Khepera®

RADIO TURRET USER MANUAL

Version 1.0

K-Team S.A.

Lausanne, 16 March 1999

Documentation author

K-Team
Ch. de Vuasset, CP111
1028 Préverenges
Switzerland

info@k-team.com
http://www.k-team.com

Trademark Acknowledgements

NOTICE:

- The contents of this manual are subject to change without notice.
- All efforts have been made to ensure the accuracy of the content of this manual. However, should any error be detected. please inform K-Team.
- The above notwithstanding, K-Team can assume no responsibility for any error in this manual.

TABLE OF CONTENT

1 INTRODUCTION

The radio turret is an intelligent compact radio modem adapted to the Khepera bus. This turret has its own local processor for management of the whole communication procedure, which includes the data encoding, transmission and reception, error detection and correction as well as the support of the protocol with Khepera using the local multiprocessor network. The radio turrets make it possible to communicate with other Khepera equipped with radio turrets as well as with a radio base station connected to a host computer.

1.1 How to use this manual

This manual is organised into six chapters and an appendix. To learn how to make the best use of your radio turret you are urged to read all of chapters 2 through 5. You need to read chapter 6 if you use the turret as a slave (not the default communication channel) and you use the software LabVIEW®. The appendix can be referred to as necessary.

Chapter 1	gives you a general introduction.
Chapter 2	describes some important warnings.
Chapter 3	explains the contents of the package.
Chapter 4	explains the functionality of the radio turret.
Chapter 5	explains how to connect the radio turret to the robot.
Chapter 6	is addressed to users of LabVIEW®. It shows simple virtual instruments (VI) to control the radio turret functionality.
Appendix A	details the commands of the communication protocol.

2 SAFETY PRECAUTIONS

Don't plug or unplug any connector or turret when the robot is powered (by batteries or external power supply). All connections and turret insertions must be made when the robot and the interface are switched OFF. Otherwise damages can occur.

Switch OFF the robot if you will not use it for more than a day. Please disconnect the power supply removing it from the wall socket.

If you have any question or problem concerning this turret, please contact your Khepera dealer.

3 UNPACKING AND INSPECTION

Please check that you have a complete package. You should find:

- Documentation.
- The radio turret equipped with an antenna.
- Disks with the software modules for LabVIEW® on SUN®, Macintosh® and PC. Please note that LabVIEW® itself is NOT included in the package.

4 THE RADIO TURRET

4.1 Overview

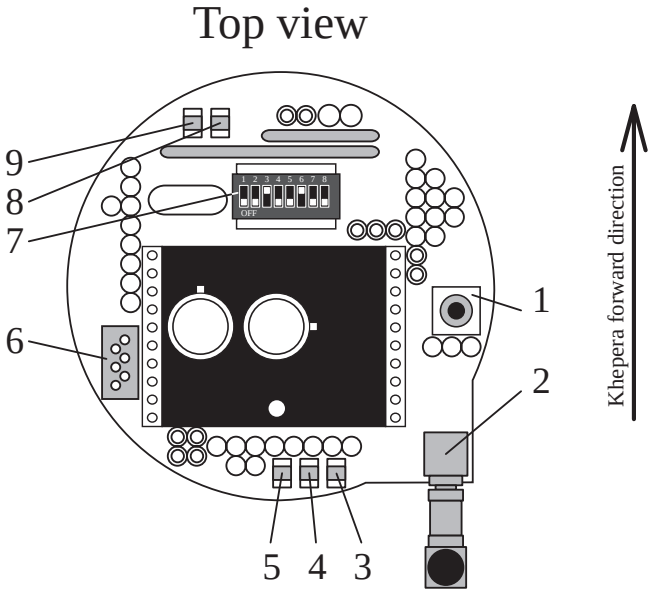

Figure 1: Overview of the turret layout.

Make an external inspection of the turret. Note the location of the following parts:

1. Reset button (function identical to the Khepera one)
2. Connector with antenna (antenna can be removed and turned)
3. Rx LED. When ON, the turret is ready to receive data.
4. Tx LED. When ON, the turret is transmitting data.
5. Carrier detect. When ON, the turret is detecting an active radio channel.
6. Serial line (S) connector.
7. Radio turret running mode and ID selector.
8. Repeat data LED. When ON, indicates that data has been lost and is repeated.
9. Lost data LED. When ON, indicates that data has been repeated 10 times without success and is considered as lost.

