

Motor Position Control

System Description

Bonden i Nol

hakanbrolin@hotmail.com

Rev PA1, 2017-12-30, Håkan Brolin

Table of Contents

1	Introduction.....	3
1.1	Purpose.....	3
1.2	Platform.....	3
1.3	Resources.....	3
2	Development tools.....	4
3	Components.....	5
3.1	DC motor.....	5
3.2	DC motor driver.....	5
4	MCU.....	6
4.1	SAM9-L9260.....	6
4.2	External connector EXT.....	6
4.3	External connector AEXT.....	6
5	PID controller test#1.....	7
5.1	Conditions.....	7
5.2	PID controller test#1.1.....	8
5.3	PID controller test#1.2.....	9
6	PID controller test#2.....	10
6.1	Conditions.....	10
6.2	PID controller test#2.1.....	11
6.3	PID controller test#2.2.....	12

1 Introduction

1.1 Purpose

This project investigates how to control the position of a DC motor shaft using PID controller.

1.2 Platform

Controller application is implemented in C and executes on a ARM9 32bit micro-controller.

A “bare metal” approach was selected to get low latency and overhead, and to achieve high real-time performance. This decision made the application more complex then when using an operating system (Linux). The learning curve is steeper and the lack of support of system resources makes it harder to get up and running in a small amount of time.

But the “bare metal” approach makes it possible to customize the application with a minimum use of system resources. It's also an interesting way to get to know the hardware and increases the knowledge about various aspects of the system, like start-up sequence, interrupt handling and more.

1.3 Resources

All resources for this project, including source code and documentation can be found at:

<https://github.com/emwhbr/dcmotor>

2 Development tools

TBD

3 Components

3.1 DC motor

Brushed DC (12V) motor with gearbox and quadrature encoder: Pololu, part#3240.

<https://www.pololu.com/product/3240>



Specifications:

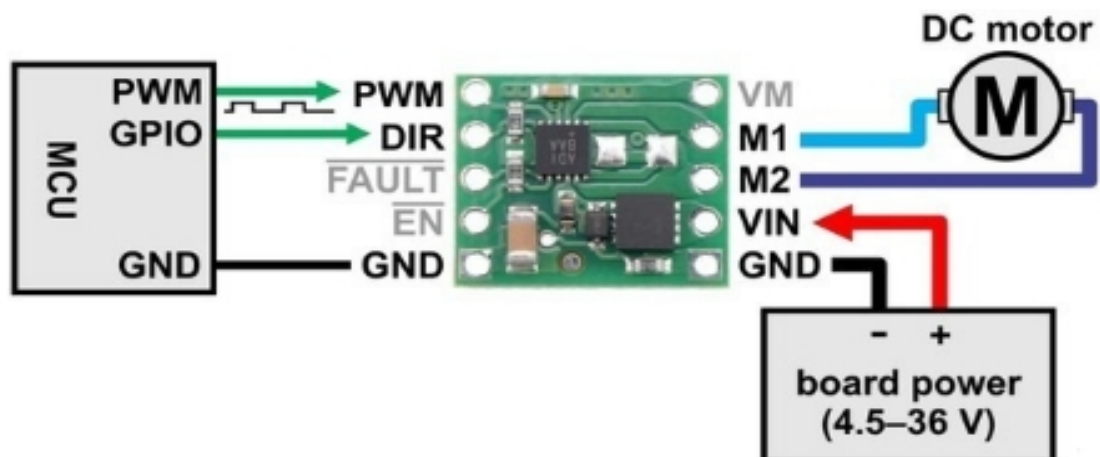
Free-run current: 200 mA (stall current: 2100 mA)
Free-run speed: 220 rpm (3.67 rev/s, 273ms/rev)
Gear ratio: 34.014:1
Encoder: 48 CPR, quadrature ($34.014 \times 48 = 1632,672$ counts per revolution)

3.2 DC motor driver

Motor driver carrier board for MAX14870: Pololu, part#2961.

<https://www.pololu.com/product/2961>

Application circuit:



TBD

4 MCU

4.1 SAM9-L9260

SAM9-L9260 development board (Olimex).

