

CHAPTER 3.0 OPERATION

3.1 GENERAL

The Micro-Tech 2000 is a menu driven machine that allows the operator easy access to the PROFIBUS-DP set-up scroll in Main Menu 5.

3.2 MICRO-TECH 2000 CONFIGURATION

Main Menu 5 is dedicated to the Micro-Tech 2000 communication options: COMM A serial communications, PROFIBUS-DP and PRINT. The PRINT scroll is only visible if the COMM A board is installed.

3.2.1 Slave Address and Buffer Dimensions

The slave address and the buffer dimensions should be defined in the PROFIBUS scroll. Repeatedly press the MENU key until Main Menu 5 appears.

```
- MAIN MENU 5 -  
Press MENU for more  
  
COMMA PROFIB PRINT
```

3.2.2 PROFIB Scroll

1. Slave Address

Defines the address of the Micro-Tech 2000 in the PROFIBUS net. Press the soft key under PROFIB and scroll down.

```
- PROFIBUS SCROLL 1 -  
Address  
1  
ENTER
```

Password: SERVICE

Default: 1
Min: 1
Max: 126

Type in the address and press ENTER.

In the next two scrolls, the operator can define independently the read and write buffer dimensions. This feature is very useful since it allows reducing the address space to the minimum need.

The dimension should be entered in number of words. This number includes the header of the telegram taking four words (see Table 3-1), so the minimum is five words.

```
- PROFIBUS SCROLL 2 -  
Read buffer dim.  
48 words  
ENTER
```

Password: SERVICE

Default: 48
Min: 5
Max: 48

Type in the number of words and press ENTER. The following screen appears.

```
- PROFIBUS SCROLL 3 -  
Write buffer dim.  
48 words  
ENTER
```

Password: SERVICE

Default: 48
Min: 5
Max: 48

Type in the number of words and press ENTER. Press RUN to return to the RUN menu or MENU for more menus.

3.2.3 Baud Rate

The baud rate is automatically detected by the Micro-Tech 2000 and is displayed in Main Menu 3, TEST Scroll. Press the MENU key repeatedly until Main Menu 3 appears.

```
- MAIN MENU 3 -  
Press menu for more  
  
DIAG TEST
```

Press the TEST key and scroll down.

```
- TEST SCROLL 9 -  
Test communication B  
19.2 kb Wait Prm
```

The baud rate and status are dynamically displayed. The status can be:

WAIT PRM The Micro-Tech is waiting for the PARAMETER message.

WAIT CFG The Micro-Tech is waiting for the CONFIGURATION message.

DATA EXC The Micro-Tech is in the DATA EXCHANGE phase, configuration and parameterization have been completed successfully.

If the baud rate displayed value changes continuously, it means there are problems on the line or the CPU is off.

Press RUN to return to the RUN menu or MENU for more menus.

3.3 DATA TRANSFER

During the communication activity, the Micro-Tech 2000 will always act as slave, meaning it will respond to a request from a master device on the line, but will never attempt to send messages out.

The PROFIBUS interface allows a remote intelligent device to read and write data from and to the Micro-Tech 2000. Data is organized in registers, these registers are collected in several groups, some of them are read only while others are write only groups. Groups are identified by a number, numbers from 0 are reserved for read groups, numbers from 100 are for the write groups.

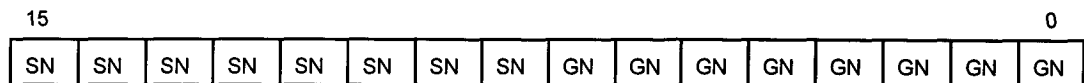
The master has the possibility to perform two types of operations: write data and read data. The first simply consists in sending to the Micro-Tech 2000 the values to write in the registers together with indications to identify what registers have to be written. In the second case, the procedure can be a little more complex since the reading operation can be preceded by a write operation to tell the Micro-Tech 2000 what registers should be read. Write and read operations are explained in detail below.

3.3.1 Write Operations

The write telegram is composed of two parts, header and data. The 'header' consists of 4 elements (see Table 3-1):

1. Group Identifier

The 'group identifier' identifies which group of data the master wants to write. In some models of the Micro-Tech line (MT2100, MT2102, MT2104), one Micro-Tech 2000 can handle two or more independent scales so the Micro-Tech 2000 keeps in memory more copies for each group. The 'group identifier' is also used to determine the scale. It is in two parts:



SN	Scale number:	0x00	scale 1
		0x01	scale 2
		0x02	scale 3
		0x03	scale 4

GN Group number

The MSB identifies the scale. In case the Micro-Tech 2000 is configured to work with a single scale, it should be set to 0. The LSB identifies the group, see Section 3.6.

For example,

Hex	0102	read group 2, scale 2
Hex	0065	write group 101, scale 1

2. First Register Number

Registers in the groups are numbered from 0 to 'n' and the master has the possibility to write only a portion of the registers composing the group. This parameter specifies the first register to write; 0 means the first register of the group.

3. Number of Registers to Write

This parameter specifies the number of registers to write. It can be set to 0 to indicate all the registers of the group. In case the dimension of the write buffer is less than the length in words of the group, putting 0 here is the same as putting the group dimension minus 4 which is the length of the header.

4. Stamp

In the PROFIBUS protocol, data is transmitted continuously, so the same telegram is sent by the master to the slave several times. To avoid the slave interpreting the same data continuously (it would create problems; for example, with the commands), the stamp is used. The slave interprets received data only if the stamp is different from the stamp received in the previous telegram. The master has only to change the stamp value when it creates a new telegram. The stamp of the last interpreted telegram is re-transmitted by the Micro-Tech 2000 in the read buffer.

The 'Data' section contains the data to write in the order MSB - LSB.