4.2 Introduction to the Khepera communication structure

This section explains the Khepera BIOS communication structure.This structure is available only with a Khepera equipped with a BIOS of revision 5 or later.

The SerCom protocol, which implements the standard RS232 protocol used to control Khepera remotely, is based on the COM (communication) module of the BIOS. The COM module has a choice between several physical communications channels. On Khepera, the two main communication channels are the RS232, managed by the SER module, and the local turrets network, managed by the MSG module. The SER module includes all calls to manage the RS232 hardware. The MSG module includes all calls to manage the hardware devoted to the communication with the extension turrets processors. The choice between SER and MSG channels is made by the COM module at boot, and depends on the turrets presents on the Khepera and their configuration. If there is an extension turret configured as communication channel, this choice is set as main communication channel. If no turrets are found, the SER module is used. On the radio turret, the running mode switch 6 allows selection of the turret as main communication channel or as simple extension turret.

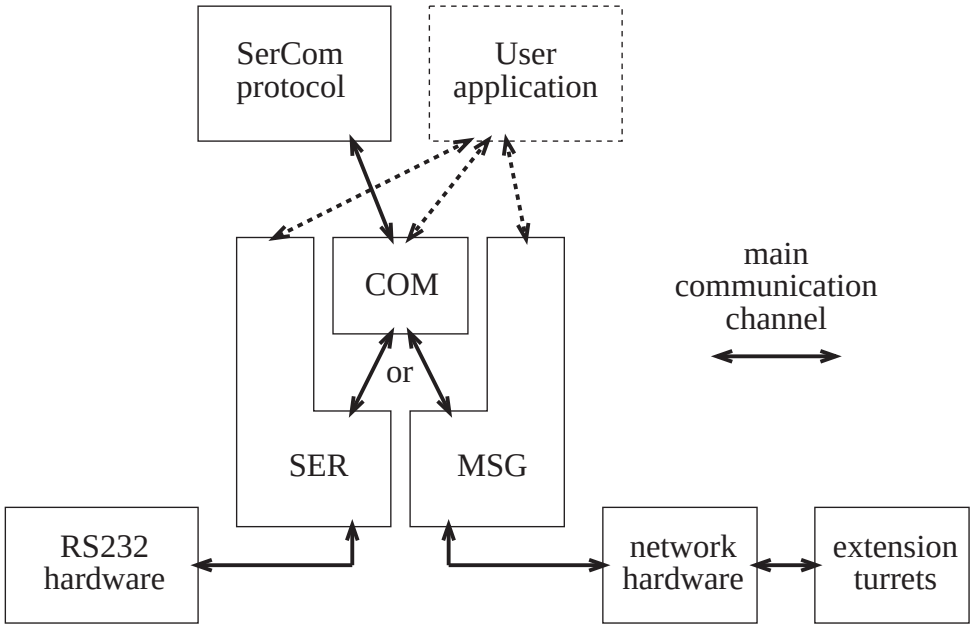

Figure 2: Communication channels in the Khepera BIOS structure

If the radio turret is set as main communication channel, all messages of the SerCom protocol are sent and received on the radio link to and from the turret with ID 0 (radio base). In this configuration, the serial link is not used by the SerCom protocol. The SER module can be used in this configuration by user applications by accessing it directly.

If the radio turret is set as simple extension turret, the SerCom protocol is available on the serial line. The radio turret can be accessed as a standard turret with its own command of emission and reception of data.

4.3 Radio network and turret ID

The radio network is composed of a maximum of 31 Khepera equipped with radio turrets and by one radio base station connected to the host computer, as illustrated in figure 3.

