8.ANEXOS

8.ANEXOS

```
%PUMA560    Load kinematic and dynamic data for a Puma 560 manipulator
%
%       Defines the matrix 'p560' which describes the kinematic and dynamic
%       characterstics of a Unimation Puma 560 manipulator.
%       Specifies armature inertia and gear ratios.
%
%       See also DH, DYN, TWOLINK, STANFORD.

%
% Notes:
%   -   the value of m1 is given as 0 here.  Armstrong found no value for it
%       and it does not appear in the equation for tau1 after the substituion
%       is made to inertia about link frame rather than COG frame.
% updated:
%       2/8/95  changed D3 to 150.05mm which is closer to data from Lee, AKB86 and Tarn
%               fixed errors in COG for links 2 and 3
%       29/1/91 to agree with data from Armstrong etal.  Due to their use
%               of modified D&H params, some of the offsets Ai, Di are
%               offset, and for links 3-5 swap Y and Z axes.
%       14/2/91 to use Paul's value of link twist (alpha) to be consistant
%               with ARCL.  This is the -ve of Lee's values, which means the
%               zero angle position is a righty for Paul, and lefty for Lee.
%               Note that gravity load torque is the motor torque necessary
%               to keep the joint static, and is thus -ve of the gravity
%               caused torque.
%
%       8/95    fix bugs in COG data for Puma 560. This led to signficant errors in
%               inertia of joint 1.
%

%       Copright (C) Peter Corke 1990

% alpha     A       theta   D       sigma   m       rx      ry      rz      Ixx     Iyy
            Izz     Ixy     Iyz     Ixz     Jm      G       B       Tc+     Tc-
p560 = [
pi/2    0       0       0       0       0       0       0       0       0       0.35    0
        0       0       0       200e-6  -62.6111        1.48e-3 .395    -.435
0       .4318   0       0       0       17.4    -.3638  .006    .2275   .13     .524    .539
        0       0       0       200e-6  107.815         .817e-3 .126    -.071
-pi/2   .0203   0       .15005  0       4.8     -.0203  -.0141  .070    .066    .086    .0125
        0       0       0       200e-6  -53.7063        1.38e-3 .132    -.105
pi/2    0       0       .4318   0       0.82    0       .019    0       1.8e-3  1.3e-3  1.8e-3
        0       0       0       33e-6   76.0364         71.2e-6 11.2e-3 -16.9e-3
-pi/2   0       0       0       0       0.34    0       0       0       .3e-3   .4e-3   .3e-3
        0       0       0       33e-6   71.923  82.6e-6 9.26e-3 -14.5e-3
```

```
0	0	0	0	0	.09	0	0	.032	.15e-3	.15e-3	.04e-3
	0	0	0	33e-6	76.686	36.7e-6	3.96e-3	-10.5e-3
];

%
% some useful poses
%
qz = [0 0 0 0 0 0];		% zero angles, L shaped pose
qr = [0 pi/2 -pi/2 0 0 0];	% ready pose, arm up
qstretch = [0 0 -pi/2 0 0 0];
```

```
%TR2ROTVEC Convert a homogeneous transform matrix to a rotation about an axis
%
%	[VEC ANGLE] = TR2ROTVEC(TR) returns a rotation pair (vector and angle corresponding
%	to the rotational part of the homogeneous transform TR.
%
%	See also  ROTVEC

%	Copright (C) A. Romeo 2012
function [vec,angle] = tr2rotvec(m)

	vec = zeros(1,3);

	angle = atan(sqrt((m(3,2)-m(2,3))^2 + (m(1,3)-m(3,1))^2 + (m(2,1)-m(1,2))^2) / (m(1,1) + m(2,2) + m(3,3) - 1));
	if abs(angle) > eps
      senangle=sin(angle);
		vec(1) = (m(3,2)-m(2,3))/(2*senangle);
		vec(2) = (m(1,3)-m(3,1))/(2*senangle);
		vec(3) = (m(2,1)-m(1,2))/(2*senangle);
	end
```

```
%PLOTBOT	Graphical robot animation
%
%	PLOTBOT(DH, Q)
%	PLOTBOT(DH, Q, OPT)
%
%	produces a graphical animation of a robot from a
%	description of the kinematics, DH, and a joint trajectory Q.
%	For an n-axis manipulator Q is mxn for an m-point trajectory.
%
%	OPT is a character string which may contain the following letters:
%		l	leave trail, that is, dont erase the previous pose
%		w	don't draw the wrist axes
