## User's Guide

BullsEye®7


The information in this document can be subject to change without prior notice and should not be regarded as an undertaking from $A B B$. $A B B$ assumes no responsibility for errors that can appear in this document.

ABB is not responsible for damage incurred by the use of this document or software or hardware described in this document.

The document, or parts there of, may not be reproduced or copied without prior permission from ABB. It may neither be imparted to another nor otherwise used without authorisation. Infringement here of will be subject to action in accordance with applicable laws.

Further copies of this document can be obtained from ABB at current prices.
© ABB Automation Technology Products AB
Article number: 505 942-102

Date: 2004-02
ABB Automation Technology Products AB
Arc Welding \& Application Equipment
S-695 82 Laxå
Sweden

## TABLE OF CONTENTS

1 Introduction ..... 5
1.1 How to use this Manual ..... 5
1.2 What you must know before you use the Robot ..... 5
2 Technical Specification ..... 7
2.1 Unit Specifications. ..... 7
2.2 Physical Dimensions ..... 7
3 Product Description ..... 9
3.1 Tool Center Point (TCP) ..... 9
3.2 Theory of Operation. ..... 10
3.3 BullsEye ${ }^{\circledR}$ Advantages ..... 11
4 BullsEye ${ }^{\circledR}$ Installation ..... 13
4.1 Mechanical Installation ..... 13
4.2 Electrical Installation ..... 14
4.3 Software Installation ..... 16
5 Operation ..... 25
5.1 Basic Operation ..... 25
6 Advanced Setup Information ..... 29
6.1 Global routines in the BullsEye.sys module: ..... 29
7 User's Guide ..... 33
7.1 Program execution ..... 33
7.2 Fault management ..... 33
8 Maintenance ..... 39
9 Reservdelsförteckning/Spare parts list ..... 41

## 1 Introduction

### 1.1 How to use this Manual

This manual provides information on installation, preventive maintenance, trouble shooting and operation of the BullsEye ${ }^{\circledR}$ product.

### 1.2 What you must know before you use the Robot

The power supply must always be switched off whenever work is carried out in the controller cabinet.

NOTE! Even though the power is switched off at the robot controller, there may be live cables which are connected to external equipment and are consequently not affected by the mains switch on the controller.

Circuit boards - printed boards and components - should never be handled without Electro-Static-Discharge (ESD) protection. Use the wrist strap located on the inside of the controller door whenever handling any of these components.

BullsEye® 7
Introduction

## 2 Technical Specification

### 2.1 Unit Specifications

Electrical
Robot Connections
Software
Repeatability
$40 \mathrm{~mA}, 24 \mathrm{VDC}$
One sensor signal, 24 VDC, and 0 VDC
Furnished with equipment (requires 250 kb free user memory)
$\pm 0.006^{\prime \prime}(0.163 \mathrm{~mm})$

### 2.2 Physical Dimensions

For variant 501 527-880:


Figure 1 BullsEye ${ }^{\circledR}$

For variant: 550880-001:


Figure 2

## 3 Product Description

### 3.1 Tool Center Point (TCP)

BullsEye ${ }^{\circledR} 7$ provides completely automated Tool Center Point (TCP) definition for the S4C and S4Cplus models of IRB robots.
TCP is defined as an invisible reference point in direct alignment and relationship to all axes of the robot arm and located at the precise point where the welding wire tip would touch the workpiece using a pre-determined wire stickout distance from the bottom of the gas nozzle.


Figure 3 Welding Torch Revolving Around A Set TCP

### 3.2 Theory of Operation

When the robot is instructed to revolve around the TCP all robot axes will move accordingly to keep the TCP constant (See Figure 4 and Figure 5). If the torch is bumped and the program is run again, the robot repeats its same movements but it is possible that the TCP will no longer follow the same programmed path due to misalignment. You now have two choices:

1 Physically move the torch back into alignment (a task that could be difficult if not impossible)

2 Adjust for the misalignment automatically by redefining the TCP to the new torch position using the BullsEye ${ }^{\circledR}$. After the BullsEye ${ }^{\circledR}$ system updates the current TCP definition, the torch will rotate around the TCP as before because the robot arm has adjusted its path to compensate for the torch misalignment.


Figure 4 Robot Arm and Torch Movement With Correct TCP


Figure 5 Robot Arm Follows Same Path But Torch Path Has Changed

Once a point has been programmed, the robot remembers the tool center point location, not what the angles of the robot joints are. When the robot replays the programmed path, it calculates what the joint angles should be to get the TCP back to where it was when the path was programmed initially. As long as the robot controller is kept informed about where the tool center point is, it will always keep the paths properly adjusted.

