

us sensor

user manual



version 1.0
JUNE 2010

Documentation Author

Frederic Lambercy
K-Team S.A.
Rue Galilee 9, Y-Park
1400 Yverdon-les-Bains
Switzerland

Email: info@k-team.com

Url: www.k-team.com

Trademark Acknowledgements:

IBM PC: International Business Machines Corp.

Macintosh: Apple Corp.

SUN Sparc-Station: SUN Microsystems Corp.

LabVIEW: National Instruments Corp.

Matlab: MathWorks Corp.

K-Junior: K-Team.

LEGAL 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 CONTENTS



1	INTRODUCTION	5
1.1	HOW TO USE THIS MANUAL.....	5
1.2	SAFETY PRECAUTIONS	5
1.3	RECYCLING	5
2	CONNECTIONS	6
2.1	INSTALL	6
2.2	UNINSTALL.....	6
3	THE K-JUNIOR ULTRASONICSENSOR	7
3.1	OVERVIEW	7
3.2	I2C ADDRESS	7
3.3	PHYSICAL PHENOMENON.....	8
3.4	REGISTER ADDRESSES.....	8
4	ULTRASONICSENSOR USE.....	10
4.1	C PROGRAMMING	10
4.2	SERIAL PORT CONTROL.....	11
A	TECHNICAL SPECIFICATION	12
B	CONNECTORS.....	13
C	SIGNAL APERTURE.....	14

1 INTRODUCTION



The IR sensors fitted to K-Junior have a maximum detection distance of approximately 20cm depending on the materials in question, and in addition, these sensors are so-called proximity sensors; in other words they do not give information on the exact distance of an obstacle. The K-Junior UltraSonicSensor module enables you not just to considerably extend the detection distance (it can be used from 3cm to 6m), but in addition the information received is in the form of a distance reading (with no fluctuations resulting from type of obstacle). This therefore allows K-Junior to find its bearings in much more complex environments, as well as detecting obstacles a lot earlier.

1.1 *How to use this Manual*

This manual is an introduction to the K-Junior UltraSonicSensor designed for the K-Junior robot. If you cannot find any information in this manual relating to a given problem that you encounter, please go to the K-Team website (<http://www.k-team.com/>), and in particular the Forum or the FAQs section.

1.2 *Safety precautions*

Here are some recommendations on how to correctly use the K-Junior Robot:

- **Always make sure it is placed in a stable position.** This will avoid any risk of it falling that might lead to its breaking or causing injury to another.
- **Keep out from the wet places!** A contact with water can make a short circuit and damage the electronics.
- **Don't plug or unplug any connector or turret when the robot is powered!** All connections and turret insertions must be made when the robot and the interface are switched OFF. Otherwise damages can occur.
- **Never leave the K-Junior powered when it is unused.** When you have finished working with K-Junior, turn it off. It will save the battery life

1.3 *Recycling*

Think about the end of life of your robot! Parts of the robot can be recycled and it is important to do so. It is for instance important to keep batteries out of the solid waste stream. When you throw away a battery, it eventually ends up in a landfill or municipal incinerator. These batteries, which contain Lithium Polymer, can contribute to the toxicity levels of landfills or incinerator ash. By recycling the batteries through recycling programs, you can help to create a cleaner and safer environment for generations to come. For those reasons please take care to the recycling of your robot at the end of its life cycle, for instance sending back the robot to the manufacturer or to your local dealer.

Thanks for your contribution to a cleaner environment!

2 Connections



Fitting and removal of the extension module is a delicate operation. Please read carefully the instructions below to prevent damage to it. K-TEAM cannot be held liable for any damage caused by improper handling.

2.1 Install

To begin with, make sure the robot is switched off; then insert the K-Junior UltraSonicSensor module keeping it as vertical as possible, but making sure it is positioned as below:



Figure 2.1: K-Junior US Sensor module on the K-Junior

Please do not attempt to insert K-Junior UltraSonicSensor in a different orientation or slightly biased, it could cause important damages to K-Junior or the extension module.