```

```
%		r	repeat, play the animation 50 times
%		b<value>	set the height of the robot's base in Z.  Must
%			be the last argument.
%
%	See also FKINE, DH.

%	Copright (C) Peter Corke 1993
%  Cambio clg por clf Sept-2000 Josechu Guerrero

% MOD.HISTORY
%	12/94	make axis scaling adjust to robot kinematic params
function plotbot(dh, q, opt)
	np = numrows(q);
	n = numrows(dh);


	if numcols(q) ~= n,
		error('Insufficient columns in q')
	end

	erasemode = 'xor';
	wrist = 1;
	repeat = 1;
	base = 0.0;

	%
	% options
	%
	if nargin == 3,
		mopt = size(opt,2);
		for i=1:mopt,
			if (opt(i) == 'l'), erasemode = 'none'; end;
			if (opt(i) == 'w'), wrist = 0; end;
			if (opt(i) == 'r'), repeat = 50; end;
			if (opt(i) == 'b'),
					base = str2num(opt(i+1:mopt));
					break;
				end;
		end;
	end;
	%
	% simple heuristic to figure the maximum reach of the robot
	%
	reach = sum(abs(dh(:,2))) + sum(abs(dh(:,4)));

	%
	% setup an axis in which to animate the robot
	%
```

```
clf
axis([-reach reach -reach reach -reach reach]);
figure(gcf);                    % bring to the top
xlabel('X')
ylabel('Y')
zlabel('Z')
set(gca, 'drawmode', 'fast');
grid
line('xdata', [0;0], 'ydata', [0;0], 'zdata', [-reach;base], 'color', 'magenta');

% create a line which we will
% subsequently modify.  Set erase mode to xor for fast
% update
%
hr = line('color', 'yellow', 'erasemode', erasemode);
if wrist,
        hx = line('xdata', [0;0], 'ydata', [0;0], 'zdata', [0;base], ...
         'color', 'red', 'erasemode', 'xor');
        hy = line('xdata', [0;0], 'ydata', [0;0], 'zdata', [0;base], ...
                'color', 'green', 'erasemode', 'xor');
        hz = line('xdata', [0;0], 'ydata', [0;0], 'zdata', [0;base], ...
                'color', 'blue', 'erasemode', 'xor');

        mag = reach/10;
end

for r=1:repeat,
   for p=1:np,
        % for every trajectory point

        x = 0;
        y = 0;
        z = base;
        % compute the link transforms, and record the origin of each frame
        % for the animation.
        t = [eye(3,3) [0;0;base];0 0 0 1];
        for j=1:n,
                t = t * linktran(dh(j,:), q(p,j));
                x = [x; t(1,4)];
                y = [y; t(2,4)];
                z = [z; t(3,4)];
        end

        if wrist,
                %
                % compute the wrist axes
                %
                xv = t*[mag;0;0;1];
```

```
			yv = t*[0;mag;0;1];
			zv = t*[0;0;mag;1];

			%
			% update the line segments, wrist axis and links
			%
			set(hx,'xdata',[t(1,4) xv(1)], 'ydata', [t(2,4) xv(2)], ...
				'zdata', [t(3,4) xv(3)]);
			set(hy,'xdata',[t(1,4) yv(1)], 'ydata', [t(2,4) yv(2)], ...
				'zdata', [t(3,4) yv(3)]);
			set(hz,'xdata',[t(1,4) zv(1)], 'ydata', [t(2,4) zv(2)], ...
				'zdata', [t(3,4) zv(3)]);
		end

		set(hr,'xdata', x, 'ydata', y, 'zdata', z);

		drawnow
	   end
	end
```

```
%JACOBN	Compute manipulator Jacobian in end-effector frame
%
%	JACOBN(DH, Q) returns a Jacobian matrix for the current pose Q.
%
%	The manipulator Jacobian matrix maps differential changes in joint space
%	to differential Cartesian motion of the end-effector.
%			dX = J dQ
%
%	This function uses the technique of
%		Paul, Shimano, Mayer
%		Differential Kinematic Control Equations for Simple Manipulators
%		IEEE SMC 11(6) 1981
%		pp. 456-460
%
%	For an n-axis manipulator the Jacobian is a 6 x n matrix.
%
%	See also DIFF2TR, TR2DIFF, DIFF

%	Copyright (C) Peter Corke 1993
function J = jacobn(dh, q)
	J = [];
	n = numrows(dh);

	T = eye(4,4);
	for i=n:-1:1,
		T = linktran(dh(i,:), q(i)) * T;
		if dh(i,5) == 0,
```

```
                    % revolute axis
                    d = [     -T(1,1)*T(2,4)+T(2,1)*T(1,4)
                              -T(1,2)*T(2,4)+T(2,2)*T(1,4)
                              -T(1,3)*T(2,4)+T(2,3)*T(1,4)];
