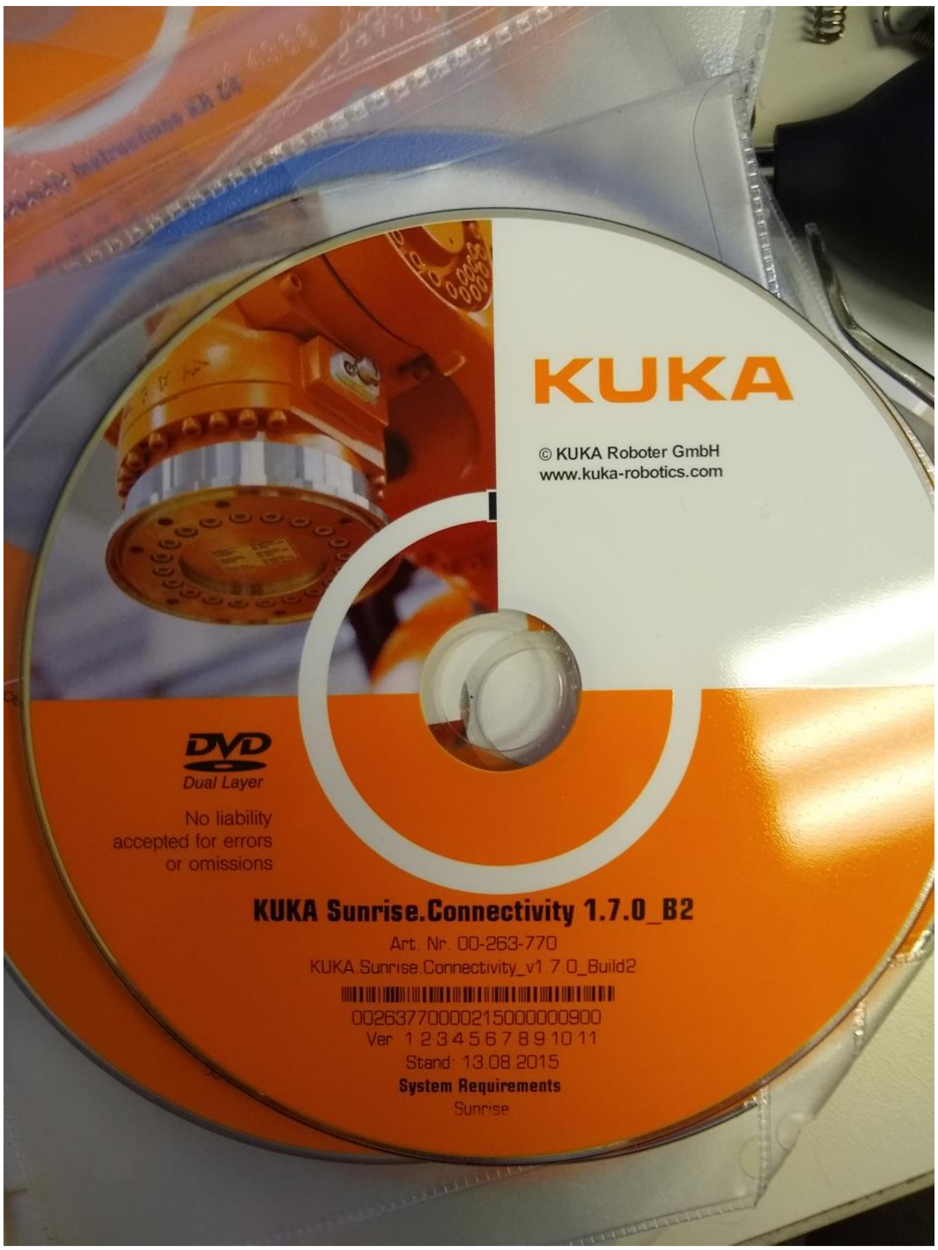
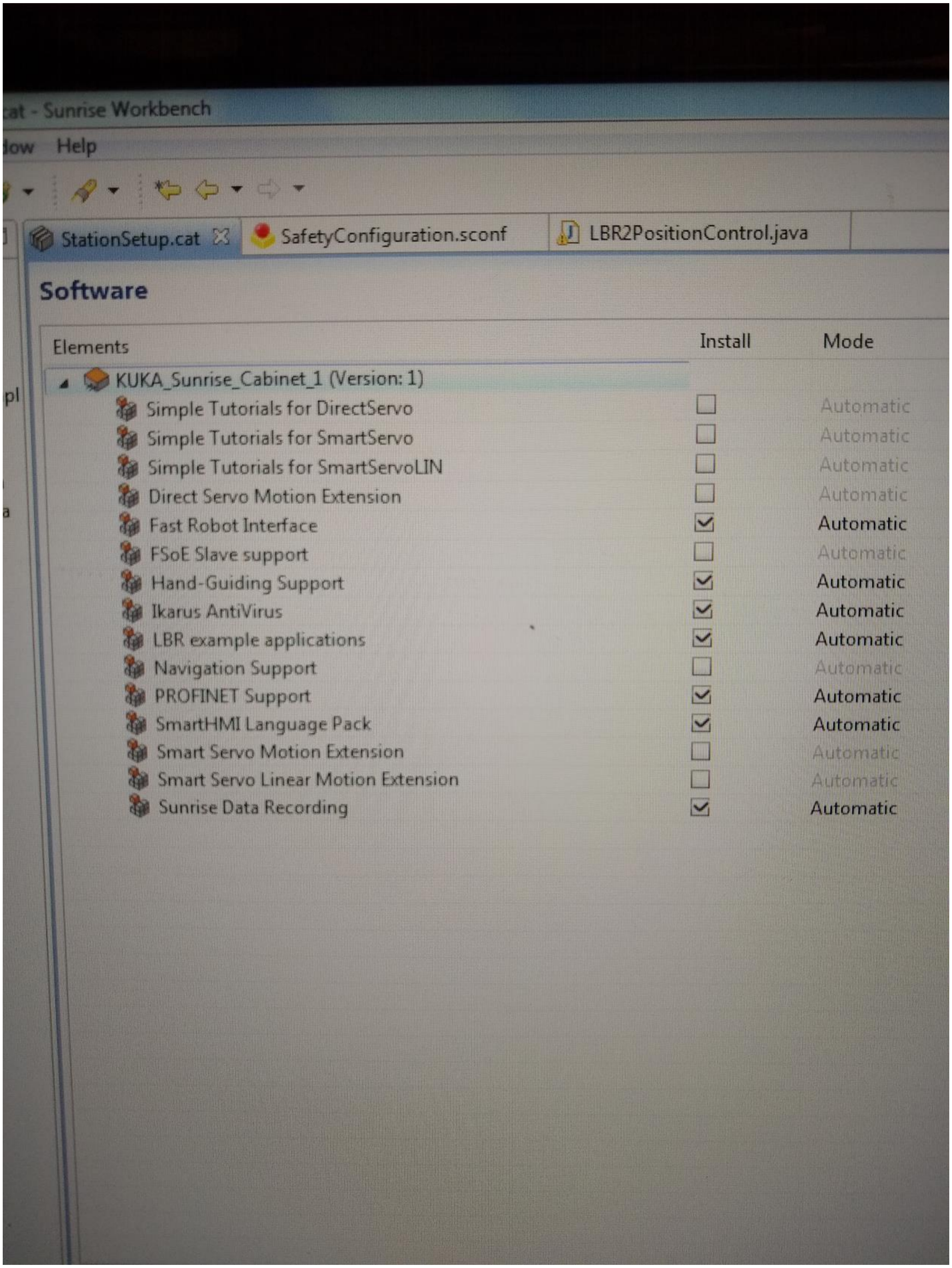
NOTE: Before attempting this tutorial, it is strongly recommended that you read through the KUKA Sunrise OS Manual, KUKA FRI Manual and the KUKA Quick Start Manual. This tutorial does not cover some basic functionality, such as using the KUKA controller, which you'll need for this tutorial. Furthermore, these manuals are critical to troubleshooting any errors that may occur while operating the KUKA.

