Intelligent Force Sensing System

User's Guide

14155-000

Copyright Notice

Table of Contents

Revision History

Revision	Date	Revised Content
A	December, 2014	Original release.
B	March, 2016	Added access from ACE.
C	May, 2017	Added adapter between XBELTIO cable and robot controller.
D	June, 2018	Removed Force Jog example, removed most English units.
E	September, 2018	Noted that end-of-arm solenoid control is not available for Vipers after installing the Force Sensor. Specified IP rating as IP40.
F	January, 2020	Removed adapter between XBELTIO cable and robot controller.

Chapter 1: Introduction

The Intelligent Force Sensing system is a hardware and software package that allows robots to react to sensed forces and moments. Tight integration of the force sensor to the robot control system dramatically reduces force overshoot and robot stopping time when forces or moments exceed preset thresholds. As a result, assembly operations can be performed at higher speeds.

This manual describes installation and use of an Intelligent Force Sensing system with a robot controlled with an eAIB or eMB-60R amplifier.

The Intelligent Force Sensing system is supported on all Standard and Pro eCobras, the Viper 650/850, the Quattros, and the Hornet robots. It is not supported on any Cobra robots other than the eCobras.

> **NOTE:** The DeviceNet port on eCobras will no longer be available after installation of an Intelligent Force Sensing system.

> **NOTE:** The Force Sensor uses the CN20 connector on the Viper robot's base. For a Viper, this will prevent a user from using an end-of-arm solenoid, because the Force Sensor and the solenoid need the same connector.

A complete set of instructions in the eV+ language will allow you to quickly take advantage of force sensing in your application. In addition, the ACE software application includes full Intelligent Force Sensing support.

Throughout this manual, we assume that you are familiar with the eV+ language and the basic operation of robots. eV+ programmers should be familiar with the eV+ Language Reference Guide.

See the ACE User's Guide and documentation on your specific robot for details on using Intelligent Force Sensing in ACE software.

1.1 Hardware Overview

> **NOTE:** The force sensor is manufactured by JR3, code 67M25A3. It is not manufactured by Omron.

The force sensor is a device that detects forces and moments applied by the end-of-arm gripper. The sensor has strain gauges mounted on flexing elements within the sensor. When a force is exerted on the sensor, the strain gauges sense the deflection of the flexing elements, causing a voltage difference across several strain gauges. On-board electronics multiplex and amplify the voltage differences, convert them into digital data, and transmit them to the robot amplifier. The sensor-resident electronics also contain information specific to the sensor, such as serial and model numbers, and a calibration matrix that relates strain gauge readings to force and moment readings.

The strain gauge readings are transmitted through a cable from the force sensor to the eAIB or eMB-60R amplifier. The strain gauge readings are then converted into useful force readings using a sensor-specific calibration matrix. Filters, scaling, translations, and rotations further condition the data before making force readings available to user programs.

The force sensor should survive unexpected collisions to several times its rated load. However, making a habit of overloading the sensor can result in shortened sensor life. You should not attempt long-term operation at loads beyond the sensor rating.

Figure 1-1. Force Sensor, Mounted and Separate

1.2 Software Overview

The Intelligent Force Sensing system is integrated into the eV+ programming language and operating system using several eV+ instructions. These instructions perform a full range of functions, including force system customization, reading forces, and controlling force operating modes.

In addition, ACE users will find that the robot includes a complete software library.

Operating Modes

The sensing system has two active operating modes: Guarded mode and Protect mode. Each mode causes a particular system response when an associated force is exceeded. The modes are completely independent and may operate concurrently or individually.

- Guarded mode stops the robot at its current position when either of two user-configurable force trip conditions is exceeded. This mode is typically used during assembly operations because it allows the robot to rapidly respond to forces during part acquisition, insertion, or post-insertion verification.

  NOTE: Stop-on-digital-signal mode behaves the same as Guarded mode, but uses an external signal rather than the force sensor to trip. The source of the digital signal is user-supplied. This mode is available even on systems without the Intelligent Force Sensing system.

- Protect mode turns off high power to the robot and applies the robot's brakes when a user-specified force or moment level is exceeded. It is typically used to provide an additional level of safety to users and tooling. When configured to trip at a level above the normal operating force levels, Protect mode will detect abnormal situations, such as crashes, and power down the robot immediately.

Built-in Crash Detection

Omron Robotics and Safety Technologies, Inc. control systems provide two built-in means of detecting crashes:

- Envelope errors, in which the robot is detected as being too far from its desired path, and
- Nulling errors, in which the robot is unable to settle to a final location within a specific amount of time.

These provide indirect crash detection through position error measurement. Protect mode provides a direct, user-controllable crash detection mechanism.

eV+ Examples and ACE Software Library

This manual contains example code to allow eV+ users to quickly incorporate an Intelligent Force Sensing system into their applications.

The ACE software library is capable of immediately providing the following capabilities:

- Simple teaching of complicated assembly processes such as guarded moves, insertion search, assembly verification, and snap-fit assembly.
- Force and position data displays. A graphical presentation of the force and position history of an operation can be displayed in a graphics window. Live bar chart displays of force readings are also available on an ACE menu display.
- Automatic force sensing system software loading, initialization, and error recovery.

1.3 How Can I Get Help?

Website

Refer to the corporate website:

http://www.ia.omron.com

Related Manuals

This manual covers the installation, startup, and operation of the Intelligent Force Sensing system. There are additional manuals that cover related topics.

Table 1-1. Related Manuals

Manual Title	Description
Robot Safety Guide	Describes safety aspects for our robots.
Robot user's guide	Describes the installation and operation of your robot.
ACE User's Guide	Describes the ACE user interface, which is used for configuration, control, and programming of the robot system. This is built into the ACE software, under Help.
eV+ Language User's Guide	A description of the structure and elements of the eV+ programming language.
eV+ Language Reference Guide	A complete description of the keywords used by the eV+ programming language. This manual contains a list of all system messages.

Support

If, after reading this manual, you are having problems with your force sensor, contact your local Omron Support.

Chapter 2: Safety

2.1 Dangers, Warnings, and Cautions

Alert Levels

There are three levels of special alert notation used in our manuals. In descending order of importance, they are:

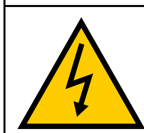
DANGER: This indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

WARNING: This indicates a potentially hazardous situation which, if not avoided, could result in injury or major damage to the equipment.

CAUTION: This indicates a situation which, if not avoided, could result in damage to the equipment.

Alert Icons

The icon that starts each alert can be used to indicate the type of hazard. These will be used with the appropriate signal word - Danger, Warning, or Caution - to indicate the severity of the hazard. The text following the signal word will specify what the risk is, and how to avoid it.

	This is a generic alert icon. Any specifics on the risk will be in the text following the signal word.
	This identifies a hazardous electrical situation.
	This identifies a hazardous burn-related situation.

Special Information

There are several types of notation used to call out special information.

IMPORTANT: Information to ensure safe use of the product.

NOTE: Information for more effective use of the product.

Additional Information: Offers helpful tips, recommendations, and best practices.

Version Information: Information on differences in specifications for different versions of hardware or software.

2.2 Safety Precautions

WARNING: PERSONAL INJURY OR PROPERTY DAMAGE
An Omron Robotics and Safety Technologies, Inc. robot can cause serious injury or death, or damage to itself and other equipment, if the following safety precautions are not observed:

- All personnel who install, operate, teach, program, or maintain the system must read this guide, read the *Robot Safety Guide*, and complete an Omron training course for their responsibilities in regard to the robot.

Figure 2-1. Read Manual and Impact Warning Labels

- All personnel who design the robot system must read this guide, read the *Robot Safety Guide*, and must comply with all local and national safety regulations for the location in which the robot is installed.
- The robot system must not be used for purposes other than described in Intended Use of the Robot on page 13. Contact your local Omron Support if you are not sure of the suitability for your application.
- The user is responsible for providing safety barriers around the robot to prevent anyone from accidentally coming into contact with the robot when it is in motion.
- Power to the robot and its power supply must be locked out and tagged out before any maintenance is performed.

2.3 What to Do in an Emergency or Abnormal Situation

Press any E-Stop button (a red push-button on a yellow background) and then follow the internal procedures of your company or organization for an emergency or abnormal situation. If a fire occurs, use CO_2 to extinguish the fire.

In case of an emergency or abnormal situation, some robot joints can be manually moved without high power. However, only qualified personnel who have read and understood the robot user's guide and *Robot Safety Guide* should manually move the robot into a safe state. Some robot joints are held by a brake, which can only be released with the Brake Release button. This requires 24 V power to the robot.

2.4 Robot Behavior

Hardstops

If the robot runs into one of its hardstops, the robot’s motion will stop completely, an envelope error will be generated, and power will be cut to the robot motors.

The robot cannot continue to move after hitting a hardstop until the error has been cleared.

The hardstops are capable of stopping the robot at any speed, load, and maximum or minimum extension.

Limiting Devices

There are no dynamic or electro-mechanical limiting devices provided by Omron Robotics and Safety Technologies, Inc. The robots do not have safety-rated soft axis or space limiting.

However, the user can install their own safety rated (category 0 or 1) dynamic limiting devices if needed, that comply with ISO 10218-1, Clause 5.12.2.

Singularities

No singularities exist that cause a hazardous situation with an Omron Robotics and Safety Technologies, Inc. robot.

2.5 Intended Use of the Robot

WARNING: PERSONAL INJURY
With the exception of Omron's TM line of collaborative robots: Omron's fixed robots are not collaborative robots. They require a dedicated work area that will prevent personnel from coming into contact with them during operation.

The normal and intended use of these robots does not create hazards.

The robots have been designed and constructed in accordance with the relevant requirements of IEC 60204-1.

See the Technical Specifications for your robot for complete specifications. Refer to the *Robot Safety Guide* for details on the intended use of our robots.

The Omron Robotics and Safety Technologies, Inc. robots are not intended for:

- Use in the presence of ionizing or non-ionizing radiation
- Use in potentially explosive atmospheres

- Use in medical or life saving applications
- Use in a residential setting. They are for industrial use only.
- Use before performing a risk assessment

2.6 Additional Safety Information

We provide other sources for more safety information:

Manufacturer’s Declaration of Incorporation

This lists all standards with which the robot complies. The Manufacturer’s Declarations for Omron Robotics and Safety Technologies, Inc. fixed robots and other products are in the *Manufacturer’s Declarations Guide*.

Robot Safety Guide

The *Robot Safety Guide* provides detailed information on safety for our robots. It also gives resources for more information on relevant standards. It ships with each robot.

Manual Control Pendant

The protective stop category for the pendant enable switch is category 1, which complies with the requirements of ISO 10218-1.

The pendant is designed in accordance with the requirements of IEC 60204-1 and ISO 13849. The E-Stop button is ISO 13850 compliant.

> **NOTE:** Omron Robotics and Safety Technologies, Inc. does not offer a cableless (wireless) pendant.

The manual control pendant can only move one robot at a time, even if multiple robots are connected to a SmartController EX, and the pendant is connected to the SmartController EX.

Chapter 3: Basic Force Sensor Concepts

3.1 Introduction

Measuring forces and moments is only one of many Intelligent Force Sensing system capabilities. This chapter is intended to provide you with a clear picture of the full range of these capabilities. The concepts covered in this chapter are:

Force measurement	
Force filter	Performs low-pass filtering to reduce noise levels.
Force Reference Frame	Establishes the coordinate system in which forces are measured.
Force zero	Establishes a permanent zero point from which forces are measured.
Force offset	Temporarily compensates for part weight and orientation.
Force scaling	Allows the user to specify forces and moments in any units desired, such as pounds instead of Newtons, etc.
Modes of operation	
Guarded mode	Halts the robot when a specified force threshold is crossed.
Stop-on-digital-signal	Halts the robot when a specific digital signal is received.
Protect mode	Performs simple crash detection by powering down the robot when a specified force threshold is crossed.

Any time that the force sensor could halt the robot in Guarded mode, a Stop-on-digital-signal could also be used to halt the robot.

3.2 Force Measurement

The analog voltages produced by the strain gauge eventually lead to force readings in the robot control system, but on their way they undergo various stages of conditioning. The most important stage is multiplication by the calibration matrix, which converts strain gauge readings to six components of force and moment. The calibration matrix is unique to each sensor. It is created during manufacture by correlating sensor readings to a series of precisely calibrated loads.

The user cannot change the way strain gauge readings are converted to raw forces and moments because the calibration matrix is recorded in the force sensor at the factory, but there are a variety of other user-settable parameters that affect the force readings the system eventually produces. The steps from gauge voltages to force and moment readings are described in the sections that follow.

Force Filter

As a result of mechanical vibration and electrical noise, the strain gauge readings will normally have some high-frequency fluctuation. User-selectable low-pass filters are provided to reduce this noise. Although filters have the advantage of reducing noise in the sensor reading, allowing more delicate operations, they have the disadvantage of adding time delay between the moment of contact and the resulting force reading.

The eV+ FORCE instructions use a default set of filters that you can change with the FORCE.MODE instruction. The default filter for a guarded move is a fast filter (light filtering with a small time delay), so that contact is quickly detected, while the default filter used for zeroing force readings is a slow filter (heavy filtering with a longer time delay) for maximum accuracy. The available force filters are listed in Force Filter Specifications on page 71.

You should exercise care in changing the default filters, since the more filtering that is done, the longer it will take the force system to sense contact. This can result in higher position, force, and moment overshoots immediately after contact. You can experiment with different settings to determine the best balance between noise reduction and fast response time, but the default setting of the sensor filters should work for most applications.

Force Reference Frame

With a six-axis force sensor, it is necessary to know the relationship between the X, Y, and Z axes of the force measurements and those of the robot gripper. The user can specify this relationship with the Force Reference Frame (FRF). A force sensor mounted directly to a standard tool flange has a default Force Reference Frame coincident with the default robot tool frame. The robot tool frame is a coordinate system centered at the robot tool flange, with the Z axis pointing away from the tool flange and the X axis pointing towards the flange key. The nominal robot tool frame and FRF X axis is labeled on the outside cover of the force sensor.

If the force sensor is not mounted on the robot’s tool flange, the user needs to specify the offset from the robot tool frame to the sensor frame. This is done by specifying a standard transformation relating the two frames, and is known as the Sensor Reference Frame, or SRF.

The force sensor reports forces and moments exerted by the robot, not on the robot. With no load, the readings in all directions should be close to 0. If the robot is pushing down against a surface (along the nominal FRF +Z axis) the Z component of the force readings will increase. If a load is hung on the robot, the robot has to pull up (along the FRF -Z axis) and the Z component of the force readings will decrease.

Moments (often called torques) are reported when a force is exerted that is not directed through the FRF origin (that is, a force that tries to twist the sensor) or when a couple (pure twist with no net force) acts on the sensor. For example, when a load is hung so that its weight acts along the FRF Z axis, there will be no moment - the FRF is being pulled, not twisted. When a load is hung out along the positive X axis, a positive moment is created about the Y axis in addition to the pull along the Z axis.

Moment, like force, is a vector quantity. Its direction is fixed by the right-hand rule, when the fingers of the right hand curl in the direction that the force tends to turn about the FRF origin, the right thumb points in the direction of the moment vector. Since moments often result from a force acting at a distance from the FRF, a moment has units of force and distance, such as Newton-meters or inch-pounds.

The following figure shows an example of forces and moments that occur when a weight is hung from two different positions in the FRF. Note that force readings are independent of where the weight is hung, but moment readings are very position-dependent. In the first case, the force's line of action (the line along which the force pushes or pulls) passes through the FRF origin, causing a 20 N force and no moment. In the second case, the line of action is 20 cm from the FRF origin, causing not only a 20 N force but also a 4.0 N·m moment.

