# Force Sensor Application #1

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>04/18/2018</td>
<td>Yannick Bertrand</td>
<td>Document Created</td>
</tr>
<tr>
<td>2</td>
<td>04/22/2018</td>
<td>Yannick Bertrand</td>
<td>Document update – added explanation</td>
</tr>
<tr>
<td>3</td>
<td>04/25/2018</td>
<td>James Gulart</td>
<td>Document update</td>
</tr>
<tr>
<td>4</td>
<td>06/15/2018</td>
<td>James Gulart</td>
<td>Updated Appendix</td>
</tr>
</tbody>
</table>
Contents

1. Introduction ........................................................................................................................................... 3
2. Applications ........................................................................................................................................... 3
3. Hardware specification ......................................................................................................................... 3
4. Reference documents .......................................................................................................................... 4
5. Getting Started .................................................................................................................................... 4
  5.1 Hardware assembly .......................................................................................................................... 4
  5.2 Wiring .............................................................................................................................................. 6
  5.3 Force Sensor Configuration ............................................................................................................. 7
6. Explanation ........................................................................................................................................... 9
7. Force Search ....................................................................................................................................... 12
8. Force Guarded Insertion ..................................................................................................................... 12
Appendix A – Sample Code .................................................................................................................. 14

Figure 1 – Side View for Viper, Sensor, Adapters, and Robot Flange ....................................................... 5
Figure 2 – Viper Cabling Overview ........................................................................................................ 6
Figure 3 – Enabling Force Sensor 1/3 ...................................................................................................... 7
Figure 4 – Enabling Force Sensor 2/3 ...................................................................................................... 7
Figure 5 – Enabling Force Sensor 3/3 ...................................................................................................... 8
Figure 6 – FireWire configuration 1/2 ..................................................................................................... 8
Figure 7 – FireWire configuration 2/2 ..................................................................................................... 9
Figure 8 – Force sensor reference frame (Left hand rule) ....................................................................... 10
Figure 9 – Force sensor operation mode ............................................................................................... 11
Figure 10 - Side detection ...................................................................................................................... 12
Figure 11 - New frame origin .................................................................................................................. 12
Figure 12 Force Guarded Part Insertion .................................................................................................. 13
1. Introduction
This document is to give a brief explanation and sample code of one the many possible use of the Force Sensor.

The force sensor solution can be used in a variety of applications like validating dimensions, applying desired force during operation and protecting against unexpected interference.

In this example, guarded mode will be used to find the position of a box that can hold three cylindrical parts. Once the target body has been located, a frame will be created and used as reference to drop the parts in their respected position.

2. Applications
In this application the force sensor will be used to locate part targets and protect the robot during part insertion with tight tolerances. The force sensor is suggested for use in the following applications:

- Deburring and Grinding
- Polishing
- Assembling
- Fitting and Inserting

Examples of such applications include:

- Engine Piston Insertion – Keep the part aligned with the tight tolerance hole via force guided motion
- Deburring of cast or molded parts such as Transmission Casing and Dashboards – Using the force sensor to keep the contour despite changes to the effective path
- Machine Tending – Protect tooling with force guarded motion
- Polishing of Class A surfaces – Applying constant force as measure by the force sensor
- Assembly of Geared Mechanisms – Using Force Guarded motion to match phases of the different gears

3. Hardware specification
This application was developed using the following hardware and firmware; Viper 650, Viper Force Sensor Assembly (P/N: 14161-200), ACE 3.8.3.0, eV+ 2.4C1

The Force Sensor comes in three kits. One for the Viper 650 & 850, one for the Cobra 600 & 800, and one for the Quattro & Hornet. Note the Force Sensor kits are not USDA nor Clean Room compatible. Also note the CN20 connector on a viper can be used for either Force Sensor or the Internal Solenoids, but not both.

<table>
<thead>
<tr>
<th>Complete Kits</th>
<th>Part Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assembly, Force Sensor, Cobra 600 &amp; 800</td>
<td>14161-100</td>
</tr>
<tr>
<td>Assembly, Force Sensor, Viper 658 &amp; 850</td>
<td>14161-200</td>
</tr>
<tr>
<td>Assembly, Force Sensor, Quattro &amp; Hornet</td>
<td>14161-300</td>
</tr>
</tbody>
</table>

For applications with existing hardware, please verify the correct cables. Use XBELTIO Triplex Cable P/N 13463-000 and not 14305-000. Part 14305-000 is not force sensor compatible. You can check this by verifying the middle
cable is labeled with "FORCE/EXPIO" and not "EXPIO". Also note the new addition of Triplex Cable Adapter, which is used to allow more power to the force sensor during initial power on. It is recommended to get the new spare parts kits since it contains the adapter plates, cables, and mounting hardware to use the force sensor on different robots. The spare parts kit can also be used to update pre-released force sensors kits.