                    delta = T(3,1:3)';          % nz oz az
          else
                    % prismatic axis
                    d = T(3,1:3)';              % nz oz az
                    delta = zeros(3,1);         %  0  0  0
          end
          J = [[d; delta] J];
end
```

```
%JACOB0       Compute manipulator Jacobian in world coordinates
%
%         JACOB0(DH, Q) returns a Jacobian matrix for the current pose Q.
%
%         The manipulator Jacobian matrix maps differential changes in joint space
%         to differential Cartesian motion (world coord frame) of the end-effector.
%                             dX = J dQ
%
%         For an n-axis manipulator the Jacobian is a 6 x n matrix.
%
%         See also JACOBN, DIFF2TR, TR2DIFF, DIFF

%         Copyright (C) Peter Corke 1993
function J0 = jacob0(dh, q)
          %
          %   dX_tn = Jn dq
          %
          Jn = jacobn(dh, q);         % Jacobian from joint to wrist space

          %
          %  convert to Jacobian in base coordinates
          %
          Tn = fkine(dh, q);          % end-effector transformation
          J0 = [Tn(1:3,1:3) zeros(3,3); zeros(3,3) Tn(1:3,1:3)] * Jn;
```

```
%FKINE  Forward robot kinematics for serial link manipulator
%
%         FKINE(DH, Q)  computes the forward kinematics for each joint space
%         point defined by Q.  DH describes the manipulator kinematics in standard
%         Denavit Hartenberg notation.
%
% DH has one row of kinematic parameters for each axis.  Each row is of
```

```
% the form
%               [alpha A theta D]           for an all revolute manipulator
%       or      [alpha A theta D sigma] for a mixed revolute/hybrid manipulator
%
% For an n-axis manipulator Q is an n element vector or an m x n matrix. The
% elements are interpretted as joint angle or link length according to
% the form of DH or the j'th sigma value (0 for revolute, other for prismatic).
%
% If Q is a vector it is interpretted as the generalized joint coordinates, and
% FKINE(DH, Q) returns a 4x4 homogeneous transformation for the final link of
% the manipulator.
%
% If Q is a matrix, the rows are interpretted as the generalized
% joint coordinates for a sequence of points along a trajectory.  Q(i,j) is
% the j'th joint parameter for the i'th trajectory point.  In this case
% FKINE(DH, Q) returns an m x 16 matrix with each row containing a 'flattened'
% homogeneous transform corresponding to the input joint state.  A row can
% be unflattened using reshape(v, 4, 4).
%
%       See also LINKTRAN, MFKINE.

%       Copright (C) Peter Corke 1993

function t = fkine(dh, q)
        %
        % evaluate fkine for each point on a trajectory of
        % theta_i or q_i data
        %

        n = numrows(dh);

        if length(q) == n,
                t = eye(4,4);
                for i=1:n,
                        t = t * linktran(dh(i,:), q(i));
                end
        else
                if numcols(q) ~= n,
                        error('bad data')
                end
                t = [];
                for qv=q',                      % for each trajectory point
                        tt = eye(4,4);
                        for i=1:n,
                                tt = tt * linktran(dh(i,:), qv(i));
                        end
                t = [t; tt(:)'];
                end
```

```
    end
```

```
%DIFF2TR    Convert a differential to a homogeneous transform
%
%    DIFF2TR(D) returns a homogeneous transform representing differential
%    translation and rotation.
%
%    See also TR2DIFF, DIFF


%    Copyright (C) Peter Corke 1993
%  Bug: Corregido Sept 2000, Josechu Guerrero

function t = diff2tr(d)
    t =[    1       -d(6)   d(5)    d(1)
            d(6)    1       -d(4)   d(2)
            -d(5)   d(4)    1       d(3)
            0       0       0       1       ];
```