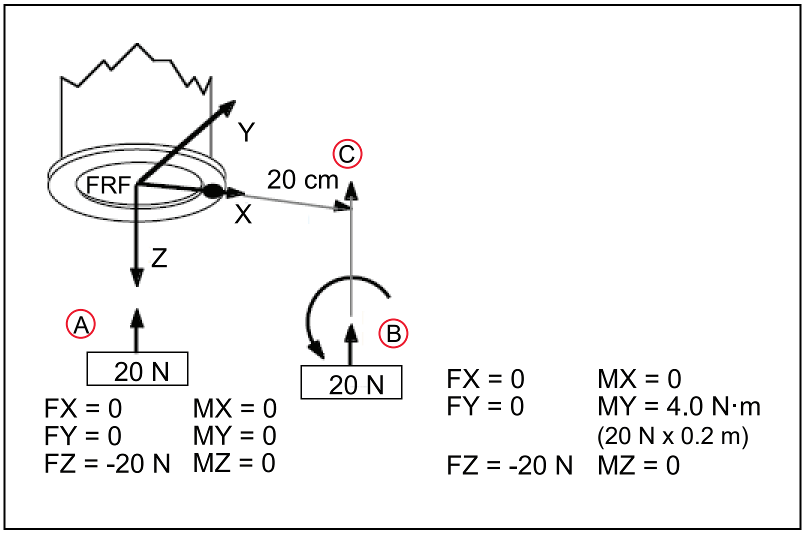


Figure 3-1. Effect of Loading Position on Forces and Moments in FRF

Key	Meaning
A	Force exerted by robot to support load
B	Force and torque exerted by robot to support load
C	Line of Action

In general, it is easiest to interpret the FRF force and moment readings if the FRF coincides with the tool axes as defined by the eV+ tool transformation (see the eV+ Reference Guide for a description of the TOOL program instruction and monitor command). By aligning the FRF and the tool frame, motion along the tool axes tends to cause forces along the same FRF axes during contact. It is common for users to always set their FRF transformation equal to their TOOL transformation.

The Force Reference Frame can be set in eV+ with the FORCE.FRAME instruction.

Force Zero

It is best to have force and moment readings close to zero when the robot is not holding a load, but the weight of the force sensor alone will initially cause the force sensor to read non-zero force levels. With a heavy gripper the initial reading will be even larger. An application with a 40 Newton gripper would cause an initial reading of about -40 Newtons of force in the FRF Z direction. If the robot is commanded to move down until an upward force of one Newton is sensed, the actual force applied would be close to 41 Newtons before the robot stops.

Performing a force-zero operation is a simple way of correcting for this. It will set the zero level for all force and moment readings at the current force and moment levels. This is typically done when the robot is at a known location, and there is no load other than the gripper itself. In the 40 Newton gripper example, performing a force zero before the motion would have set the force levels to zero despite the gripper weight, and the robot would have stopped at the commanded force level of one Newton.

Over time, accuracy may decrease slightly if the temperature varies widely. It is a good idea to zero the force sensor periodically to counter thermal effects.

The force zero level can be set with the FORCE.ZERO eV+ instruction.

Force Offset

A force offset is a set of force and moment levels that are added to all force readings. This is often used in conjunction with the force-zero operation to compensate for multiple loads on the end of the arm. For example, if an application might use one of three end-of-arm grippers weighing 8, 10, and 12 Newtons, then a single force zero will not compensate for the weight of each of them. In this case, perform a force zero with no gripper attached, and invoke a force offset of 8, 10, or 12 Newtons in the Z direction whenever the appropriate gripper is held. Force readings will then initially be close to zero with each of the grippers.

In practice, it usually works best to zero the force sensor readings for each gripper when that gripper is installed. This will compensate exactly for the weight of the gripper.

The force offset level can be set with the FORCE.OFFSET eV+ instruction.

Force Scaling

The force sensors by default return force readings in Newtons and moment readings in Newton-meters. You can change the scaling to any units you like, such as pounds and inch-pounds, by setting the scaling factors using the FORCE.MODE eV+ instruction. All subsequent force and moment readings will then be reported in the new units.

3.3 Modes of Operation

The Intelligent Force Sensing system provides two ways of responding to sensed forces.

- Guarded mode rapidly halts the robot when a force or moment crosses a specified threshold.
- Protect mode is a simple crash-detection scheme that turns off robot power when a specified force level is exceeded. It operates like an E-Stop button that is tripped whenever a sufficiently hard crash occurs.

A summary of the two operating modes is given in the following table.

Since each of these modes may influence robot motion, any force system hardware error that occurs while the modes are active will cause the robot to stop with a “*Force sensor communication error*”. Once the problem is fixed, the user can enable robot power again, and proceed. Protect mode is then automatically re-enabled, but Guarded mode must be explicitly re-enabled (with a FORCE.MODE instruction) after a force system hardware error.

Table 3-1. Intelligent Force Sensing Operation Modes

	Guarded Mode	Protect Mode
Cause of trip	One or two conditions can be active: • directional, planar, or resultant • X, Y, or Z for directional or planar • force or moment for each • threshold for each • above or below threshold flag for each	Force or moment resultant exceeds preset threshold
Effect of trip	Halt robot at trip position (return to trip position if overshoot)	"Panic stop" stops robot movement
Recovery required	None - ready for next motion	Robot power must be re-enabled
State after trip	Guarded mode disabled (must be invoked for next Guarded move)	Protect mode remains enabled

Both modes can operate at the same time. For example, you might enable Protect mode immediately after the system is powered up to detect crashes. You can then perform Guarded mode moves without having to disable Protect mode.

Guarded Mode

When a Guarded mode trip occurs, the current robot position is automatically recorded and the robot is commanded to return to the trip location as quickly as possible. Some overshoot will naturally occur, since it is impossible to stop a mechanical system instantaneously. The overshoot is a function of robot speed, load, direction of the motion, and the "give" of the surfaces in contact. Once the robot returns to the trip location, the next motion instruction can be processed.

Guarded mode can have two specified trip conditions operating concurrently. A force trip occurs when either of the conditions trip. After the motion, the force system reports which (if any) condition has tripped. The application program must decide how to respond to each of the possible trip conditions. You might use only one of the two conditions because you only require contact/no-contact information. But it is also possible to use multiple trip conditions. For example, you can branch to different subroutines depending on which of the conditions tripped. Or you could consider the first trip condition to signify a normal force trip and the second trip condition to signify an error condition.

Guarded mode is used to implement guarded moves, which are robot motions that are halted if a force trip occurs. To perform a guarded move, you typically send the force system the information that configures the trip conditions for Guarded mode, start the move, and then enable Guarded mode. Guarded mode will remain enabled until a force trip occurs, or until it is explicitly disabled. As a result, if a force trip does not occur during a guarded move and Guarded mode is not disabled, all subsequent moves are guarded until a trip occurs.

There are three types of trip conditions that can cause a Guarded mode trip:

- directional (1-dimensional)
- planar (2-dimensional)
- resultant (3-dimensional)

Each trip condition will trip on either force or moment, and each has lower and upper thresholds. If the force or moment level falls below the lower threshold or rises above the upper threshold, the condition trips.

Directional trip conditions measure forces or moments in a single direction (1 dimension). You can select the X, Y, or Z direction as specified by the Force Reference Frame. Directional trip conditions are particularly useful in applications where it is helpful to ignore forces in certain directions, such as X-Y forces that can occur while decelerating towards a location where only Z direction forces are important to an insertion.

Planar trip conditions select a plane (2 dimensions) in which to monitor forces or moments. You can select the X, Y, or Z direction as specified by the Force Reference Frame. The specified direction is perpendicular to the plane in which forces and moments are measured. For example, suppose the robot should stop if it senses more than 5 N of lateral force (any force in the X-Y plane). The X-Y plane is selected by specifying the Z axis (which is perpendicular to the X-Y plane).

Note that the same direction has very different meanings for directional and planar trip conditions. Directional trip conditions monitor forces along the specified axis; planar trip conditions monitor forces in the plane perpendicular to the specified axis. The measured force can be negative in the case of the directional trip condition if it is in the opposite direction from the axis. The measured force is never negative for a planar trip condition since it does not distinguish between different directions on the surface of the plane.

Resultant trip conditions measure net force or moment in all directions (3 dimensions). Since this type of trip condition is non-directional, no direction needs to be specified. Like planar trip conditions, the measured force can never be negative since this condition measures only the magnitude of the force. Resultant conditions are the easiest to use of the three possibilities, since the user need only specify two thresholds and whether to monitor force or moment.

An application example follows. Assume that you would like to put a package onto a pallet and secure it against a corner of that pallet. You could perform the operation, shown in Trip Conditions for Locating a Part in a Corner on page 21, in three stages:

1. Guarded move to surface.

   Since you may be approaching the location with high XY deceleration, you perform a directional guarded move, monitoring forces in the Z direction only.

2. Guarded move to side of pallet.

   If the point of contact with the surface is more than 125 mm from the center of the force sensor, the sensor is more sensitive to assembly moments than to forces. Since a moment will be created about the FRF origin, you perform a resultant guarded move, monitoring moments about any axis. (This move could also be performed with a directional guarded move, measuring forces in the FRF X direction or moments about the FRF Y axis.)

3. Guarded move to corner of pallet.

   Since you already have a moment caused by pressing up against one wall, you perform a directional guarded move, measuring moments about the FRF X axis. (This move could also be performed with a directional guarded move, measuring forces in the FRF Y direction.)

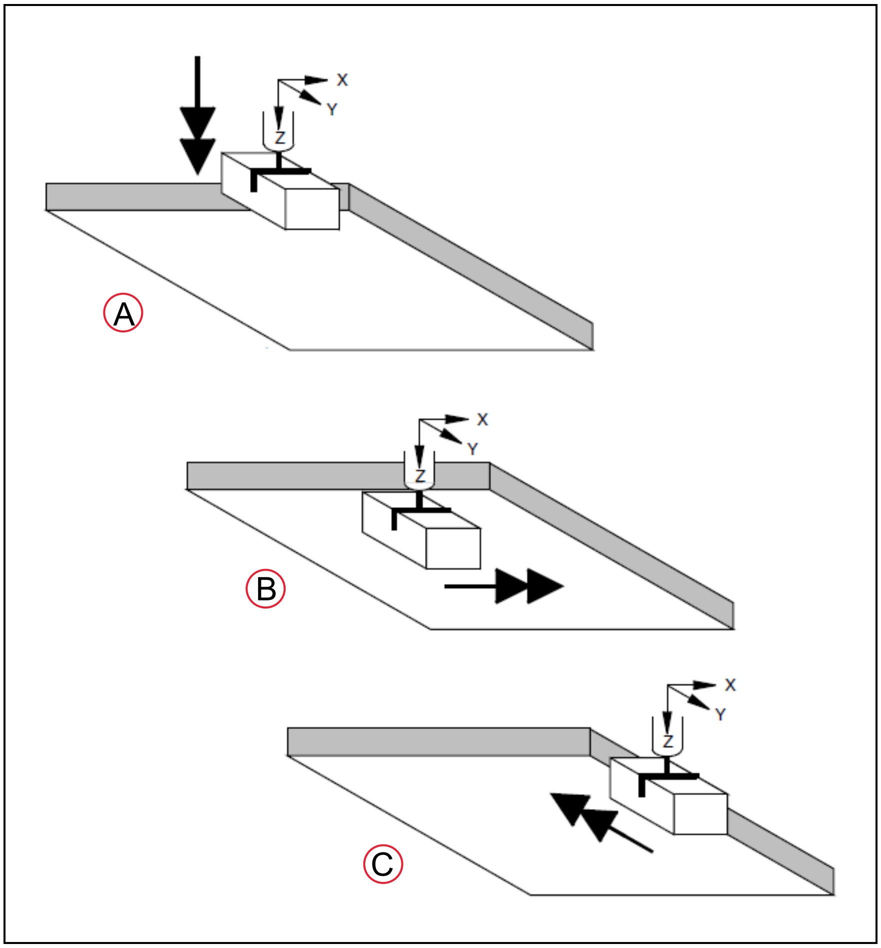


Figure 3-2. Trip Conditions for Locating a Part in a Corner

Key	Meaning
A	Step 1. MOVE TO SURFACE Force/Moment: Force Trip dimension: 1 (along axis) Trip direction: Z Threshold: > 2 lb
B	Step 2. SLIDE TO EDGE Force/Moment: Moment Trip dimension: 3 (resultant) Trip direction: don't care Threshold: > 10 in-lb
C	Step 3. SLIDE TO CORNER Force/Moment: Moment Trip dimension: 1 (about axis) Trip direction: Y Threshold: > 10 in-lb

Protect Mode

Guarded mode is typically in effect for only a small percentage of the total cycle time for part acquisition and placement. Although it is unlikely that an unexpected collision will occur during the rest of the cycle, Protect mode can provide a level of emergency sensing of such forces.

Protect mode requires specifying a force level which, if exceeded, will apply robot brakes and turn off motor power. This differs from Guarded mode in that a Guarded mode trip leaves the robot able to begin its next move immediately. After a Protect mode trip, the user must re-enable motor power. Once power is re-enabled, robot motion can be continued.

Chapter 4: Installation

If any problems are encountered, check the Troubleshooting chapter for information or call your local Omron Support.

The mechanical installation of the Intelligent Force Sensing system involves physically mounting the force sensor between two adapter plates to the robot tool flange, and installing cables to connect the force sensor to the eAIB or eMB-60R.

In the ACE software, in Configure Firewire Nodes, you will need to make sure that each force sensor gadget is mapped to a unique ID.

4.1 Environment

The force sensor is rated IP40. It is not intended for use in wet environments, or in any situation where it could be exposed to water.

It is designed to operate in temperatures from -40 to +65°C, non-condensing.

4.2 Installation Tools

Cable ties are needed for mounting the sensor cable to your robot.

The force sensor comes with a hex key (4 mm) that matches its retained screws.

You will need a 2 mm hex key for the strain relief clamp.

Depending on the robot, you will also need a 3 and 5 mm hex key.

4.3 Hardware Installation

This section covers the installation of the force sensor. The sensor is sandwiched between two adapter plates, which hold the sensor to the robot tool flange and provide the user with the same hole and thread pattern as the robot tool flange provides.

CAUTION: EQUIPMENT DAMAGE
Do not leave robot or amplifier power on during the installation of any force system components. Damage to the components or the system could result.

NOTE: You can uninstall a force sensor by following the installation steps in reverse order.

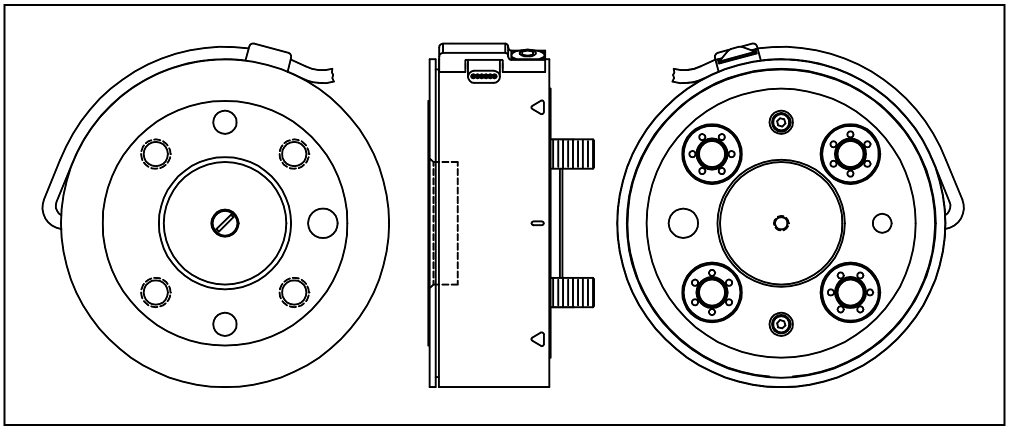


Figure 4-1. Sensor. The Left View is the Gripper Side. The Right View is the Robot Side.