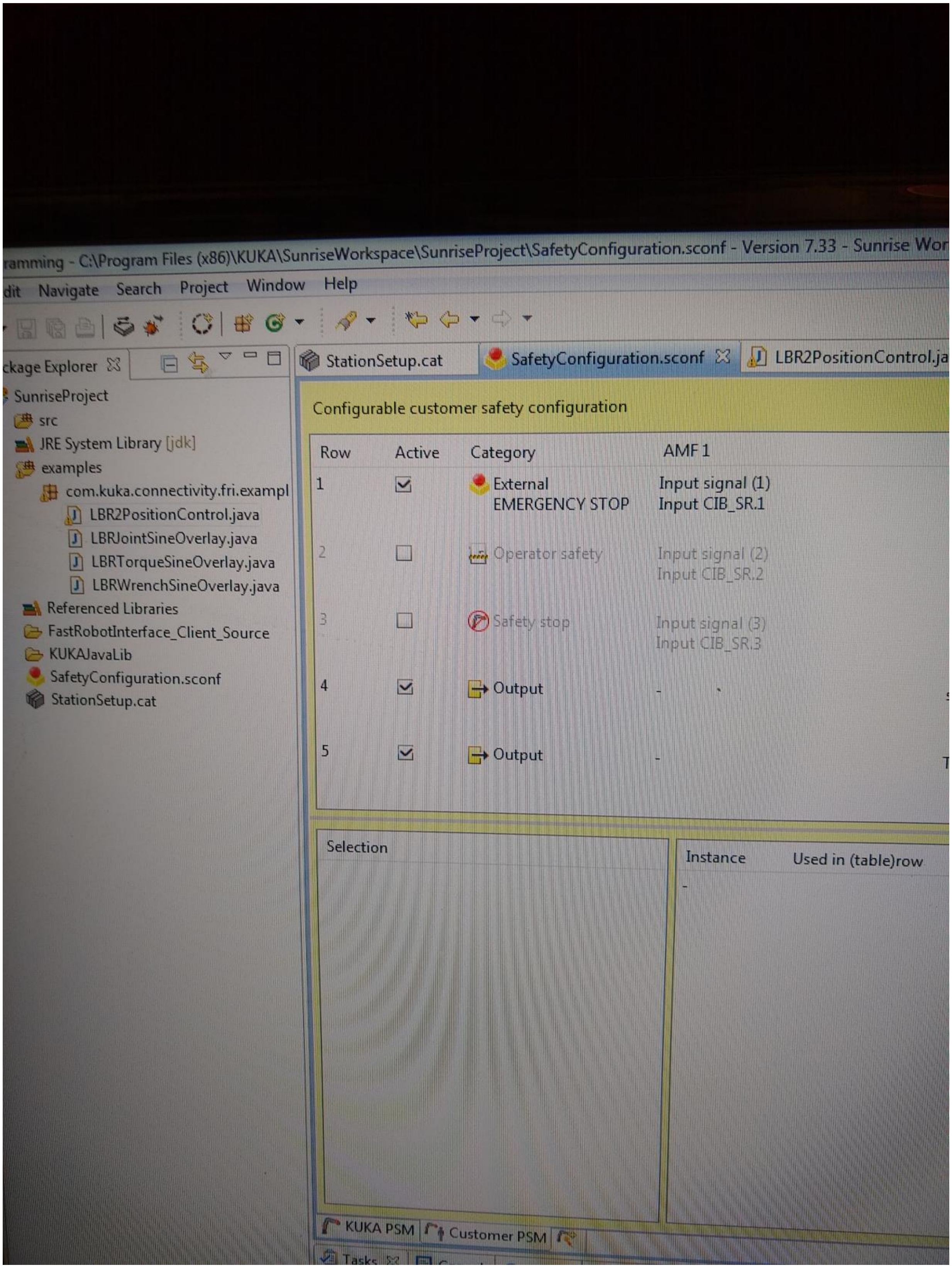
KUKA Setup: Installing Sunrise

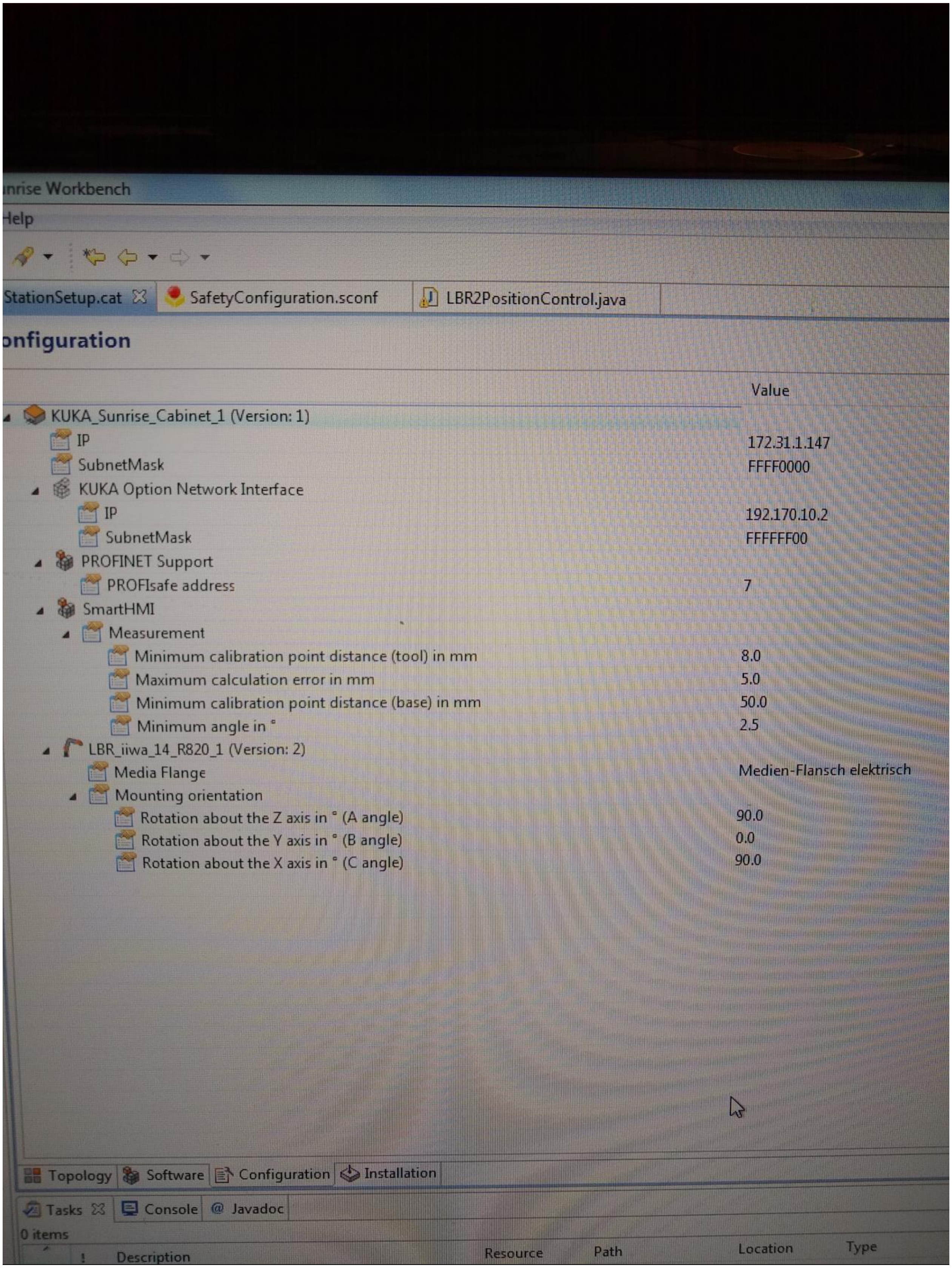
Sunrise is the software that will allow you to program and upload codes to the KUKA cabinet. It must be installed on an off board computer running Windows 7. You will need to install is called KUKA Sunrise Connectivity 1.7.0_B2. If you do not install the connectivity version of Sunrise, you will not have Fast Robotic Interface (FRI) functionality.



