



# GoDroid

Application Notes

Motor Drive

v1.1



# Copyright Statement

Copyright in this document is owned by GoWarrior Community. Any person is hereby authorised to view, copy, print and distribute this document subject to the following conditions:

- The document may be used for informational purposes only
- The document may be used for non-commercial purposes only
- Any copy of this document or portion thereof must include this copyright notice

This document is provided "as is" without any warranty of any kind, either express or implied, statutory or otherwise; without limiting the foregoing, the warranties of satisfactory quality, fitness for a particular purpose or non-infringement are expressly excluded and under no circumstances will GoWarrior Community be liable for direct or indirect loss or damage of any kind, including loss of profit, revenue, goodwill or anticipated savings.

This document is intended only to assist the reader in the use of the product. GoWarrior Community makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, which is used at your own risk and should not be relied upon. The information could include technical inaccuracies or typographical errors. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property right is granted by this document.

The product described in this document is subject to continuous development and improvements. GoWarrior Community also reserves the right to make changes to the specifications and product description at any time without notice.

Third-party brands and names mentioned in this publication are for identification purpose only and may be the property of their respective owners.

Android™ is a registered trademark of Google Inc. Linux® is a registered trademark of Linus Torvalds. Microsoft® and Windows® are registered trademarks of Microsoft Corporation. Supply of this product does not convey a license nor imply a right under any patent, or any other industrial or intellectual property right of Google Inc., Linus Torvalds, and Microsoft Corporation to use this implementation in any finished end-user or ready-to-use final product. It is hereby notified that a license for such use is required from Google Inc., Linus Torvalds and Microsoft Corporation.

For the latest version of this document refer to:

[www.gowarriorosh.com](http://www.gowarriorosh.com)

**Copyright © 2016 GoWarrior Community All Rights Reserved.**

# Table of Contents

<b>Preface .....</b>	<b>1</b>
Overview .....	1
Audience .....	1
Applicable Products.....	1
Reference Documents.....	1
Conventions.....	2
How to Contact Us.....	3
 <b>1   Introduction .....</b>	 <b>4</b>
 <b>2   Controlling Servo Motor .....</b>	 <b>5</b>
2.1   Connect SG90 Servo Motor to TIGER Board .....	5
2.2   Sample Codes for Controlling.....	6
 <b>3   Controlling Stepper Motor .....</b>	 <b>8</b>
3.1   Connect 28BYJ-48 Stepper Motor with ULN2003 Control Board to TIGER Board 8	
3.2   Reference Codes for Controlling .....	9
 <b>Revision History .....</b>	 <b>12</b>
Document Change History.....	12
Software Changes.....	12

## List of Tables

Table 1. Typographical Conventions.....	2
Table 2. Symbol Conventions .....	2
Table 3. Document Change History .....	12
Table 4. Software Change History.....	12

## List of Figures

Figure 1. SG90 Servo Motor Wires.....	5
Figure 2. 28BYJ -48 Stepper Motor with ULN2003 Control Board .....	9

# Preface

## Overview

This manual provides a general overview and procedures about how to use two different motors with the GoWarrior TIGER Board.

- **Chapter 1: Introduction**

This chapter gives brief introduction about motor drive.

- **Chapter 2: Controlling Servo Motor**

This chapter provides a general overview and procedure to connect servo motor with the TIGER Board and drive it.

- **Chapter 3 Controlling Stepper Motor**

This chapter gives a step-by-step procedure on how to connect Stepper motor with TIGER Board and drive it.

## Audience

This manual is primarily written to provide complete guidance for those who wants to exploit GoWarrior TIGER Board, such as makers, tinkers, innovators, students, etc.

## Applicable Products

This manual is applicable for the GoWarrior TIGER Board.

## Reference Documents

N/A

## Conventions

### Typographical Conventions

Item	Format
codes, keyboard input commands, file names, equations, and math	Courier New, Size 10.5
Variables, code variables, and code comments	<i>Courier New, Size, Italic</i>
Menu item, buttons, tool names	Ebrima, Size 10.5, Bold e.g. Select USB Debugging
Screens, windows, dialog boxes, and tabs	<b>Ebrima, Size 10.5, Bold</b> <b>Enclosed in double quotation marks</b> e.g. Open the “Debug Configuration” dialog box

Table 1. Typographical Conventions

### Symbol Conventions




Item	Description
 <b>Caution</b>	Indicates a potential hazard or unsafe practice that, if not avoided, could result in data loss, device performance degradation, or other unpredictable results.
 <b>Note</b>	Indicates additional and supplemental information for the main contents.
 <b>Tip</b>	Indicates a suggestion that may help you solve a problem or save your time.

Table 2. Symbol Conventions

## How to Contact Us

Submit all your comments and error reports to the author at:

[info@gowarriorosh.com](mailto:info@gowarriorosh.com)

Tel: +886-2-8752-2000

Fax: +886-2-8751-1001

For questions regarding GoWarrior, contact our support team at the email listed below:

[support@gowarriorosh.com](mailto:support@gowarriorosh.com)

# 1 Introduction

This chapter provides a compact description to students or hobbyists to get started with driving different motors on the GoWarrior TIGER Board. The principle hardware required to run motors on a TIGER Board consists of the following components.

- A GoWarrior TIGER Board with GoDroid
- A kind of motor driver board
- One SG90 servo motor or 5-Wire "28BYJ-48" Stepper Motor
- Object orientated Python driver code.



