

**Documentation Author**

Arnaud Maye ( K-Team S.A. )  
Ch. du Vuasset, CP 111  
1028 Préverenges  
Switzerland

email: [amaye@k-team.com](mailto:amaye@k-team.com)

Url: [www.k-team.com](http://www.k-team.com)

**TRADEMARK ACKNOWLEDGMENTS:**

**IBM PC:** International Business Machine Corp.

**Macintosh:** Apple Corp.

**SUN Sparc-Station:** SUN Microsystems Corp.

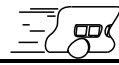
**Koala:** K-Team and LAMI

**LEGAL NOTICE:**

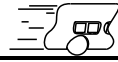
- The content of this manual is subject to change without notice.
- All effort have been made to ensure the accuracy of the content of this manual. However, should any error be detected, please inform K-Team S.A.
- The above notwithstanding K-Team can assume no responsibility for any error in this manual.

# TABLE OF CONTENTS

---



<b>1</b>	<b>Introduction</b>	<b>3</b>
1.0.1	LedRangeFinder . . . . .	3
1.0.2	PBS Module . . . . .	3
1.0.3	PBS Communication . . . . .	3
<b>2</b>	<b>Led Range Finder Hardware</b>	<b>5</b>
2.1	Overview . . . . .	5
2.2	LedRangeFinder Connections . . . . .	6
2.2.1	Communication Connections . . . . .	6
2.2.2	Power Connections . . . . .	6
2.3	Hardware Protection . . . . .	6
2.3.1	Electrostatic Discharge Protection . . . . .	6
<b>3</b>	<b>LedRangeFinder Software</b>	<b>7</b>
3.1	Overview . . . . .	7
3.2	LedRangeFinder Programming on KoreBot . . . . .	7
3.2.1	simple LRF programming . . . . .	7
3.2.2	advanced LRF programming . . . . .	8
3.2.3	Supported LRF APIs . . . . .	10



## 1.0.1 LedRangeFinder

The LedRangeFinder board is an extension that provides necessary hardware to use the Hokuyo PBS-03JN module on the K-Team Koala robot. Even if this board may be used with an external desktop personal computer, we recommend to use the KoreBot cpu board with the KoreBotAdaptor board.

## 1.0.2 PBS Module

The Infrared Scanner module allows to instant reliable measurement of the whole front 180 degrees, this even if one can retrieve from 18 deg minus to 198 deg. Measurement principle of this module isn't known. Only things that are known is that internal leds are rotating at a 600 turns per minute rate while measuring reflected light with an 1.8 degree resolution. Every received measure block will gives more than hundred millimeters values. The module delivers tens measures per minute, but we recommend to average several measurements to increase the reliability of the results.

## 1.0.3 PBS Communication

The Infrared Scanner module using some system to ensure that the software has been compiled for the specific module.

### Communication procedure:

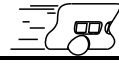
1. Host asks 'ID code' from the PBS-03JN module, there is one 'ID code' by sensors
2. Sensor then returns its 8 bytes 'ID code'
3. Host then send 16 bit crc back to the sensor
4. Sensor then confirm link is certified

At this time, the host requires to certify link every three seconds (maximum) otherwise the sensor will close the link and the whole link certification have to be done again.

5. The host ask for a measure
6. The sensor returns a measure

Note that programmer doesn't requires to deal with such low level functions for a normal situation, because top level apies doing the whole link certification.

## 2 LED RANGE FINDER HARDWARE



### 2.1 Overview

LedRangeFinder.

Figure 2.1 describes the board main components.