### 3.3 BullsEye ${ }^{\circledR}$ Advantages <br> New for version 7.0 BullsEye ${ }^{\circledR}$ :

1 Operation remains similar to predecessor, version 6.3 BullsEye ${ }^{\circledR}$.
2 Teach pendant text now resides in a separate system module. This change allows provision for the text to be translated into other languages by other ABB divisions.

3 Version 7.0 is now compatible with Motion Supervision available in BaseWare 3.2 rev. 20 and higher.

BullsEye® 7
Product Description

## 4 BullsEye ${ }^{\circledR}$ Installation

### 4.1 Mechanical Installation

The BullsEye ${ }^{\circledR}$ should be either bolted to the floor (freestanding) or in a position where the robot can reach it and where it is not in the way of personnel working around the robot.


Figure 6 Base Coordinates

When the software is installed and executed (see Section 4.3), the robot will swivel around the weld wire plus and minus 30 or 45 degrees (See Figure 7). Forty-five (45) degrees will achieve the best accuracy. The position chosen for mounting the BullsEye ${ }^{\circledR}$ must not only allow the robot to reach it, but this position must also allow the robot to easily maneuver around its TCP within the BullsEye ${ }^{\circledR}$ work envelope without causing the robot to exceed its joint limits.


Figure 7 Alignment Angle Of $45^{\circ}$ Works Best

The best procedure for installation is to place the Freestanding BullsEye ${ }^{\circledR}$ in a desired position without securing it permanently. Perform the software installation and execute the BullsEye ${ }^{\circledR}$ setup procedures as described in section 4.3. If the robot is able to complete the full TCP update process and successfully completes a Quick Check routine while meeting all other location requirements, then bolt it solidly in position.
Side Mount BullsEye's must also be mounted in a manner to allow unrestricted robot access.

### 4.2 Electrical Installation

The BullsEye ${ }^{\circledR}$ is pre-wired at the factory for easy assembly. Connect the cable provided from the receptacle on the BullsEye ${ }^{\circledR}$ unit to the controller cabinet.
Use the following diagram for installation.
Figure 8 shows the termination points for the wires in the robot controller cabinet at the I/O Module and 24VDC terminal blocks.


Figure 8 Wiring Diagram for Male Receptacle on BullsEye, 550 880-001.

## Connecting BullsEye, 550 880-001.

1 The white cable (2014045) wire goes to 24 VDC
2 The black cable (2014045) wire (signal to sensor 1) goes to I/O
3 The green cable (2014045) wire goes to 0 VDC

## Connecting BullsEye, 501 527-880, with three cables.

1 The blue cable wire is connected to 24 V DC, terminal 4 on the terminal block in TC 96 and 1 in Binzel BRS-LC.

2 The brown wire is connected to 0 V , terminal 8 on the terminal block in TC 96 and 3 in Binzel BRS-LC.

3 The black wire to the sensor input, connection 16 on the terminal block in TC 96 and 10 in Binzel BRS-LC.

## Connecting BullsEye, 501 527-880, new sensor with four cables (2004-02).

1 The brown cable wire is connected to 24 V DC, terminal 4 on the terminal block in TC 96 and 1 in Binzel BRS-LC.

2 The blue wire is connected to 0 V , terminal 8 on the terminal block in TC 96 and 3 in Binzel BRS-LC.

3 The white wire to the sensor input, connection 16 on the terminal block in TC 96 and 10 in Binzel BRS-LC.

When the BullsEye ${ }^{\circledR}$ is correctly wired, the sensor light and input on the I/O in the robot control cabinet should be illuminated only when the light beam of the BullsEye ${ }^{\circledR}$ is broken.

### 4.3 Software Installation

With the Program Window of the Teach Pendant displayed, select FILE, then OPEN, and choose the "BullsEye" System Module from the floppy disk or other drive location where the module has been stored. This will load th program from the floppy into the robot controller where it will reside in memory. Acknowledge any reference errors displayed on the Teach Pendant and follow the same procedurees to load the "TCPData" Program Module from the floppy disk. (EasyArc users will load TCP_Data Program Module.) Finally load the BE_text System Module in the same way. Alteratively, BE_text may be installed as a Buildin module. Note that the system must have at least 250 kb free memory to run this software.

## Required modules:

BullsEye.sys, BE_text.sys, and one of the following: TCPdata.mod or TCP_Data.mod.