The following settings are what you'll need for the KUKA in its current wall mounted setup.



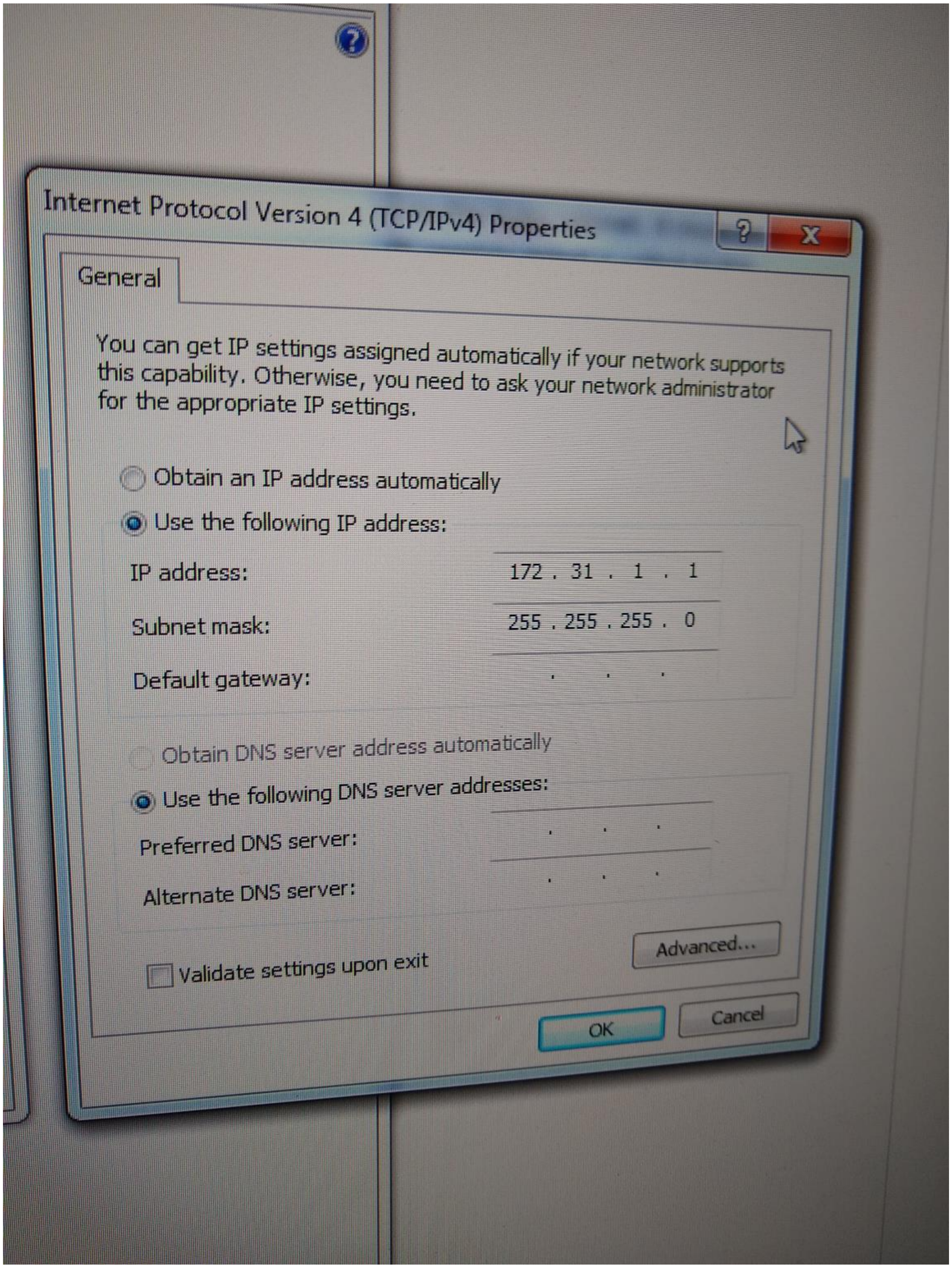

ramming - C:\Program Files (x86)\KUKA\SunriseWorkspace\SunriseProject\SafetyConfiguration.sconf - Version 7.33 - Sunrise Wor
dit Navigate Search Project Window Help
ckage Explorer
SunriseProject
src
JRE System Library [jdk]
examples
com.kuka.connectivity.fri.exampl
LBR2PositionControl.java
LBRJointSineOverlay.java
LBRTorqueSineOverlay.java
LBRWrenchSineOverlay.java
Referenced Libraries
FastRobotInterface_Client_Source
KUKAJavaLib
SafetyConfiguration.sconf
StationSetup.cat
StationSetup.cat
SafetyConfiguration.sconf
LBR2PositionControl.ja
Configurable customer safety configuration
Row Active Category AMF 1
1 External EMERGENCY STOP Input signal (1) Input CIB_SR.1
2 Operator safety Input signal (2) Input CIB_SR.2
3 Safety stop Input signal (3) Input CIB_SR.3
4 Output -
5 Output -
Selection
Instance Used in (table)row
KUKA PSM
Customer PSM
Tasks



inrise Workbench
Help
StationSetup.cat
SafetyConfiguration.sconf
LBR2PositionControl.java
onfiguration
Value
KUKA_Sunrise_Cabinet_1 (Version: 1)
IP 172.31.1.147
SubnetMask FFFF0000
KUKA Option Network Interface
IP 192.170.10.2
SubnetMask FFFFFF00
PROFINET Support
PROFIsafe address 7
SmartHMI
Measurement
Minimum calibration point distance (tool) in mm 8.0
Maximum calculation error in mm 5.0
Minimum calibration point distance (base) in mm 50.0
Minimum angle in ° 2.5
LBR_iiwa_14_R820_1 (Version: 2)
Media Flange Medien-Flansch elektrisch
Mounting orientation
Rotation about the Z axis in ° (A angle) 90.0
Rotation about the Y axis in ° (B angle) 0.0
Rotation about the X axis in ° (C angle) 90.0
Topology
Software
Configuration
Installation
Tasks
Console
Javadoc
0 items
Description
Resource
Path
Location
Type

KUKA Setup: Connecting Sunrise computer to KUKA Cabinet

A manual connection will need to be setup between the Sunrise PC and the KUKA Cabinet. Use the settings below.

