

# **Implementation of Conventional and Neural Controllers Using Position and Velocity Feedback**

System Level Block Diagram

By:  
Christopher Spevacek  
and  
Manfred Meissner

Advisor:  
Dr. Gary Dempsey

### Overview:

Our project objective is to design and evaluate different controllers for a small robot arm-motor platform from Quanser Consulting. Our main effort will be to work in a software environment on a 200 MHz or higher Pentium Computer to develop the controllers and signal processing algorithms in C language. An internal A/D and D/A converter card is connected to the external plant as shown in Fig. 1. The plant consists of an amplifier, DC motor assembly, external gear train, external load, and potentiometer for the position sensor (also from Quanser Consulting). The feedback signal will be passed through an antialiasing filter, to the A/D converter, into the computer. The feedback voltage signal is proportional to the position of the robotic arm. The arm position, output position on Fig. 1, is the primary output of the system, although arm velocity will also be of interest.

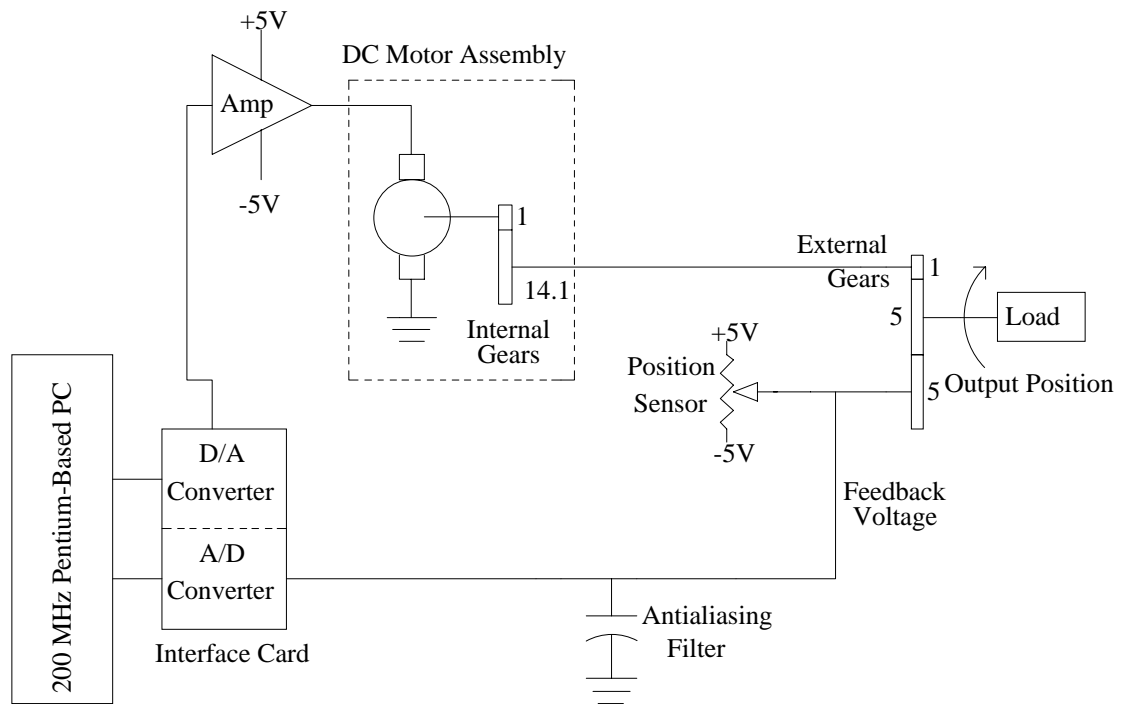


Fig. 1. Low Level System Block Diagram

### Hardware:

The high-level system block diagram in Fig. 2 shows the main parts of the project. The robotic arm system is expanded in Fig. 1. It consists of mechanical devices and the circuitry to convert the input and output of the A/D and D/A converters to the proper analog signals. The D/A converter converts the PC signals to analog signals that will perform actions on the robot arm. The position sensor, where a velocity feedback loop will be added later, sends the position of the robot arm through an A/D converter back to

the PC. The Pentium-based PC will perform all the calculations, update display information, and generate the command signal. If the optional joystick is present in the system, the command signal will be generated by the user's interactions with the joystick. The programs will be written in C and used under windows. All the mechanical components are ready to use and Quanser Consulting supplied some PC programs.