4.2 External connector EXT

TBD

Pin	MCU pin	MCU function	Application
1, 2	-	-	3.3V
4	-	-	5V
39, 40	-	-	GND
5	PB0	PIO Controller B Peripheral B TC3, TIOA3	PWM output to MAX14780 PWM input.
7	PB1	PIO Controller B PIO, output	Motor direction.
9	PB2	PIO Controller B PIO, output	Debug pin#1.
11	PB3	PIO Controller B PIO, output	Debug pin#2.
13		PIO Controller B PIO, input	Encoder A input. Interrupt enabled.
15		PIO Controller B PIO, input	Encoder B input. Interrupt enabled.

TBD

4.3 External connector AEXT

TBD

Pin	MCU pin	MCU function	Application
1	GNDANA	Analog ground	GND
2	-	-	-
3	PC0	PIO Controller C Peripheral A, AD0	Analog input channel.
4	-	-	-
5	ADVREF	Analog reference	3.3V
6	VDDANA	Analog power supply	3.3V

TBD

5 PID controller test#1

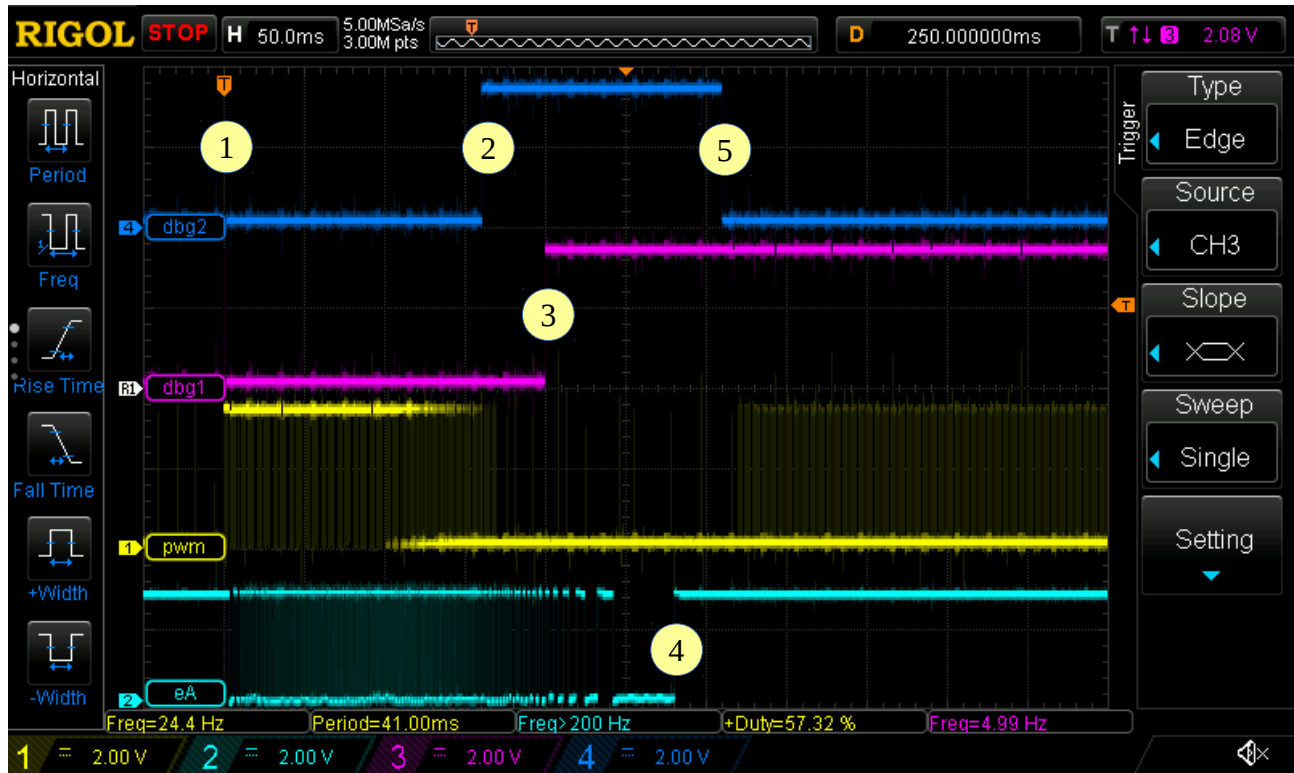
5.1 Conditions

- PID controller frequency: 100Hz (T=10ms)
- PID parameters: P=250, I=0, D=700

5.2 PID controller test#1.1

Conditions:

- Motor shaft position set point: $0x330 = 816$ (180°)



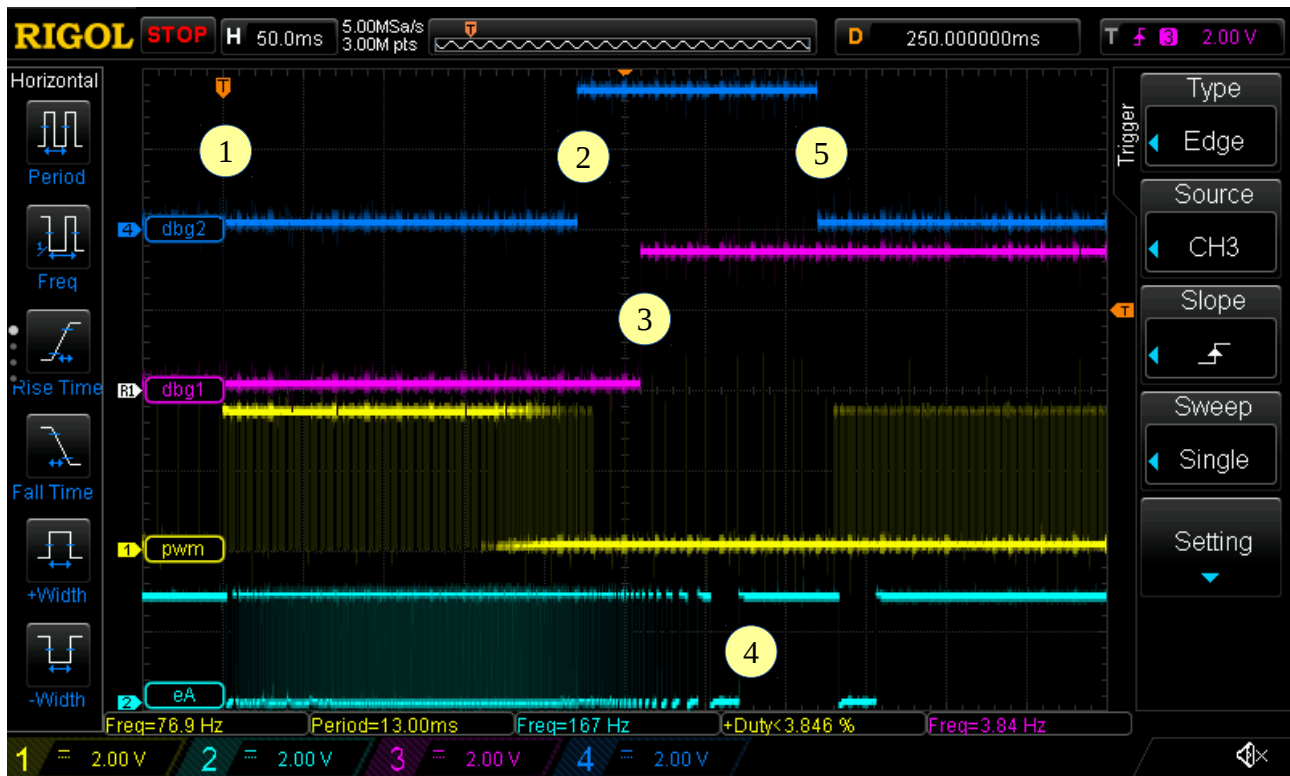
Description:

1. $T=0\text{ms}$
First PID controller cycle after calibration.
Motor shaft starts moving 3.2ms after PWM activation.
2. $T=160\text{ms}$
Entering stop state due to request to change motor direction.
Duration of stop state is 150ms.
3. $T=200\text{ms}$
Motor shaft position is within set point $\pm 1.5\%$.
This position keeps stable within limits.
4. $T=280\text{ms}$
Motor shaft completely stopped 120ms after entering stop state.
5. $T=310\text{ms}$
Leaving stop state with very low PWM duty, not enough to move motor shaft.