## Module Details:

| BullsEye.sys | Loaded | Task 0 | noview/encrypted |
| :--- | :--- | :--- | :--- |
| BE_text.sys | Loaded of Built-in | Task 0 | noview/encrypted |
| TCPdata.mod | Loaded | Task 0 | open (non-EasyArc) |
| TCP_Data.mod | Loaded | Task 0 | open (EasyArc) |

## Software Setup:

Prepare the robot, the torch and BullsEye ${ }^{\circledR}$ for use. With the Program Window visible on the Teach Pendant, select the VIEW Menu Key, select MODULE, highlight "BULLSEYE" and press ENTER. Now highlight the routine "SetupBullsEye" and press ENTER. Press the TEST Function Key. With the robot ready to operate and the enabling devise activated, you may begin the program by pressing the START Function Key.

The BullsEye ${ }^{\circledR}$ setup routines can also be called from the EasyArc program. Refer to the appropriate EasyArc instructions for starting and running this program.

Upon starting the setup program, the first screen displayed will tell you the module version number and that the program running is "...for setting up the BullsEye".
Press PROCEED.
The next screen is regarding Copyright and Patent information concerning the BullsEye ${ }^{\circledR}$ software. Press PROCEED.

Now the Teach Pendant will display the Present Selections Screen. To change any of the parameters shown, select the corresponding Function Key. To keep the parameters shown press OK.

| Program Waiting for Data! |  |  |
| :---: | :---: | :---: |
| Present setup number | =1 |  |
| 1. Degree of rotation |  |  |
| 2. Robot mounting pos. |  | VER |
| 3. Gun alignment |  | ETU |
| 4. Stick-out distance (mm) |  | M |
| Select which parameter to change to keep present values; Press OK |  |  |
| $\begin{array}{lll}1 & 2 & 3\end{array}$ | 4 | OK |

## Present setup number

This value represents which tool is selected, and therefore which TCP is going to be updated. Changing this value must be performed off-line on a text editor.

## 1 Degree of rotation

Making this selection elicits the following menu:

| Program Waiting for Data! |
| :--- |
| This parameter defines how much the |
| welding torch will rotat between |
| each measurement |
| (Try to use 45 degree if possible) |
| Select degree of rotation |

The robot will calculate the TCP by finding the welding wire at different rotation locations of the welding gun. A larger rotation will result in slightly better accuracy in the measurement (See Figure 7).

## 2 Robot mounting position

Making this selection elicits the following menu:

| Program Waiting for Data! |
| :--- |
| This parameter defines how the robot |
| is mounted, STANDING or INVERTED |
| Select robot mounting pos. |
|  |

This parameter defines how the robot and BullsEye $\circledR$ are mounted in relation to each other, either STANDING or INVERTED. If the base of both the robot and the BullsEye ${ }^{\circledR}$ are mounted on the same plane, select STANDING. If for example the robot is hanging upside down from a tower and the BullsEye ${ }^{\circledR}$ is mounted conventionally on the floor, or visa versa, select INVERTED.

## 3 Gun alignment

This parameter is for selecting when to align the welding gun to World Z (torch angle realignment) - Always is recommended.

| Program Waiting for Data! |  |  |
| :--- | :--- | :--- |
| This parameter defines if and when |  |  |
| the welding torch should be aligned |  |  |
| with World Z (torch angle re-alignment) |  |  |
| (ALWAYS is recommended if possible) |  |  |
| Select robot mounting pos. |  |  |
|  |  |  |
|  | NEVER |  |

NEVER - If torch is difficult to align.
SETUP - Aligns during setup only.
ALWAYS - Recommended method.

## 4 Stick-out distance

Allows you to indicate stick-out distance from bottom of the gas cup to the TCP in millimeters. Use the keypad to input the desired length.

| Program Waiting for Data! |
| :--- |
| Stick-out is the distance between the |
| bottom of the gas cup and the TCP. |
| Min. 5 mm , Max. 30 mm |
| Select stick-out distance ( mm ) |
|  |



Figure 9 Defining The TCP

Once all the set up questions have been answered, the next screen you will see is:

| Program Waiting for Data! |
| :--- |
| Robot needs to be at a point where the |
| gun is positioned straight up and down |
| with the gas cup breaking the beam. |
| STOP, Jog robot manually, start again |
| LAST, Moves to last Setup position |
| Make a selection please |

To continue with the set up, the robot needs to be moved to a point were the welding gun is positioned straight up and down with the gas cup breaking the beam. The beam needs to be placed a location were the robot is able to rotate the welding gun around its gas cup axis $+/-45$ or 30 and make a 30 tilt motion without running out of reach (See Figure 7 and Figure 10). Also, the beam needs to be parallel to the plane of the robot base.