Unimation®

A Westinghouse Company

Unimation Incorporated
Shelter Rock Lane
Danbury, CT 06810

November 1984
Supersedes DB-22-523
Dated May 1984
Mailed to: E.C.D/22-505A,
B,C,E,J

A compact, computer-controlled robot for medium-to-lightweight assembly, welding, materials handling, packaging and inspection applications.

UNIMATE® PUMA® Series 500 Industrial Robot

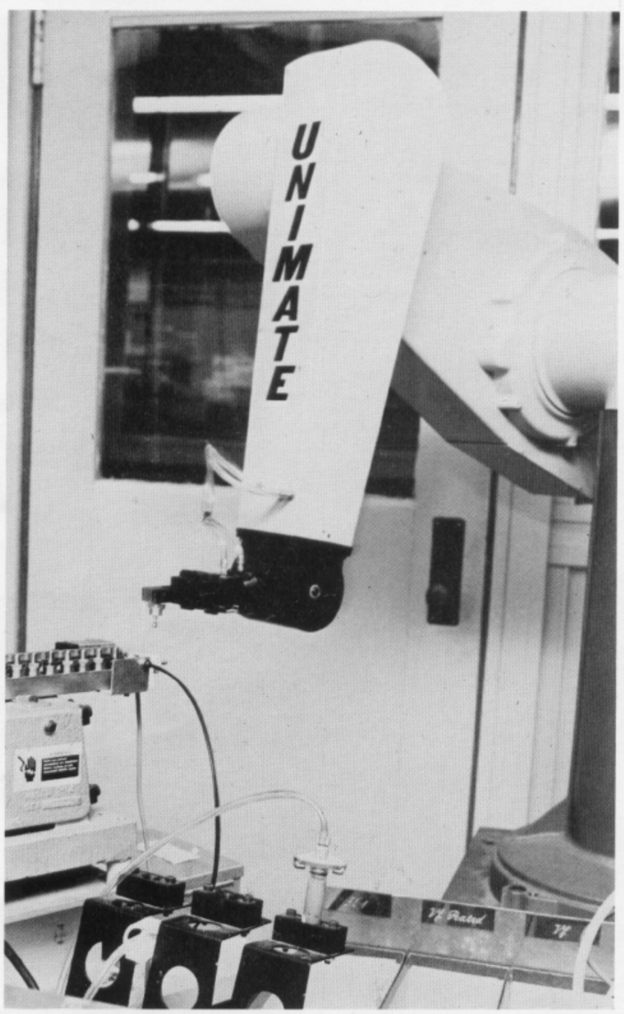
FEATURES

The Series 500 is the most widely used model in the UNIMATE PUMA line of electrically driven industrial robots. With a 36 inch reach and 5-pound payload capacity, the PUMA Series 500 robot is designed for assembly and applications requiring high degrees of flexibility and reliability.

With over 1,000 units in the field, its capabilities are particularly suited to the requirements of the electronics and other industries where light-to-medium-weight parts handling or processed functions are carried out.