5.3 PID controller test#1.2

Conditions:

- Motor shaft position set point: $0x4c8 = 1224$ (270°)



Description:

1. $T=0ms$
First PID controller cycle after calibration.
Motor shaft starts moving 4ms after PWM activation.
2. $T=220ms$
Entering stop state due to request to change motor direction.
Duration of stop state is 150ms.
3. $T=260ms$
Motor shaft position is within set point $\pm 1.5\%$.
This position keeps stable within limits.
4. $T=320ms$
Motor shaft completely stopped 100ms after entering stop state.
5. $T=370ms$
Leaving stop state with low PWM duty, just enough to move motor shaft.

6 PID controller test#2

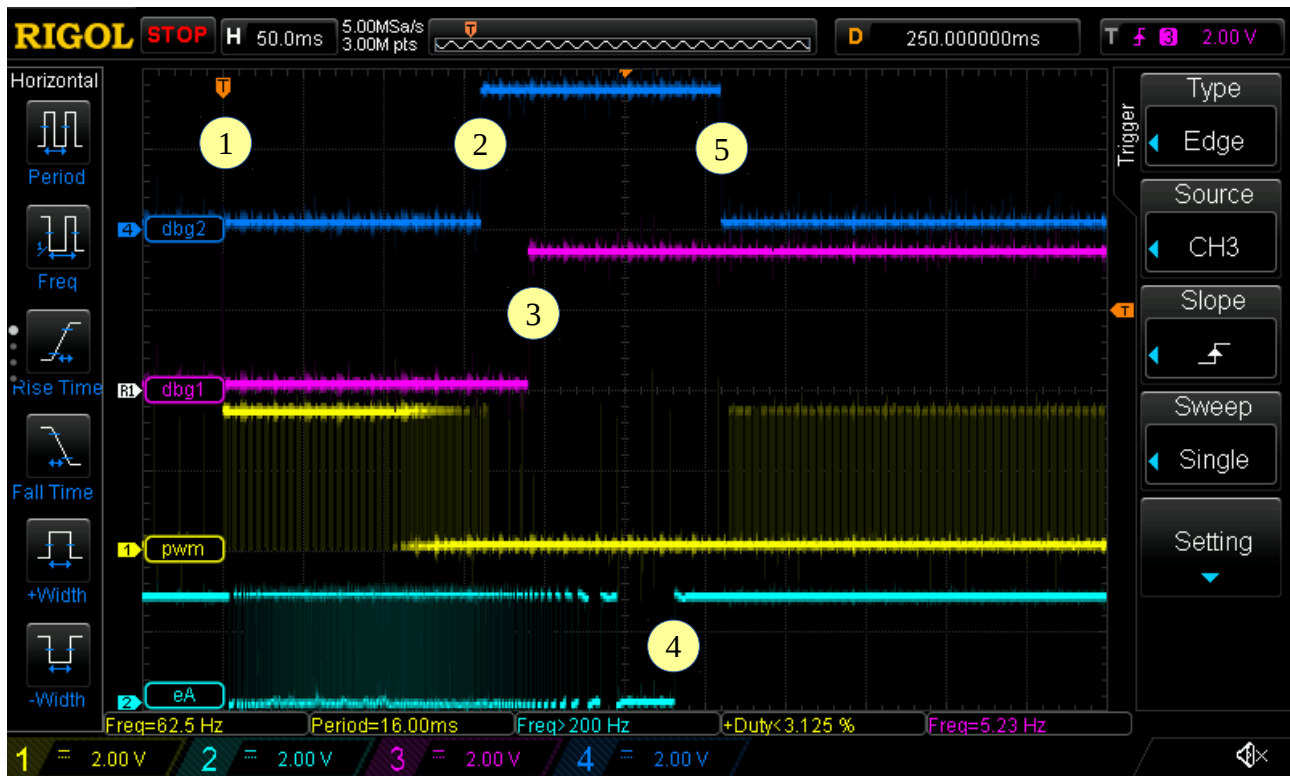
6.1 Conditions

- PID controller frequency: 200Hz (T=5ms)
- PID parameters: P=300, I=0, D=1650

6.2 PID controller test#2.1

Conditions:

- Motor shaft position set point: $0x330 = 816$ (180°)



