## Val3 Implementation

To gain access to the set of VAL3 alter commands, modifications were made to the Robot Raconteur (RR) service to pass the desired joint angles to the TX-40 through a socket. The VAL3 program listens and interprets the commands received. Below is the VAL3 psuedocode.

**Code Snippet - Val3**

```
begin

  //Gripper DIO Link
  //Socket Link

  do
    inStr = socket //Get Data from socket

    if inStr == "cgrip"
     //Close Gripper
    elseIf inStr == "ogrip"
     //Open Gripper
    elseIf inStr =="stop"
      //Stop Robot
    else

      mMdesc.accel = 5
      mMdesc.decel = 5
      mMdesc.blend = joint
      mMdesc.leave = 1
      mMdesc.reach = 5
                  ...
      {  Parse Incoming Joint Angle  }
                  ...

      if(isEmpty())
        movej(jJoint,gripper,mMdesc)
        alterEnd()
      else
        alterBegin(world,mMdesc)
        alterMovej(jJoint,gripper,mMdesc)
      endIf

    endIf
    delay(0)
  until false

end
```

The robot's "movement parameters" were tuned to help improve the fluidity of the robot's motion. Specifically, the acceleration and deceleration of the robot was decreased to 5% the nominal value, along with the enabling of joint blending. For our purposes, only a small amount of blending was required to improve the robot's smoothness while moving. Too much blending caused the robot to overshoot its desired position.

Initially, when the robot's command queue is empty, the 'movej' command is issued. As the robot's desired path is updated, the 'alterMovej' command is used to change the robot's path without stopping translation. The VAL3 script continues to use the alter command until the robot's queue is empty. Once the queue is cleared, the alter command is concluded by issuing the 'alterEnd' command. Issuing 'alterEnd' before the robot has altered its path results in choppy motion.

## Summary

By tuning the robot's movement parameters and using the alter command set, the fluidity of the robot's motion was greatly improved.