Getting Started (Part III)

Approximate Time Investment: 1 hour

Installing ROS2 Humble, Create Workspace, and Setup bashrc for Better Quality of Life
At this stage, we have exhausted the checks we can perform without ROS2. ROS2 is the critical middleware that we will use going forwards for working with the vehicle. It is assumed that the reader has watched the beginners guide in the prerequisites and has basic knowledge of ROS2 (creating packages, creating nodes, building a workspace, publishers/subscribers, topics, etc.)

1. Install ROS2 Humble by following the official installation guide. This step should be a repeat of the installation process from the prerequisite ROS2 beginner's guide by Robotics Back-End.
    a. ROS2 Humble for Beginners by Robotics Back-End (11 video series)
    b. Official ROS2 Humble Installation Guide

2. Install colcon for building ROS packages. Colcon is an open-source, command-line tool that automates the building, testing, and installation of multiple software packages.
    a. Open a new terminal and run the following command

```
sudo apt install python3-colcon-common-extensions
```

3. Create and build a ROS2 workspace by running the following commands. I am creating a folder called "ros2_ws" in the Home directory.

```
cd                  # this will bring us to the home directory
mkdir ros2_ws       # this will create a folder called "ros2_ws" which will be our workspace
cd ros2_ws
mkdir src
colcon build
```

Our workspace should successfully build and you should see a screen similar to that in the next figure.