2.2 Uninstall

Check first that the K-Junior Robot is switched off, hold firmly the K-Junior in your hand, pull as much vertical as possible the K-Junior UltraSonicSensor module with the other hand. Please take extra care in not bending connector pins.

3 The K-Junior UltraSonicSensor



3.1 Overview

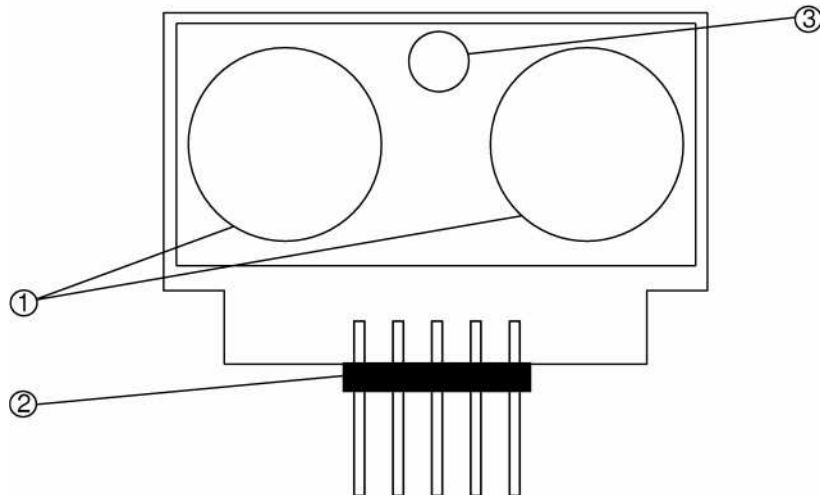


Figure 3.1: Overview of the UltraSonicSensor turret layout

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

1. Ultrasound transmitter and receiver
2. K-Junior Extension Connector
3. Luminosity sensor

For more information on connector details, please consult the Appendix at the end of this manual.

3.2 I2C Address

The K-Junior Robot control the USsensor through an I²C bus. The turret address is defined by 7 bits + 1 bit for the mode selection "write" or "read"

1	0	0	1	0	1	1	R/W
---	---	---	---	---	---	---	-----

The GenIO I²C address is 0xE1 in "read" mode and 0xE0 in "write mode".

3.3 Physical Phenomenon

Unlike infrared sensors, which are so-called proximity sensors, ultrasound modules enable distance measurements. The principle is to transmit a "packet" of ultrasound waves and measure the time until the echo is received on the receiver. The distance d to the object producing the echo can be measured, provided we know the speed of sound propagation in an environment c and the elapsed time t :

$$D = \frac{c \cdot t}{2}$$

In the air, at normal atmospheric pressure and 20°C, the speed of sound is about $c=343\text{m/s}$. Ultrasonic waves, typically of between 40 and 180 kHz frequency, are in general produced by a piezzo transducer.

3.4 Register Addresses

Different registers allow configuration of the module or activation of measurements.

I ² C Address	Reading	Writing
0x00	Firmware Version	Main register
0x01	Luminosity sensor	-
0x02	MSB of 1 st echo	Distance control
0x03	LSB of 1 st echo	
...	...	
0x22	MSB of 17 th echo	
0x23	LSB of 17 th echo	

Note:

- MSB = Most Significant Bit
- LSB = Least Significant Bit

3.4.1 Firmware Version Register

All this register does is send back the firmware version of the ultrasound module.

3.4.2 Main Register

This is the register used to activate a measurement; for this different values can be sent:

Value	Description
80 (0x50)	Activate a measurement with a result in inches
81 (0x51)	Activate a measurement with a result in centimetres
82 (0x52)	Activate a measurement with a result in microseconds

So writing 0x51 in the register 0x00 enables activation of a measurement. Once the measurement is activated, the result is available at most 65ms later (this time depends on the settings within the distance register). If you do not want to wait a certain given time, but prefer to check if the measurement is ready, all you need to do is read-access the register 0x00 (Firmware Version); if the module is still taking measurements it will send back the value 0xFF (255).