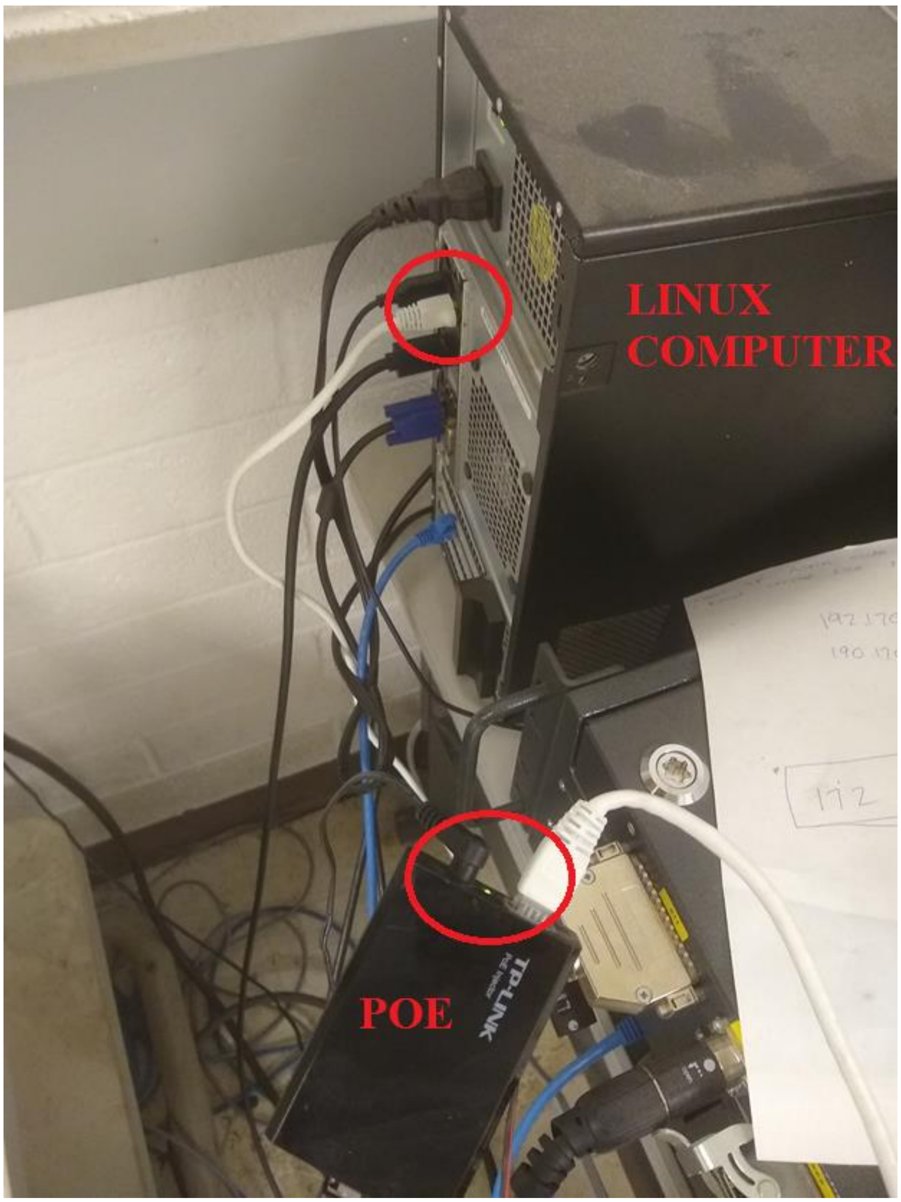


Booting the Linux/OpenSUSE Computer

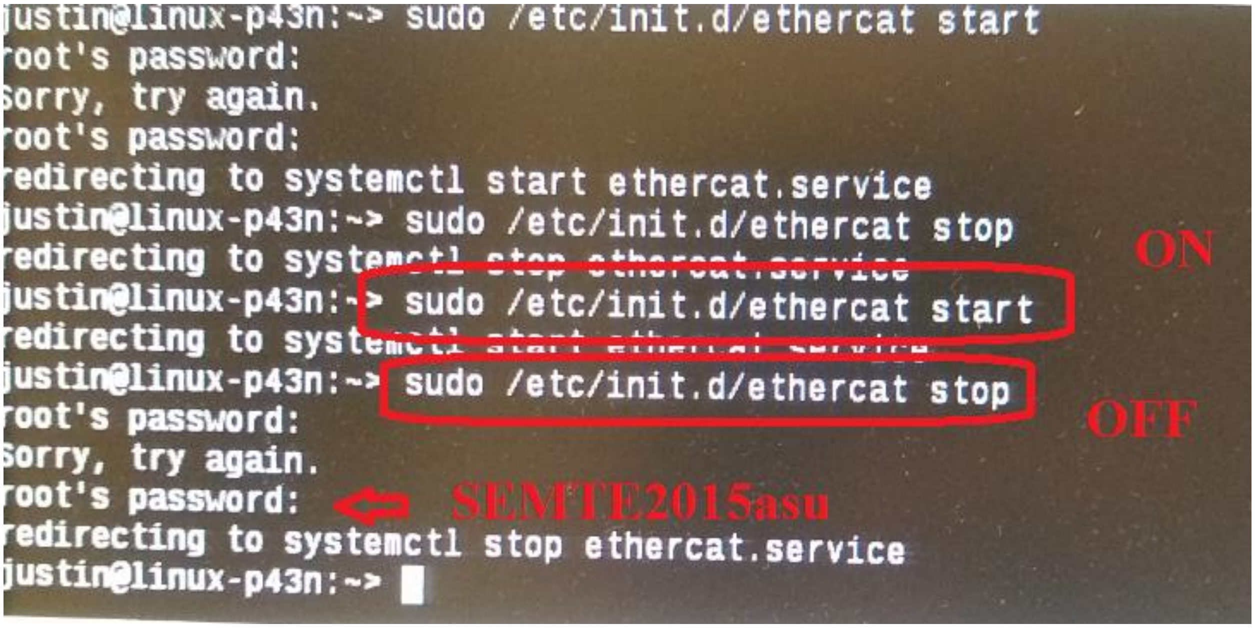
When booting the Linux machine, be sure to boot using the real-time kernel found under advance setup. This option is available immediately after turning the power on. This will allow the ATI 6-axis force sensor to stream in real time at 2 kHz.