When the Micro-Tech 2000 processes the 'package', before writing the received data in the registers, it checks that all the parameters are correct (Example: Group ident. refers to an existing group). The data to write does not overlap specified maximum and minimum limits (each writeable register has its own maximum and minimum limit, see Table 3-4).

If it detects errors, the write operation fails and a DIAGNOSTICS message will be activated.

**TABLE 3-1
COMPOSITION OF THE WRITE PACKAGE**

elem. #	register	type	offset (words)	note
1	Group identifier	integer	0	
2	First register	integer	1	
3	Nr of registers	integer	2	
4	Stamp	integer	3	
5	Data #1		4	
...	
...	Data #n		..	

3.3.2 Read Operations

In the PROFIBUS protocol, the master continuously reads data from the slave. Since the Micro-Tech 2000 has a large number of registers that can be sent to the master, the readable registers are collected in groups. Therefore, the master should have the possibility to tell the Micro-Tech 2000 what group and what registers of the group it needs to read.

It does it by performing a write operation, in the mode explained above, setting as 'group identifier' the number of a read group. After the Micro-Tech 2000 receives and interprets this write request, it starts to fill the read buffer with the requested data.

NOTE: In a write operation, if the group identifier refers to a read group, only the header of the telegram is interpreted. The data is ignored.

The Micro-Tech 2000 will remember which registers have been required the last time and it will continue to update the read buffer with their actual values.

What the Master receives after a read operation is a buffer of data having the same structure of the write telegram (see Table 3-1). The first 4 words will contain the 'group identifier', 'first register', 'number of register' and 'stamp', then the 'data'.

After power on, the Micro-Tech 2000 starts to update the read buffer with the register of the read group 0 (scale 1). The master receives:

**TABLE 3-2
COMPOSITION OF THE READ BUFFER AFTER POWER ON**

Structure				Read Data	
elem.#	register	type	offset	Value	Note
1	Group identifier	integer	0	0	Ident for read group 0
2	First register	integer	1	0	From the first reg
3	Nr of registers	integer	2	nnn	Number of word of the group 0, it depends by the MT model.
4	Stamp	integer	3	0	
5	Data #1		4	xxx	Contents first reg.
...	xxx
nnn+5	Data #nn		..	xxx	Contents last reg

In the Micro-Tech 2000, the group 0 is the 'dynamic' group. It contains all the variables that change dynamically; for example, the actual weight or speed, the totalizers. Other groups contain set-up data; for example, the weight threshold set. Normally, the master will ask the group 0, only when necessary will it ask other groups.

3.4 TIMINGS

The Micro-Tech 2000 updates almost all its basic variables (e.g.; weight, rate, totals) every 100 milliseconds. With the same frequency the Micro-Tech 2000 updates the read buffer. If the master performs more communications in this period, it will receive the same data more times.

The write request is interpreted by the Micro-Tech 2000 in polling with a period of 100 milliseconds.

3.5. COMMUNICATION ERRORS

The alarm condition 'PROFIBUS comm error' is activated by the Micro-Tech 2000 in two conditions:

1. If the SPC3 controller installed on the 'PROFIBUS interface board' does not recognize successful data transfer within the watch dog control interval.
2. If the received data contains errors (value overlaps limits, register number does not exist, group number does not exist).

The alarm can be defined as SHUT DOWN and the shut down digital output can be used to set the system in a safety condition.

In the second case, the Micro-Tech 2000 also activates an EXPANDED DIAGNOSTIC request to the master.

3.5.1. Diagnostics Data

The master system has the possibility to detect a 'communication error' condition by checking the diagnostics. In case of an error, the Micro-Tech 2000 will activate the request for diagnostic acquisition, indicating the expanded device related diagnostics data is present.

When the master requires the diagnostics, it receives:

**TABLE 3-3
COMPOSITION OF THE DIAGNOSTICS DATA**

Byte	Bit position								Diagnostics data
	7	6	5	4	3	2	1	0	
0									StatStatus1
1									StatStatus2
2									StatStatus3
3									MasterAdd
4									IdentNumberHigh
5									IdentNumberLow
6	0	0	0	0	0	0	1	0	Ext Diag - Header
7	0	0	0	0	GE	RE	LE	CE	Ext Diag - Data

GE Group Error

Invalid group identifier, the group does not exist.

- | | | |
|----|-----------------------|---|
| RE | Register number Error | The number of requested registers is wrong. |
| LE | Limits Error | A write operation has been performed but the value to write overlaps the limits. |
| CE | Coherency Error | The master has tried to write or read partially a variable composed by more registers (e.g., only one word of a floating variable). |

The individual bits in Table 3-3 have the following meaning:

Bit 7: Diag.Master_Lock

The DP-Slave has been parameterized from another master. This bit is set by the DP-Master (class 1), if the address in octet 4 is different from 255 and different from the own address. The DP-Slave sets this bit to zero.

Bit 6: Diag.Prm_Fault

This bit is set by the DP-Slave if the last parameter frame was faulty (e.g., wrong length, wrong Ident_Number, invalid parameters).

Bit 5: Diag.Invalid_Slave_Response

This bit is set by the DP-Master as soon as receiving a not plausible response from an addressed DP-Slave. The DP-Slave sets this bit to zero.

Bit 4: Diag.Not_Supported

This bit is set by the DP-Slave as soon as a function is requested, which is not supported from this DP-Slave.

Bit 3: Diag.Ext_Diag

This bit is set by the DP-Slave. It indicates a diagnostic entry exists in the slave specific diagnostic area (Ext_Diag_Data) if the bit is set to one. If the bit is set to zero, a status message can exist in the slave specific diagnostic area (Ext_Diag_Data). The meaning of this status message depends on the application and will not be fixed in this standard.

Bit 2: Diag.Cfg_Fault

This bit is set by the DP-Slave as soon as the last received configuration data from the DP-Master are different from these which the DP-Slave has determined.