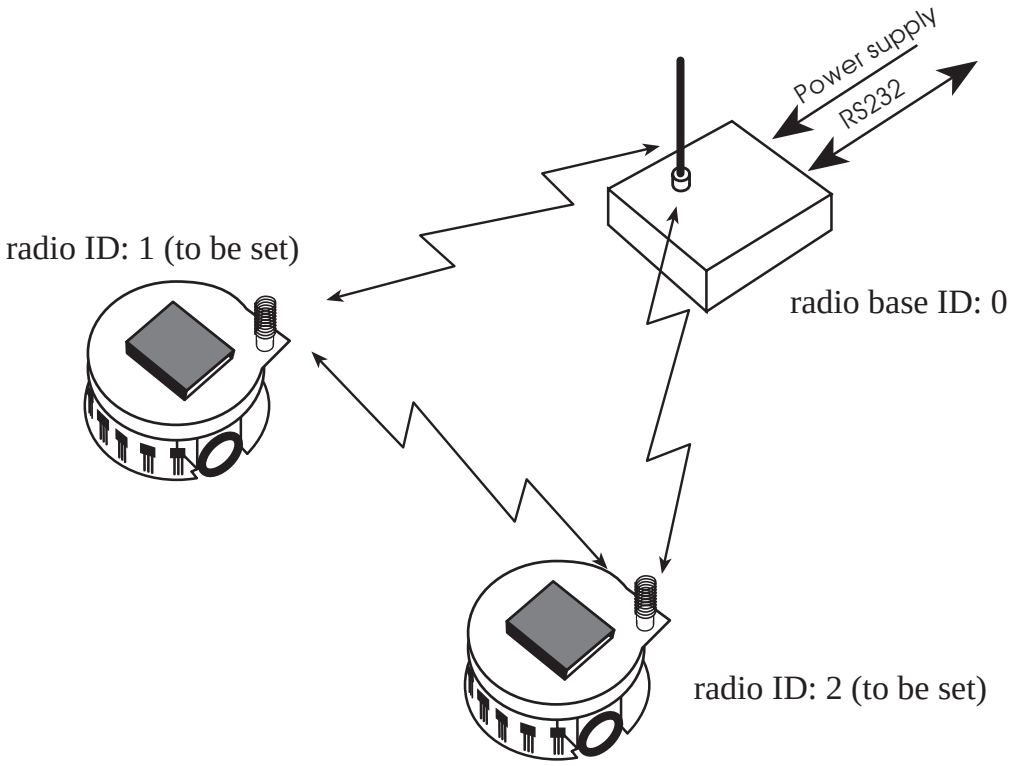

Figure 3: Radio network typical configuration

The network is based on ID numbers associated with each robot present in the radio network. The radio base station ID is always 0. The ID of a radio turret can be set on the radio turret itself using the ID selector (see section 4.5 for more details). **The radio turret ID should be never set to 0.** The ID selector is read at boot or after reset and should be different on every radio turret present inside the network.

4.4 Radio properties and correct data transmission

The radio channel is a half duplex channel, which means that reception and transmission of data are mutually exclusive. The default state of the radio module is reception. As soon as data has to be transmitted, the state changes and the data is transmitted.

The state of the radio module is visualized on the Rx (receive) LED (item 3 in figure 1) and on the Tx (transmit) LED (item 4 in figure 1).

The data transmitted on the radio channel is encapsulated in messages including information on the type of message, the sender ID, the destination ID, length of data and a checksum for error correction. The receiver acknowledges the reception of the mes-

sage. The sender considers the message as correctly sent only if acknowledged by the receiver. If no acknowledgment is received in a given timeout, the message is sent again. The same procedure is repeated until a maximum number of repetitions is reached. If all repetitions fail, the message is considered lost.

This protocol process is visualized on the “repeat data” LED (item 8 in figure 1) and “lost data” LED (item 9 in figure 1).

The data encapsulation and transmission protocol add an important quantity of information and time delay to simple data transmission. This ensures correct data transmission and is therefore necessary. Due to the fact that the added information and time delay is independent from the quantity of data, optimal results in data transmission speed are achieved with data having a length of 16 bytes (max length of data in a message).

WARNING: environmental conditions can drastically influence the transmission quality. Distance of transmission and quantity of data lost depend on several factors: proximity of metallic structures, other radio devices and noise emission from computers can all cause interferences. In a good environment no data should be lost. If data is lost, please check the environmental conditions.

4.5 Running mode and ID selector

The running mode and the radio turret ID are specified by 6 switches placed on the turret board (item 7 in figure 1, see detailed view in figure 4). Switches 1 to 6 are used for this purpose, switches 7 and 8 are not used and have to be always set to 0.