## 2 Controlling Servo Motor

This chapter describes how to connect SG90 servo motor to TIGER Board and drive it.

### 2.1 Connect SG90 Servo Motor to TIGER Board

There are three wires with different colors for SG90 servo motor (Figure 1.), and the function of each wire:

- Brown wire is a ground wire
- Orange wire is a power wire
- Yellow wire is a signal wire.



**Figure 1. SG90 Servo Motor Wires**

Connect the signal wire to TIGER Board's pin 43, connect the power wire to TIGER Board's pin 2, and connect the ground wire to TIGER Board's pin 6.

## 2.2 Sample Codes for Controlling

After connection is done, you can control the SG90 servo motor. Here is an example for controlling it to transfer from 0 degrees to 180 degrees, and then comes back to 0 degrees continuously. The complete codes of the Python script are shown as below:

```
import RPi.GPIO as gpio

import time

import signal

import atexit

atexit.register(gpio.cleanup)

pin = 18

gpio.setmode(gpio.BCM)

gpio.setup(pin, gpio.OUT, initial=False)

p = gpio.PWM(pin, 50) #50HZ

p.start(0)

time.sleep(2)

while(True):

    for i in range(0,181,10):

        p.ChangeDutyCycle(2.5 + 10 * i/180)

        time.sleep(0.02)

        p.ChangeDutyCycle(0)

        time.sleep(0.2)

    for i in range(181,0,-10):

        p.ChangeDutyCycle(2.5 + 10 * i/180)

        time.sleep(0.02)
```

```
p.ChangeDutyCycle(0)  
time.sleep(0.2)
```

You need to save the codes as `servo.py`, you also need to run it as the super user. Also run it with the command `(sudo) python servo.py`.

# 3 Controlling Stepper Motor

This chapter describes how to connect SG90 servo motor to GoWarrior tiger board and drive it.

## 3.1 Connect 28BYJ-48 Stepper Motor with ULN2003 Control Board to TIGER Board

The motor connects to the controller board with a pre-supplied connector. The controller board has 4+2 pins that need to be connected to the header on the TIGER Board.

- 5V (P1-02)
- GND (P1-06)
- Inp1 (P2-03)
- Inp2 (P2-05)
- Inp3 (P2-07)
- Inp4 (P2-09)

The PX-XX references above represent the Pi header pins we used. These are defined in the Python example in chapter 3.2 in the StepPins list so if you use different pins be sure to update the Python list as well. You can also use other GPIO pins if required just remember to update your Python script.



**Figure 2. 28BYJ -48 Stepper Motor with ULN2003 Control Board**

## 3.2 Reference Codes for Controlling

The reference codes are show as below:

```
# Import required libraries
import sys
import time
import RPi.GPIO as GPIO

# Use BCM GPIO references
# instead of physical pin numbers
GPIO.setmode(GPIO.BCM)

# Define GPIO signals to use
# Physical pins 11,15,16,18
# GPIO17,GPIO22,GPIO23,GPIO24
#StepPins = [17,22,23,24]
StepPins = [18,26,22,20]

# Set all pins as output
for pin in StepPins:
    print "Setup pins"
    GPIO.setup(pin,GPIO.OUT)
    GPIO.output(pin, False)
```

```
# Define advanced sequence
# as shown in manufacturers datasheet
Seq = [[1,0,0,0],
        [1,1,0,0],
        [0,1,0,0],
        [0,1,1,0],
        [0,0,1,0],
        [0,0,1,1],
        [0,0,0,1],
        [1,0,0,1]]

StepCount = len(Seq)-1
StepDir = 2 # Set to 1 or 2 for clockwise
            # Set to -1 or -2 for anti-clockwise

# Read wait time from command line
if len(sys.argv)>1:
    WaitTime = int(sys.argv[1])/float(1000)
else:
    WaitTime = 10/float(1000)

# Initialise variables
StepCounter = 0

# Start main loop
while True:

    for pin in range(0, 4):
        xpin = StepPins[pin]
        print StepCounter
        print pin
        if Seq[StepCounter][pin]!=0:
            print " Step %i Enable %i" %(StepCounter,xpin)
            GPIO.output(xpin, True)
        else:
            GPIO.output(xpin, False)

    StepCounter += StepDir

    # If we reach the end of the sequence
    # start again
    if (StepCounter>=StepCount):
        StepCounter = 0
```

```
if (StepCounter<0):  
    StepCounter = StepCount  
  
# Wait before moving on  
time.sleep(WaitTime)
```

To specify a different wait time you can pass a number of milliseconds as an argument on the command line below:

```
python stepper.py 20
```

Where 20 represents the number of milliseconds

In this example the default wait time is set to 0.01 seconds (10 milliseconds). You can change this value to change the speed of rotation.

If the script runs too fast the motor controller can't keep up. This performance may vary depending on your motor and its controller.

# Revision History

## Document Change History

Revision	Changes	Date
v1.1	Updated document version to 1.1.	February 29, 2016
v1.0	Initial Release	September 07, 2015

**Table 3. Document Change History**

## Software Changes

Revision	Changes	Date
v1.1	Install and start GoDroid with a MicroSD card.	February 29, 2016
v1.0	Initial Release	September 07, 2015

**Table 4. Software Change History**





[www.gowarriorosh.com](http://www.gowarriorosh.com)

Headquarters

Tel: +886-2-8752-2000

Fax: +886-2-8751-1001