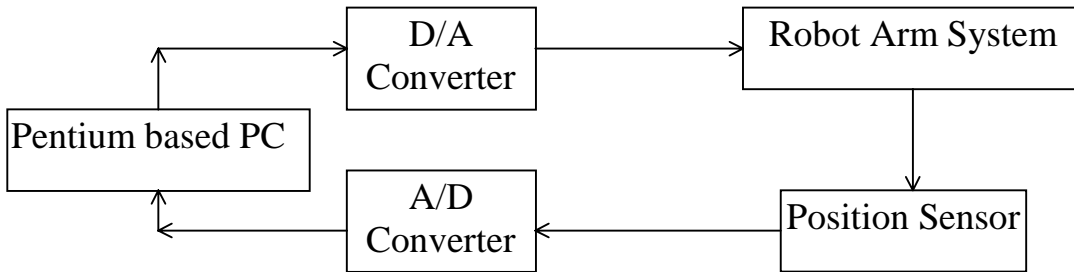


Fig. 2. High Level Block Diagram

**Modes of Operation:**

Two modes of operation will be incorporated into the product. The first mode is an option of connecting a joystick to one A/D channel and will allow the user to control (command) the robot arm position. An external digital input can be used to signal the software that a joystick is present. The other mode (default) will use an internal software command signal to control robot arm position. The user via keyboard will be able to change the set point (desired final position) and the slope of the command signal (velocity).

The signals that are important in this system are described below. They are separated into three categories (1) inputs, (2) outputs, and (3) internal signals to the computer that are important for controller evaluation and testing.

(1) Input signals to computer:

The letters in brackets refer to the letters in Fig. 3:

Signal	Description
Voltage from position sensor (proportional to position)[f]	Used to help create the new control signal
Voltage from joystick	Used to create the command signal[a]
Digital input	Used to signal if joystick is present

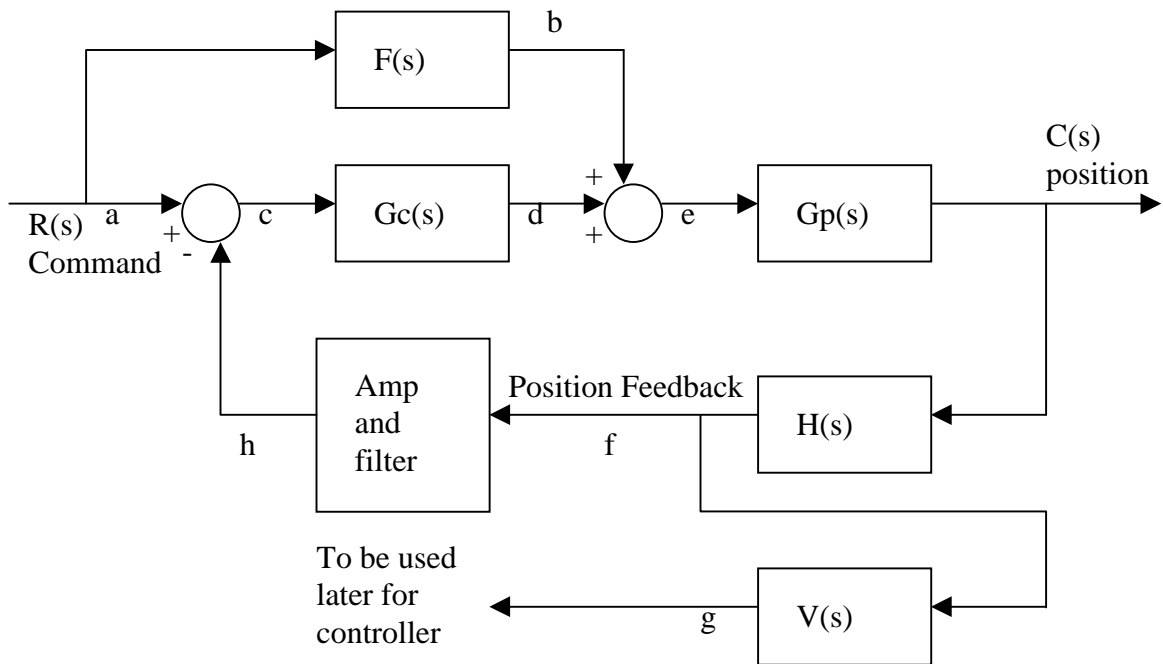
(2) Output signals from computer:

The letters in brackets refer to the letters in Fig. 3:

Signal	Description
Control (Actuating) Signal[e]	Used to drive the robot arm
Calculated velocity from robot arm[g]	Used to help create the new control signal

(3) Internal signals (in software, each will be displayed on computer monitor):  
 The letters in brackets refer to the letters in Fig. 3:

Signal	Description
Command Signal[a]	Desired signal of robot arm position
Feed-Forward Signal[b]	The signal output from the feed-forward compensator
Error Signal[c]	The difference between the desired position and the calculated actual position
Conventional Controller Signal[d]	The signal output from the PID-type controller
Control (Actuating) Signal[e]	The signal created by the feed forward signal and the PID controller signal
Filtered and amplified position sensor output signal[h]	The filtered and amplified position signal
Calculated velocity signal[g]	The signal that is calculated by using the filtered position output signal



$F(s)$  is the feed-forward compensator implemented in software

$G_c(s)$  is the PID-type controller implemented in software

$G_p(s)$  is the plant this is part of the hardware

$H(s)$  is the position sensor this is part of the hardware

Amp is an amplifier part of the hardware

$V(s)$  is an algorithm to be determined for velocity feedback

$C(s)$  robot arm position output

$R(s)$  command generated in software

**Fig.3. Control Block Diagram**

**Software:**

In Fig. 4, the software flowchart is shown. In the table below the blocks are listed and described.