<table>
<thead>
<tr>
<th>Spare Parts Kits without Force Sensor</th>
<th>Part Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>eCobra Spare Kit</td>
<td>14161-105</td>
</tr>
<tr>
<td>Viper Spare Kit</td>
<td>14161-205</td>
</tr>
<tr>
<td>Hornet/Quattro Spare Kit</td>
<td>14161-305</td>
</tr>
</tbody>
</table>

4. Reference documents

<table>
<thead>
<tr>
<th>Document Description</th>
<th>Part Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligent Force Sensing System – User guide</td>
<td>14155-000 Rev. D</td>
</tr>
<tr>
<td>eV+ Language – User’s Guide</td>
<td>1604-E-01</td>
</tr>
<tr>
<td>ACE (Automation Control Environment) – User’s Guide</td>
<td>1603-E-01</td>
</tr>
</tbody>
</table>

5. Getting Started

Before using the code, the force sensor must be wired and configured in the robot using the ACE software.

5.1 Hardware assembly

Refer to section 3 of the Intelligent Force Sensing System – User Guide for the necessary parts and installation according to the robot that will be used. For this test a Viper 850 was used and the force sensor and adaptor plate configuration are shown in Figure 1.
Figure 1 - Side View for Viper, Sensor, Adapters, and Robot Flange
5.2 Wiring
The viper force sensor cable schematic is shown in Figure 2.

Figure 2 – Viper Force Sensor Cable Schematic
5.3 Force Sensor Configuration


![Image of Force Sensor Configuration Process]

**Figure 3 – Enabling Force Sensor 1/3**

![Image of Force Sensor Configuration Settings]

**Figure 4 – Enabling Force Sensor 2/3**
Figure 5 – Enabling Force Sensor 3/3

The Force sensor must be assigned a number in the FireWire network.

Figure 6 – FireWire configuration 1/2
The Force Sensor number can be selected by right clicking on the Force Sensor and choosing a number. If multiple robots are connected to a SmartController EX, be sure each force sensor has a unique firewire configuration. Do NOT have two Force Sensor 1, configure them as Force Sensor 1 and Force Sensor 2.

![Configure FireWire Nodes](image)

*Figure 7 – FireWire configuration 2/2*

6. **Explanation**

Since the force sensor is offset from the robot flange, a configuration must be executed at the start of the routine.

It is important to enter a rotation configuration to align the force sensor reference frame to the tool reference frame. The Force sensor reference frame is according to the left-hand rule. There is also side makes on the force sensor. The one marked by the circle represents the X axis.
To align the Force sensor to the default tool frame, a rotation of -67.5° in Z must be configured.

```plaintext
SELECT FORCE = 1 ;Selecting Force sensor 1
FORCE.FRAME (1) TRANS(,,,-67.5) ; Aligning Force frame to the tool frame
```
The force sensor has many modes of operation.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>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 4-4, and the bit descriptions are documented in Figure 3-1. After a Guarded mode trip occurs, the LATCHED(1) function returns the number 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.</td>
</tr>
<tr>
<td>2</td>
<td>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.</td>
</tr>
<tr>
<td>3</td>
<td>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 45. Once enabled, Protect mode remains enabled until explicitly disabled with a FORCE.MODE(-3) instruction.</td>
</tr>
<tr>
<td>4</td>
<td>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.</td>
</tr>
</tbody>
</table>

Figure 9 – Force sensor operation mode
7. Force Search
Using the Force Sensor Guarded mode, the robot will search to find the edges of the two side surface of the support block.

![Figure 10 - Side detection](image)

Once the two side surfaces are found, the slope and angle of both are found and the point of intersection is isolated and will be used as the origin of the frame.

![Figure 11 - New frame origin](image)

The drop position will be dependent of the new frame that has been found and all 3 positions can be used.