Changes will take place only at boot or after a reset.

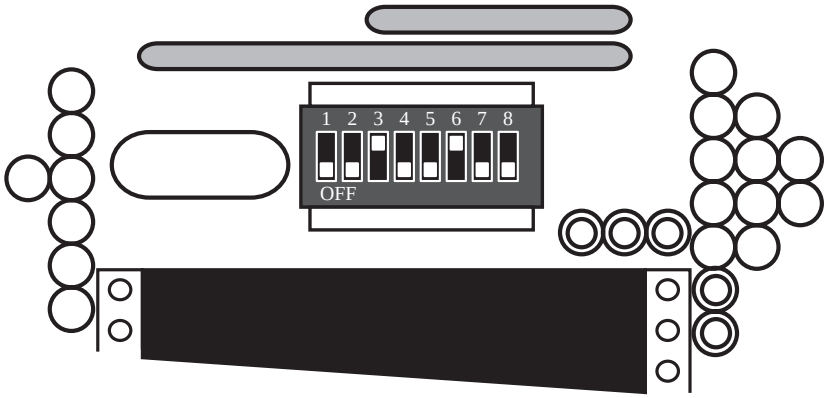

Figure 4: Running mode and ID selector

- Switches 1 to 5 define the 5 bits of the turret ID.
- Switch 6 defines the running mode of the turret: when this switch is in the OFF position, the turret is a normal extension turret. When this switch is in the ON position, the turret is used as main communication channel of the module COM. This functionality is available only with a BIOS of version 5.

- Switches 7 to 8 need to be set always to OFF.

Figure 5: Examples of selector settings

5 CONNECTIONS

Assembling and disassembling additional turrets is a delicate operation. Try to avoid it as much as possible and perform it carefully. Please follow the following instructions to avoid damage to your modules. K-Team can assume no responsibility for any damage caused by improper manipulation.

5.1 Assembling

Assembling is the easier operation, but it is also necessary to perform it carefully:

- First of all choose the parameters of the module on to which you plan to plug the new turret (the running mode that you will use on the basic Khepera configuration, for instance) and set the jumpers if necessary. When the turrets are assembled, it is impossible to access to the modules that are inside the robot without disassembling it.
- Assemble the turret with the basic configuration in two steps: First, place the module on the extension connector checking that all pins are seated

correctly. Second, apply force to insert the turret into the extension connector.

- If you want to connect the robot to your workstation, use the serial connector of the topmost turret.
- Operate as normal.

5.2 Disassembling

This is the most difficult operation for people that are not accustomed working with this type of hardware.

- First switch OFF the robot or disconnect the power supply.
- Separate the turret from the rest of the robot. To perform this without damage to the connections, it must be removed carefully such that all pins are disconnected simultaneously. One way to do this is to insert a large plastic screwdriver between adjacent modules and gently ease the boards apart, being careful not to push on delicate components. First open one side a bit, then the other, alternating sides until the module is free.

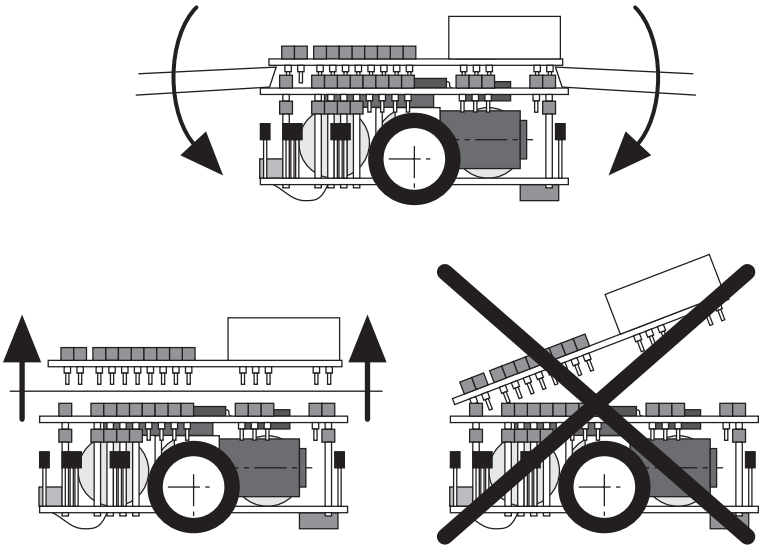

Figure 6: How to disassemble an additional turret.