Blocks	Description
Initialization	Interrupts will be setup to generate a 200Hz sampling (50ms).
Interrupt Service routine and Performing of PID controller	Send signals from the calculated values of the main program and interrupt service routine to the robot arm and to the monitor at user specified times.
Main Program	Will calculate values for the display, check the keyboard, read the joystick, and generate the command signal.
Keyboard	Gives the users a choice of which signals are to be displayed. Type of controller can also be selected.
Display	Will show all internal and external signals chosen by the user.
Joystick Check	There is the possibility that the robot arm can be driven by a joystick or by PC commands. The connection will be checked and if yes the joystick position will be read and saved for the interrupt service routine. If not, the PC will generate the command signal.
Generate Command Signal	This option occurs when the joystick is not connected. The PC will calculate the command signal and save its value for the interrupt service routine.

# Pentium-based PC

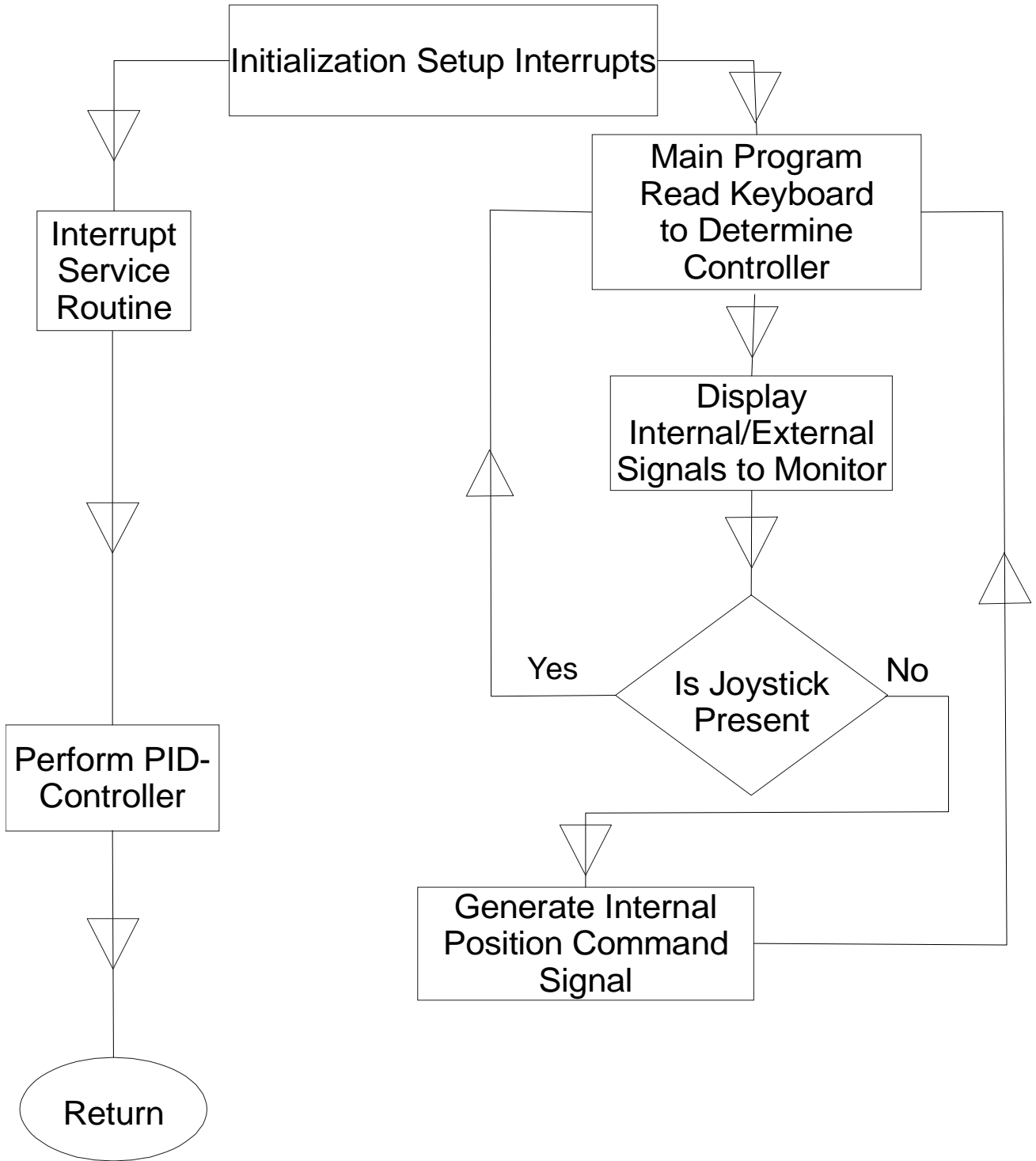


Fig. 4. Software Flowchart