5 Robot Programming Packages

5.1 Introduction

Essentially a robot is an assembly of mechanical actuators and sensors which are driven by their respective control circuitry, for example, the driving circuitry of a servo motor, the amplifier circuits of a sonar device, the electronic receptors of a camera, etc. Ultimately an entire program drives this hardware. As a programmer, however, one would not be as interested in wasting time and energy in dealing with the intricacies of the low-level functioning of the robot, as one should be in trying to manipulate the behavior of the robot at a higher level, guided by the task. Experimental robots usually have microcontroller driven hardware circuitry and the microcontroller takes care of the low-level device control. Besides that, it needs to act as the interface between the robot actuators and sensors and the user programs. The operating system of the robot serves this purpose, by interpreting the programs running on the microcomputer of the robot and mapping them into a sequence of instructions for the microcontroller. Similarly the feedback by the microcontroller is transferred to the user programs.

Here the programmer again faces the problem of knowing and repeatedly using code or commands for interacting with the operating system of the microcontroller. To make the task simpler for the programmer, the basic functionality is wrapped into classes or functions and is made available to the programmer in the form of new programming languages or libraries for existing languages such as C++ or JAVA. Some examples are cited here such as the ARIA library (for C++), the Saphira and Colbert programming environments to handle motor and sensor functions, the SVS library (for C and C++) to handle camera functions, the BotSpeak library (for C) for speech recognition and synthesis. As C++ is a popular and powerful OOP language, the programs discussed in this book have been tested in C++. The ARIA library for C++ has been used to implement modules to control the movement of the robot as well as its gripper. The
SVS classes have been used to interface the camera, and the BotSpeak library functions have been used to synthesize voice feedback from the robot. This chapter discusses the tools required for robot programming. This should facilitate better understanding of the codes that are presented in the subsequent chapters.

5.2 Robot Hardware and Software Resources

The Pioneer 2-DX ActivMedia mobile robot shown in Fig. 5.1, contains basic components for sensing and navigation in a real-world environment. It also includes battery power, drive motors and wheels, position–speed encoders, integrated sensors and accessories like a gripper and stereo camera. The robot is controlled by an onboard microcontroller and robot server software [Saphira, 1999]. Pioneer 2-DX also contains an addressable I/O bus for 16 devices, two RS-232 serial ports, eight digital I/O ports, and five A/D ports, which are accessible through a common application interface to the robot server software, Pioneer 2 Operating System (P2OS). The weight of Pioneer 2-DX is 9 kg and it can carry extra payload of up to 23 kg.

Fig. 5.1. Pioneer 2-DX from ActivMedia Robotic LLC, USA