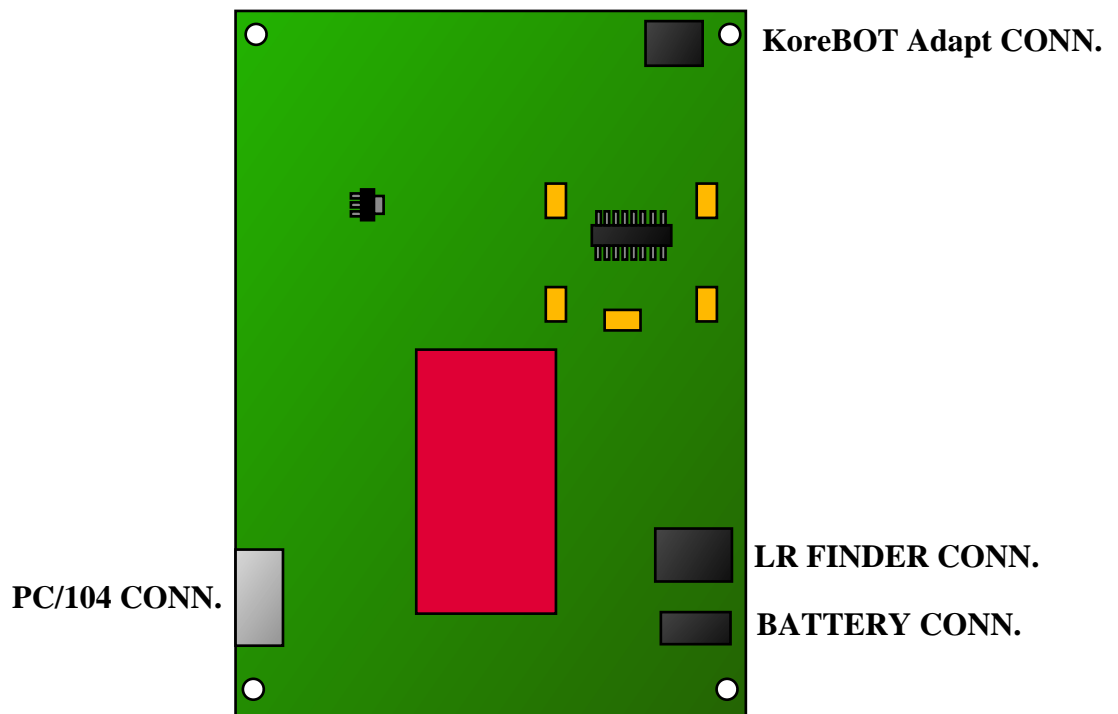


Figure 2.1: LedRangeFinder hardware overview

## **2.2 LedRangeFinder Connections**

### **2.2.1 Communication Connections**

LedRangeFinder offers two communication connectors.

The 'PC/104' connector is used as a RS232 input (+/-9V min), because its type, is straightforward to build your own cable to plug your own desktop computer. However, best results are achieved via the KoreBOT adaptor connector.

The 'KoreBOT adaptor' connector is used as main communication channel. DO NOT CONNECT the KoreBOT without to use the KoreBOT adaptor, that would just damage your Korebot.

The 'LR Finder' connector is used to connect the PBS-03JN module

### **2.2.2 Power Connections**

The LedRangeFinder card is powered by the KoreBot via the KoreBot Adaptor board.

The BATTERY connector is used to power the DC/DC converter which will requires at least 9V dc to function properly. Be attentive that wire should resist to about 1A. This because the on-board fuse is 1A as well

## **2.3 Hardware Protection**

### **2.3.1 Electrostatic Discharge Protection**

As any electronic device, the KoreSound may be damaged by Electrostatic Discharge.



### 3.1 Overview

This chapter presents how to use LedRangeFinder from a KoreBot ( with a KoreBot Adpator ) using a Linux Operating System. It requires:

- a KoreBot (with a KoreBot Adpator)
- a LedRangeFinder
- a terminal emulator software (such as minicom or HyperTerminal)
- a host computer with at least one RS-232 connector (such as a standard PC running Linux or Windows)
- GNU tools for KoreBot (assembler, linker and C/C++ compiler for Intel XScale processor) running on your host computer (only for programming lrf application)

### 3.2 LedRangeFinder Programming on KoreBot

This section shows how to access to the led reange finder from your standard Linux application. The examples given are in standard C and use the standard C library. It requires some knowledge about the file functions on Linux.

