9.2 Keyboard control of ROS robot

1. Install teleop_twist_keyboard

Wiki：http://wiki.ros.org/teleop_twist_keyboard

Source code：https://github.com/ros-teleop/teleop_twist_keyboard

This feature pack can be installed directly into the system. If you use the official image of jetbot-mini, you can skip this step.

```
sudo apt-get install ros-melodic-teleop-twist-keyboard
```

2. start up

```
roslaunch jetbot_ros jetbotmini_keyboard.launch
```

3. control method

**Note: The initial speed is 0, you need to press [u] or [j] to accelerate*

key	Car	key	Speed change
【w】	Advance	【u】	Left wheel speed increase10%
【a】	Turn left	【i】	Left wheel speed reduction10%
【s】	Back	【j】	Right wheel speed increase10%
【d】	Turn right	【k】	Right wheel speed reduction10%

4. Code analysis

Mainly use select module, termios module and tty module

```
    import sys, select, termios, tty
```

-The select module is mainly used for socket communication.
-The termios module provides an IO-controlled POSIX call interface for tty
-The tty module is mainly used to change the mode of the file descriptor fd

Get current key information

```
def getKey():
    # tty.setraw():Change the file descriptor fd mode to raw; fileno(): returns
an integer file descriptor (fd)
    tty.setraw(sys.stdin.fileno())
    # select():Directly call the IO interface of the operating system; monitor
all file handles with fileno() method
    rlist, _, _ = select.select([sys.stdin], [], [], 0.1)
    # Read a byte of input stream
    if rlist: key = sys.stdin.read(1)
    else: key = ''
    # tcsetattr sets the tty attribute of the file descriptor fd from the
attribute
    termios.tcsetattr(sys.stdin, termios.TCSADRAIN, settings)
    return key
```

Control flow

```
            # Get current key information
            key = getKey()
            #print(key)
            # Key string to determine whether it is in the dictionary
            if key in moveBindings.keys():
                leftdir = moveBindings[key][0]
                rightdir = moveBindings[key][1]
            elif key in speedBindings.keys():
                leftspeed = leftspeed + speedBindings[key][0]
                if leftspeed >= 255:
                    leftspeed = 255
                elif leftspeed <= 0:
                    leftspeed = 0
                rightspeed = rightspeed + speedBindings[key][1]
                if rightspeed >= 255:
                    rightspeed = 255
                elif rightspeed <= 0:
                    rightspeed = 0
                #print(leftspeed)
                #print(rightspeed)
            bus.write_i2c_block_data(ADDRESS,0x01,
[leftdir,int(leftspeed),rightdir,int(rightspeed)])
            # Press q to exit control
            if key == 'q':
                bus.write_i2c_block_data(ADDRESS,0x01,[1,0,1,0])
                break
```

Code path：/home/jetson/workspace/catkin_ws/src/jetbot_ros/scripts/jetbotmini_keyboard.py