All Supported Robots

The HDB26 end of the XBELTIO cable connects to the XBELTIO port on the eAIB or eMB-60R. The other end splits into three branches. The EXPIO branch is used to connect the force sensor to the robot. The pinouts for the EXPIO branch are given in the following table:

Table 4-1. XBELTIO Cable

EXPIO	HDB26	Description
Pin 1	Pin 7	CLK+
Pin 2	Pin 8	CLK-
Pin 5	Pin 15	Gnd
Pin 6	Pin 6	Belt_5 V
Pin 8	Pin 16	Data+
Pin 9	Pin 17	Data-
Shell	Shell	Shield
Pin 7	NC	9 V

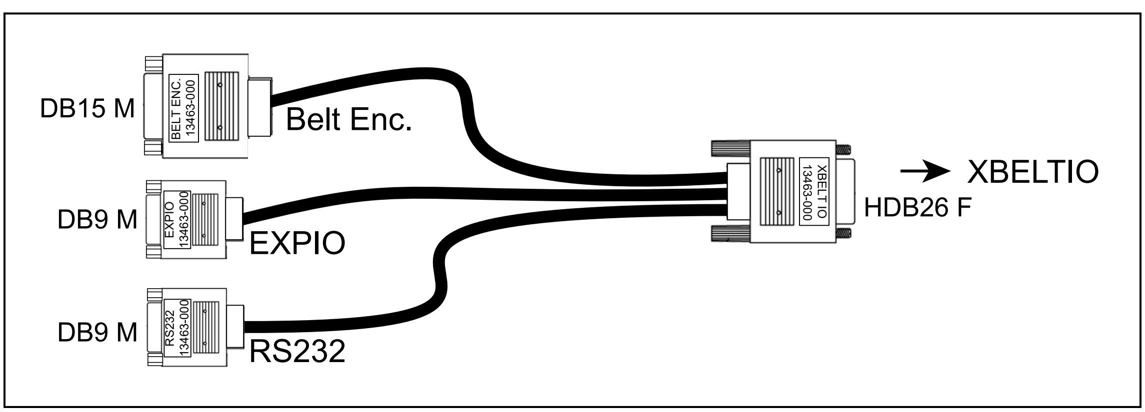


Figure 4-2. XBELTIO Cable

eCobra, Quattro, and Hornet (eAIB Robots)

NOTE: Cobra robots other than the eCobra do not support the Intelligent Force Sensing system.

The installation of the force sensor, with adapter plates, for these three families of robots is the same. The cabling is different.

eCobra Force Sensor Kit

- Force Sensor
- eCobra Force Sensor Parts Kit

eCobra Force Sensor Parts Kit

- Intelligent Force Sensing User's Guide
- XBELTIO Cable (HDB26 F to eAIB or eMB-60R XBELTIO)
- Adapter Plate (inner)
- Adapter Plate (outer)
- Screws for adapter plates, eight M6 x 12 mm
- Dowel pins, H6, four M6 x 12 mm
- Upper Cobra Cable (RJ11 6-pin to DB25 M)
- Lower Cobra Cable (Robot base to XBELTIO EXPIO - DB25F - DB9)

eCobra-Only Parts

- Flex conduit, 7.7 mm long x 2.8 mm OD, 2.3 mm ID
- Rubber grommet, (DeviceNet hole) 0.89 mm (5/16 inch) ID, 2.1 mm (13/16 inch) OD, 0.3 mm (1/8 inch) H

Quattro and Hornet Force Sensor Kit

- Force Sensor
- Quattro and Hornet Force Sensor Parts Kit

Quattro and Hornet Force Sensor Parts Kit

- Intelligent Force Sensing User's Guide
- XBELTIO Cable (HDB26 F to eAIB or eMB-60R XBELTIO)
- Adapter Plate (inner)
- Adapter Plate (outer)
- Screws for adapter plates, eight M6 x 12 mm

- Dowel pins, H6, four M6 x 12 mm
- Upper Cobra Cable (RJ11 6-pin to DB25M)
- Lower Cobra Cable (Robot base to XBELTIO EXPIO - DB25F - DB9)

Quattro and Hornet-Only Parts

- Hex Standoff, Upper Cobra cable to Lower Cobra cable, x2
- Cable tie anchors, for inner joint and robot base

Pinouts

Table 4-2. Upper Cobra Cable | *Table 4-3. Lower Cobra Cable*

RJ11	DB25 M	Description		DB25 F	DB9 F	Description
Pin 1	Pin 1	CLK+		Pin 1	Pin 1	CLK+
Pin 2	Pin 2	CLK-		Pin 2	Pin 2	CLK-
Pin 3	Pin 7	9 V		Pin 7	Pin 7	9 V
Pin 4	Pin 8	Gnd		Pin 8	Pin 5	Gnd
Pin 5	Pin 11	Data+		Pin 11	Pin 8	Data+
Pin 6	Pin 12	Data-		Pin 12	Pin 9	Data-
		Shield		Shell	Shell	Shield

Upper Cobra Cable Installation

This section does not apply to Quattro or Hornet robots.

The Upper Cobra cable runs from the DB25 User Electrical connector on the outer link, internally through the link, and down through the quill, where it is plugged into the force sensor.

1. Move the quill to its uppermost position.
2. Turn off all power to the robot.
3. Remove two screws (eCobra 600) or three screws (eCobra 800) on each side of the outer link cover. Remove two screws at the top rear of the cover and remove the cover.

   Save the screws for reassembly.
4. Remove the screws from the rear cover of the outer link.

   You will need access to the area under this cover for routing the Upper Cobra cable and securing the DeviceNet connector. Save the screws for reassembly.
5. Remove the DeviceNet nut, shown in the following figure.

Figure 4-3. Joint 2, with DeviceNet Connector (A)

6. Push the DeviceNet connector down inside the joint 2 rear cover.

   Reattach the nut just removed to the loose DeviceNet connector, in case it ever needs to be reinstalled.

   Use a cable tie to secure the loose DeviceNet connector to one of the lines inside joint 2, so that it won't be free to move when the robot moves.

7. Plug the DB25 end of the Upper Cobra cable into the User Electrical port at joint 2.

8. Push the RJ11 end of the Upper Cobra cable down through the DeviceNet hole.

9. Route the Upper Cobra cable through the outer link, avoiding moving parts, to the vertical bracket next to the quill.

10. Thread the Upper Cobra cable through the flex tube and then down through the quill.

    The RJ11 end of the Upper Cobra cable should come out through the hole in the robot flange.

11. Attach the flex tube to the vertical bracket, which is parallel to the quill, with a cable tie. Align the top of the flex tube with the top of the vertical bracket.

12. Attach the Upper Cobra cable to the vertical bracket, below the flex tube, at the bottom of the bracket.

The cable tie should go around the blank white label on the Upper Cobra cable.

This will leave enough slack above the quill so that, when the quill is at the top of its stroke, the Upper Cobra cable almost touches the inside of the robot cover.

Cable-tie any excess cable into a service loop under the rear joint 2 cover.

13. Reinstall the outer link covers and tighten the screws to 1.6 N·m (14 in-lb) of torque.
14. Use the included split rubber grommet to seal the DeviceNet hole around the Upper Cobra cable.

eCobra, Quattro, and Hornet Force Sensor and Adapter Plates

This procedure involves the installation of the inner adapter plate, the force sensor, and the outer adapter plate.

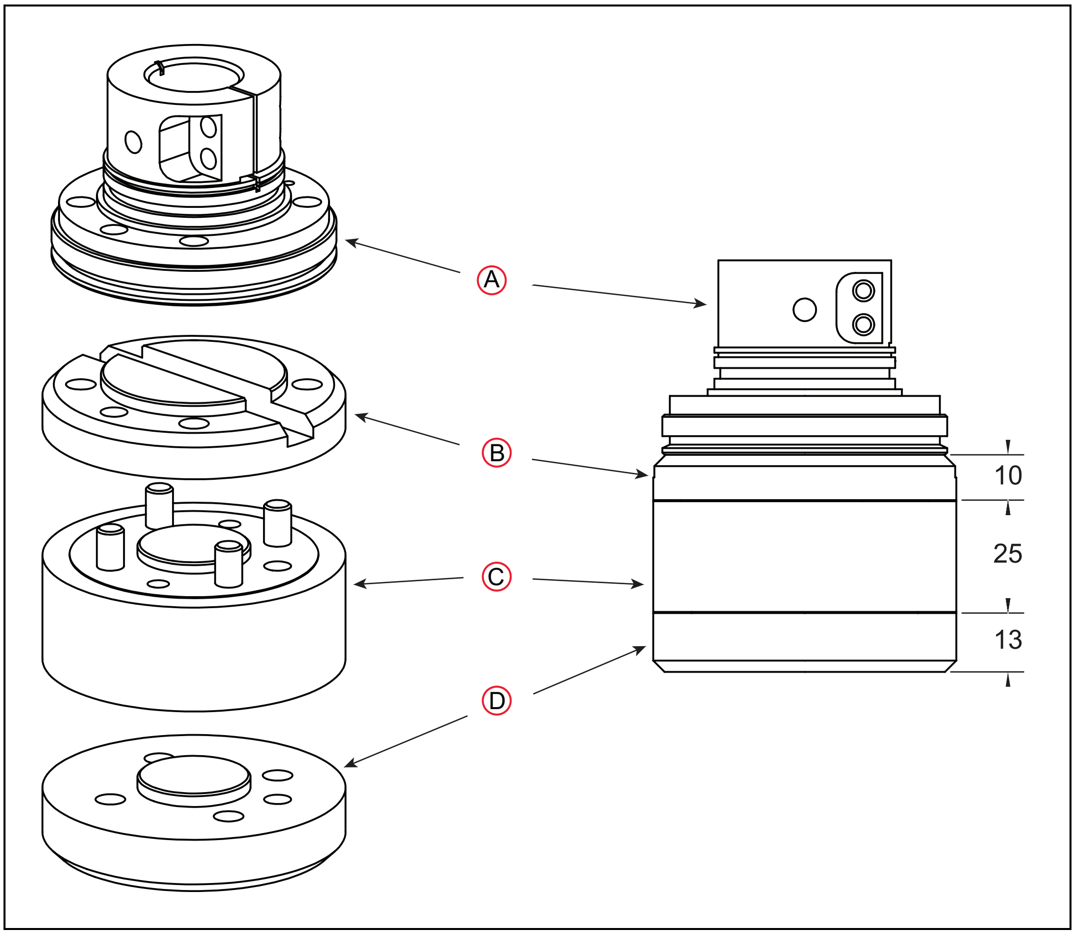


Figure 4-4. Side View, eCobra, Quattro, and Hornet Sensor, Adapters, and Robot Flange. Units in mm.

Key	Meaning	Key	Meaning
A	Robot Tool Flange	C	Sensor
B	Inner Adapter	D	Outer Adapter

1. Install an M6 dowel pin into the robot tool flange 6 mm hole.

   This will match a corresponding hole in the inner adapter plate.

2. Install an M6 dowel pin in the sensor side of the inner adapter plate.

   This will match a 6 mm hole in the sensor. (There are two 5 mm holes, too.)

3. Attach the inner adapter plate to the robot tool flange.

   One side of the adapter matches the hole pattern of the tool flange.

   The channel for the cable goes against the tool flange. See the following figure.

   For a Quattro or Hornet robot, route the RJ11 end of the Upper Cobra cable down through the hole in the platform and tool flange.

   For all robots, lay the cable in the inner adapter plate channel so the RJ11 end of the cable exits the channel on the side nearest the sensor dowel pin. Leave 140 mm of the cable extending beyond the edge of the adapter plate, measured from where the cable enters the RJ11 connector. The cable will be connected later.

   Use four of the provided M6 x 12 screws to attach the adapter plate to the tool flange.

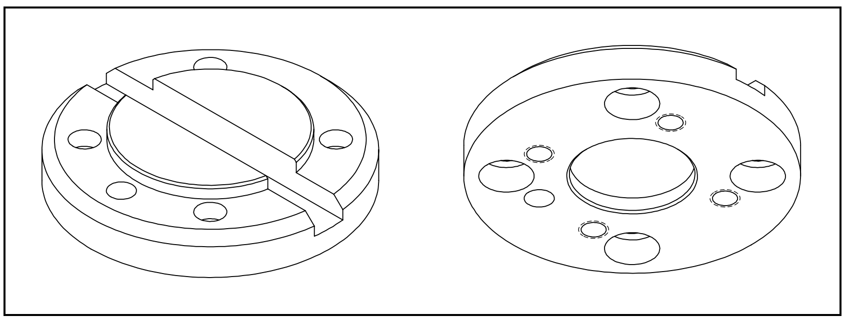


Figure 4-5. eCobra/Quattro/Hornet Inner Adapter Plate.
The Robot Side is shown on the Left, the Sensor Side on the Right.

4. Attach the force sensor to the inner adapter plate.

   The raised center of one side of the sensor fits into the inner adapter plate.

   The sensor uses M6 captive button-head screws to mount to the inner adapter plate. The screws are tightened with a hex key (provided with the sensor) through the screw holes in the tool side of the sensor. See Figure 4-1.

   CAUTION: SENSOR DAMAGE
   Do not turn one sensor screw all the way in at a time. This can damage the sensor.

   In many cases these screws will protrude from the sensor even when fully retracted.

   a. Turn each screw in a few turns at a time, then go to the next screw.

      Repeat this with all screws until the sensor mounting surface is flat against the adapter plate mounting surface.

b. Torque the screws in three stages until 10 N·m (88 in-lb) is reached.

5. Install an M6 dowel pin in the gripper side of the sensor.

   This will match a 6 mm hole in the outer adapter plate.

6. Attach the outer adapter plate to the gripper side of the force sensor.

   The raised center of one side of the outer adapter plate fits into the force sensor. The other has the same mounting holes as the robot's tool flange, so you can mount your grippers to it.

   Use the remaining four provided M6 x 12 screws for this.

CAUTION: SENSOR DAMAGE
The maximum allowed screw penetration into the gripper side of the sensor is 8 mm. More penetration into this side will damage the sensor.

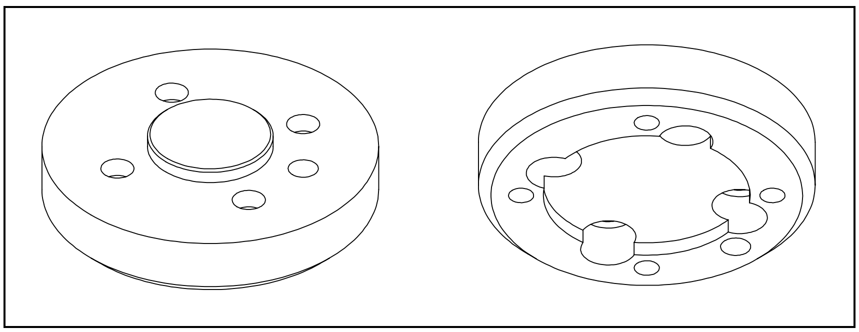


Figure 4-6. eCobra/Quattro/Hornet Outer Adapter Plate.
The Sensor Side is shown on the Left, the Gripper Side on the Right.

Torque these screws in three stages until 10 N·m (88 in-lb) is reached.