8. Force Guarded Insertion
Once the robot finds the orientation of the target body, it picks up parts from fixed part locations and attempts to place them into the target. The robot does not know how deep the holes are and does not know if the holes are obstructed. The robot approaches the part, zeros the sensor, and set’s the force sensor into force guard. The robot then descends into the target and if at any point the sensor feels too much resistance, such as the bottom of the hole, it releases the part and moves on to the next part in the sequence.
Figure 12 Force Guarded Part Insertion
Appendix A - Sample Code

This is the main code used in the sample application

```assembly
.PROGRAM rob.main()
;
; ABSTRACT:      Main robot application program
;
; INPUTS:        None
;
; OUTPUTS:       None
;
GLOBAL REAL rob.run
AUTO LOC temp.loc
AUTO REAL decompl[6]
AUTO REAL traj, move[6], mem1

;CALL err.init()
SET temp.loc = HERE
DECOMPOSE decompl[] = temp.loc

SET temp.loc = TRANS(decompl[0], decompl[1], 450, decompl[3], decompl[4], decompl[5])
MOVES temp.loc
BREAK
APPRO search.y, 200

CASE TASK(1,1) OF
VALUE -1: ;Invalid task number
VALUE 0: ;Idle
        EXECUTE 1 force.main()
VALUE 1: ;Stopped due to program completion.
        PROCEED 1
VALUE 2: ;Stopped due to program execution error (for example, undefined value).
        PROCEED 1
VALUE 3: ;Stopped due to ABORT, panic button pressed, robot error, or watchpoint.
        PROCEED 1
VALUE 4: ;Executing.
        ; Nothing to do for the moment.
VALUE 5: ;Stopped due to PAUSE or breakpoint
        PROCEED 1
VALUE 7: ; Stopped due to single-step execution
        PAUSE
        PROCEED 1
ANY PAUSE
END
SIGNAL -41, 42
FORCE.MODE (-1)

MOVES search.y ;moving to the first search position
BREAK

CLEAR.LATCHES (0) ;clearing all latch position in buffer

; Force sensor configuration
; Set force scale (for N put 4.45) (for lbs put 1)
```

Force Sensor Application #1, Rev 4
FORCE.MODE (10) 1
;Set length (for mm put 25.4) (for inch put 1)
FORCE.MODE (11) 25.4

;Guarded mode setup
FORCE.MODE (1) AH130, 10 ;Search for moment 3D, 10lbs0m
FORCE.OFFSET (1) ;Zero the force sensor
WAIT.EVENT , 0.7

SPEED 30 MMPS ; fixing moving speed to 30mm/s
MOVES search.y:TRANS(0,150,0)
WAIT STATE(2) <> 1 ; Waiting for motion to be stopper

;Checking if force sensor was triggered
IF LATCHED(1) THEN
  FORCE.MODE (-1)
  SET pos.find[1] = LATCH(0)
  MOVES search.y
  BREAK
  SET #position = #PLATCH(1) ;retrieving joint position
ELSE
  PAUSE ; edge not detected
END

;-----------------------------------------

MOVES search.y2
BREAK

;Guarded mode setup
FORCE.MODE (1) AH130, 35 ;Search for moment 3D, xx lbs0m
FORCE.OFFSET (1) ;Zero the force sensor
WAIT.EVENT , 0.7
SPEED 30 MMPS ; fixing moving speed to 30mm/s
TYPE "Start Y2 Search"
MOVES search.y2:TRANS(0,150,0)
WAIT STATE(2) <> 1 ; Waiting for motion to be stopper
TYPE "Stopping Y2 Search"

;Checking if force sensor was triggered
IF LATCHED(1) THEN
  FORCE.MODE (-1)
  SET pos.find[2] = LATCH(0)
  MOVES search.y2
  BREAK
ELSE
  PAUSE ; edge not detected
END

;-----------------------------------------

MOVES search.trans
MOVES search.x1
BREAK

;Guarded mode setup
FORCE.MODE (1) AH130, 25 ;Search for moment 3D, xx lbs0m
FORCE.OFFSET (1) ;Zero the force sensor
WAIT.EVENT, 0.7

SPEED 30 MMPS; fixing moving speed to 30mm/s
TYPE "Start X1 Search"

MOVES search.x1:TRANS(0,150,0)

WAIT STATE(2) <> 1 ; Waiting for motion to be stopper
TYPE "Stopping X1 Search"

; Checking if force sensor was triggered
IF LATCHED(1) THEN
    FORCE.MODE (-1)
    SET pos.find[3] = LATCH(0)
    MOVES search.x1
    BREAK
ELSE
    PAUSE; edge not detected
END