EASE OF USE

VAL,™ a revolutionary advance in robot control systems, is used to control and program PUMA robots. The system uses an LSI-11 as a central processing unit and communicates with individual joint processors for servo control of robot arm motions. The results are ease in set up, high-tolerance repeatability, and greater application versatility.



VAL combines a sophisticated, easy-to-use robot programming capability with advanced servo control methods. Intuititive English- language instruction provides fast, efficient program generation and editing capabilities. All servo path computations are performed in real time, which makes it possible to interface with sensory based systems.

EASE OF INSTALLATION

PUMA 500 robots are easily integrated into existing production lines because of the ease with which they are programmed. In addition, the robot can be easily and quickly reprogrammed for changes in the product or production process.

Programs can be written either on- or off-line using a CRT or teletype terminal, or they can be generated manually by guiding the robot arm through program paths using a microprocessor-based teach pendant. With either method, position data can be added or changed by key input, manual control or floppy-disk input, without affecting the overall task program.

VAL reduces memory requirements and permits complex programs, such as palletizing, to be easily written with a minimum number of taught positions.

Significant time savings can also be realized by integrating predefined subroutines and tasks into complex operations. These tasks can also be stored on floppy disks to build a library of routines, and even whole programs, that can be loaded into the memory of any PUMA controller so that specific tasks and repetitive routines need to be written only once.

With VAL, the Series 500 can also respond to fluctuations in the rate or other parameters of on-going production processes. Since all servo-path computations are performed in real time, changes in the arm path, or even task sequencing, can be initiated by feedback from various sensors and vision systems. As a result, the 500 can interact flexibly and efficiently as part of a large and complex manufacturing system.

APPLICATIONS

With its high speed, repeatability and flexibility the PUMA 500 robot is suited to a wide range of small parts-handling applications, and VAL control makes it easy to design application programs to carry out the most difficult robotic tasks.

Current assembly applications include automotive instrument panels, small electric motors, printed circuit boards, subassemblies for radios, television sets, appliances and more. Other applications include packaging functions in the pharmaceutical, personal care, and food industries. Palletizing of small parts, inspection, and electronic parts handling in the computer, aerospace and defense industries round out the present installed base.

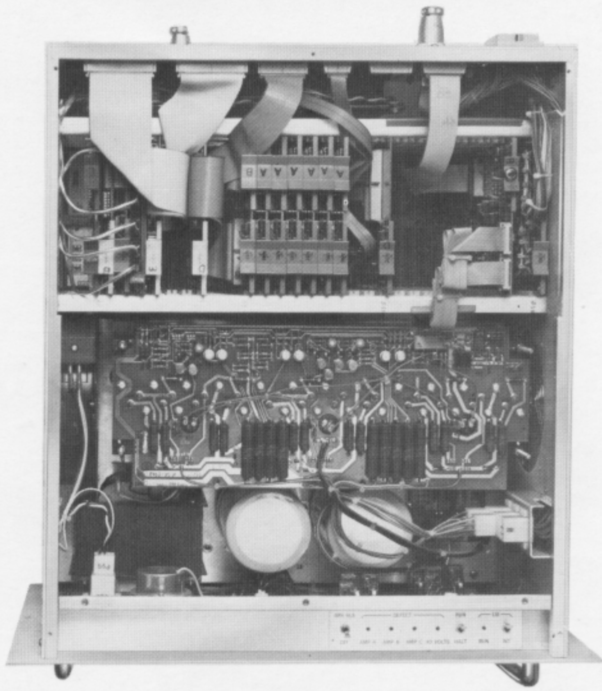
ADVANCED ENGINEERING

Modular, straightforward layout facilitating easy board replacement and plug-in expansion.

High-volume ventilation system.

10 bit digital converter offering highly reliable repeatability of arm positions to ±0.004 inch.

Digital servo components designed for high-temperature operation and state-of-art dc motor control.



Compact packaging (19 in. rack mountable.).

Power amplifiers designed for high energy output with sensors for high temperature.

Auto-start button for automatic operation.

Easy-access panel location for self-diagnostic indicators and troubleshooting switches.