A fourth M6 dowel pin is provided, which will fit in the 6 mm hole in the outer adapter plate. This matches where a dowel pin would be if the sensor and adapter plates were not present, and is for alignment of your gripper.

eCobra Force Sensor Cabling

NOTE: The DeviceNet port on eCobras will no longer be available after installation of an Intelligent Force Sensing system.

From	To	Cable
Sensor	User Electrical, outer link	Upper Cobra cable, RJ11 6-pin to DB25M
User Electrical, outer link (J2)	User Electrical, Robot base (J1)	internal to robot
User Electrical,	XBELTIO cable,	Lower Cobra

From	To	Cable
Robot base	EXPIO branch	cable, DB9 to DB25
XBELTIO cable, HDB26F end	eAIB, XBELTIO port	XBELTIO cable, DB9 - HDB26F

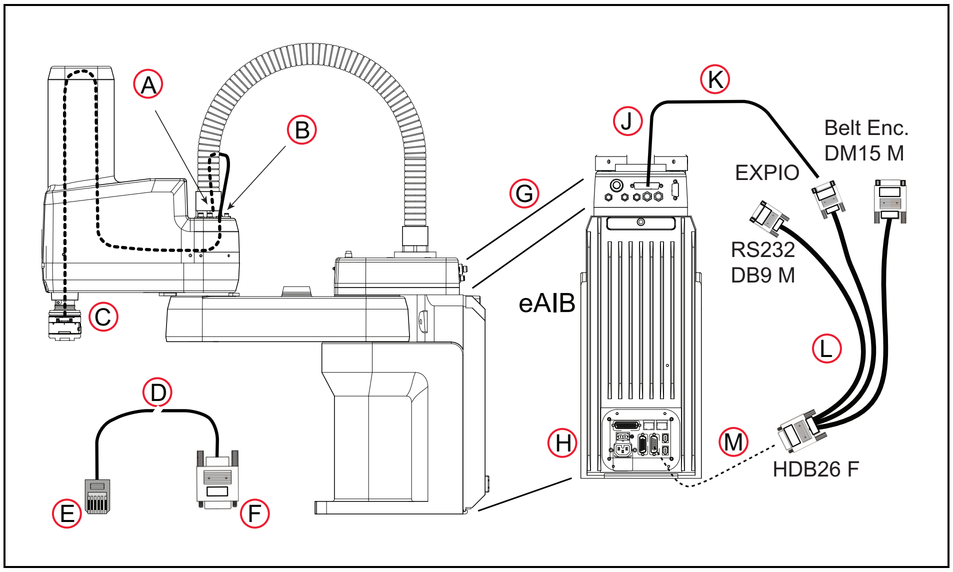


Figure 4-7. eCobra Cabling Overview

Key	Meaning	Key	Meaning
A	J2 User Electrical	G	J1 Connectors
B	Grommet in DeviceNet Hole	H	Robot Interface Panel
C	RJ11, 6-pin, at Sensor	J	to User Electrical DB25
D	Upper Cobra Cable	K	Lower Cobra Cable
E	Sensor	L	XBELTIO Cable
F	User Electrical	M	to XBELTIO

1. Plug the RJ11 plug into the force sensor.
2. Loosen the strain relief tab, using a 2 mm hex key.
3. Route the Upper Cobra cable under the strain relief tab, and tighten the tab on the cable. The cable should be snug around the force sensor up to the strain relief tab.

   There will be a small service loop in the Upper Cobra cable as it enters the channel in the inner adapter plate.
4. Plug the DB25 end of the cable into the robot base.

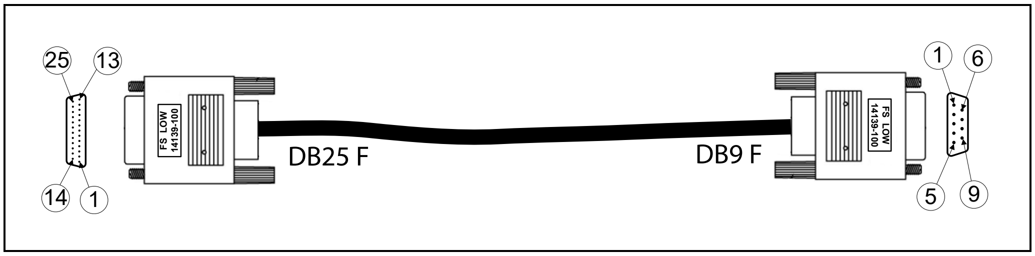


Figure 4-8. Lower Cobra Cable

5. Plug the DB9 end of the cable into the EXPIO branch of the XBELTIO cable.

6. Plug the DB25 end of the XBELTIO cable into the XBELTIO port on the eAIB.

Quattro/Hornet Force Sensor Cabling

NOTE: The Quattro, Hornet, and eCobra robots use the same cables for the force sensor installation.

From	To	Cable
Sensor	Lower Cobra cable	Upper Cobra cable, RJ11 6-pin to DB25M
Upper Cobra cable	XBELTIO cable, EXPIO branch	Lower Cobra cable, DB9 to DB25
XBELTIO cable, HDB26F end	eAIB, XBELTIO port	XBELTIO cable, DB9 - HDB26F

1. Plug the RJ11 plug into the force sensor.

2. Loosen the strain relief tab, using a 2 mm hex key.

3. Route the Upper Cobra cable under the strain relief tab, and tighten the tab on the cable. The cable should be snug around the force sensor up to the strain relief tab.

   There will be a small service loop in the Upper Cobra cable as it enters the channel in the inner adapter plate.

4. Connect the DB25 ends of the Upper Cobra cable and Lower Cobra cable.

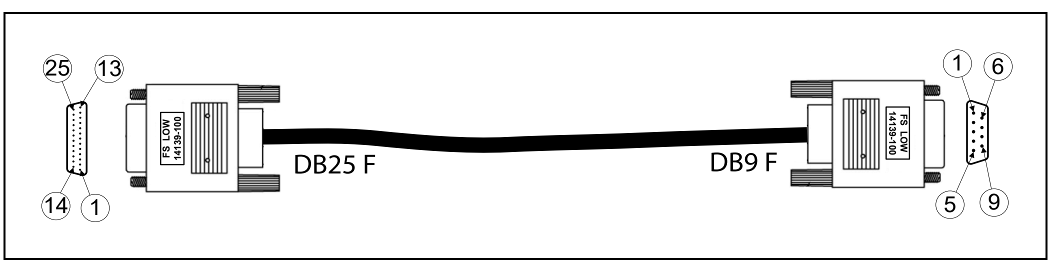


Figure 4-9. Lower Cobra Cable

The thumbscrews from each of the DB25 connectors screw into one of the two hex standoffs provided. See the following figure.

Figure 4-10. Top: Standoffs Between Ends of DB25 Connectors. Bottom: Assembled Connectors.

5. Route the connected cables from the platform to one of the outer arms.

   Cable-tie the cable to the outer arm at two locations on the outer arm.

   It is important to route the cable so that it does not interfere with the platform-outer arm ball joints.

6. Route the cable straight up that outer arm.

7. Attach an adhesive cable tie anchor midway up the upper flat surface of the inner arm that is connected to the outer arm where you attached the cable.

   Route the cable up to this anchor, and attach it there.

8. Plug the DB9 end of the Lower Cobra cable into the EXPIO branch of the XBELTIO cable.

9. Plug the DB25 end of the XBELTIO cable into the XBELTIO port on the eAIB.

10. Attach a second adhesive cable tie anchor to the top of the robot base casting surface near the eAIB.

    Cable-tie the XBELTIO cable to that anchor.

11. Secure any excess cable so that it will not interfere with the movement of the arms.

Viper 650 and Viper 850 Robots

Viper Force Sensor Kit

- Force Sensor
- Viper Force Sensor Parts Kit

Viper Force Sensor Parts Kit

- User's Guide
- XBELTIO Cable (DB9 M to eAIB or eMB-60R XBELTIO)
- Adapter Plate (inner)
- Adapter Plate (outer)
- Screws for inner adapter plate, four M5 x 16 mm
- Screws for outer adapter plate, four M6 x 12 mm
- Dowel pins, H6, two M5 x 12 mm
- Dowel pins, H6, two M6 x 12 mm
- Upper Viper Cable (RJ11 6-pin to CN21)
- Lower Viper Cable (CN20 to XBELTIO EXPIO- DB9 F)

Pinouts

Table 4-4. Upper Viper Cable

RJ11	Hirose 10 M (CN21)	Description
Pin 1	Pin 1	CLK+
Pin 2	Pin 2	CLK-
Pin 3	Pin 3	9 V
Pin 4	Pin 4	Gnd
Pin 5	Pin 5	Data+
Pin 6	Pin 6	Data-

Table 4-5. Lower Viper Cable

Hirose 24 F (CN20)	DB9 F	Description
Pin 1	Pin 1	CLK+
Pin 2	Pin 2	CLK-
Pin 3	Pin 6	9 V
Pin 4	Pin 5	Gnd
Pin 5	Pin 8	Data+
Pin 6	Pin 9	Data-
Shell	Shell	Shield

Viper Force Sensor and Adapter Plates

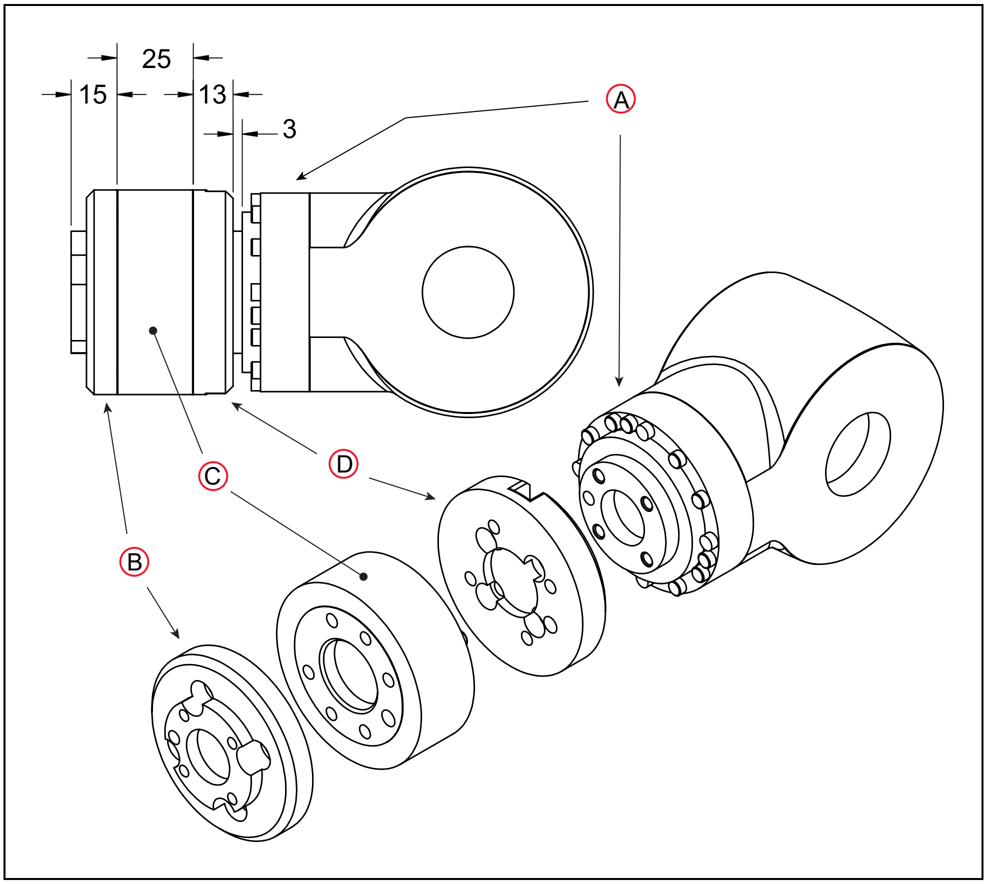


Figure 4-11. Side View for Viper, Sensor, Adapters, and Robot Flange. Units in mm.

Key	Meaning	Key	Meaning
A	Robot Flange	C	Sensor
B	Outer Adapter	D	Inner Adapter

This procedure involves the installation of the inner adapter plate, the force sensor, and the outer adapter plate.

1. Install an M5 dowel pin into the robot tool flange 5 mm hole.
2. Install an M6 dowel pin in the sensor side of the inner adapter plate.

   This will match a 6 mm hole in the sensor. (There are two 5 mm holes, too.)
3. Attach the inner adapter plate to the tool flange.

   One side of the adapter matches the hole pattern of the tool flange. The 5 mm hole in the adapter plate lines up with the dowel pin.

   Lay the cable in the inner adapter plate channel, from one side to the other, before the plate is attached, so the RJ11 end of the cable exits the channel on the side nearest the sensor dowel pin. Leave 178 mm of the cable extending beyond the edge of the adapter plate, measured from where the cable enters the RJ11 connector. The cable will be connected later.

Use the four provided M5 x 16 screws for this.

See the following figure.

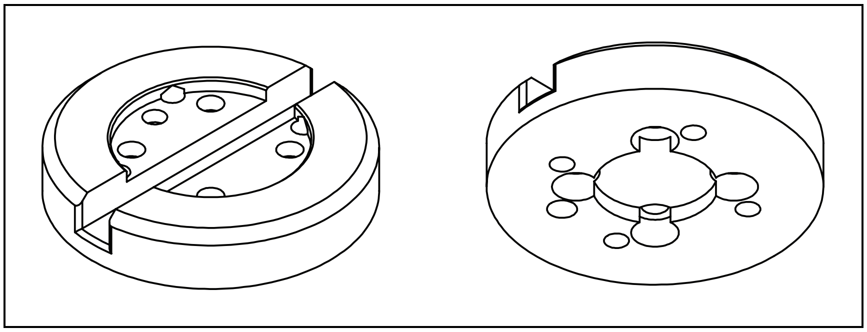


Figure 4-12. Viper 650/850 Inner Adapter Plate.
The Robot Side is shown on the Left, the Sensor Side on the Right.

4. Attach the force sensor to the inner adapter plate.

   The raised center of one side of the sensor fits into the inner adapter plate.

   The sensor uses M6 captive button-head screws to mount the sensor. The screws are tightened with a hex key (provided with the sensor) through the screw holes in the gripper side of the sensor. See Figure 4-1.

   CAUTION: SENSOR DAMAGE
   Do not turn one sensor screw all the way in at a time. This can damage the sensor.

   In many cases these screws will protrude from the sensor even when fully retracted. Turn each screw in a few turns at a time, then go to the next screw until the sensor mounting surface is flat against the adapter mounting surface. Then tighten the screws in a sequence, moving from side to side. Torque these screws in three stages until 10 N·m (88 in-lb) is reached.

5. Install an M6 dowel pin in the gripper side of the sensor.

   This will match a 6 mm hole in the outer adapter plate.

6. Attach the outer adapter plate to the gripper side of the force sensor.

   The raised center of one side of the outer adapter plate fits into the force sensor. The other has the same mounting holes as the robot’s tool flange, so you can mount your grippers to it.

   Use the four provided M6 x 12 screws for this.

CAUTION: SENSOR DAMAGE
The maximum allowed screw penetration into the gripper side of the sensor is 8 mm (0.32 inch). More penetration into this side will damage the sensor.

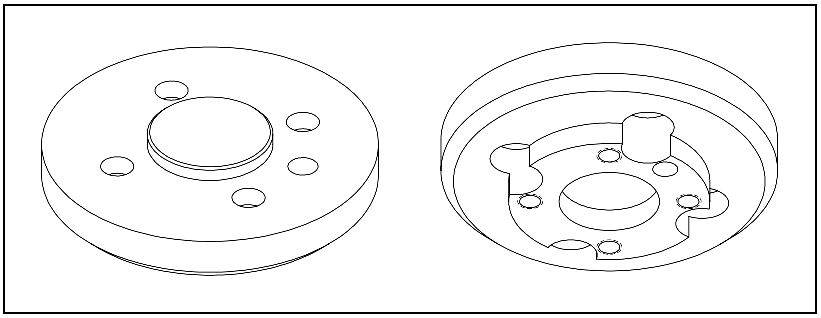


Figure 4-13. Viper 650/850 Outer Adapter Plate.
The Sensor Side is shown on the Left, the Gripper Side on the Right.