6 USING LABVIEW®

This chapter explains how to use the radio turret from the LabVIEW® environment in the context of Khepera use.

WARNING: access to the radio turret as an slave device can be made only if the radio turret is configured as extension turret and NOT as main communication channel. If configured as main communication channel, the turret becomes transparent and cannot be accessed as an independent device.

LabVIEW® is a product of National Instruments (http://www.natinst.com). Please refer to the LabVIEW manuals for more information about this software.The following examples and the files distributed with this product are based on LabVIEW® version 5.

LabVIEW® runs on PC, Macintosh® or SUN® workstations, and can control the functionality of the Khepera robot using the serial communication protocol described the Khepera User Manual and in Appendix A.

6.1 Configuration

Set your environment as described in the Khepera User Manual for running LAb-VIEW®.

To enable the exchange of information between your computer and the robot, you have to configure the serial link of your host computer, according to the setting chosen on the Khepera robot. Be sure that the connection cable is connected at both ends (Khepera and interface), that the robot is powered (power adaptor), then start LabVIEW® and run the Set-up VI as for the basic Khepera robot.

6.2 Status of the radio module

The basic VI used is the “get_status” VI. Open it and you should get the following panel:

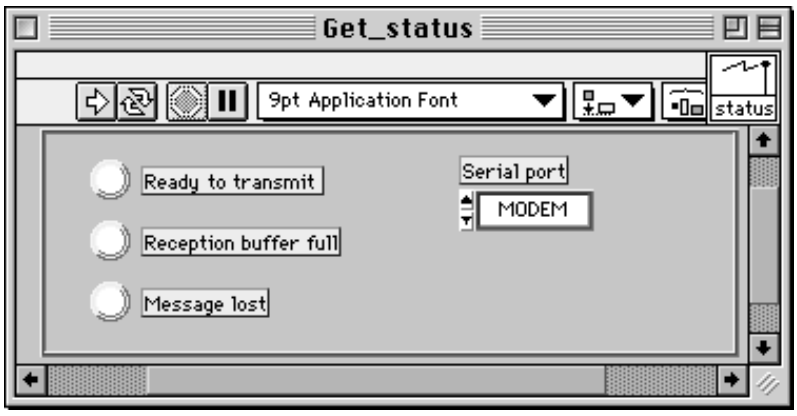

Figure 7: Get_status panel.

This panel displays the three radio turret status flags. The first flag indicates that the emission buffer is empty and that the radio is ready to transmit. The second flag indicates that a message has been received. The third flag indicates that the last message sent has been lost. This flag keeps its value until a new message is sent.

Run this VI continuously for real-time status of your radio turret. The emission ready flag should be active.

6.3 Emission of data

The emission of data is done with the VI “Send_buffer” illustrated below:

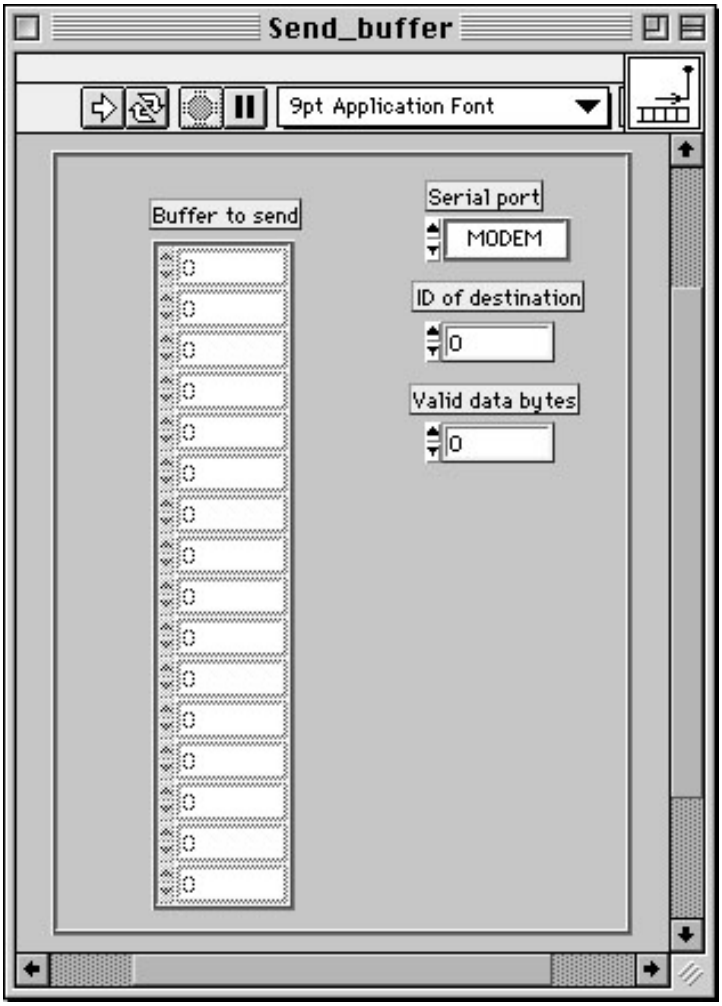

Figure 8: Send_buffer panel.

To transmit a data buffer, just fill in the buffer in the array control (Buffer to send), indicate the number of valid data (Valid data bytes) and the ID of the destination. The radio base ID is always 0. Run the VI. The data should be sent. On the status, the first flag should desactivate briefly. If the message is sent correctly, the first flag becomes active again after a short delay and the “Message lost” flag keeps its inactive value. If the mes-

sage is lost because of communication problems, both the ready to send and message lost LEDs become active. The radio is then ready to send a new message.

6.4 Reception of data

If another turret or the radio base send a message to your turret, the flag “Reception buffer full” of the status VI becomes active. To read the buffer, please use the “Get_buffer” VI illustrated below:

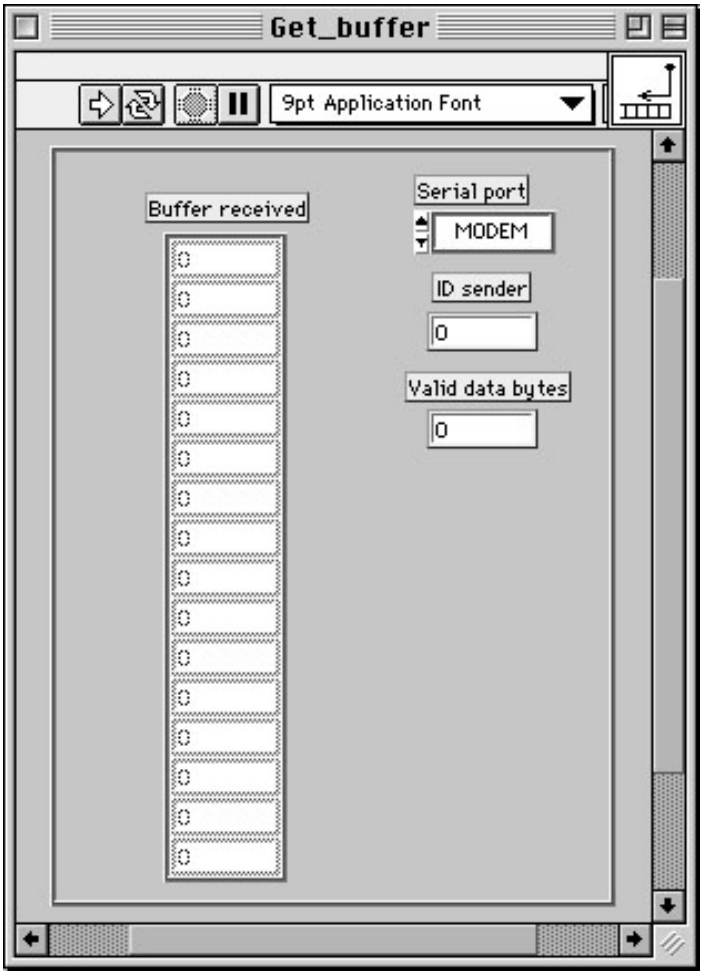

Figure 9: Get_buffer panel.

Running it you will see the buffer and the status flag “Reception buffer full” will be desactivated. The VI will show the buffer, the size of the valid message and the ID of the sender of the message. Executing this VI again will retrieve the same old buffer again. For this reason it is necessary to check the status before reading the buffer.