Bit 1: Diag.Station_Not_Ready

This bit is set by the DP-Slave if the DP-Slave is not yet ready for data transfer.

Bit 0: Diag.Station_Non_Existent

This bit is set by the DP-Master if the respective DP-Slave cannot be reached over the line. If this bit is set, the diagnostic bits contain the state of the last diagnostic message or the initial value. The DP-Slave sets this bit to zero.

- high limit The maximum acceptable value for the variable. Higher values are considered as errors.
- refresh t. The time between two updates of the variable in the Micro-Tech 2000's memory.
- format Can be: integer or float

**TABLE 3-4
VARIABLES AND THEIR CHARACTERISTICS**

Register	Type	Low Limit	High Limit	Refresh Time [mS]	Format
Commands					
Commands	WO	0	32767	100	Integer
Sim_Key	WO	-	-	-	Integer
Set/Reset Out#	WO	-	-	-	Integer
Recipe write					
RW_Rec	RW	1	_MaxRec_	-	Integer
RW_Line	RW	1	504	-	Integer
RW_Comp	RW	1	_MaxComp_	-	Integer
RW_Label	RW	0	65535	-	Integer
RW_Set	RW	0	scale_cap	-	Float
RW_Dev	RW	0	100.0 %	-	Float
Dynamic values and status					
Display	RO	-	-	200	Integer
Alarms	RO	-	-	100	Integer
Status	RO	-	-	100	Integer
IO_Image	RO	-	-	100	Integer
Running_Rec	RW	1	_MaxRec_	-	Integer
Run./Set_Cycle	RW	0	9999	-	Integer
Running_Comp	RO	-	-	-	Integer
Actual_Set	RO	-	-	-	Float
Gross_Weight	RO	-	-	100	Float
Net_Weight	RO	-	-	100	Float
Cal_Error	RO	-	-	-	Float

**TABLE 3-4
VARIABLES AND THEIR CHARACTERISTICS**

Register	Type	Low Limit	High Limit	Refresh Time [mS]	Format
Sets and thresholds					
Scale_Cap	RW	1	20000	-	Float
Set_of_Zero	RW	1	20000	-	Float
Low_Weight_Set	RW	0	105.0 %	-	Float
High_Weight_Set	RW	0	105.0 %	-	Float
Recipe and component totals					
Tot_Comp#	RW	1	100	1	Integer
Tot_Rec#	RW	1	40	1	Integer
Comp#_Tot	RO	-	-	-	Float
Comp#_LastTot	RO	-	-	-	Float
Rec#_Tot	RO	-	-	-	Float
Rec#_LastTot	RO	-	-	-	Float

MaxComp is the max number of components defined in the setup

MaxRec is the max number of recipes defined in the setup

Registers in the table are explained below and are sorted in alphabetical order.

Actual_Set

Set point of component actually in use if a batch is running or last used.

Alarms

In the alarms register, each bit represents the status of an alarm. If the alarm is active, the relevant bit will contain '1', otherwise it will contain '0'.

Alarms 1

15	14	NU	NU	NU	NU	09	08	07	NU	NU	NU	NU	02	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

A1.15 CLOCK FAIL

A1.14 CELL FAIL

A1.09 RAM FAIL

A1.08 ROM FAIL

A1.07 HIGH WEIGHT

A1.02 LOW WEIGHT

Alarms 2

NU	NU	13	12	11	10	NU	NU	NU	NU	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

A2.13 WARM START

A2.12 COLD START

A2.05 EXTERN ALAM 1

A2.04 EXTERN ALAM 2

A2.11 PWD DURING CALIB
 A2.10 CAL TIME ELAPSED

A2.03 EXTERN ALAM 3
 A2.02 HW CNF CHANGE SLOT 1
 A2.01 HW CNF CHANGE SLOT 2
 A2.00 HW CNF CHANGE SLOT 3

Alarms 3

15	14	13	12	11	10	09	08	NU	NU	NU	NU	03	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

A3.15 HW CNF CHANGE SLOT 4
 A3.14 HW CNF CHANGE SLOT 5
 A3.13 HW CNF CHANGE SLOT 6
 A3.12 BCD OVERFLOW
 A3.11 MATHEMATIC ERROR
 A3.10 PRINTER ERROR
 A3.09 COMMUNICATION ERROR

A3.08 NOT ZERO
 A3.03 STOP CYCLE

Alarms 4

NU	14	NU	NU	NU	NU	NU	NU	NU	NU	NU	04	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

A4.14 END CYCLE
 A4.09 ABORT CYCLE

A4.04 COMP. TIME OUT

Alarms 5

15	NU	NU	NU	NU	NU	09	NU	NU	NU	NU	NU	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

A5.15 DEVIATION
 A5.09 PROFIBUS COMM ERROR

Cal_Error

It is the error in percent resulting from a remote calibration. The master can evaluate it before accepting the result of a remote autozero or autospan.

Commands

Each bit of the commands register is specified as follows. In order to give a command, the Host must set the relevant bit to 1 and write (send) the register to the Micro-Tech 2000. The action will be performed if the write message is accepted.

Commands

15	14	13	12	11	10	09	08	NU	NU	NU	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

C1.15 AUTOSPAN RCAL
 C1.14 AUTOSPAN WTS
 C1.13 AUTOZERO
 C1.12 CONFIRM ZERO/SPAN
 C1.11 END CYCLES
 C1.10 ABORT BATCH
 C1.09 STOP BATCH
 C1.08 START BATCH

C1.02 RESET REC. TOTALS
 C1.01 RESET COMP. TOTALS
 C1.00 RESET ALARM

3.6.2 Calibration Commands

It is possible to start a calibration function from a remote location. At the end of the calibration (calibration flag in status registers should be tested to determine when cal function ends), it is possible to verify the calibration error ('cal_error' register) and eventually accept the new zero or span. Calibration commands are not executed if the Micro-Tech 2000 is batching.