Force Sensor Setup: ON/OFF

Make sure the force sensor is plugged into the top Ethernet port on the Linux computer and that the POE is powered on.

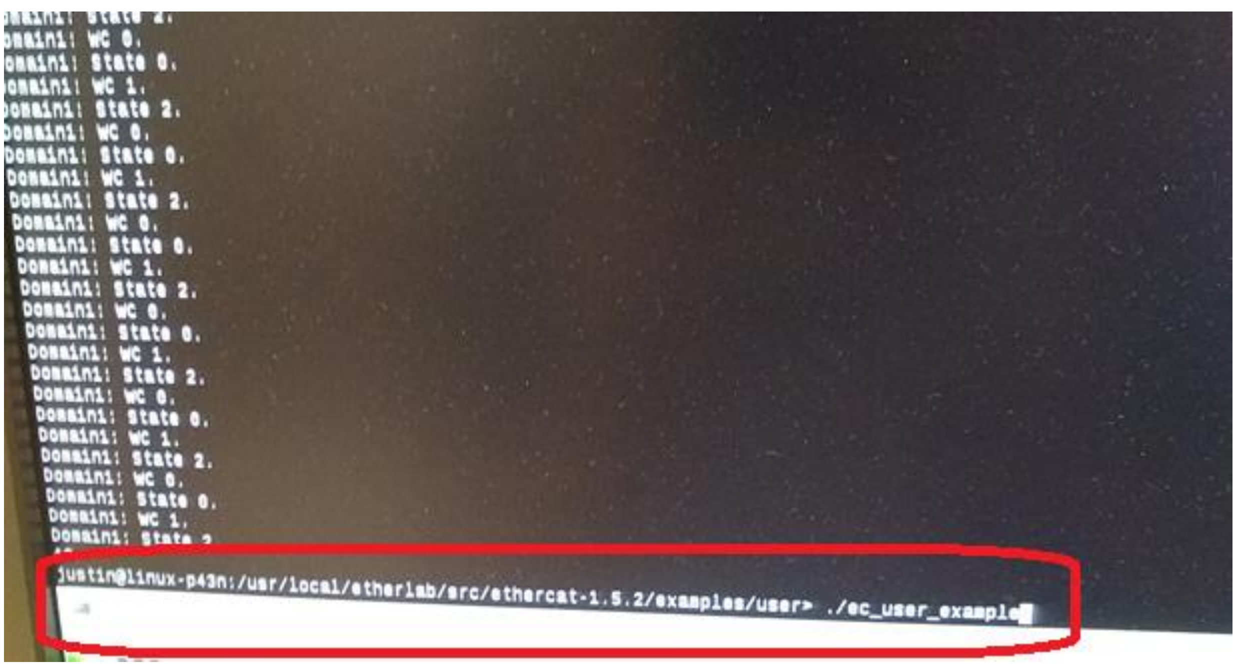


Open a terminal in the Linux computer and enter the following commands to turn the force sensor on/off. The system will prompt you for a password. Use “SEMTE2015asu”. The connection is working if the lights on the force sensors blink rapidly.



Force Sensor Setup: Streaming Data to Shared Memory

A code to stream the force sensor data to shared memory at 2 kHz is set up if you need it. The file and directory are shown below. Currently the only Fx and Fy data are streamed to shared memory, but this can be easily changed to Fz, Tx, Ty, Tz or all of them if needed.