APPENDIX A COMMUNICATION PROTOCOL TO CONTROL THE RADIO TURRET

This communication protocol allows complete control of the functionnalities of the radio turret through a RS232 serial line and the robot main processor. The connection configuration needed is presented in section 5.2 of the Khepera USER MANUAL. The setup of the serial line of your host computer must correspond to the one set on the robot with the jumpers (running modes 1 to 3).

The protocol used to control the radio turret from the robot should not be confused with the protocol used to control the robot from a host computer. We call the robot control protocol “main protocol” and the radio turret control protocol “turret protocol”.

The “T” command of the main protocol transmits a command of the turret protocol to the additional turret with the given identification number (see appendix A of the Khepera USER MANUAL). The identification number of the radio turret is 4.

The turret protocol is constituted by commands and responses, with an header in ASCII codes, like for the main protocol. The string of the turret protocol must be inserted in the command field of the T order of the main protocol (see appendix A of the Khepera USER MANUAL). A command of the turret protocol is constituted by a capital letter followed, if necessary, by 8 bit (0 to 255) numerical parameters separated by a comma. The response is transmitted in the response field of the command T of the main protocol. The response is constituted by the same letter of the command but in lower case, followed, if necessary, by 8 bit numerical parameters separated by a comma.

To better understand this protocol we propose a very simple test as following:

- Set the jumpers of the robot for running mode number 1.
- Plug in the radio turret.
- Set the connection configuration presented in section 5.2 of the Khepera User Manual.
- Start a terminal emulator (for instance VT100) on your host computer with the serial line set to 9600 Baud, 8 bit data, 1 start bit, 2 stop bits, no parity.
- Type the text T,4,B followed by a carriage return or a line feed.
- The robot must respond with t,4,b, followed by an indication of the version and revision of the software running on the radio turret and terminated by a line feed.
- Try other commands:

Command of the main protocol for the control of an additional turret

(See also Appendix A of the Khepera User Manual)
∏ indicates CR (carriage return) **or** LF (line feed). ¶ indicates CR **and** LF.

T Send a message to an additional turret

Format of the command: T, turret_ID, command_turret_protocol∏

Format of the response: t, response_turret_protocol¶

Effect: Send a command and return the response of the additional turret with turret_ID. The turret_ID of the radio turret is 4. The turret protocol takes the same form as a standard main protocol. Every command of the turret protocol includes an identification capital letter followed, if necessary, by numerical parameters separated by commas. The response takes the same format, starting with the same letter but in lower case, followed, if necessary, by numerical parameters separated by commas. The command and response formats are specific for every turret and are described in the turret protocol.

List of available commands of the turret protocol for the radio turret

B Read software version

Format of the command: B

Format of the response: b,version_of_software, revision_of_software

Effect: Read the version and revision of the radio turret software running on the local 68HC11 processor.

F Read status

Format of the command: F

Format of the response: f,status

Effect: Read the 8 bit status of the radio module. Bit 0 of the status byte is set to 1 if the emission buffer is empty and the sender is ready to send a new message. Otherwise this bit is set to 0. Bit 1 of the status byte is set to 1 if there is a message in the reception buffer. Bit 2 of the status byte is set to 1 if a message was sent 10 times without success. In this case the message was not sent successfully and is considered lost. This bit is cleared when the next message is sent. All other bits can take the value 0 or 1.

R Read reception buffer

Format of the command: R

Format of the response: r, source_ID, mess_size, byte1, byte2, ..., byte15, byte16

Effect: Read the complete reception buffer composed of 16 bytes. “source_ID” indicates the sender of the message and mess_size defines the effective size of the message stored in the fixed size buffer. Bytes 1 to 16 compose the buffer. All 16 bytes have to be read.

S Send buffer

Format of the command: S, dest_ID, mess_size, byte1, byte2, ..., byte15, byte16

Format of the response: s

Effect: Send a buffer composed of 16 bytes. “dest_ID” indicates the ID of the destination of the message and mess_size defines the effective size of the message stored in the fixed size buffer. Bytes 1 to 16 compose the buffer. All 16 bytes have to be specified, even if unused.