Autospan RCAL	Start the autospan function with RCAL method. The RCAL is automatically connected by the Micro-Tech 2000 at the beginning of the function and disconnected at its end. AutoSpan has a built in delay of 0.5 seconds after START is pressed. In case a RCAL remote calibration is aborted, the calibration flags will turn off after this delay.
Autospan WTS	Start the autospan function with WTS method. An output of the Micro-Tech 2000 can be programmed to automatically load the test weights. This adds a 10 second delay at the beginning and at the end of the calibration function. In case a WTS remote calibration is aborted, the calibration flags will turn off after this time.
Autozero	Start the autozero function.
Confirm zero/span	At the end of the calibration function, this command tells the Micro-Tech 2000 to save the new zero or span. If this command is sent during a remote calibration, it aborts the running function.

3.6.3 Batch Commands

The batch can be controlled by serial.

Stop Batch	Temporarily stops a batch. Batch can be resumed by a start command.
Abort Batch	Abort a batch.
Start Batch	Starts a new batch.

3.6.4 Reset Commands

When the Micro-Tech 2000 receives a reset command, it executes the command for all the scales.

Reset comp totals	Reset the component totals
Reset rec totals	Reset the recipe totals
Reset Alarms	Reset any pending alarm

In order to give a command, the Host must set the relevant bit to 1 and write (send) the register to the Micro-Tech 2000.

- Comp#_LastTot** Amount of material the component # has batched the last time it has been activated. The value is cleared at the start of a new batch.
- Comp#_Tot** Amount of material the component # has batched on the whole all the times it has been activated. It is in TONNES if METRIC units has been selected, in LONG TONS if ENGLISH.
- Display** The ASCII image of the Micro-Tech 2000 display.
- Gross_Weight** The instantaneous gross weight in engineering units.
- High_Weight_Set** The set point for the alarm of high weight. Entered or downloaded by the operator.
- IO_Image** The Micro-Tech 2000 has physical inputs and outputs to which logical input and output functions are associated. As far as communication is concerned, only the status of physical inputs and outputs are transferred.

IO_image 1 - mother board inputs

NU	NU	NU	NU	11	10	09	08	NU	NU	NU	NU	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

- IO1.11 in 4 - mother board
 IO1.10 in 3 - mother board
 IO1.09 in 2 - mother board
 IO1.08 in 1 - mother board

IO_image 2 - 4in/16out boards inputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

- | | |
|-----------------------------|-----------------------------|
| IO2.15 in 4 board 4i/16o #1 | IO2.07 in 4 board 4i/16o #3 |
| IO2.14 in 3 board 4i/16o #1 | IO2.06 in 3 board 4i/16o #3 |
| IO2.13 in 2 board 4i/16o #1 | IO2.05 in 2 board 4i/16o #3 |
| IO2.12 in 1 board 4i/16o #1 | IO2.04 in 1 board 4i/16o #3 |
| IO2.11 in 4 board 4i/16o #2 | IO2.03 in 4 board 4i/16o #4 |
| IO2.10 in 3 board 4i/16o #2 | IO2.02 in 3 board 4i/16o #4 |
| IO2.09 in 2 board 4i/16o #2 | IO2.01 in 2 board 4i/16o #4 |
| IO2.08 in 1 board 4i/16o #2 | IO2.00 in 1 board 4i/16o #4 |

IO_image 3 - 16in/4out boards #1 inputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 4 - 16in/4out boards #2 inputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 5 - 16in/4out boards #3 inputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 6 - 16in/4out boards #4 inputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IOx.15 in 8 board 16i/4o #x	IOx.07 in 16 board 16i/4o #x
IOx.14 in 7 board 16i/4o #x	IOx.06 in 15 board 16i/4o #x
IOx.13 in 6 board 16i/4o #x	IOx.05 in 14 board 16i/4o #x
IOx.12 in 5 board 16i/4o #x	IOx.04 in 13 board 16i/4o #x
IOx.11 in 4 board 16i/4o #x	IOx.03 in 12 board 16i/4o #x
IOx.10 in 3 board 16i/4o #x	IOx.02 in 11 board 16i/4o #x
IOx.09 in 2 board 16i/4o #x	IOx.01 in 10 board 16i/4o #x
IOx.08 in 1 board 16i/4o #x	IOx.00 in 09 board 16i/4o #x

IO_image 7 - mother board outputs

NU	NU	NU	NU	NU	10	09	08	NU	NU	NU	NU	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO7.10 out 3 - mother board
 IO7.09 out 2 - mother board
 IO7.08 out 1 - mother board

IO_image 8 - 4out/16in boards outputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO8.15 out 4 board 4o/16i #1	IO8.07 out 4 board 4o/16i #3
IO8.14 out 3 board 4o/16i #1	IO8.06 out 3 board 4o/16i #3
IO8.13 out 2 board 4o/16i #1	IO8.05 out 2 board 4o/16i #3
IO8.12 out 1 board 4o/16i #1	IO8.04 out 1 board 4o/16i #3
IO8.11 out 4 board 4o/16i #2	IO8.03 out 4 board 4o/16i #4
IO8.10 out 3 board 4o/16i #2	IO8.02 out 3 board 4o/16i #4
IO8.09 out 2 board 4o/16i #2	IO8.01 out 2 board 4o/16i #4
IO8.08 out 1 board 4o/16i #2	IO8.00 out 1 board 4o/16i #4

IO_image 9 - 16out/4in boards #1 outputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 10 - 16out/4in boards #2 outputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 11 - 16out/4in boards #3 outputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IO_image 12 - 16out/4in boards #4 outputs

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IOx.15 out08 board 16o/4i #x	IOx.07 out16 board 16o/4i #x
IOx.14 out07 board 16o/4i #x	IOx.06 out15 board 16o/4i #x
IOx.13 out06 board 16o/4i #x	IOx.05 out14 board 16o/4i #x
IOx.12 out05 board 16o/4i #x	IOx.04 out13 board 16o/4i #x

IOx.11 out04 board 16o/4i #x	IOx.03 out12 board 16o/4i #x
IOx.10 out03 board 16o/4i #x	IOx.02 out11 board 16o/4i #x
IOx.09 out02 board 16o/4i #x	IOx.01 out10 board 16o/4i #x
IOx.08 out01 board 16o/4i #x	IOx.00 out09 board 16o/4i #x

Low_Weight_Set The set point for the low weight alarm. Entered or downloaded by the operator.