3.4.3 Luminosity Sensor Register

Reading of this register gives a value proportional to the amount of light present in the environment. The value of this sensor is only refreshed if a measurement is requested on the main register.

3.4.4 Distance Control Register

The module is capable of measuring distances up to 6 metres. If you only want to detect objects at 30cm, there is no point in the module waiting for an echo that might come from an object detected at 6m. So you can specify for it the distance of the obstacles that you want to detect, and the module will carry out the measurement more quickly. The maximum detectable distance is specified by $((\text{value written in the register} \times 43 \text{ mm}) + 43\text{mm})$; so writing the value 24 (0x18) allows you to get a maximum detection distance of $(24 \times 43\text{mm}) + 43\text{mm} = 1075\text{mm}$.

3.4.5 Echo Registers

If you set up K-Junior opposite you and you have a wall behind you, two echoes are sent, one reflecting your presence and another caused by the wall. So the echoes allow detection of the nearest obstacle as well as the ones after that. Since the distance values are 16-bit, you have to read the MSB register and the LSB register to get the distance in your chosen unit.

4 UltraSonicSensor Use



There are two ways of using HemUltraSonicSensor with K-Junior:

- C Programming
- Serial port control (e.g. SysQuake LE)

4.1 C Programming

To program K-Junior in C, you first need to acquire the CCS compiler (available from the software section of the K-Team website). We first recommend you read the latest version of the KJOs manual (<http://ftp.k-team.com/K-Junior/KJOSManual.doc>); in this document you will find all the information you need on the KJOs as well as how to install CCS.

- Then download the latest available version of KJOs, as well as the latest version of K-Junior UltraSonicSensor Lib (containing HemUltraSonicSensor.h).
- Open the project in CCS

The HemUltraSonicSensor.h library includes the following functions:

4.1.1 Void HemUltraSon Init(void)

Purpose: Initialisation of the ultrasound module. This is the first function that needs to be called at the start of your code.

Example: HemUltraSon_Init();

4.1.2 Char HemUltraSon Read Version(void)

Purpose: Read the firmware version.

Example: char Version;
Version = HemUltraSon_Read_Version();

4.1.3 Char HemUltraSon Read Brightness(void)

Purpose: Read the luminosity sensor.

Example: char Brightness;
Brightness = HemUltraSon_Read_Brightness();

4.1.4 Void HemUltraSon Start Measure(void)

Purpose: Activate a distance measurement. This function already includes a waiting time of 65ms

Example: HemUltraSon_Start_Measure();

4.1.5 Unsigned int16 HemUltraSon Read Values(char Echo)

Purpose: Read the echo obtained; the echo number is parameterised (from 0 to 16)

Example: unsigned int16 Distance;
Distance = HemUltraSon_Read_Values(0);

4.1.6 Void HemUltraSon Init Range Register(Undsigned char Value)

Purpose: Initialise the maximum detectable distance.

Example: HemUltraSon_Init_Range_Register(0x10);

NB: To use these functions, you must first prefix an include command for the library at the beginning of your code: #include "HemUltraSonicSensor.h" (Clearly the file needs to be put in the same register.)

4.2 Serial Port Control

With the serial protocol of the K-Junior, you can access I2C modules via RS-232 commands. To do this you should consult K-Junior user manual.

Examples:

- R,E0,00,01 : Reads the firmware version.

You can use any development environment that can access serial port to interface to K-Junior USSensor. For instance, we developed a graphical interface to read and configure the K-Junior USSensor. You must first plug your K-Junior USSensor to the K-Junior, turn on the robot and then connect the PC to the K-Junior with the USB cable.