Using the FRI

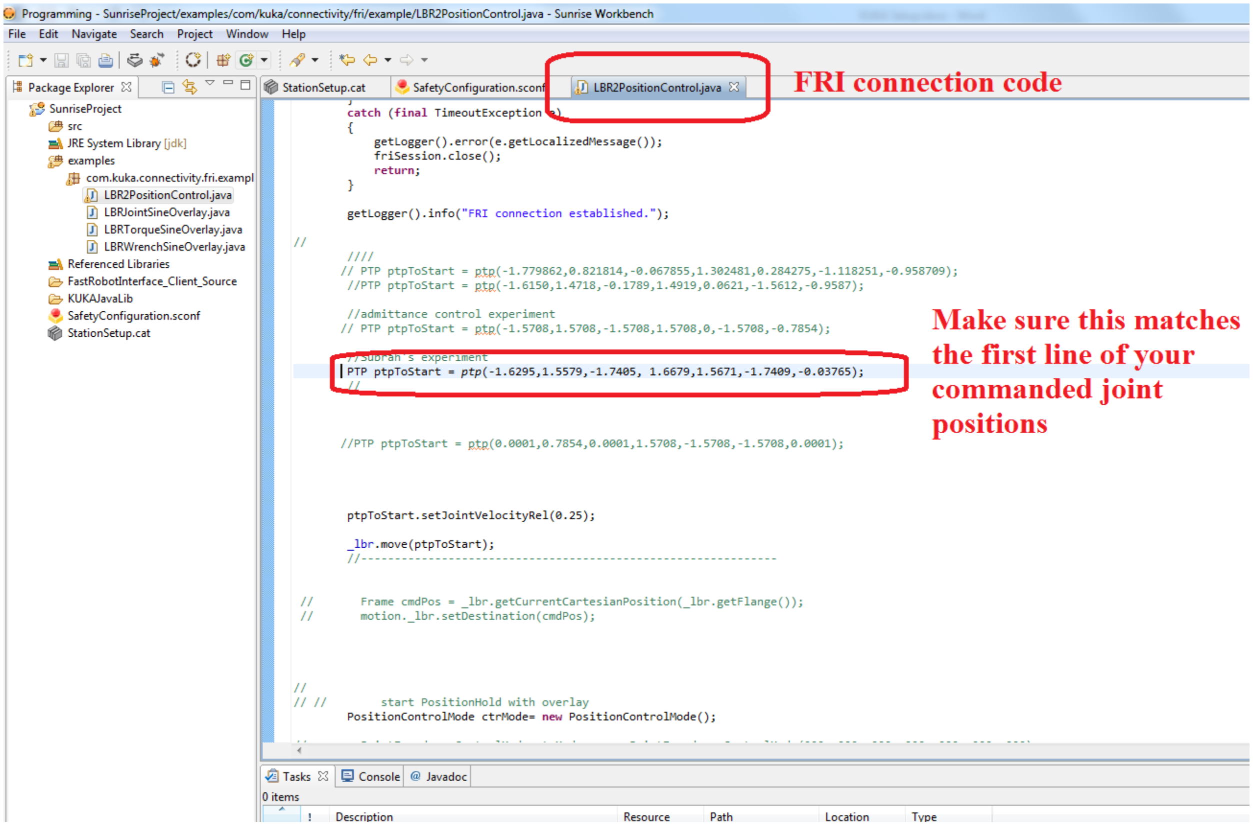
NOTE: do not attempt to use the FRI unless you have a clear understanding of robot kinematics. Logic errors can causes the robot to move at dangerous speeds that can damage the robot and/or operator. Safety features such as virtual workspace boundaries and joint velocity checks should be implemented in all FRI codes.

The Fast Robotic Interface (FRI) is a way of controlling the KUKA at high frequencies (1kHz) using any programming language you like (C++, Python, etc). The FRI connection accepts joint commands from Linux computer and passes them to the KUKA. Therefore, all calculations would need to either be completed ahead of time or done in real time (<1 ms) on the Linux computer. If the cycle time is greater than 1 ms, then the FRI connection will likely crash. Depending on your experiment, there are at least three options for commanding the robot using the FRI:

1. **Perform calculations ahead of time:** If possible, it is suggested that you use Matlab's Robot Toolbox to perform the kinematic calculations ahead of time and store the joint solutions in a txt file that can be read by the Linux computer. This will ensure that the cycle time does not exceed 1 ms.
2. **Treat the KUKA as a 6-DoF robot and solve the kinematics in real-time:** For experiments that don't need multiple kinematic solutions (such as obstacle avoidance), it is possible to treat the KUKA as a 6-DoF robot with a singular kinematic solution by ignoring the last joint. This means the Jacobian is square and therefore a single analytical solution can be found. This allows the kinematics to be solved very quickly (<1 ms) and thus ensure that the FRI does not crash.
3. **Solve the full 7-DoF KUKA kinematics in real time:** This solution is hard to do, but the most versatile. The full 7-DoF of the KUKA does not have a closed form solution, so a numeric solver must be used. Getting a numeric solver to converge consistently under 1 ms is not easy, but can be done. If this is your only option, good luck.

FRI Example code: Sunrise

The first step toward using the FRI is to ensure that the Sunrise code you are using to setup the FRI connection has the same starting joint position as the first line of your commanded positions via the FRI. The following code to connect the Linux computer to the FRI is available if you need it.



FRI Example Code: LInux

To help you create your first robot control code using the FRI, an example can be found on the Linux machine in the directory shown below.