Net_weight The instantaneous net weight in engineering units.

Rec#_LastTot Amount of material the recipe # has totaled the last time it has been executed. The value is cleared at the start of a new batch with the same recipe.

Rec#_Tot Amount of material the recipe # has totaled on the whole all the times it has been executed. It is in TONNES if METRIC units has been selected, in LONG TONS if ENGLISH.

Running_Rec It is the recipe selected to run.

Run/Set_Cycles When read, it is the number of cycles executed from the last start. When write, it is the number of cycles to execute.

Running_Comp It is the component active at the moment or the last one activated.

RW_Comp Should be used to write or read a recipe into or from the Micro-Tech 2000 memory. It is the component number of the selected recipe (RW_Rec) at the selected line (RW_Line).

RW_Dev Should be used to write or read a recipe into or from the Micro-Tech 2000 memory. It is the maximum deviation of the selected recipe (RW_Rec) at the selected line (RW_Line).

RW_Label Should be used to write or read a recipe into or from the Micro-Tech 2000 memory. It is the label of the selected recipe (RW_Rec) at the selected line (RW_Line).

RW_Line Should be used to write or read a recipe into or from the Micro-Tech 2000 memory. Defines which line of the recipe has to be read or write.

RW_Rec Should be used to write or read a recipe into or from the Micro-Tech 2000 memory. Defines which recipe has to be read or write.

RW_Set It is the set point of the selected recipe (RW_Rec) at the selected line (RW_Line).

Scale_Cap The scale capacity. Entered by the operator at first start up. It is the reference value for the high and low weight set expressed in percent. It can only be read.

Set_of_Zero The set of zero for the discharge and the start recipe.

Set/Reset out# It is a variable that allows the master to directly handle the spare digital outputs of the Micro-Tech 2000. There are 5 variables in the write group so the master has the possibility to set or reset 5 maximum outputs simultaneously.

To set or reset an output, it transmits the number of the physical output with the most significant bit set to 0 to reset and 1 to set. If the content is 0, the command is ignored. The Micro-Tech 2104 does not control the output the master is setting/resetting if it is a spare. If the master tries to change the state of an output that is used by the Micro-Tech 2000, a conflict could happen.

Sim_Key The master has the possibility to send a key code to the Micro-Tech 2000. Interpreted key codes are:

**TABLE 3-5
KEY CODES**

Key	Key Code	Key	Key Code
UP ARROW	0048 H	EIGHT	0038 H
DOWN ARROW	0050 H	NINE	0039 H
DEC. POINT	002E H	MENU	004D H
CLEAR	0053 H	RUN	0052 H
ZERO	0030 H	F1	003F H
ONE	0031 H	F2	0040 H
TWO	0032 H	F3	0041 H
THREE	0033 H	AUTO/MAN	003B H
FOUR	0034 H	LOC/REM	003C H
FIVE	0035 H	START	003D H
SIX	0036 H	STOP	003E H
SEVEN	0037 H		

Status The actual status of the Micro-Tech 2000 is resumed in four words, each bit has its own meanings. When the bit is 1, the associated status is true.

Status 1 - Generals

15	14	13	12	11	10	09	08	07	06	05	04	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S1.15 Cumulative shut down	S1.07 Batch end cycle
S1.14 Cumulative Alarms	S1.06 Batch abort
S1.13 Calibration running	S1.05 Batch stop
S1.12 Auto	S1.04 Batch run
S1.10 High Load	
S1.09 Low load	
S1.08 Ready	

Status 2 - Batch Run and Batch Stop

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S2.15-S2.08 Batch run flags	S2.07-S2.00 Batch stop flags
-----------------------------	------------------------------

Status 3 - Batch Abort

15	14	13	12	11	10	09	08	NU	NU	NU	NU	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

S3.15-S3.08 Batch ABORT Flags

Status 4 - For further use

NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU	NU
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

The Batch Run Flags give the actual phase of the running batch.

Batch Run	
<u>Value</u>	<u>Phase</u>
0	Not running
1	Start batch
2	Charge batch - wait charge interlock
3	Charge batch - wait delay at the start
4	Charge batch - charging at high rate
5	Charge batch - charging at low rate
6	Charge batch - charge end, wait stabilization time
7	Refill - wait charge interlock
8	Refill - wait delay at the start
9	Refill - refilling
10	Refill - refill end, wait stabilization time
11	AutoRefill - wait charge interlock
12	AutoRefill - wait delay at the start

Batch Run	
<u>Value</u>	<u>Phase</u>
13	AutoRefill - refilling
14	AutoRefill - refill end, wait stabilization time
15	Discharge batch - wait discharge interlock
16	Discharge batch - wait delay at the start
17	Discharge batch - discharging at high rate
18	Discharge batch - discharging at low rate
19	Discharge batch - discharge end, wait stab.time
20	Discharge - wait discharge interlock
21	Discharge - wait delay at the start
22	Discharge - discharging
23	Discharge - discharge end, wait stab.time
24	Time - wait interlock
25	Time - wait start delay
26	Time - batching
27	Time - end charge
28	Recipe line end
29	Next recipe line

The Batch Stop Flags give indication about the cause of a batch stop condition.