A Technical Specification



- Weight : NC
- Power Supply : 5 [V]
- Minimum detectable distance: 3 [cm]
- Maximum detectable distance: 600 [cm]

Power consumption varies depending on modes of use:

Description	Consumption	Duration
The module is in the process of carrying out a measurement.	15mA	65ms max.
All other modes	3mA	Undefined

B Connectors

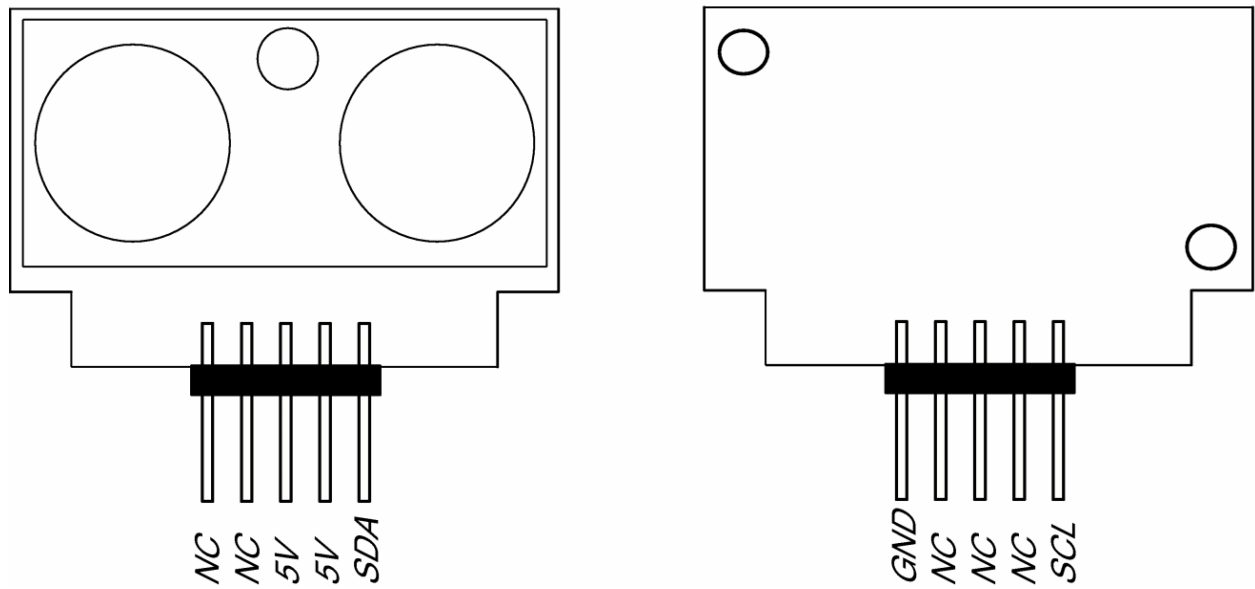
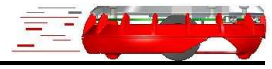