High-density, double-sided floppy-disk drive unit for program storage at 9,600 baud with 10,000-hour MTBF componentry.

High-precision gearing designed for ultra-low backlash and specially hardened for long life.

External lubrication fittings for ease of maintenance.

Durable, industrially hardened design utilizing monocoque construction for stiff, accurate motions.

Servo motors incorporating square wave encoders and high precision, laser-trimmed potentiometers offer the most proven and sophisticated drive systems.

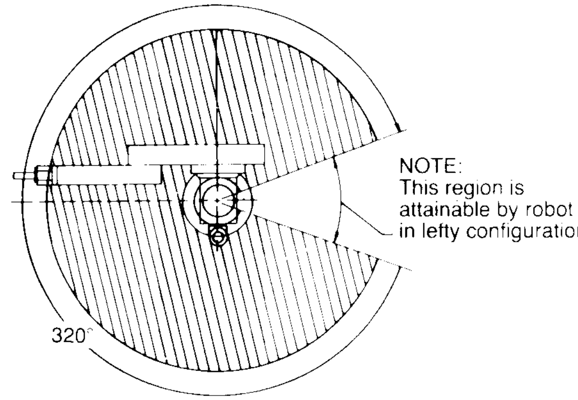
Industrial-grade, membrane-type teach pendant and CRT.

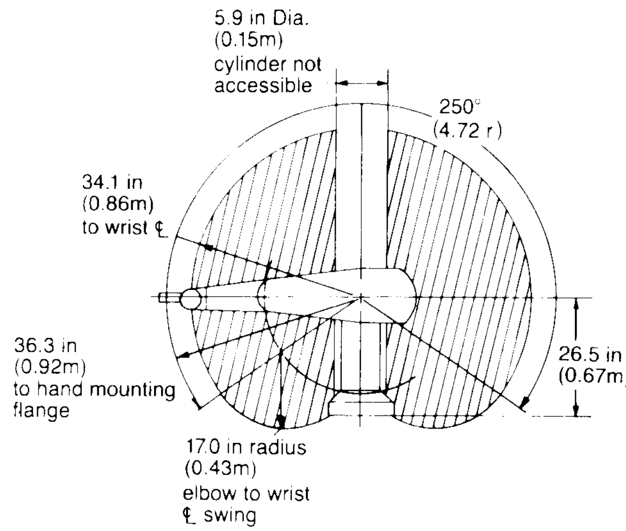
GENERAL	
Configuration	Up to 6 degrees of motion
Drive	Electric DC Servos
Controller	System computer (LSI-11)
Teaching Method	By teach control and/or computer terminal
Program Language	VAL PLUS or VAL II
Program Capacity	8K CMOS user memory in VAL PLUS 24K CMOS user memory in VAL II Options for add'l. user memory
External Program Storage	Floppy-disk
Gripper Control	4-way pneumatic solenoid
Power Requirement	110-130 VAC, 50-60 Hz, 1500 Watts
Optional Accessories	CRT or TTY terminals, I/O module (8 input/ 8 output signals isolated AC/DC levels) up to 32, I/O capacity, pneumatic grippers without fingers, software packages
PERFORMANCE	
Repeatability	±0.004 in. (0.1 mm)
Maximum Payload	
Static Load	5.5 lbs. (2.5 kg)
Dynamic Load Around Joint 5	137.5 lb-in² (403.2 kg-cm²) (5.5 lb (2.5 kg) concentrated load at 5 inches (12.7cm) from Joint 5
Dynamic Load Around Joint 6	12.4 lb-in² (35.3 kg-cm²) (a 5.5 lb (2.5 kg) concentrated load at 1.55 inches (3.76cm) from Joint 6
Straight Line Velocity	20 in/sec. max. (0.5m/sec.)