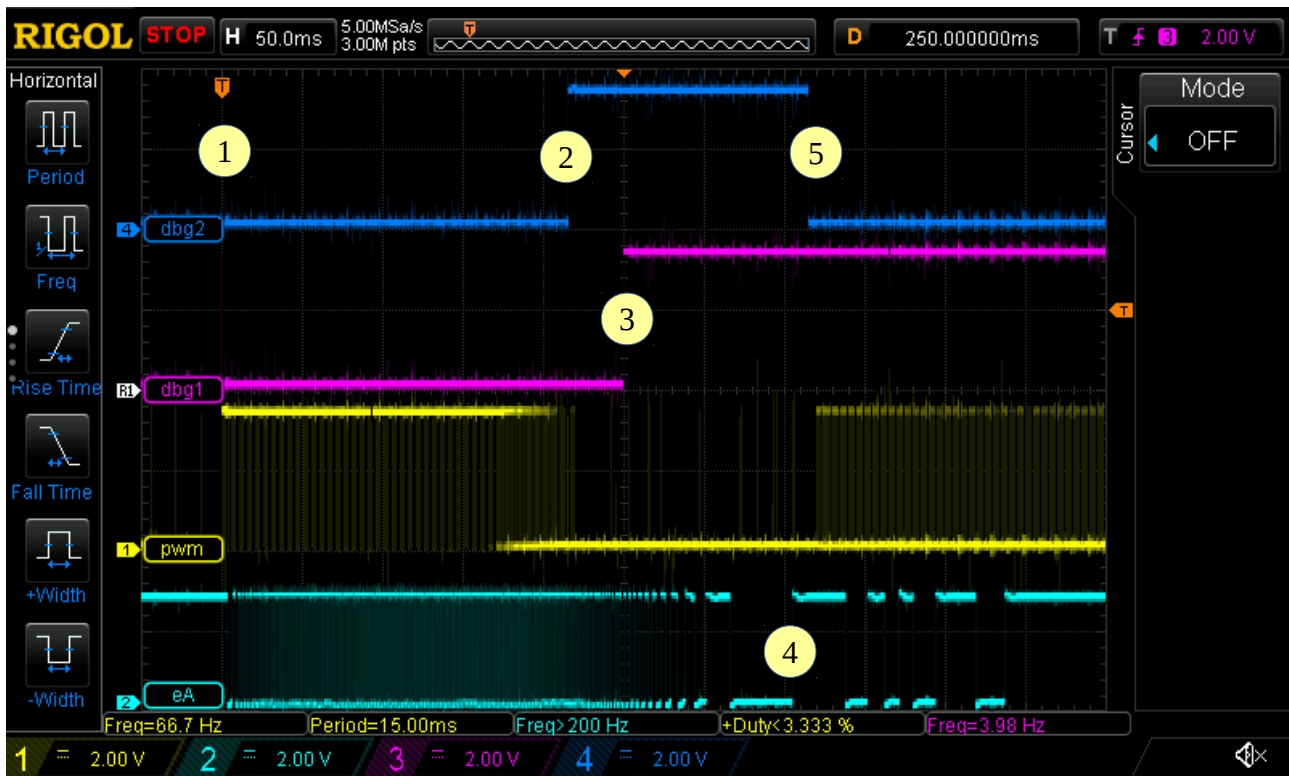
Description:

1. $T=0\text{ms}$
First PID controller cycle after calibration.
Motor shaft starts moving 4ms after PWM activation.
2. $T=160\text{ms}$
Entering stop state due to request to change motor direction.
Duration of stop state is 150ms.
3. $T=190\text{ms}$
Motor shaft position is within set point $\pm 1.5\%$.
This position keeps stable within limits.
4. $T=280\text{ms}$
Motor shaft completely stopped 120ms after entering stop state.
5. $T=310\text{ms}$
Leaving stop state with very low PWM duty, not enough to move motor shaft.

6.3 PID controller test#2.2

Conditions:

- Motor shaft position set point: $0x4c8 = 1224$ (270°)



Description:

1. $T=0ms$
First PID controller cycle after calibration.
Motor shaft starts moving 4ms after PWM activation.
2. $T=215ms$
Entering stop state due to request to change motor direction.
Duration of stop state is 150ms.
3. $T=250ms$
Motor shaft position is within set point $\pm 1.5\%$.
This position keeps stable within limits.
4. $T=355ms$
Motor shaft completely stopped 100ms after entering stop state.
5. $T=365ms$
Leaving stop state with low PWM duty, just enough to move motor shaft.