Torque these screws in three stages until 10 N·m (88 in-lb) is reached.

A second M5 dowel is provided, which will fit in the 5 mm hole in the outer adapter plate. This matches where a dowel pin would be if the sensor and adapter plates were not present, and is for alignment of your gripper.

Viper 650 and 850 Force Sensor Cabling

NOTE: The Force Sensor uses the CN20 connector on the robot's base. This will prevent a user from using an end-of-arm solenoid, because the Force Sensor and the solenoid need the same connector.

From	To	Cable
Sensor	CN21, upper link	Upper Viper Cable, RJ11 6-pin to CN21
CN21, upper link	CN20, robot base	Wired inside robot
CN20, robot base	XBELTIO cable, EXPIO branch	Lower Viper Cable, CN20 - DB9
XBELTIO cable, HDB26 F	eMB-60R, XBELTIO port	XBELTIO

Externally, the sensor cabling needs to connect the sensor to the CN21 connector on the robot upper link. That is pre-wired inside the robot to the robot’s base, where it comes out as the CN20 connector. From there, it needs to be connected to the XBELTIO port on the eMB-60R amplifier.

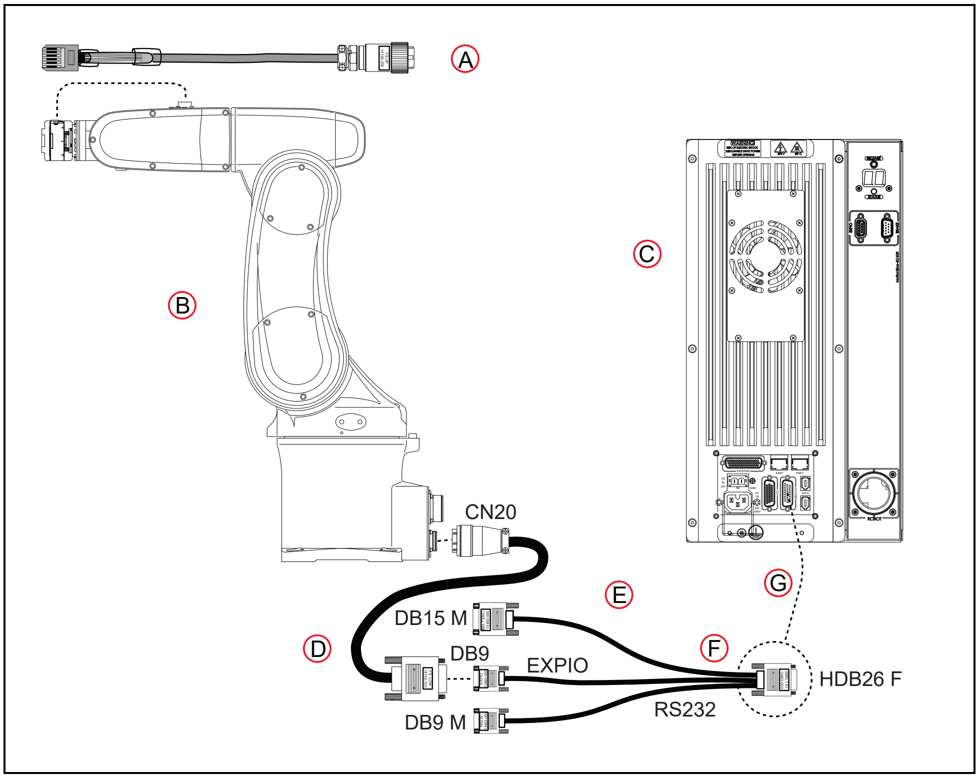


Figure 4-14. Viper Cabling Overview

Key	Meaning	Key	Meaning
A	Upper Viper Cable	E	XBELTIO Cable
B	Viper Robot	F	Belt Encoder
C	eMB-60R	G	to XBELTIO
D	Lower Viper Cable		

1. Connect the RJ11 6-pin end of the Upper Viper cable to the sensor.
2. Loosen the strain relief tab, using a 2 mm hex key.
3. Route the Upper Viper cable under the strain relief tab, and tighten the tab on the cable. The cable should be snug around the force sensor up to the strain relief.

   There will be a small service loop in the Upper Viper cable as it enters the channel in the inner adapter plate.
4. Plug the CN21 end of the Upper Viper cable into the CN21 plug on link 4 of the robot.

   From there, the cabling is pre-wired inside the robot to the CN20 plug on the robot base.
5. Plug the CN20 end of the Lower Viper cable into the robot base.
6. Plug the DB9 end of the cable into the EXPIO branch of the XBELTIO cable.
7. Plug the DB25 end of the XBELTIO cable into the XBELTIO port on the eMB-60R.

Chapter 5: Configuration and Programming

Prolonging Cable Life

CAUTION: CABLE DAMAGE
Rotating a force sensor beyond 180° in either direction or using excessive acceleration or deceleration will shorten the life of cables connecting the force sensor to the amplifier. These behaviors will have the most effect on the cable nearest the sensor.

Cable life can be extended by adding a SINGLE ALWAYS to any program with a force sensor, which will limit the rotation to ±180°. As long as your cabling can go ±180°, this keyword will keep it from rotating more than that. This isn't mandatory, but will help prolong your cable life. This is more important for Viper robots than the other robots that support a force sensor.

Syntax

SINGLE ALWAYS

Function

Limit rotations of the robot wrist joint to the range -180 to +180°.

5.1 Intelligent Force Sensing and eV+

Intelligent Force Sensing is tightly integrated into eV+ for a convenient, high-performance user interface. In addition to augmenting the functionality of existing eV+ instructions for force sensing, Intelligent Force Sensing incorporates several eV+ keywords dedicated to force sensing. A summary of all the keywords related to force sensing is shown in the following table. The sections that follow describe the keywords in more detail. This is followed by a section containing example programs for a simple Intelligent Force Sensing application.

Table 5-1. Intelligent Force Sensing Keywords

eV+ Keyword	Description
ALTER	Specify the magnitude of the real-time path modification that is to be applied to the robot path during the next trajectory computation.
FORCE.FRAME	Defines Force Reference Frame (FRF), the system of coordinates in which forces are measured.
FORCE.MODE	Controls Guarded and Protect modes of operation.
FORCE.OFFSET	Offsets force readings, usually to compensate for loads held by the robot. This instruction is primarily used to set the zero value for the force sensor (calibration).
FORCE.READ	Returns force readings.

eV+ Keyword	Description
ID	Returns force system ID information. See ID Real-Valued Function on page 40 for more information concerning this keyword.
LATCH LATCHED #PLATCH	Returns Guarded mode trip information. See LATCH Transformation Function, LATCHED Real-Valued Function, and #PLATCH Precision-Point Function on page 41 for more information on these keywords.
SELECT	Selects the force sensor to be used by a given program task in multi-sensor systems. See SELECT Real-Valued Function and SELECT Program Instruction and Monitor Command on page 42 for more information concerning this keyword.
STATE	Returns the robot state, which indicates if the robot is decelerating or stopped due to a Guarded mode trip. See STATE Function on page 42 for more information concerning this keyword.

5.2 Standard eV+ Keywords Used with Force Sensing

This section summarizes the standard eV+ keywords that have special functions when used with an Intelligent Force Sensing system. See the eV+ *Language Reference Guide* for details on each of these keywords.

ID Real-Valued Function

The ID real-valued function has been augmented to return information about the currently-selected force sensor, using device number 50. See the SELECT program instruction documentation (SELECT Real-Valued Function on page 42) for an explanation of selecting among multiple force sensors. The acceptable values for the component parameter and the corresponding data returned are listed in the following table.

Table 5-2. ID function use with Intelligent Force Sensing

Component	Result of ID (component, 50)
1	Model number of force sensor (0 if no force sensor is connected)
2	Serial number of force sensor (0 if no force sensor is connected)

LATCH Transformation Function

The LATCH(1) transformation function returns the location of the robot at the last Guarded mode trip. (The LATCHED(1) real-valued function can be used to determine if a Guarded mode trip has occurred.) A program example follows.

```
MOVE goal                    ;Start move to goal
FORCE.MODE (1) ^H30, 1       ;Trip at 1 Newton of force
WAIT STATE(2) <> 1           ;Wait for motion to
                             ; terminate
FORCE.MODE (-1)              ;Disable Guarded mode
IF LATCHED(1) THEN           ;If Guarded trip,
SET trip.loc = LATCH(1)      ; record trip location,
FORCE.READ (1) tripforce[1]  ; and log trip forces
TYPE “Tripped at Z-height:”, DZ(trip.loc)
TYPE “Tripped at Z-force:”, tripforce[3]
END
```

LATCHED Real-Valued Function

The LATCHED(1) real-valued function returns the status of the last stop-on-force trip. See the LATCH and #PLATCH function descriptions for programming examples.

Value	Meaning
1	Guarded mode trip condition 1 has tripped
2	Guarded mode trip condition 2 has tripped
n[1]	Stop-on-force has occurred due to an external digital signal interrupt
0	No trip has occurred since the last time the LATCHED(1) function was used

[1]n is the signal number that was detected.

#PLATCH Precision-Point Function

The #PLATCH(1) precision-point function returns the joint angles of the robot at the last Guarded mode trip. The LATCHED(1) real-valued function can be used to determine if a Guarded mode trip has occurred. A program example follows.

```
MOVE goal                    ;Start move to goal
FORCE.MODE (1) ^H30, 1       ;Trip at 1 Newton of force
WAIT STATE(2) <> 1           ;Wait for motion to terminate
FORCE.MODE (-1)              ;Disable Guarded mode
IF LATCHED(1) THEN           ;If Guarded trip,
SET #joints = #PLATCH(1)     ; record trip location,
DECOMPOSE j[1] = #joints     ; put into array,
FORCE.READ (1) tripforce[1]  ; and log trip forces
TYPE “Tripped at Joint 3 position:”, j[3]
TYPE “Tripped at Z-force:”, tripforce[3]
END
```

SELECT Real-Valued Function

The SELECT real-valued function returns the number of the selected force sensor for the current task, or the total number of force sensors available. Program examples:

```
our_sensor = SELECT(FORCE)      ;Force sensor number
                                ;  selected by this task
num_sensors = SELECT(FORCE,-1)  ;Total number of force
                                ;  sensors
```

SELECT Program Instruction and Monitor Command

The SELECT program instruction selects the number of the force sensor for access by the current task. It only needs to be used if there are multiple force sensors connected to your system controller. Multiple tasks can have the same force sensor selected at any one time.

The following program example selects force sensor 2, and might output something like “Current forces on sensor 2 are 0.1 -1.2 0.4”.

```
SELECT FORCE = 2       ;Select force sensor 2
FORCE.READ (0) f[]     ;Read sensor 2 forces
TYPE “Current forces on sensor”, SELECT(FORCE), /S
TYPE “ are ”, /F0.1, f[0], /X1, f[1], /X1, f[2]
```

STATE Function

These two STATE function values return information on stop-on-force robot motion termination. They are:

- STATE(2) == 9 Decelerating due to a stop-on-force condition.
- STATE(2) == 10 Stopped due to a stop-on-force condition.

eV+ automatically waits for the robot to stop (STATE(2) == 10) after a force trip before the next motion begins.

5.3 Dedicated eV+ Keywords for Force Sensing

The keywords listed on the following pages are dedicated for use by eV+ systems equipped with the Intelligent Force Sensing system.

FORCE.FRAME Program Instruction

Syntax

FORCE.FRAME (mode) trans_write, trans_read

Function

Set or read the transform from the robot tool flange to the Force Reference Frame or to the Sensor Reference Frame.

Usage Considerations

This option is available only if your system is equipped with the Intelligent Force Sensing option.

This instruction applies only to the currently-selected force sensor. In multiple force-sensor systems, the SELECT program instruction can be used to change the selected force sensor.

Parameters

mode Real value, variable, or expression that specifies which reference frame to define. Allowed values of mode follow:

Mode	Function	trans_write	trans_read
1	Set tool-to-FRF transform	Required	Unused
-1	Read tool-to-FRF transform	Unused	Required
2	Set tool-to-sensor transform	Required	Unused
-2	Read tool-to-sensor transform	Unused	Required

trans_write Transform variable or expression required only if mode is positive. This parameter must be a standard transformation, with distances in millimeters and angles in degrees.

trans_read Transform variable or expression required only if mode is negative. This parameter must be a standard transformation, with distances in millimeters and angles in degrees.

Details

A typical robot tool flange may have axes that do not align with the axes in which forces are computed by the force sensor (the Sensor Reference Frame, or SRF). By properly setting the transform from the robot tool flange to the sensor reference frame, pushing on the tool tip in the tool X, Y, or Z direction will cause the force sensor to report a force in the same direction.

Example

The following program line initializes the transformation from the nominal robot flange tool frame to the sensor reference frame, assuming that the sensor frame is displaced by a 12.5 mm tool Z translation, followed by a 90 degree tool Z rotation, followed by a 180 tool Y rotation. Such a tool-to-SRF transform is typically set during force system initialization.

```
FORCE.FRAME (2) TRANS(,,12.5,90,180     ;Tool-to-SRF transform
```

Once the tool-to-SRF transform sets the offset from the robot tool frame to the sensor frame, forces are reported in the robot tool frame. When the robot tool frame is selected to align with the tool tip (using the TOOL instruction), force readings will also align with the tool tip if the user specifies “FORCE.FRAME TOOL”.

As an example, the following program segment might be used to align both the robot TOOL frame and the Force Reference Frame with the tool tip:

```
TOOL TRANS(,,40)        ;Tool tip is 40 mm beyond tool flange
FORCE.FRAME (1) TOOL    ;Align force frame with tool tip
```

Related Keywords

NULL (program instruction)

SELECT (program instruction)

TOOL (transformation function)

TRANS (transformation function)

FORCE.MODE Program Instruction

Syntax

FORCE.MODE (**mode**, , stt) parm0, parm1, parm2, parm3

Function

Sets up and controls force operating modes. Configures and enables Guarded and Protect modes, and allows changing various force system defaults.

Usage Considerations

All mode options are available if your system is equipped with the Intelligent Force Sensing option. Only modes 2 and -2 (stop-on-digital-signal) can be used without the Intelligent Force Sensing system.

When used with the Intelligent Force Sensing system, this instruction applies only to the currently-selected force sensor. In multiple force-sensor systems, the SELECT program instruction can be used to change the selected force sensor.

Parameters

mode	Real value, variable, or expression that specifies which operation to perform. Valid mode values are listed in the table that follows.
stt	Optional real variable that is set to an eV+ error code if an error occurred in execution of this instruction. This variable is set to 0 if no error occurs. If this parameter is omitted, any error stops program execution.
parm0, parm1, parm2, parm3	Optional real values, variables, or expressions, whose values are interpreted as described in the following table.

Details

This instruction has two primary functions. First, it controls the force system operating modes, such as Guarded or Protect modes. Second, it is used to change various force system operating parameters such as force filters, force units, etc. The function to be performed depends on the mode parameter. All mode parameters are valid for an Intelligent Force Sensing system; only mode 2 can be used by a system not equipped with the Intelligent Force Sensing system.

The force system operating mode selections are shown in the following table.