Batch Stop	
<u>Value</u>	<u>Cause of the stop</u>
0	Not stop
1	Stop command
2	Discharge device open
3	Charge interlock
4	Discharge interlock
5	Shut down active
6	Power down

The Batch Abort Flags give indication about the cause of a batch abort condition.

Batch Abort	
<u>Value</u>	<u>Cause of the abort</u>
0	Not abort
1	Abort command
2	Bad recipe
3	Bad external cycles set
4	Bad external set point set
5	Manual

Tot_Comp# Define the first component for the access to the component total registers. See Comp#_Tot and Comp#_LastTot.

Tot_Rec# Define the first recipe for the access to the recipe total registers. See Rec#_Tot and Rec#_LastTot

3.7 READ DATA GROUPS COMPOSITION

3.7.1 Read GROUP 0 - Dynamic Variables, Status and Sets

Read Group 0 is the default group the Micro-Tech 2000 sends to the master for a read request if the master does not change request (Table 3-6).

It contains dynamic values (weight) and information about the status of the Micro-Tech 2000.

**TABLE 3-6
READ GROUP 0 [VAT1]**

elem. #	register	type	offset (words)	Note
5	Status (1)	integer	4	
6	Status (2)	integer	5	
7	Status (3)	integer	6	
8	Status (4)	integer	7	
9	Alarms (1)	integer	8	
10	Alarms (2)	integer	9	
11	Alarms (3)	integer	10	
12	Alarms (4)	integer	11	
13	Alarms (5)	integer	12	
14	IO_Image(1)	integer	13	

**TABLE 3-6
READ GROUP 0 [VAT1]**

elem. #	register	type	offset (words)	Note
15	IO_Image(2)	integer	14	
16	IO_Image(3)	integer	15	
17	IO_Image(4)	integer	16	
18	IO_Image(5)	integer	17	
19	IO_Image(6)	integer	18	
20	IO_Image(7)	integer	19	
21	IO_Image(8)	integer	20	
22	IO_Image(9)	integer	21	
23	IO_Image(10)	integer	22	
24	IO_Image(11)	integer	23	
25	IO_Image(12)	integer	24	
26	Running_Rec	integer	25	
27	Running_Cycle	integer	26	
28	Running_Comp	integer	27	
29	Actual_Set	float	28	
30	Net_Weight	float	30	
31	Gross_Weight	float	32	
32	Cal_Error	float	34	

3.7.2 Read GROUP 1 - Sets and Thresholds

It contains the definition of the thresholds and the set of the reference values (Table 3-7).

**TABLE 3-7
READ GROUP 1 [VAT2]**

elem. #	register	type	offset (words)	Note
5	Scale_Cap	float	4	
6	Set_of_Zero	float	6	
7	High_Weigh_Set	float	8	

**TABLE 3-7
READ GROUP 1 [VAT2]**

elem. #	register	type	offset (words)	Note
8	Low_Weight_Set	float	10	

3.7.3 Read GROUP 2 - Display

It contains the image of the Micro-Tech 2000 display (Table 3-8).

**TABLE 3-8
READ GROUP 2 [VAT3]**

elem. #	register	type	offset (words)	Note
5	Display(1)	integer	4	
6	Display(2)	integer	5	
7	Display(3)	integer	6	
8	Display(4)	integer	7	
9	Display(5)	integer	8	
10	Display(6)	integer	9	
11	Display(7)	integer	10	
12	Display(8)	integer	11	
13	Display(9)	integer	12	
14	Display(10)	integer	13	
15	Display(11)	integer	14	
16	Display(12)	integer	15	
17	Display(13)	integer	16	
18	Display(14)	integer	17	
19	Display(15)	integer	18	
20	Display(16)	integer	19	
21	Display(17)	integer	20	
22	Display(18)	integer	21	
23	Display(19)	integer	22	
24	Display(20)	integer	23	
25	Display(21)	integer	24	

**TABLE 3-8
READ GROUP 2 [VAT3]**

elem. #	register	type	offset (words)	Note
26	Display(22)	integer	25	
27	Display(23)	integer	26	
28	Display(24)	integer	27	
29	Display(25)	integer	28	
30	Display(26)	integer	29	
31	Display(27)	integer	30	
32	Display(28)	integer	31	
33	Display(29)	integer	32	
34	Display(30)	integer	33	
35	Display(31)	integer	34	
36	Display(32)	integer	35	
37	Display(33)	integer	36	
38	Display(34)	integer	37	
39	Display(35)	integer	38	
40	Display(36)	integer	39	
41	Display(37)	integer	40	
42	Display(38)	integer	41	
43	Display(39)	integer	42	
44	Display(40)	integer	43	

3.7.4 Read GROUP 3 - Reading of the Component Total Registers

**TABLE 3-9
READ GROUP 3 [VAT4]**

elem. #	register	type	offset (words)	Note
5	First_Component (#)	integer	4	It is the component at which refers the first total register (Comp#_Tot).

