# Lab 2 Comprehension Quiz

Chengyu ZHANG

Q1.

When the proportional gain is small, there will be a huge error between the target value and the value controller output.

When the proportional gain increase, the error during steady state will be smaller and the position will approach the steady state much faster, but there will be overshooting during the first approach.

Q2.

The speed will be hugely reduced when the position is approaching close to the target value. The overshoot is disappeared.

Q3.

Because those complex operations are only for calculating the initial value of all these parameters, which means they will only be triggered when a new target value is set using the serial communication from computer. Therefore, the complex of these won't delay during the control.

Q4.

From the code, it is clear that computeNewSpeed() will be called repeatedly after a certain interval during the control process, and this function send the instructions to the stepper motor driver. If there is a delay in this function, the open loop control might be impossible to keep a certain constant speed. Therefore, it is necessary to reduce/avoid the complex operation.

Q5.

There was random delay in generating the pulse when using a timed loop, which lead to unsmooth output.

The timer comparation (when counter equal the setting value) triggers the interrupt. Therefore, when using the interrupt, the stepper motor will move one step exactly after the setting interval.

While this interrupt is triggered, the code will check if there are more steps need to go (if no, the flag in TIMSK1 for trigger the interrupt will be disabled), and call the moveOneStep() function to send a pulse, which triggers the stepper motor to move. After this, the register OCR1A will be set to new value of interval which calculated in pervious interrupt (the prescaler will be set to most reasonable mode too). After these actions, the computeNewStep() will be called to get next step interval.

The lengthy block's purpose is to enhance the precision of the timer interrupt when the p is small. The larger prescaler is, the less precision it can offered. Thus, the interval is measured to set a most proper prescaler for next step.

Q6.

The stepper motor will try to accelerate to that high speed but when the speed is higher than some value, the speed of stepper motor will suddenly stall.

Q7.

The servo motors need a closed loop to control the position of output, as the output speed is controlled by the voltage (most cases, PWM is used to simulate different voltage). The current output position is measured by the encoder and an error will be calculated using the target value. Thus, a PID control can be applied to control its motion.

The stepper motors use an open loop to control the output, for each step of the stepper motors are under certain angle and the move of the stepper motor is usually triggered by the pulse. Thus, the stepper motor usually calculates the interval of each step for required speed and hold the interval after sending one pulse.

## Listings of Programs

Arduino Programs

/LeibRampStepperCZ/ LeibRampStepperCZ.ino

/PIDClosedLoopCZ/ PIDClosedLoopCZ.ino

/ProportionalClosedLoopCZ/ ProportionalClosedLoopCZ.ino

C programs for Unit test

/PureC/ LeibRampStepperCZ.c

/PureC/ SimplisticRampStepper.c