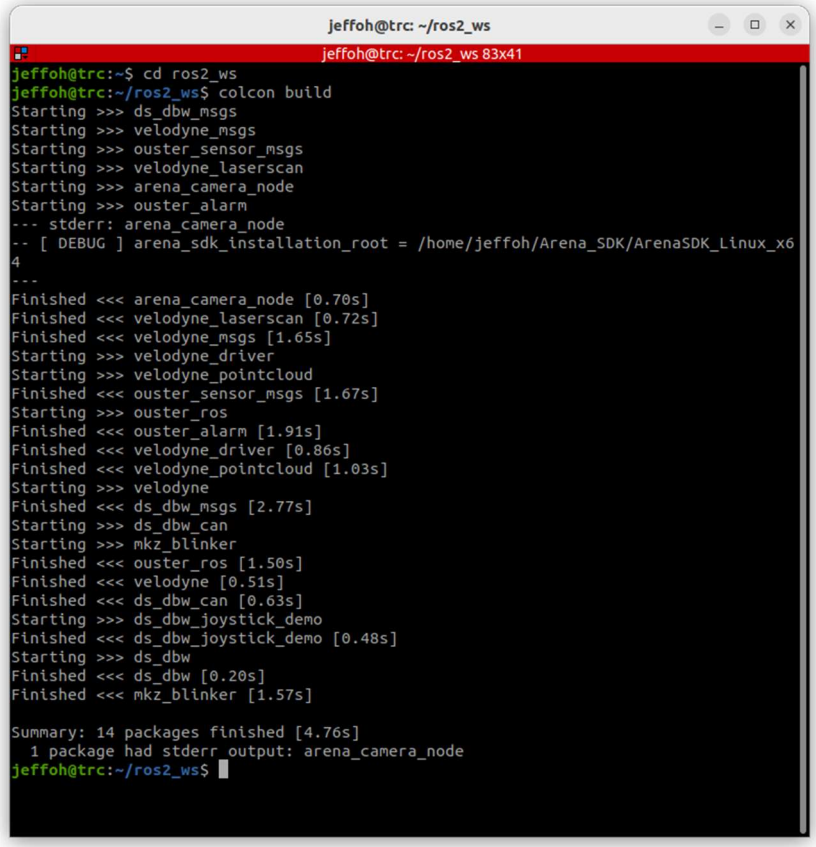


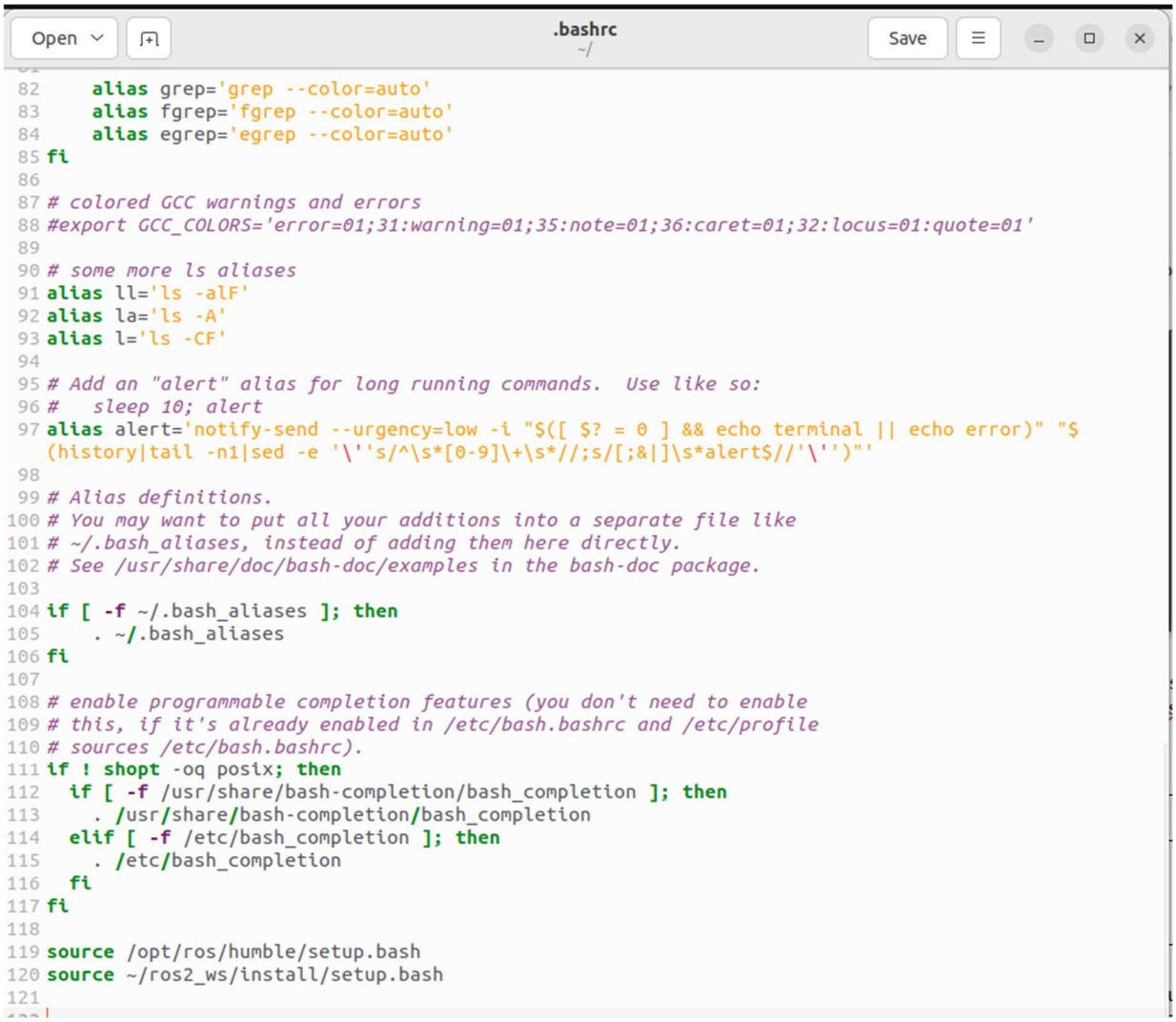
Figure 1: Our ros2_ws colcon build result

4. You may recall from the ROS2 beginner's guide that we must source the ROS2's setup.bash file and our workspace's setup.bash file every time we need to use ROS. We will edit our bashrc file so that those files are automatically sourced every time we open a new terminal.
    a. In the terminal open bashrc with the following command

```
gedit ~/.bashrc
```

    b. Inside bashrc, add the following lines at the bottom:

```
source /opt/ros/humble/setup.bash
source ~/ros2_ws/install/setup.bash   # this location may vary depending on where you created the workspace
```




```
    alias grep='grep --color=auto'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
fi

# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
#   sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$
(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

source /opt/ros/humble/setup.bash
source ~/ros2_ws/install/setup.bash
```

Figure 2: The bottom of your bashrc file should have the two source lines.