Figure B.1: Connector details

C Signal Aperture

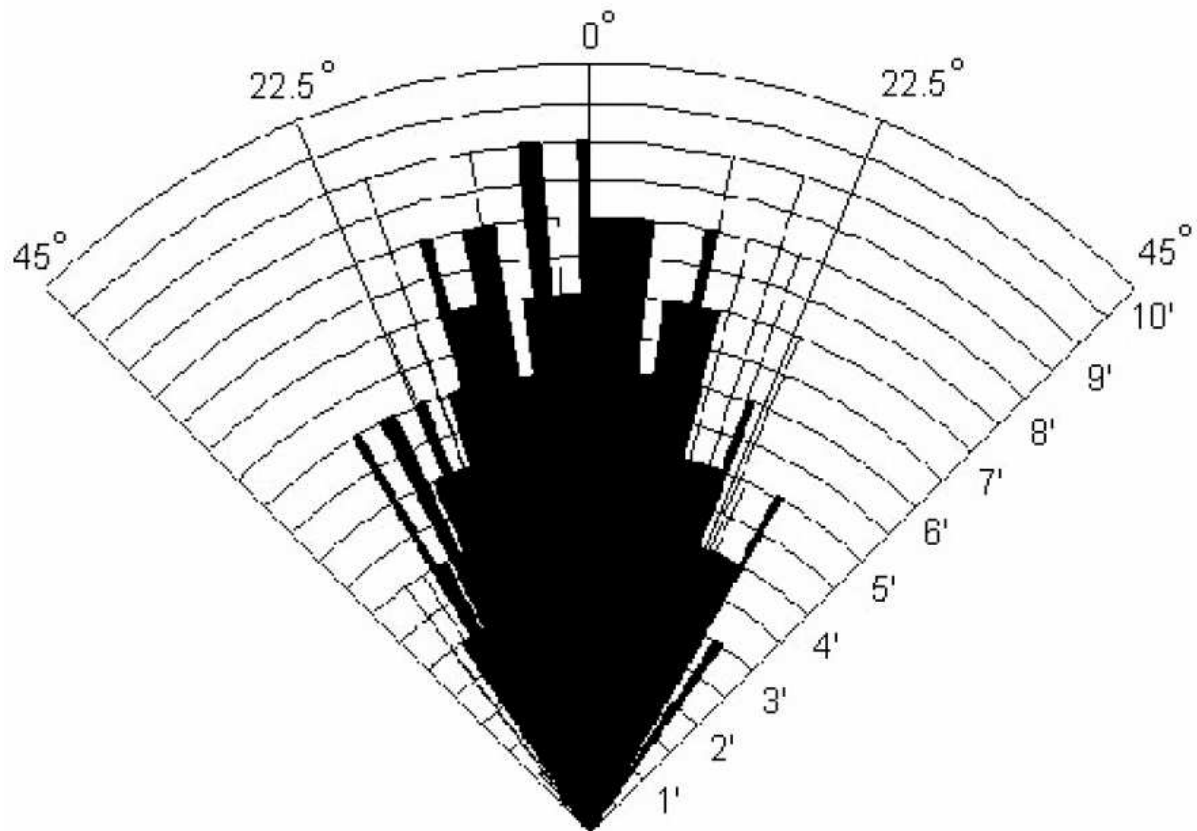
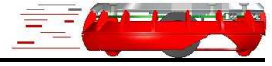
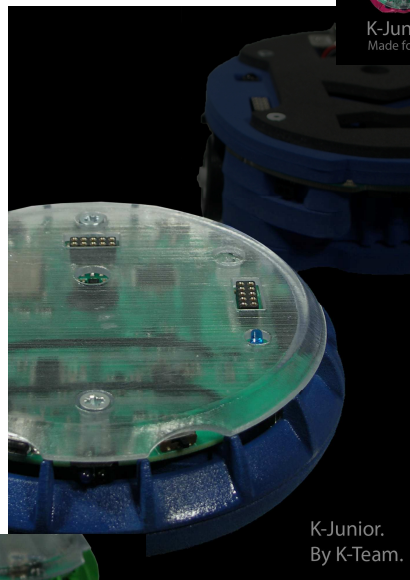
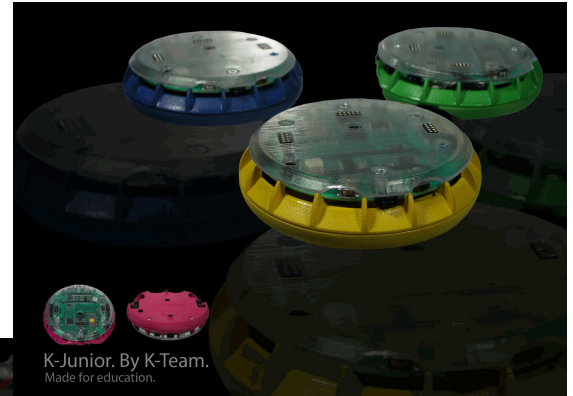


Figure C.1: Spatial Distribution of Signal



K-Team S.A.
RUE GALILEE 9
1400 YVERDON-LES-BAINS
SWITZERLAND