Table 5-3. FORCE.MODE instructions for setting force operating modes

Mode	Description
1	Enables Guarded mode (mode value -1 disables it). This causes the currently-SELECTed robot to come to an immediate halt when either of two trip conditions is tripped. The trip conditions are configured if any of the parm values are specified; otherwise, the previous configuration is left unchanged. The meanings of the parm values for mode 1 are shown in Table 5-4. , and the bit descriptions are documented in Figure 3-1. After a Guarded mode trip occurs, the LATCHED(1) function returns the number

Mode	Description
	of the trip condition (either 1 or 2), the LATCH(1) transformation function returns the trip location, the #PLATCH(1) precision point function returns the joint angles of the trip, and the FORCE.READ(1) instruction returns the force readings at the instant of the trip. Guarded mode must be re-enabled with another FORCE.MODE (1) instruction before another trip can occur.
2	Enables Stop-on-digital-signal mode (mode value -2 disables it). This causes the currently SELECTed robot to come to an immediate halt when a digital input transitions from low-to-high. The digital input must be configured to cause a position latch on the servo boards using the ACE software. After a Stop-on-digital-signal occurs, the LATCHED(1) function returns the value -1, the LATCH(1) transformation function returns the trip location, and the #PLATCH(1) precision point function returns the joint angles of the trip. Stop-on-digital-signal mode must be re-enabled with another FORCE.MODE(2) instruction before another trip can occur.
3	Enables Protect mode (mode value -3 disables it). When enabled, if the Protect mode trip force is exceeded, robot power is turned off. Power must then be turned back on (e.g., by typing ENABLE POWER) before the robot can be moved again. The trip conditions are configured if any of the parm values are specified. Otherwise, the previous configuration is left unchanged. The meanings of the parm values for mode 1 are shown in the following table. The bit descriptions are documented in Bit Settings for a Resultant Force Guarded Move on page 47 Once enabled, Protect mode remains enabled until explicitly disabled with a FORCE.MODE(-3) instruction.
4	Enables Soft Guarded mode (mode value -4 disables it). This is the same as mode 1, except when moving to the latched robot position, it will do it smoothly. Only use this if your application does not require instant guarded motion reaction, and can allow time for the robot to move to the guarded stop safely.

Table 5-4. Guarded Mode (mode value 1) parameter description

Parameter	Description
parm0	Bit settings that describe trip condition 1. The low 4 bits specify direction of force monitoring. The next 4 bits specify whether to monitor forces along a direction (1-D), in a plane (2-D), or as a force resultant (3-D). The highest three bits specify whether to monitor forces or moments, whether to trip when force exceeds or falls below the specified threshold, and whether to compare the threshold against the absolute value of the force readings. See Figure 3-1 for a description of valid bit settings.
parm1	Trip threshold (in force or moment units) for trip condition 1.
parm2	Bit settings that describe trip condition 2. See Figure 3.1 for a description of valid bit settings.
parm3	Trip threshold (in force or moment units) for trip condition 2.

An example of setting the trip flag bits (arguments parm0 or parm2) for a guarded move is shown in the following figure. In the example, it is assumed that the user wants to perform a guarded move in which the robot is to stop when the force resultant exceeds a threshold value. In this case, the direction bits are ignored, the dimension bits have the value 3 (for 3-D), and the high bits are all clear, for a net value of ^H30 (^H is eV+ notation for hexadecimal).

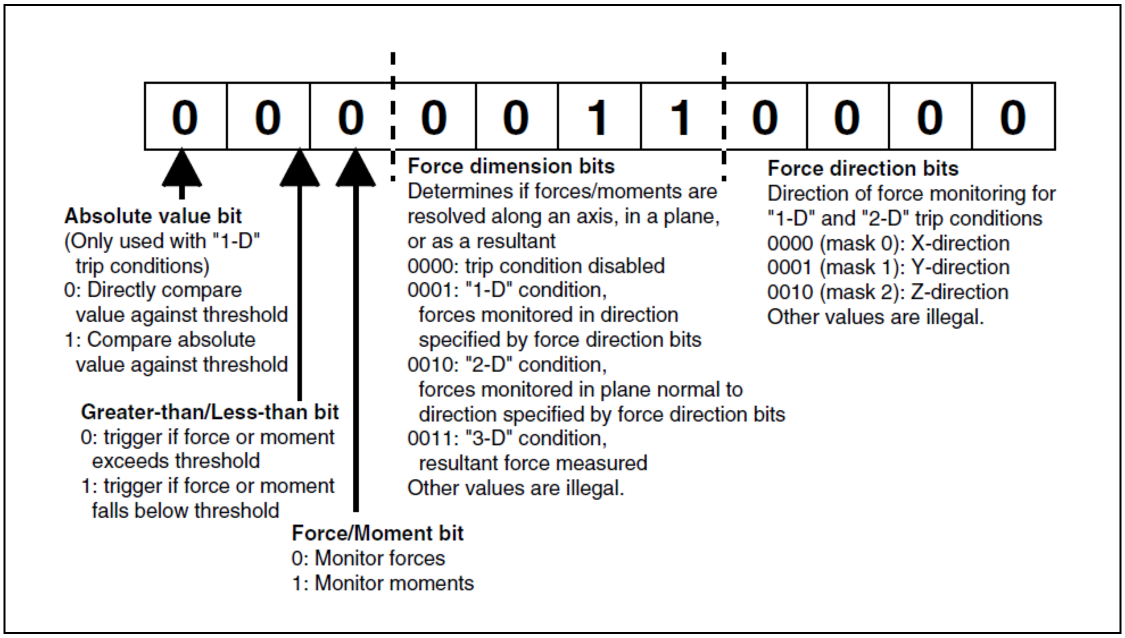


Figure 5-1. Bit Settings for a Resultant Force Guarded Move

Other mode values allow the user to change force defaults (see the following table) or to perform various functions (see Table 5-6.).

Table 5-5. FORCE.MODE instructions for changing force defaults

Mode	parm0 Description	Default value
10	User force scale. The sensor units are pounds. This default converts forces to Newtons. Change to 1 for pounds.	4.45
11	User length scale. The sensor units are inches. This default converts inches to mm. Change to 1 for inches.	25.4
12	Force filter number. This filter is used for data collection, current force readings, and Guarded moves. Acceptable values for filter numbers are 1 to 7: raw data corresponds to filter 1, heavily filtered data to filter 7.	2
13	Protect mode filter number. This filter is used for Protect mode. Use a low value for quickest response, a high value for least noise sensitivity. This value indicates the number of consecutive trips (computed each trajectory tick) it will take to trigger a protect mode. Must be > 0.	3
14	Force zero filter number. This filter is used when zeroing force readings. A high value provides the most accuracy but has higher time lag. The value here indicates the number of seconds the sensor will collect readings to calculate its zero. Must be > 0.	4

Table 5-6. FORCE.MODE instructions for miscellaneous functions

Mode	Description
20	Reset peak monitor levels to the current force and moment levels. Using the FORCE.READ instruction, you can find the minimum and maximum levels detected by the sensor since the last time this instruction was issued.
21	Reinitialize sensor. This puts the sensor back to its default state. The following operations are performed: • If Guarded move is enabled, it is tripped. • Protect mode is disabled. • The force filter and user force and length units are reset to their default levels. • All force offsets are cleared. • The default Force Reference Frame is restored.

Examples

The following program segment configures and enables Protect mode so that high power is turned off if the resultant force exceeds forty Newtons.

```
FORCE.MODE (3) ^h30, 40
```

Protect mode can be disabled with the instruction:

```
FORCE.MODE (-3)
```

The following program segment starts a move, then configures and enables Guarded mode so that the robot comes to a high-speed stop if the absolute value of the Z-direction force exceeds 2.5 Newtons:

```
MOVE goal              ;Start motion
FORCE.MODE (1) ^H412, 2.5 ;Enable Guarded mode
WAIT STATE(2) <> 1     ;Wait till move terminates
FORCE.MODE (-1)        ;Disable Guarded mode
trigger = LATCHED(1)   ;Determine if trip occurred
```

Configuration of an operating mode is “sticky”; that is, it can be re-enabled without reconfiguring. So the operation above can be restarted with the program segment:

```
MOVE goal              ;Start motion
FORCE.MODE (1)         ;Re-enable Guarded Mode
```

The following is an example of changing the force and moment scaling factors from English to Metric units. Once changed, all subsequent force and moment readings are returned in the new units.

```
FORCE.MODE (10) 1      ;Use sensor raw data of pounds
FORCE.MODE (11) 1      ;Use raw sensor data of inches
```

An example of using the Stop-on-digital-signal capability is shown in the following example. It can be used with a touch sensor to stop the robot quickly on contact.

```
MOVE goal              ;Start motion
FORCE.MODE (2)         ;Enable Stop-on-digital-signal mode
WAIT STATE(2) <> 1     ;Wait till move terminates
FORCE.MODE (-2)        ;Disable Stop-on-digital-signal mode
trigger = LATCHED(1)   ;Determine if trip occurred
```

Related Keywords

FORCE.FRAME (program instruction)

FORCE.MODE (program instruction)

FORCE.OFFSET (program instruction)

LATCH (transformation function) LATCHED (real-valued function)

#PLATCH (precision-point function)

FORCE.OFFSET Program Instruction

Syntax

FORCE.OFFSET (mode, , stt) offset[ind]

Function

Set either temporary or permanent force offset. This instruction is primarily used to zero the force sensor readings.

Usage Considerations

This option is available only if your system is equipped with the Intelligent Force Sensing option.

This instruction applies only to the currently-selected force sensor. In multiple force-sensor systems, the SELECT program instruction can be used to change the selected force sensor.

FORCE.OFFSET 2 should be called as: FORCE.OFFSET (2) a[] Where a[] is a 6 entry input array of values to set the offset at.

Parameters

mode Real value, variable, or expression that specifies which offset operation to perform. Valid mode values are as follows:

0	Sets the permanent force zero by offsetting all future force readings. This is done at the driver level, and will take up to [Force zero filter] seconds, based on what was set in FORCE.MODE for the force zero filter number.
1	Sets a temporary force zero using the most recent force sensor reading. This is done by eV+.
2	Sets a temporary zero by offsetting force readings by the amount specified in the offset[ ] array, starting at index ind.

stt Optional real variable that is set to an eV+ error code if an error occurred in execution of this instruction. This variable is set to 0 if no error occurs. If this parameter is omitted, any error stops program execution.

ind Optional real value, variable, or expression that specifies the first array element in the offset[ ] array in which the force offsets are stored. If this parameter is omitted, the value 0 is assumed.

offset[ ] Optional real array containing the amount to offset each force reading (in user force and moment units) if mode has the value 2. If not supplied, all values are assumed zero. The force offset should be as follows:

offset[ind]	X-direction force offset
offset[ind+1]	Y-direction force offset
offset[ind+2]	Z-direction force offset
offset[ind+3]	Moment offset about X-axis
offset[ind+4]	Moment offset about Y-axis
offset[ind+5]	Moment offset about Z-axis

Details

The Intelligent Force Sensing system is normally used in relative force measurements, that is, the force difference between the instant before contact and after contact. A force offset can be used to cancel the weight of the tool to produce accurate relative force measurements. Either the permanent or temporary force zero offset functions (mode values 0 or 1) will zero the force readings for such relative force measurement. For most applications, you need only use the permanent force offset.

The use of a permanent offset lets you set a base offset, which you can easily return to by zeroing the temporary offset. The temporary offset is used to account for dynamic changes in the force sensor environment.

Example

The following program segment performs an approach to a target location, zeros the force sensor, then performs a Guarded mode move. The forces are zeroed after the final move begins, so that any forces incurred during the move to the approach location (such as cables pulling on the tool) are canceled:

```
APPROS target, 25      ;Straight-line approach to 25 mm
SPEED 10               ;Slow down for Guarded Move
MOVES target           ;Start straight-line move to target
FORCE.OFFSET (0)       ;Zero force readings
FORCE.MODE (1) ^H30, 3.1 ;Guarded move with trip
                       ; at 3.1 Newtons force resultant
FORCE.FRAME (program instruction)
FORCE.MODE (program instruction)
FORCE.READ (program instruction)
```

FORCE.READ Program Instruction

Syntax

FORCE.READ (mode, , stt) forces[ind]

Function

Returns current force readings, forces at instant of most recent Guarded move trip, current force offset, force sensor ratings, or peak force readings. All forces and moments are returned in user units, which can be changed with a FORCE.MODE instruction.

Usage Considerations

This option is available only if your system is equipped with the Intelligent Force Sensing option.

This instruction applies only to the currently-selected force sensor. In multiple force-sensor systems, the SELECT program instruction can be used to change the selected force sensor.

Parameters

mode Real value, variable, or expression that specifies which set of force values to return. Valid mode values are:

0	Return current force readings.
1	Return force readings at instant of most recent Guarded mode trip.
2	Return current force offset.
3	Return force sensor ratings. These are the maximum forces and moments that the sensor is designed to detect. The sensor will detect forces above these ratings, but accuracy may degrade.
4	Return minimum force readings since the peak detectors were last reset with a FORCE.MODE instruction.
5	Return maximum force readings since the peak detectors were last reset with a FORCE.MODE instruction.

stt Optional real variable that is set to a eV+ error code if an error occurred in execution of this instruction. This variable is set to 0 if no error occurs. If this parameter is omitted, any error stops program execution.

ind Optional real value, variable, or expression that specifies the first array element in the forces[ ] array in which the force readings will be returned. If this parameter is omitted, the value 0 is assumed.

forces [ind] Real array in which force readings will be returned. The forces are returned as follows:

forces[ind+0]	X-direction force
forces[ind+1]	Y-direction force
forces[ind+2]	Z-direction force
forces[ind+3]	Moment about X-axis
forces[ind+4]	Moment about Y-axis
forces[ind+5]	Moment about Z-axis

Details

The force readings returned by this instruction are always those applied BY the robot in the Force Reference Frame (FRF). Therefore, motion in the FRF +X direction will produce a positive X force if the robot contacts an object. By aligning the TOOL and FRF reference frames using the FORCE.FRAME instruction, motion in the TOOL +X direction will also produce positive X forces.

Example

The current forces can be continuously displayed with the program segment:

```
WHILE 1 DO
        FORCE.READ (0) f[]
        TYPE /F8.2, f[0], f[1], f[2], f[3], f[4], f[5]
END
```

Related Keywords

FORCE.FRAME (program instruction)

FORCE.MODE (program instruction)

FORCE.OFFSET (program instruction)

ALTER Program Instruction

Syntax

ALTER Dx, Dy, Dz, Rx, Ry, Rz

Function

Specify the magnitude of the real-time path modification that is to be applied to the robot path during the next trajectory computation. The coordinate values specified will automatically be superimposed on the nominal path computed by the eV+ trajectory generator. The corrections can be applied in all six degrees of freedom, and they can be specified as cumulative or non-cumulative values in World or Tool coordinates.

- Cumulative mode: Parameter values are interpreted as incremental and are accumulated (added to the previous values).
- Non-cumulative mode: Parameter values are interpreted as the total (noncumulative) correction to be applied to the trajectory. Previous values are discarded.

Use the ALTON instruction to enable alter mode, to select between cumulative and non-cumulative modes, and to select between WORLD and TOOL coordinate systems.

This instruction can be executed by the task that is controlling a robot in alter mode (see the ALTON instruction), or by any other task that has SELECTed the robot.

This instruction is ignored if the selected robot is not in alter mode.

When alter mode is enabled, this instruction should be executed once during each trajectory cycle. If this instruction is executed more often, only the last set of values defined during each cycle will be used.

Usage Considerations

This option is available only if your system is equipped with the Intelligent Force Sensing option.

This instruction applies only to the currently-selected robot.