ENVIRONMENTAL
OPERATING RANGE
50°-120°F (10°-50°C)
10-80% relative humidity (non-condensing)
Shielded against industrial line fluctuations and human electrostatic discharge

PHYSICAL CHARACTERISTICS	
Arm Weight	120 lbs. (54.5 kg)
Controller Size	12.5"H × 17.5" W × 19.6" D. (317.5 mm H × 444.5 mm W × 500.0 mm D) (19 in. rack mountable)
Controller Weight	80 lbs. (36.4 kg)
Controller Cable Length	15 ft. (4.57m) std 50 ft (15.24m) max.

Specifications subject to change without prior notification.





For technical installation information, request UNIMATION Dwg. No. 560-0050

UNIMATION TOTAL CAPABILITY

UNIMATION offers a full line of computer controlled programmable robots that provide manufacturers with the cost savings and productivity increases needed in today's economic environment. The company is dedicated to a sustained program of product improvement, taking advantage of the latest proven developments in technology, together with production quality-control procedures and field performance surveillance reports.

In addition to on-going development engineering, UNIMATION's System Engineering Division consists of a staff of knowledgeable applications engineers with experience in virtually all manufacturing disciplines. Activities range from design and manufacture of end-of-arm tooling and simple, single-robot installations up through multi-robot systems. A large applications laboratory allows customer demonstrations on all robot models, and system run-offs prior to installation.

A staff of skilled field service and installation engineers and a customer training and technical publications department round out the organization.

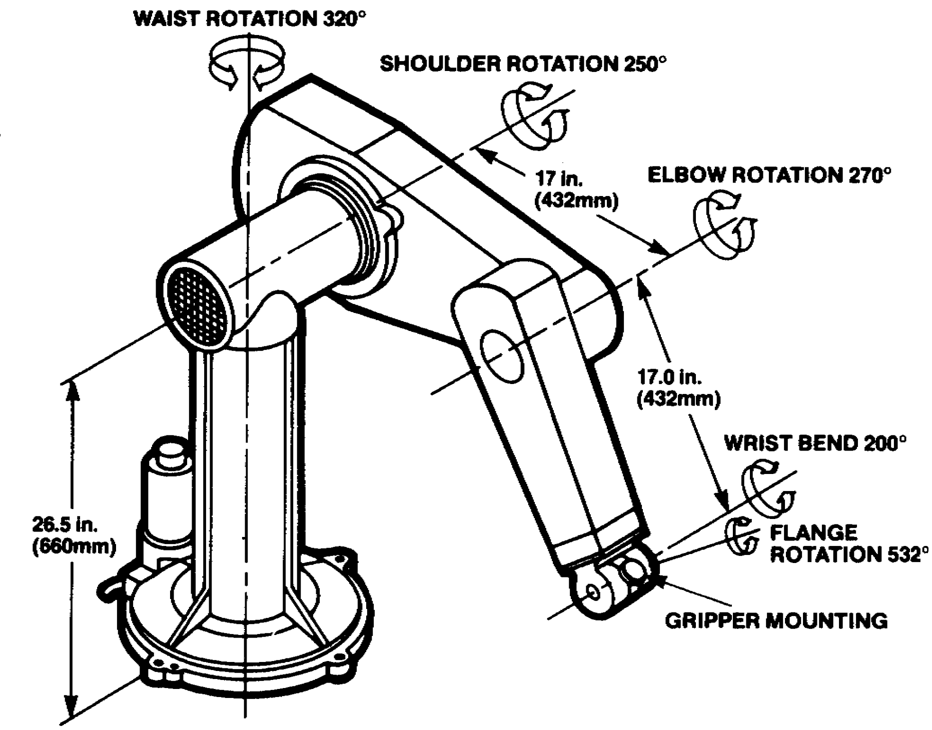
The breadth and depth of UNIMATION's total organization is its assurance to customers of full support and dependable products.