#### 3.2.1 simple LRF programming

There is no need to load a particular driver, what you requires is to use the predefined APIs provided with the standard korebot library. Start with to add korebot header to your source code. And then initialise the led range finder module.

```
#include <korebot/korebot.h>

int LRF_DeviceHandle; /* Handle to the device */

LRF_DeviceHandle = kb_lrf_Init();
```

if kb\_lrf\_init returns with a negative value, there was an issue. So if LRF\_Device.Handle isn't negative, we can continue with retrieving some values from the LRF. For the example, let's loop 10 times while retrieving

and printing out measurement. We finishing with closing the led range finder module This is the simplest possible software code and the usage of kb\_lrf\_distancedata gives averaged results.

the measure is stored in an array called kb\_lrf\_DistanceData that is defined as an external variable within the kb\_lrf software.

```
if (LRF_DeviceHandle > 0)
{
    for (int i=0; i<10; i++)
    {
        if (kb_lrf_GetDistances(LRF_DeviceHandle) >= 0)
        {
            for (int j=0; j<121; j++)
            {
                printf ("Dir %3.3d: ", j);
                printf ("%6.6d ", kb_lrf_DistanceData[j]);
            }
        }
        printf("\n");
    }

    kb_lrf_Close(LRF_DeviceHandle);
}
```

### 3.2.2 advanced LRF programming

Of course, there is more APIs available for programmers. Especially because the led range finder module requires much power programmer may use kb\_lrf\_pwrOff to turn off the on board DC/DC converter which just turn down the module. Note that the kb\_lrf\_init function does calling kb\_lrf\_pwrOn and that kb\_lrf\_close function does calling kb\_lrf\_pwrOff

A simple measure that turns on, get five measures and turns off the module.

```
int getMeasure( unsigned char[] [] * ptr )
{
    int rc = -1; /* variable for return code */
    int LRF_DeviceHandle; /* Handle to the device */

    LRF_DeviceHandle = kb_lrf_Init();

    if (LRF_DeviceHandle > 0)
    {
        for (int i=0; i<10; i++)
```



```
    {
        if (kb_lrf_GetDistances(LRF_DeviceHandle) >= 0)
        {
            memcpy(ptr[i][0], kb\_lrf\_DistanceData, 121);
        }
    }
    rc = 0; /* success */

    kb_lrf_Close(LRF_DeviceHandle);
}

return rc;
}
}
```

We encourage programmers to check the kb\_lrf.c from the libkorebot's src folder to get more infos about the other api

### 3.2.3 Supported LRF APIs

We encourage programmers to check the `kb_lrf.c` from the `libkorebot's src` folder to get more infos about the apies.

Function	Description
<code>kb_lrf_OpenComPort</code>	Opens the LRF com port.
<code>kb_lrf_FlushSerPort</code>	Empty the serial port hardware buffer.
<code>kb_lrf_GenerateCRC</code>	Generates 16 bit ccit crc from a given data and length.
<code>kb_lrf_GetDataFromLRF</code>	Retrieves some data from the led range finder sensor.
<code>kb_lrf_SendCommand</code>	Send a command to the led range finder module.
<code>kb_lrf_AcquireCertifiedCode</code>	Ask led range finder module to returns its 'ID code'.
<code>kb_lrf_CertifyLink</code>	Check that link is still certified. This function need to be called regularly.
<code>kb_lrf_pwrOn</code>	Turns on the led range finder module power.
<code>kb_lrf_pwrOff</code>	Turns off the led range finder module power.
<code>kb_lrf_Init</code>	Initialises the led range finder module, this function turning power on as well.
<code>kb_lrf_Close</code>	Closes a handle to the led range finder module, this fuction turning power off as well.
<code>kb_lrf_FetchDistanceFromLRF</code>	Retrieves an non averaged measure from the led range finder module.
<code>kb_lrf_GetDistances</code>	Retrieves an averaged measure from the led range finder module.