Parameters

Dx Dy Dz Rx Ry Rz

Optional real values, variables, or expressions that define the translations along, and the rotations about, the X, Y, and Z axes.

In cumulative mode, omitted coordinates are interpreted as zero. In non-cumulative mode, omitted coordinates default to the values specified in the previous ALTER instruction.

Distances are interpreted as millimeters; angles are interpreted as degrees.

Details

After alter mode has been enabled (see the ALTON instruction), this instruction is executed once each trajectory-generation cycle to specify the amount by which the path is to be modified. The coordinates defined by this instruction are interpreted according to the modes specified by the ALTON instruction that initiated alter mode.

Example

The following pair of instructions can be embedded in a program loop that uses sensor data to control the motion of the robot.

```
; Modify the current robot path by (0.1*sx) millimeters
; along the X axis and (0.2*sz) millimeters along the
; Z axis,

; where "sx" and "sz" are variables that contain current
; sensor input data.

      ALTER 0.1*sx, , 0.2*sz

; Wait for the next trajectory cycle to give the trajectory

; generator a chance to execute.

      WAIT
```

Related Keywords

ALTON (program instruction)

ALTOFF (program instruction)

STATE (real-valued function)

5.4 eV+ Programming Examples

The following set of programming examples is designed to help you understand how to use the Intelligent Force Sensing system in a common insertion application. These programs are not supplied to you on the software disk, but you can enter them as you see them here to familiarize yourself with using the system.

Initialization Example

```
.PROGRAM demo.init()
; ABSTRACT: Demonstration of force system initialization.
;
; 1. Sets the robot tool-to-sensor transform. As an example, we will
;    assume the sensor is mounted on a 25.4 mm (1 inch) spacer from
;    the tool flange.
; 2. Enables Protect mode, setting the trip level at the sensor
;    Z force rating, so exceeding that level will cause the
;    robot to panic stop.

    AUTO ratings[5]

    FORCE.MODE (21) ;Re-initialize the sensor
    FORCE.FRAME (1) TRANS(,,25.4) ;Set tool-to-sensor transform

; Read the force ratings and enable protect mode at the
;  Z-force rating.

    FORCE.READ (3) ratings[]           ;Read force ratings
    FORCE.MODE (3) ^h30, ratings[2]    ;Enable protect mode

    RETURN
.END
```

Guarded Move During Insertion Example

```
.PROGRAM demo()
; ABSTRACT: Demonstration of performing a guarded move during
;           insertion.
;
; This program assumes the user is performing a vertical insertion,
; where if the forces are too high the insertion is jammed
; (force > 2 N). A high-speed approach to 50 mm above the assembly
; location "goal" is attempted first, followed by fast a approach
; to 2 mm above the insertion location and a slow guarded move to
; the goal location. Using a second, low approach point minimizes
; the length of the relatively slow guarded move.
;
; The programs "demo.init" and "demo.moveg" are described below.
; The program "acquire.part" is assumed to have been created by
; the user to acquire the part to be inserted.

 AUTO forces[5], trip, trip.pos

; Start by initializing the force sensing system. This should only
; have to be executed once after the system is booted to create
; necessary global variables.

    CALL demo.init()    ;Initialize force sensing system
    CALL acquire.part() ;Your custom acquisition routine

; Configure (but don't enable) Guarded mode using only one trip
;       condition:
```

```
; 1. Trip direction is Z-axis (mask value 2)
; 2. Dimension 1 to monitor force in specified direction
;       (mask value ^H10)
; 3. Trip on force (default) rather than moment
; 4. Trip when force is greater than specified threshold (default)
; 5. Trip threshold is 2 Newtons

   FORCE.MODE (-1) ^H12, 2
          ;Set trip flags and threshold--don't enable!

   SPEED 200   ;Perform high-speed initial approach
   APPROS goal, 50

   SPEED 50    ;Perform approach to point just above goal
   APPROS goal, 2

   SPEED 5     ;Perform actual guarded move at slow speed
   CALL demo.moveg(goal, TRUE, trip, forces[], trip.pos)
          ;Perform move
; Handle results of insertion attempt. If no trip occurred ("trip" has
; value 0), the insertion was a success and the part is released.

   IF NOT trip THEN ;Success
      TYPE "Insertion succeeded."
      OPENI   ;Release part
   ELSE ;Insertion jammed. Print trip position.
      TYPE "Insertion jammed", DISTANCE(goal,trip.pos)," mm from goal"
   END
   DEPART 50   ;Depart

   RETURN
.END
```

Guarded Move Example

```
.PROGRAM guardedmove()
;
; ABSTRACT: Will attempt to place a pointer in a hole by waiting for
;       the force to be below a threshold as it inserts
;* Copyright (c) 2020 by Omron Rob otics and Safety Technologies, Inc.

      AUTO count, inserted, trip

      ; Select force sensor and robot to use on a multirobot system
      DETACH ()
      SELECT ROBOT = 1
      ATTACH ()
      SELECT FORCE = 1

      ; Invoke tool transform for the mounting plates rotation
      FORCE.FRAME (1) TRANS(,,,,,112.5)
      FORCE.FRAME (2) TRANS(0)

      ; Make sure filter is set to 4 (out of 7)
      FORCE.MODE (12) 4

      ; Set force to lbs instead of N
      FORCE.MODE (10) 1
      ; Set length to inches instead of mm
      FORCE.MODE (11) 1

      ; Clear the latch buffer
```

```
WHILE LATCHED(1) DO
   WAIT
END

; Zero force sensor
FORCE.OFFSET (0)
WAIT.EVENT , 1         ;Leave time for the new data to go
                       ;through the filter
FORCE.OFFSET (1)       ;Zero offsets

; Disable previously invoked Protect mode and Guarded mode
FORCE.MODE (-3)
FORCE.MODE (-1)

; Enable protect mode at 15 lbs
FORCE.MODE (3) ^H30, 15

; Move to the starting position (defined outside of this
; program)
MOVE start
BREAK

; Initialize looping variables
count = 0
inserted = FALSE
trip = FALSE

; Try to insert the pointer into the hole 25 times
WHILE NOT inserted AND (count < 25) DO

   ; Do not shift on the first move to find the hole,
      ; after the first attempt, move over 1mm each attempt
   IF (count > 0) THEN
      ACCEL (0) 100, 100
      SPEED 100
      MOVE SHIFT(DEST BY 0, 1)
      BREAK
   END

   ; Initiate a guarded move that triggers at 1 lb of force
   FORCE.MODE (1) ^H30, 1
   WAIT

   ; Move slowly to prevent overshoot
   SPEED 2
   ACCEL (0) 50, 50

   ; Perform guarded move in Z moving down 25mm and stopping
      ; if guarded move is triggered
   MOVE SHIFT(start BY 0,count*1,-25)

   ; Wait for motion to terminate
   WAIT STATE(2) <> 1

   ; Set trip variable based on whether the guarded move
      ; was triggered or not
   trip = LATCHED(1)

   ; If the guarded move was not triggered, the 25mm Z move
      ; was succesful and the pointer was inserted
   IF NOT trip THEN
      TYPE "SUCCESS"
```

```
            inserted = TRUE
        ELSE
            ; If the guarded move was triggered, depart slightly,
            ; increment the counter, and try again
            ACCEL (0) 100, 100
            SPEED 100
            DEPART 0.5
            BREAK
            count = count+1
        END

        ; Disable Guarded Mode
        FORCE.MODE (-1)
    END

    ; Retract from the hole
    SPEED 50 ALWAYS
    DEPART 50
    BREAK

    ; Disable Protect mode
    FORCE.MODE (-3)

.END
```

Constant-Force Example

```
.PROGRAM forcemove()
;
; ABSTRACT: Will touch down on surface, and maintain a constant
; specified force by altering in Z
;
;* Copyright (c) 2020 by Omron Robotics and Safety Technologies, Inc.

    AUTO traj, forceval, offset, a[6]

    ; Select force sensor and robot to use on a multirobot system
    DETACH ()
    SELECT ROBOT = 1
    ATTACH ()
    SELECT FORCE = 1

    ; Invoke tool transform for the mounting plates rotation
    FORCE.FRAME (1) TRANS(,,,,,112.5)
    FORCE.FRAME (2) TRANS(0)

    ; Make sure filter is set to 1 (out of 7) for fastest
    ; response time
    FORCE.MODE (12) 1

    ; Set force to lbs instead of N
    FORCE.MODE (10) 1
    ; Set length to inches instead of mm
    FORCE.MODE (11) 1

    ; Move to starting location above the surface
    SPEED 50
    MOVE start
    BREAK

    ; Zero force sensor
```

```
FORCE.OFFSET (0)
WAIT.EVENT , 1          ;Leave time for the new data to go
                        ; through the filter
FORCE.OFFSET (1)        ;Zero offsets

; Enable protect mode at 25 lbs (it is always good to have
; extra safety)
FORCE.MODE (3) ^H30, 25
WAIT

; Initiate a guarded move that triggers at 1 lb of force to
; find the surface
FORCE.MODE (1) ^H30, 1

; Move slowly to prevent overshoot
SPEED 2
ACCEL (0) 10, 10

; Move down until guarded move is triggered
MOVES SHIFT(HERE BY ,,-25)

; Wait for motion to terminate
WAIT STATE(2) <> 1

; Begin altering in Z after the next move is commanded
ALTON () 1

; Obtain current trajectory rate to scale the gain relative to        ; obtain a constant speed
traj = 1/STATE(13)

; Compute time the move should take to allow sufficient
; adjustment in Z, this will vary based on the surface
DURATION (DISTANCE(end,HERE)*(0.016))*3

; Initiate move to end location
MOVES end

; Force in lbs. that is desired to be maintained
forceval = 3

; Loop until the move is complete, altering in Z the entire
; time based on the force sensor Z reading
WHILE (STATE(10) < 100) DO

  ; Read the force sensor
  FORCE.READ (0, , stt) a[]

  ; Compute the offset difference between the actual force
    ; and the desired force in Z
  offset = (-a[2]-forceval)

  ; Based on the offset, either adjust by moving up or down
    ; to compensate
  IF (offset < 0) THEN
    ALTER () , , 0.03*(traj/0.016)
    WAIT
  END
  IF (offset > 0) THEN
    ALTER () , , -1E-03*(traj/0.016)
    WAIT
  END
END
```

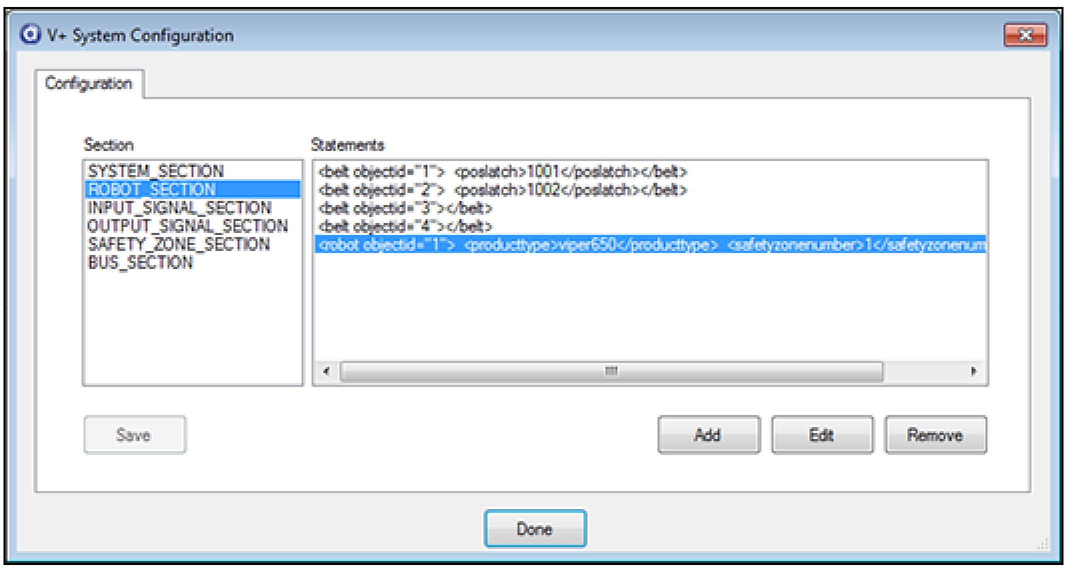
```
    ; Stop altering
    ALTOFF

    ; Disable protect mode
    FORCE.MODE (-3)

    DEPART 25
    BREAK

.END
```

5.5 ACE Software and Force Sensors

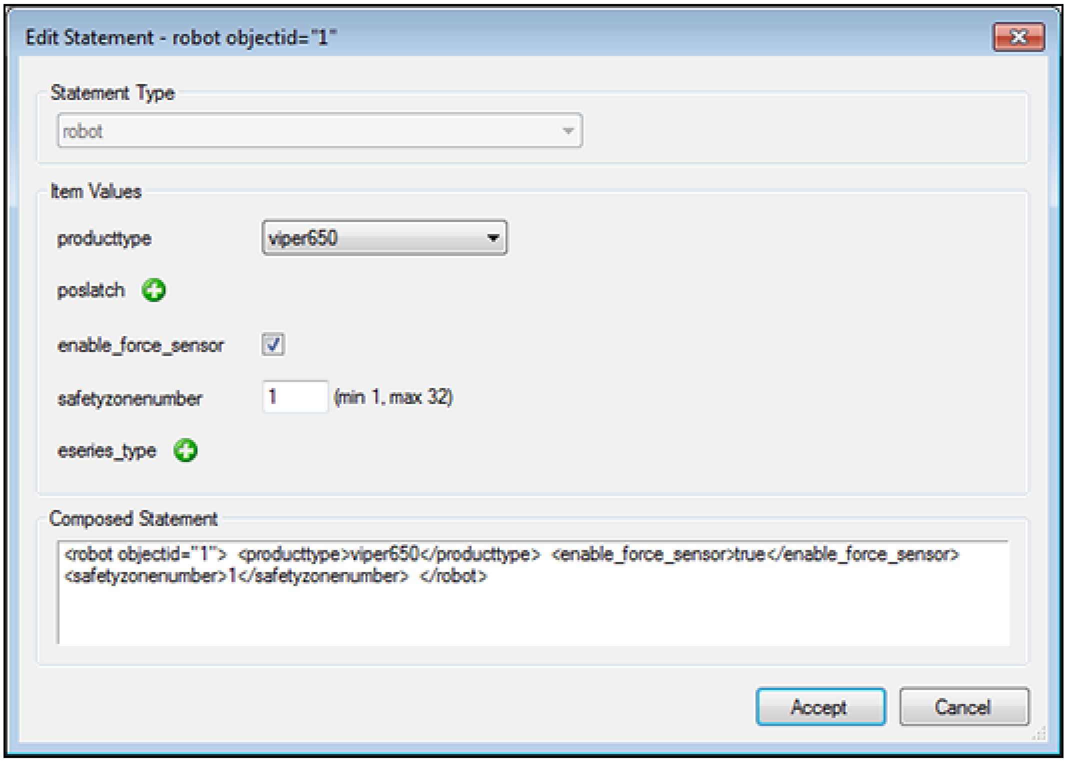
Enabling the Force Sensor

After you connect the force sensor, you need to enable it in the ACE software:

1. Start the ACE software.
2. Open the Controller object.
3. In the Configure tab, select Configure Controller, then click Finish.
4. Select ROBOT_SECTION, then the <robot object> for your robot.



5. Click Edit (or double-click the selection).
6. Check the enable_force_sensor box. It defaults to unchecked.



7. Save the configuration by clicking Accept and then Yes.

Configuring the 1394 Interface

To configure the force sensor, you will need the ACE software. In the ACE software, follow these steps:

1. Open the controller window.
2. Open the Configure Firewire Nodes window.

   You should see all of the nodes on your 1394 network, and the force sensor gadget attached to each node.
3. Configure each force sensor in the network so their IDs are different.