For more information, call or write:

Unimation®
A Westinghouse Company

UNIMATION Incorporated
Shelter Rock Lane
Danbury, Connecticut 06810
(203) 744-1800

UNIMATE PUMA
500 Series



Performance

REPEATABILITY	±0.004 in. (±0.1 mm)
LOAD CAPACITY	5.5 lbs. (2.5 Kg)
STRAIGHT LINE VELOCITY	20 in/s max. (0.5 m/s max.)
ENVIRONMENTAL	50-120° F (10-50° C)
REQUIREMENTS	80% humidity (non-condensing). Shielded against industrial line fluctuations and human electro-static discharge

Physical Characteristics

ARM WEIGHT	120 lbs. (54.5 Kg)
CONTROLLER SIZE	19" x 12.5" x 23.6" (475 mm x 312.5 mm x 590 mm)
CONTROLLER WEIGHT	80 lbs. (176 Kg)
CONTROLLER CABLE LENGTH	15 ft. (4.5 m) Standard 50 ft. (15 m) Optional

General Specification

CONFIGURATION	5 revolute axes or 6 revolute axes
DRIVE	Electric DC Servo
CONTROLLER	System Computer (LSI-11/2 or 11/23)
Teaching Method	By manual control and/or computer terminal
Program Language	VAL or VAL II
Program Capacity	16K CMOS user memory std. (32K for VAL II)
External Program Storage	Floppy-disk (optional)
GRIPPER CONTROL	4-way pneumatic solenoid
POWER REQUIREMENT	110-130 V AC 50-60 Hz, 1500 W
OPTIONAL ACCESSORIES	CRT or TTY terminals, floppy-disk memory storage, I/O module, 8 input/8 output signals (max. 32)—isolated AC/DC levels. Pneumatic gripper w/o fingers

SPECIFICATIONS
PUMA 200, 500, 700 Series Robots

Individual Specifications

	Configuration	Repeatability	Load Capacity	Straight Line Velocity	Arm Weight/Mounting
200	6 Revolute Axes	0.002 in. (±0.05 mm)	Not to exceed: At Flange Rotation, 0.5 in-oz-sec^2 At Wrist Bend and Rotation, 1.8 in-oz-sec^2	49.0 in/s max. (1.25 m/s max.)	15 lbs. Designed for floor or overhead mounting only.
500	5 or 6 Revolute Axes	0.004 in. (±0.1 mm)	Not to exceed: At Flange Rotation, 0.5 in-oz-sec^2 At Wrist Bend and Rotation, 5.7 in-oz-sec^2	20 in/s max. (0.5 m/s max.)	120 lbs. Designed for floor or overhead mounting only.
700	6 Revolute Axes	0.008 in. (±0.2 mm)	Not to exceed: At Flange Rotation, 14.1 in-oz-sec^2 At Wrist Bend and Rotation, 56.7 in-oz-sec^2	40 in/s max. (1.0 m/s max.)	660 lbs. Designed for floor or overhead mounting only.

General Specifications

Drive	Electric D.C. Servo
Controller	System Computer (LSI-11/2 or 11/23)
Size	200 & 500: 19" x 12.5" x 23.6" (475 mm x 312 mm x 590 mm) 700: 72" x 25.5" x 32" (1830.3 mm x 636.5 mm x 801.6 mm)
Weight	200 & 500: 80 lbs.—700: 600 lbs.
Teaching Method	By manual control and/or computer terminal
Program Language	VAL or VAL II
Program Capacity	16K CMOS user memory std. (32K for VAL II)
External Program Storage	Floppy Disk (Optional)
Cable Length (arm-controller)	15 ft. (4.5 in) of standard 50 ft. (15.24 m) optional
Mounting (arm)	Designed for floor or overhead mounting only.
Power Requirements	200: 110-130 VAC, 50-60 Hz, 500 W 500: 110-130 VAC, 50-60 Hz, 1500 W 700: 220 or 440 VAC, 50-60 Hz, 3 Phase, 6300 W
Options	CRT or TTY terminals, floppy disk memory storage, pneumatic grippers w/o fingers (parallel or toggle action), UNIVISION™, 200 & 500: I/O module (8 input/8 output signals-isolated AC/DC levels) up to 32 I/O capability (4 modules) 700: I/O modules (16 input/16 output, up to 32 I/O capability (2 modules)
Environmental Operating Range	50-120°F (10-50°C) 10-80% relative humidity (non-condensing) Shielded against industrial line fluctuations and human electrostatic discharge.