Table of Contents

1. Brain node . . . 4

1.1. Repository . . . 4

1.2. Components . . . 4

2. Sensor node . . . 6

2.1. Repository . . . 6

2.2. Components . . . 6

3. SI node . . . 8

3.1. Repository . . . 8

3.2. Components . . . 8

4. Power node . . . 10

4.1. Repository . . . 10

4.2. Components . . . 10

5. Willy WRT . . . 12

5.1. Connection . . . 13

5.2. Physical . . . 13

5.3. VPN Tunnel . . . 13

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity
- Sponsors

Getting started

- Development Guide
- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node
- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain

- Sonar
- Lidar
- Localization and navigation
- Motor controller
- Joystick
- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab
- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity
- Sponsors

Getting started

- Development Guide
- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node
- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain
- Sonar
- Lidar
- Localization and navigation
- Motor controller
- Joystick
- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab

- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

1. Brain node

The brain node contains all the components for the main operation of Willy.

1.1. Repository

Windesheim-Willy/brain_node

1.2. Components

- ROS master
- Brain :numbered: :toc: :toclevels: 5 :icons: font

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity
- Sponsors

Getting started

- Development Guide

- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node
- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain
- Sonar
- Lidar
- Localization and navigation
- Motor controller
- Joystick
- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab
- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

2. Sensor node

The sensor node contains components to process sensor information and publish this information on a topic.

2.1. Repository

Windesheim-Willy/sensor_node

2.2. Components

- Sonar
- OpenMV
- IMU :numbered: :toc: :toclevels: 5 :icons: font

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity

- Sponsors

Getting started

- Development Guide
- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node
- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain
- Sonar
- Lidar
- Localization and navigation
- Motor controller
- Joystick

- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab
- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

3. SI node

The SI node contains all the components for social interaction of Willy.

3.1. Repository

Windesheim-Willy/si_node

3.2. Components

- Social interaction
- Speech recognition
- Speech :numbered: :toc: :toclevels: 5 :icons: font

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity
- Sponsors

Getting started

- Development Guide
- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node
- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain
- Sonar
- Lidar

- Localization and navigation
- Motor controller
- Joystick
- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab
- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

4. Power node

The power node contains all main components for driving of Willy.

4.1. Repository

Windesheim-Willy/power_node

4.2. Components

- Motor controller
- Human detection

- Brain
- Joystick
- Lidar
- Ros Master
- Navigation stack :numbered: :toc: :toclevels: 5 :icons: font

Welcome

Project Willy

- History of Willy
- Project Willy
- Publicity
- Sponsors

Getting started

- Development Guide
- Driving Willy
- Documentation

Build of Willy

- Design history
- Requirements
- Design reference
- Physical build
- Hardware

Robotic Operating System

- Introduction to ROS
- ROS Tutorials
- Multi master

Architecture

- Software Architecture
- Hardware Architecture
- Skylab Architecture
- ROS topic design

Hardware nodes

- sensor node
- si node
- power node

- WillyWRT

Components

- ROS master
- New ROS master on Lubuntu
- Brain
- Sonar
- Lidar
- Localization and navigation
- Motor controller
- Joystick
- Social interaction
- Speech
- Speech recognition

Skylab

- Setup Skylab
- Python scripts
- Webserver
- Functions of the webserver
- Skylab servers
- ROS installation on Ubuntu VMs in Skylab
- DNS,DHCP, pfSense & Ubuntu

Radeffect App

- Radeffect App

Lessons learned

- Todo & Advice
- Lessons Learned

Archive

- Previous Groups
- Research Archive

5. Willy WRT

In order to let all the hardware nodes communicate between eachother (and to Skylab) we used a OpenWRT router, named WillyWRT.

The router is installed on a Raspberry Pi and we used OpenWRT as the hardware is easily capable of running on a low powered device. We prefered to use another Raspberry Pi for this functionality, since next to its power requirements, it has a small formfactor and easily replacable aswell as the low purchase cost.

link::https://downloads.openwrt.org/releases/[Stable releases OpenWRT]

> The current stable release of OpenWRT 18.06.1 is compatible with the Raspberry Pi however, the Wireless/blutooth chip isn't supported. Therefor we used a Raspberry Pi 3B (with an older chipset).

5.1. Connection

We requested Windesheims ICT department to give WillyWRT Wireless acces to the internet, so all phereplas can communicate to the outside world. The MAC-address is whitelisted and in the password documentation you can find the SSID and password.

5.2. Physical

The Wireless interface connects to the SSID provided by Windesheim, the LAN interface connects to a switch and provides (static) DHCP leases to the various hardware on physical Willy. We also applied a costum etc/hosts entry to make sure the ROS master can always resolve the FQDNs and/or hardware.

> ROS communicates via the ROS master suggest to read up on this process aswell. Make sure the diffrent hardware nodes can communicate on DNS name and ip-adress. This can take a lot of time troubleshooting.

5.3. VPN Tunnel

Another reason for us to choose for OpenWRT is to use an openVPN tunnel between Skylabs PFsense firewall and Willys LAN. Since configuring openVPN on openWRT can be challenging we have also exported the working configuration. WillyWRT current configuration.