;---------------------------------------------

MOVES search.x2
BREAK

; Guarded mode setup
FORCE.MODE (1) AH 30, 35; Search for moment 3D, xx lbsOm
FORCE.OFFSET (1); Zero the force sensor
WAIT.EVENT , 0.7

SPEED 30 MMPS; fixing moving speed to 30mm/s
TYPE "Start X2 Search"

MOVES search.x2:TRANS(0,150,0)

WAIT STATE(2) <> 1 ; Waiting for motion to be stopper
TYPE "Stopping X2 Search"

; Checking if force sensor was triggered
IF LATCHED(1) THEN
    FORCE.MODE (-1)
    SET pos.find[4] = LATCH(0)
    MOVES search.x2
    BREAK
ELSE
    PAUSE; edge not detected
END

;---------------------------------------------

WAIT.EVENT , 1
ABORT 1

APPROS search.trans, 200

DECOMPOSE decomp.y1[] = pos.find[1]
DECOMPOSE decomp.y2[] = pos.find[2]
DECOMPOSE decomp.x1[] = pos.find[3]
DECOMPOSE decomp.x2[] = pos.find[4]

m.x = (decomp.x2[1]-decomp.x1[1])/(decomp.x2[0]-decomp.x1[0])
angle.x = ATAN2((decomp.x2[1]-decomp.x1[1]),(decomp.x2[0]-decomp.x1[0]))+90-180
b.x = decomp.x1[1]-m.x*(decomp.x1[0]-60)

m.y = (decomp.y2[1]-decomp.y1[1])/(decomp.y2[0]-decomp.y1[0])
angle.y = ATAN2((decomp.y2[1]-decomp.y1[1]),(decomp.y2[0]-decomp.y1[0]))+90-180
b.y = decomp.y1[1]+60-(m.y*decomp.y1[0])
;Find frame origin
f.x = (b.y-b.x)/(m.x-m.y)
f.y = m.x*f.x+b.x

SET part.frame = TRANS((f.x),(f.y),0,0,180,-angle.x)

FOR iloop = 1 TO 2
  :pick part
  APPROX pick.loc[iloop], 80
  BREAK
  SIGNAL 41, -42 ;Open gripper
  MOVES pick.loc[iloop]
  BREAK
  SIGNAL -41, 42
  WAIT.EVENT , 0.5
  APPROX pick.loc[iloop], 100

;Going to drop position
APPROX part.frame:drop.loc[drop.ar[iloop]], 150
BREAK

FORCE.MODE (1) AH30, 10 ; Search for force 3D, xx lbsOm
FORCE.OFFSET (1); Zero the force sensor
WAIT.EVENT , 0.7
MOVES part.frame:drop.loc[drop.ar[iloop]]

WAIT STATE(2) <> 1

SIGNAL 41, -42 ;Open gripper
WAIT.EVENT , 0.5
APPROX part.frame:drop.loc[drop.ar[iloop]], 80
BREAK

END
; Used to teach robot drop position according to the test.frame.
; SET drop.loc[1] = INVERSE(part.frame):HERE

APPROX search.y2, 80
BREAK

PAUSE

.END
Controllers & I/O
- Machine Automation Controllers (MAC) • Motion Controllers
- Programmable Logic Controllers (PLC) • Temperature Controllers • Remote I/O

Robotics
- Industrial Robots • Mobile Robots

Operator Interfaces
- Human Machine Interface (HMI)

Motion & Drives
- Machine Automation Controllers (MAC) • Motion Controllers • Servo Systems
- Frequency Inverters

Vision, Measurement & Identification
- Vision Sensors & Systems • Measurement Sensors • Auto Identification Systems

Sensing
- Photoelectric Sensors • Fiber-Optic Sensors • Proximity Sensors
- Rotary Encoders • Ultrasonic Sensors

Safety
- Safety Light Curtains • Safety Laser Scanners • Programmable Safety Systems
- Safety Mats and Edges • Safety Door Switches • Emergency Stop Devices
- Safety Switches & Operator Controls • Safety Monitoring/Force-guided Relays

Control Components
- Power Supplies • Timers • Counters • Programmable Relays
- Digital Panel Meters • Monitoring Products

Switches & Relays
- Limit Switches • Pushbutton Switches • Electromechanical Relays
- Solid State Relays

Software
- Programming & Configuration • Runtime