**TABLE 3-9
READ GROUP 3 [VAT4]**

elem. #	register	type	offset (words)	Note
6	Comp#_Tot	float	5	
7	Comp(#+1)_Tot	float	7	
...	
16	Comp(#+10)_Tot	float	25	

3.7.5 Read GROUP 4 - Reading of the Component Last Total Registers

**TABLE 3-10
READ GROUP 4 [VAT4]**

elem. #	register	type	offset (words)	Note
5	First_Component (#)	integer	4	It is the component at which refers the first total register (Comp#_LastTot).
6	Comp#_LastTot	float	5	
7	Comp(#+1)_LastTot	float	7	
...	
16	Comp(#+10)_LastTot	float	25	

3.7.6 Read GROUP 5 - Reading of the Recipe Total Registers

**TABLE 3-11
READ GROUP 5 [VAT4]**

elem. #	register	type	offset (words)	Note
5	First_Component (#)	integer	4	It is the recipe at which refers the first total register (Rec#_Tot).
6	Rec#_Tot	float	5	
7	Rec(#+1)_Tot	float	7	
...	
25	Rec(#+10)_Tot	float	25	

3.7.7 GROUP 6 - Reading of the Recipe Last Total Registers

**TABLE 3-12
READ GROUP 6 [VAT4]**

elem. #	register	type	offset (words)	Note
5	First_Component (#)	integer	4	It is the recipe at which refers the first total register (Rec#_LastTot).
6	Rec#_LastTot	float	5	
7	Rec(#+1)_LastTot	float	7	
...	
16	Rec(#+10)_LastTot	float	25	

3.8 WRITE DATA GROUPS COMPOSITION

3.8.1 Write GROUP 100 - Commands

It gives the possibility to send commands and set/reset outputs (Table 3-13).

**TABLE 3-13
WRITE GROUP 100 [VAT5]**

elem. #	register	type	offset (words)	note
5	Commands	integer	4	
6	Set/Res Out#	integer	5	
7	Set/Res Out#	integer	6	
8	Set/Res Out#	integer	7	
9	Set/Res Out#	integer	8	
10	Set/Res Out#	integer	9	
11	Sim_Key	integer	10	

3.8.2 Write GROUP 101 - Dynamic Data and Sets

**TABLE 3-14
WRITE GROUP 101 [VAT6]**

elem. #	register	type	offset (words)	note
5	Rec#_Tot	integer	4	
6	Comp#_Tot	integer	5	
7	Running_Rec	integer	6	
8	Set_Cycle	integer	7	
9	Set_of_Zero	float	8	
10	High_Weight_Set	float	10	
11	Low_Weight_Set	float	12	

3.8.3 Write GROUP 102 - Writing a Recipe Line

**TABLE 3-15
WRITE GROUP 102 [VAT7]**

elem. #	register	type	offset (words)	note
5	RW_Rec	integer	4	
6	RW_Line	integer	5	
7	RW_Comp	integer	6	
8	RW_Label	integer	7	
9	RW_Set	float	8	
10	RW_Dev	float	10	

APPENDIX A/1 FLOATING POINT NOTATION

Micro-Tech 2000 stores floating point data types using the IEEE single precision format. The format contains a sign bit, an exponent field and a fraction field or mantissa.

The represented value

The value of the number being represented is equal to the exponent multiplied by the fractional part with the sign specified by the bit sign field:

$$(-1)^{\text{sign}} * (1.0 + \text{fraction}) * 2^{(\text{exp} - \text{bias})}$$

For detailed information, refer to the *ANSI IEEE Standard for Binary Floating Point Arithmetic*.

Sign Bit

The sign of the number being represented is stored in the sign bit. If the number is positive, the sign bit contains the value 0. If it is negative, it contains the value 1. The sign bit is stored in the most significant bit of a floating point value.

Exponent Field

Using an exponent increases the range of representable numbers. The exponent field of the number contains a 'biased' form of the exponent. A bias is subtracted from the exponent field, letting the actual exponent represent both positive and negative exponents. The value of this bias is hexadecimal 7F; therefore, the effective exponent (Rexp) of the number can be obtained as:

$$\text{Rexp} = \text{Exp} - 7F$$

NOTE: If both the exponent field and the fraction field are equal to zero, the number being represented is zero.

Fraction field (Mantissa)

IEEE floating point format stores the fractional part of a number in a "normalized" form. It assumes that all nonzero numbers are of the following form:

$$1.\text{xxxxxxxx (binary)}$$

The character x represent either 0 or 1 (binary).

Since all floating point numbers begin with 1, the 1 becomes the implicit normalized bit. It is the most significant bit of the fraction and is not stored in memory. The binary point is located immediately to the right of the normalized bit. All bits after the binary point represent values less than 1 (binary).

Example:

The number 1.625 (dec) can be represented as:

$$1.101 \text{ (bin)}$$

This number can be represented in decimal as:

$$2^0 + 2^{-1} + 2^{-3} \text{ (dec) Or } 1 + 0.5 + 0.125$$

Examples

Some examples have been added to help in the understanding of the rules explained above.

Ex 1.

The number 100.0 is represented as: 42 c8 00 00 (hex) or:

	0	1 0 0 0 0 1 0 1	1 0 0 1 0	
Bit	31	30	23 22	16
				0

The sign is positive (bit 31 is 0).

The Exponent Field is : 1 0 0 0 0 1 0 1 (bin), 85 (hex). The exp is : $exp = 85(hex) - bias (7F hex) = 6$. Thus, the first implicit bit to 1 has weight 6.

At this point the mantissa can be represented in this way, with the weight of each bit.

		2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	
		1	1	0	0	1	0	0	0	0	0	0
Bit	Imp												
		22											0

Now it is enough to sum the weight of the bits at 1.

Value = $2^6(imp) + 2^5 + 2^2 = 100.0$

Ex 2.

Value = 0.15, binary representation is:

	0	0 1 1 1 1 1 0 0	0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 1 0	
Bit	31	30	23 22	0

Sign = 0 positive

Exp = 7C hex - 7F hex = -3

Mantissa =

		2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	
		1	0	0	1	1	0	0	1	1	0
Bit	Imp										
		22									0

Value = $2^{-3} + 2^{-6} + 2^{-7} + 2^{-10} + 2^{-11} + \dots = 0.15$ (approx.)

Ex 3.