   This is a similar concept to how Robot Numbers are configured.
4. When using the force sensor, use SELECT FORCE = n, where n is the node you configured in the 1394 network.

Configuring Multiple Force Sensors

NOTE: The SmartController EX motion controller supports multiple force sensors. Each eAIB or eMB-60R supports a single force sensor.

If you use only one force sensor, your system should come configured properly. When you add multiple force sensors to an existing system, you must use the ACE software to specify which sensor is force sensor number 1, which is number 2, and so on. Once properly configured, the SELECT function in eV+ will allow communication with the specified sensor.

NOTE: Ensure that each force sensor is mapped to a unique ID.

Chapter 6: Troubleshooting

This chapter is designed to help quickly identify and solve problems associated with the Intelligent Force Sensing system. The following sections separate the two types of errors that may occur: those accompanied by eV+ error messages, and those that do not have an associated error message.

6.1 Errors with No Messages

Force sensor readings are not zero just after starting up eV+ system

Explanation

Force sensor readings should not be expected to be zero when the system is initialized. The sensor is calibrated to be accurate in measuring relative forces, not absolute forces.

Another possible cause is uneven sensor loads resulting from unbalanced moment of the robot adapter plate or tool adapter plate screws. Non-planarity of the sensor mounting surfaces can also cause large force reading offsets.

User Action

When the system is started, zero the sensor to cancel out any preloads. Also, make sure the flange to which the sensor is connected is smooth and flat, and that the attachment bolts are tightened evenly.

Force trips occur on the move just BEFORE the desired guarded move

Explanation

To provide faster processing, eV+ processes past the current move instruction to the next move instruction before it waits for the current move to complete. If you enable Guarded mode just before issuing a move instruction, it may therefore become enabled almost immediately after starting the previous move.

User Action

Generally, you should enable Guarded mode just after starting the guarded move, not just before. Alternatively, you can issue a BREAK or DELAY program instruction just before enabling in order to force the current motion to complete.

Noise level high in sensor readings

Explanation

A certain amount of sensor noise is normal, even with the robot power off. The robot generates additional electrical and mechanical noise when it is powered on. The small amount mechanical vibration that occurs even when the robot is not moving also causes some noise.

User Action

The simplest way to reduce noise in the force readings is to select a higher force filter number using the FORCE.MODE instruction.

If you suspect the sensor of having a hardware problem, you need to compare the noise against the force sensor specifications. With the robot motionless and powered off, you can run the following example code to compute standard deviation of a number of force samples in each direction.

```
.PROGRAM test.force

    AUTO DOUBLE av[5], f[5], i, j, num.samp, sq[5]
    num.samp = 1000 ;Number of samples
    FOR j = 0 TO 5
    av[j] = 0 ;Average force
    sq[j] = 0 ;Average squared force
    END
    FOR i = 1 TO num.samp
    FORCE.READ (0) f[]
    FOR j = 0 TO 5
    av[j] = av[j]+f[j]/num.samp
    sq[j] = sq[j]+SQR(f[j])/num.samp
    END
    END
    TYPE “Dir Average Std Dev”
    FOR j = 0 TO 5
    TYPE j,/F10.2,av[j],SQRT(sq[j]-SQR(av[j]))
    END

.END
```

Overshoot is too large when the robot stops at the end of a guarded move

Explanation

Overshoot will always occur when a guarded move trips, since it is impossible to instantly stop a mechanical system. The magnitude of the overshoot is a function of robot speed, end-effector weight, and the robot motor strength.

User Action

First, try to minimize the end-effector weight. This will allow higher speed motions for a given amount of overshoot. If the weight is already minimized, begin lowering the motion speed until the overshoot is acceptable. You can use the Advanced Servo Library to optimize guarded move performance by selecting appropriate integrator gain, settling tolerance, and nulling tolerance. See the Advanced Servo Library Reference Guide for more information.

Protect mode trips occur too frequently

Explanation

Protect mode, intended to power-down the robot after inadvertent crashes, can also trip during normal operation if the trip level is too low or if the load is high enough to exceed the trip force level during normal operation. Slewing motions in the X-Y direction are particularly prone to tripping Protect mode during aggressive acceleration and deceleration with high loads.

User Action

The easiest solution is to increase the Protect mode limit or simply disable Protect mode. Also, try to minimize the end-effector weight to reduce the load on the force sensor, as the load may be too heavy to avoid saturation.

6.2 Standard Errors and Messages

The following table lists force system error codes. The errors are described in detail below.

Table 6-1. Force System Error and Message Codes

Code	Text of Message
-623	*Stop-on-force tripped*
-624	*Force protect limit exceeded*
-634	*Force sensor communication error*
-658	*Device hardware not present*

Force protect limit exceeded (-624)

Explanation

At least one force sensor strain gauge reading has exceeded the preset limit, causing a robot panic stop. This may happen due to high forces experienced during an insertion, a crash, or high acceleration.

User Action

If a crash occurred, ensure that the work area is cleared. If the limit was exceeded in normal operation, it should be increased or Protect mode should be disabled. Enable power with the pendant or from the terminal and continue operation.

Force sensor communication error (-634)

Explanation

The data returned from the force sensor has been corrupted either by noise or an electrical failure. If your system starts to experience sensor errors with increasing frequency, then it is very likely caused by worn sensor cables.

User Action

Inspect the sensor cables carefully for signs of wear. Check each connection. Replace the cables if necessary. Call your local Omron Support if the problem persists.

Stop-on-force triggered (-623)

Explanation

A force system Guarded mode trip occurred when the robot was not under program control.

User Action

Arm power must be re-enabled before robot motion may continue. If the trip was not desired, make sure that Guarded mode is disabled before the program relinquishes control of the robot to the pendant.

Device hardware not present (-658)

Explanation

The force sensor was not detected by the amplifier.

User Action

Ensure that the cables are correctly secured and the selected force sensor is correctly mapped and selected by the using task.

If the problem persists, contact your local Omron Support.

Chapter 7: Technical Specifications

NOTE: The force sensor is manufactured by JR3, code 67M25A3. It is not manufactured by Omron.

7.1 Dimensions

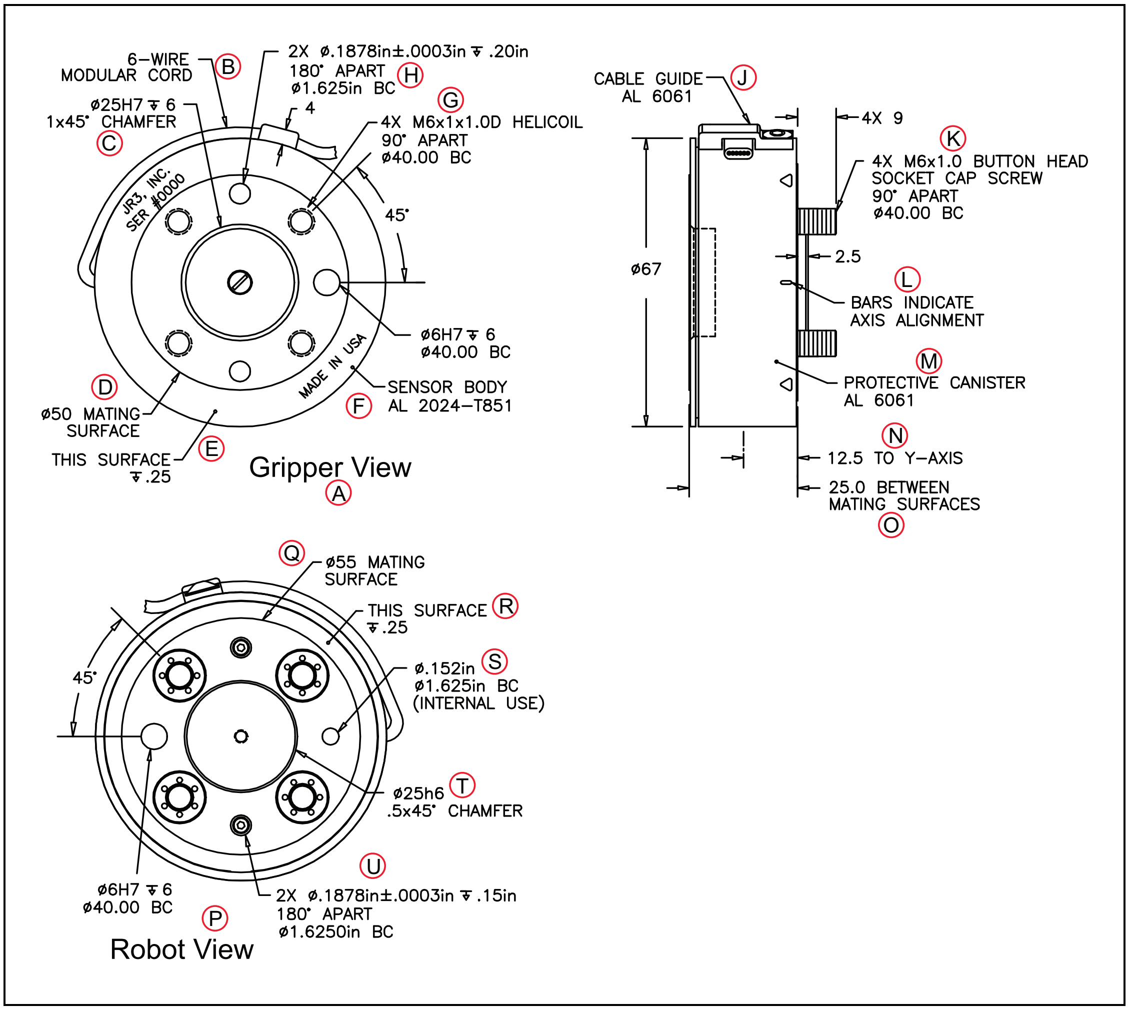


Figure 7-1. Sensor Dimensions, Units in mm unless noted.

NOTE: We have not converted the dimensions in this drawing to metric, although some of the dimensions in the original were already metric.

Key	Meaning	Key	Meaning
A	Gripper View	L	Bars indicate axis alignment
B	6-wire modular cord	M	Protective canister AL 6061
C	25 H7 depth 6, 1x45° chamfer	N	12.5 to Y Axis
D	Dia. 50 mating surface	O	25.0 between mating surfaces
E	This surface, depth 0.25	P	Robot View
F	Sensor body, AL 2024-T851	Q	Dia. 55 mating surface
G	4x M6x1x1 OD Helicoil	R	This surface depth 0.25
H	2x dia. 0.1878 ±0.0003 inch, depth 0.20 inch, 180° apart, dia. 1.625 inches BC	S	Dia. 0.152 inch, dia. 1.625 inches BC (Internal Use)
J	Cable guide AL 6061	T	Dia. 25 h6, 0.5 x 45° chamfer
K	4x M6x1.0 Button Head Socket Cap Screw, 90° apart, dia. 40.00 BC	U	2x dia. 0.1878±0.0003 inch, depth 0.15 inch, 180° apart, dia. 1.6250 inches BC

7.2 Performance

Property	Value
Update rate	8 KHz
Force signature buffer size	Configurable (up to full amount of memory of processor)
Force Filters	Choice of seven, with approximate filter time delays of 0 (raw forces), 1 ms, 6 ms, 25 ms, 100 ms, 400 ms, and 1.6 sec.
Amplifier compatibility	eAIB or eMB-60R
Robot compatibility	eCobra, Quattro, and Hornet robots controlled with an eAIB or Viper robots controlled with an eMB-60R amplifier
Operating system compatibility	eV+ 2.3 and later

7.3 Force Filter Specifications

The force sensors report forces at an 8 KHz rate. These forces are filtered by a first order filter in software. The user can change the default filters using the FORCE.MODE instruction. Filter 1 gives the raw force readings directly from the sensor. The filter cutoff frequencies are given in the following table.

Filter	Cutoff Frequency
1	4,000 Hz (Nyquist frequency)
2	1,000 Hz
3	250 Hz
4	62.5 Hz
5	15.625 Hz
6	3.906 Hz
7	0.977 Hz

7.4 Force Sensor Specifications

Table 7-1. Thickness and Weights of Complete Assemblies

eCobra/Quattro/Hornet Robots	
Thickness with adapters[1]	48 mm
Weight with adapters[2]	356 g
Viper Robots	
Thickness with adapters[1]	53 mm
Weight with adapters[2]	371 g

NOTE: [1]The extended length of the tool flange must be accounted for in the robot's configuration.

NOTE: [2]The payload capacity of any given robot will be reduced by the combined weight of the adapters and sensor.

Table 7-2. JR3 Multi-Axis Force-Torque Sensor Technical Specifications (SI Units)

Sensor Model Mechanical Load Rating	**67M25A3 200N**
Diameter (mm)	67
Thickness[1] (mm)	25
Material	AL 2024
Weight[2] (g)	175
Nominal Accuracy, all axes (% measuring range)	±1.0
Operating Temperature Range, non-condensing (°C)	-40 to +65
F_x, F_y	
Standard Measurement Range (N)	±200
Digital Resolution (N)	0.050
Stiffness (N/m)	13e6
Single-axis Overload (N)	930
Multi-axis Overload Coefficient, a (N)	930
Multi-axis Overload Coefficient, b (N)	1200
Multi-axis Overload Coefficient, c (N)	970
F_z	
Standard Measurement Range (N)	±400
Digital Resolution (N)	0.100
Stiffness (N/m)	130e6
Single-axis Overload (N)	3870
Multi-axis Overload Coefficient, d (N)	3870
M_x, M_y	
Standard Measurement Range (Nm)	±12
Digital Resolution (Nm)	0.0032
Stiffness (Nm/rad)	53000
Single-axis Overload (Nm)	58
Multi-axis Overload Coefficient, e (Nm)	79
Multi-axis Overload Coefficient, f (Nm)	215
Multi-axis Overload Coefficient, g (Nm)	58

Sensor Model Mechanical Load Rating	67M25A3 200N
M_z	
Standard Measurement Range (Nm)	±12
Digital Resolution (Nm)	0.0032
Stiffness (Nm/rad)	15000
Single-axis Overload (Nm)	48
Multi-axis Overload Coefficient, h (Nm)	48

Standard Measurement Range

This is the range of loads that each sensor model is ideally suited to measure. Factory adjustments to internal electronics allow custom measurement ranges to meet application-specific needs.

Bolt Patterns

The sensors are available standard with the ISO 9409-1 Ø40 mm bolt pattern. The two adapters provided with the kits allow the sensor to be attached to the robot tool flange, and then provide the standard Omron Robotics and Safety Technologies, Inc. tool flange bolt pattern for connecting the user's end-effector.

Multi-axis Overloads

- Insert your applied loads and the coefficients from the above table into the equations below to determine safe loading:

  $$\mathbf{F_x/a + F_y/a + F_z/d + M_x/e + M_y/e + M_z/h \leq 1}$$

  and

  $$\mathbf{F_x/b + F_y/c + F_z/d + M_x/f + M_y/g + M_z/h \leq 1}$$

  and

  $$\mathbf{F_x/c + F_y/b + F_z/d + M_x/g + M_y/f + M_z/h \leq 1}$$

  All three equations must be satisfied to avoid damage.

Omron Robotics and Safety Technologies, Inc.
4225 Hacienda Drive
Pleasanton, CA 94588 U.S.A.
Tel: (1) 925-245-3400/Fax: (1) 925-960-0590
Contact: www.ia.omron.com

Authorized Distributor:

Printed in USA
Cat. No. 11455-000 Rev. F
0120 (1214)