Value = -5.5, binary representation is:

	1	1 0 0 0 0 0 0 1	0 1 1 0	
Bit	31	30	23 22	0

Sign = 1 negative

Exp = 81hex - 7F hex = +2

Mantissa =

2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}
1	0	1	1	0	0	0	0	0	0	0	0	0	0
—	—	—	—	—	—	—	—	—	—	—	—	—	—
Bit	Imp	22											0

Value = 2^2 (imp.) + 2^0 + 2^{-1} = 5.5 (negative)

**APPENDIX A/2
MICRO-TECH 2000 CONFIGURATION
THE GSD FILE**

PROFIBUS devices have different performances characteristics. Features differ in regards to available functionality or possible bus parameters, such as baud rate and time monitoring. These parameters vary individually for each device type and vendor. For this reason, the PROFIBUS Organization has standardized a database file, it contains all the characteristics of the PROFIBUS device and must be delivered by the vendor together with the device. This file has a name that depends on the vendor and an extension that is normally GSD, but can be different depending on the country of origin of the device. The third letter of the extension defines the country:

- D - Default, is valid for each country
- I - Italian
- G - German
- E - English
- F - French
- P - Portuguese
- S - Spanish

Refers to the manual of the device you are using as Master of your PROFIBUS network to get information about how to use the GSD file.

These are the main PROFIBUS characteristics of the Ramsey Micro-Tech 2102's.

Vendor	Ramsey
Model	Micro-Tech 2102
Ident number	0x2102
Protocol	Standard DP
Baud rate supported	from 9.6 kb to 12 Mb
Freeze Mode supp	NO
Sync Mode supp	NO
Auto Baud supp	YES
Set Slave Add_supp	NO
Station Type	Compact
Prm Data Length	7
Diag Data Length	8
Config Data Length	9
Input data length	48 word, coherency over word
Output data length	48 word, coherency over word

To change I/O data length

The Micro-Tech 2000 gives the possibility to define the write and read buffer dimensions. Default values are 48 words for both. If the operator changes them, he should also modify the GSD file so the configuration in the Micro-Tech 2000 setup and in the file corresponds.

To make it is easy, the GSD file should be edited with a normal text editor and the value at the directive 'module =' at the bottom of the file should be changed. It normally appears as follows:

Module = "MT2000" 0xC0,0x6F,0x6F

The first value (0xC0) should be left unchanged, the second value specifies the dimension of the write buffer, the third of the read buffer.

The value to set should be computed as follows:

$$\text{Read_Buffer_Dim} = 63 + \text{RBD}$$

$$\text{Write_Buffer_Dim} = 63 + \text{WBD}$$

where:

RBD is the read buffer dimension defined in the Micro-Tech 2000 setup.

WBD is the write buffer dimension defined in the Micro-Tech 2000 setup.

The value should be then converted and written in hexadecimal notation.

The meaning of the default values 0x6F is:

$$6F \text{ hex} = 111 \text{ dec} = 63 + 48$$

48 is the default dimension of the buffers.

Instructions are also contained in the GSD file.

The following is the contents of the Ramsey Micro-Tech 2102 GSD file.

R_MT2102.GSD :

```
#Profibus_DP
GSD_Revision          = 1
;
Vendor_Name           = "Ramsey"
Model_Name            = "MT2102"
Revision              = "Rev.A"
Ident_Number          = 0x2102
Protocol_Ident        = 0
Station_Type          = 0
FMS_supp              = 0
Hardware_Release      = "Rev.A"
Software_Release      = "45.GSD.01"
;
9.6_supp              = 1
19.2_supp              = 1
93.75_supp            = 1
187.5_supp            = 1
500_supp              = 1
1.5M_supp             = 1
3M_supp               = 1
6M_supp               = 1
12M_supp              = 1
;
MaxTcdr_9.6           = 60
MaxTcdr_19.2          = 60
MaxTcdr_93.75         = 60
MaxTcdr_187.5         = 60
MaxTcdr_500           = 100
MaxTcdr_1.5M          = 150
MaxTcdr_3M            = 250
```

```

MaxTsd_r_6M           = 450
MaxTsd_r_12M          = 800
;
Redundancy            = 0
Repeater_Ctrl_Sig     = 0
24V_Pins              = 0
Implementation_Type   = "ifak"
Bitmap_Device         = "dev.dib"
Bitmap_Diag           = "diag.dib"
Bitmap_SF             = "sf.dib"
;
Freeze_Mode_supp      = 0
Sync_Mode_supp        = 0
Auto_Baud_supp        = 1
Set_Slave_Add_supp    = 0
Min_Slave_Intervall   = 120
;
Modular_Station       = 0
Modul_Offset          = 1
Max_User_Prm_Data_Len = 7
;
Fail_Safe             = 0
Slave_Family          = 5
Max_Diag_Data_Len     = 8
;
Unit_Diag_Bit(0)      = "Coherency Error"
Unit_Diag_Bit(1)      = "Limits Error"
Unit_Diag_Bit(2)      = "Register Number Error"
Unit_Diag_Bit(3)      = "Group ID Error"
;

```

```
;;
```

```
-----
```

```
; Buffers dimension
```

```
; Standard buffer dimension is :    48 words read buffer (0x6F)
;                                  48 words write buffer (0x6F)
```

```
; Dimensions can be changed by modifying the last two values in the
; 'Module' definition below.
```

```
; The value should be obtained as follows:
```

$$63 + WB_WN \quad \text{or} \quad 63 + RB_WN$$

```
; where WB_WN is the wished words number for the write buffer, RB_WN is
; the wished words number for the read buffer.
```

```
; The value should be converted in hexadecimal format. Default value
; hex 6F corresponds to:
```

$$63 + 48 \quad \rightarrow \quad 111 \text{ dec} \quad \rightarrow \quad 6F \text{ hex}$$