## STOP

STOP allows the operator to manually jog the robot to a point where the torch is positioned straight up and down with the gas cup breaking the beam of the BullsEye ${ }^{\circledR}$. Once the robot is in a correct position. the operator then presses START on the Teach Pendant to resume the program.

## LAST

LAST causes the robot to move to the last point defined as the Startup position. This position is stored in the TCPData module and is inaccessible to edit except through the Setup routine. It is not recommended that the operator chose LAST, if it is the first time for the BullsEye ${ }^{\circledR}$ to be set up on that particular system.

## Caution:

The robot is not able to verify that the last setup location can be reached before trying to go there


Figure $10 \quad 30^{\circ}$ tilting motion.

If LAST was selected, the operator will be asked if the location that the robot moved to is indeed in correct orientation for the BullsEye ${ }^{\circledR}$ to begin updating the TCP:

| Program Waiting for Data! |  |
| :--- | :--- |
| Robot needs to be at a point where the |  |
| gun is positioned straight up and down |  |
| with the gas cup breaking the beam. |  |
| YES, Program will stop, you can then |  |
| jog robot manually, start again |  |
| NO, Program will continue |  |
| Is adjustment needed? |  |

YES will stop the program and allows the operator to manually jog the robot to a correct position. Once the robot is in a correct position, the operator then presses START on the Teach Pendant to resume the program.
Pressing NO allows the program to continue with the tool center point set up.
The above screen would also appear had STOP been chosen instead of selecting LAST at the previous screen. In either case if the torch is not breaking the beam when the program is allowed to resume with the set up, the following message and request for action appears:

| Program Waiting for Data! |
| :--- |
| The system has detected that the beam |
| is not broken. |
| Program will move to previous screen. |

The robot is now ready to begin calculating the TCP for the welding gun. The first step is to determine the beam orientation and generate an approximate TCP.

| Program Waiting for Data! |  |
| :--- | :--- |
| The Robot is now ready to search |  |
| for the beam orientation and |  |
| approximate TCP values. |  |
| Select SKIPTCP If you already have |  |
| good TCP values. |  |
| If your using a MIG weld gun, make |  |
| sure that 15 mm wire is fed out. |  |
|  | SKIPTCP | PROCEED | PR |
| :--- |

The approximate TCP measurement is excluded if the SKIPTCP button is pressed. If the system has a MIG weld gun, use the wire feeder manual button to increase the wire stick-out, cut to length as required and then press PROCEED. Note that on some plasma cutting torches a special extension needs to be added to the torch to simulate a wire. Now the BullsEye ${ }^{\circledR}$ begins automatic estimation of the TCP.

Once an estimated TCP has been defined, a screen similar to the figure below will be displayed. Values can be altered if necessary. Tool load can also be entered here.

| Program Waiting for Data! |  |  |  |
| :---: | :---: | :---: | :---: |
| Defined approximate TCP values |  |  |  |
| Values for estimated tcp are: |  |  |  |
| $X$ value (mm) | =0 |  |  |
| $Y$ value (mm) | $=0$ |  |  |
| $Z$ value (mm) | $=0$ |  |  |
| Select which value to change for present. Press OK |  |  |  |
| X | Y | Z | OK |

Once the data has been acknowledged, the following screen appears:

| Program Waiting for Data! |
| :--- |
| The Robot will now continue with |
| the setup by measuring the TCP |
| accurately. |

The automatic TCP and gun alignment routine will begin when PROCEED is selected. As the BullsEye ${ }^{\circledR}$ is functioning, the operator is kept informed of the update process by messages displayed on the Teach Pendant. This will take several minutes. The robot will come to a stop with the torch gun just above the BullsEye ${ }^{\circledR}$.

A new screen will be displayed prompting the operator how to define the manual calibration position at the pointer of the BullsEye ${ }^{\circledR}$.

| Program Waiting for Data! |
| :--- |
| Define manual calibration pointer. |
| The robot needs to be manually |
| moved to the pointer. |
| 1.Stop the program by pressing PROCEED |
| 2. Move the robot to the pointer |
| 3. Restart the program |
|  |

It is important to jog the robot so the TCP is directly above and on the pointer at the base of the "horseshoe" of the BullsEye ${ }^{\circledR}$. This position will be stored and can be used in subsequent checks of the TCP to manually move the torch neck if needed in the event of a severe crash. The BullsEye ${ }^{\mathbb{R}}$ program will prompt the operator if this is necessary.

After re-starting the program with the START Function Key, the operator will be asked if the robot is on the pointer. Answering YES allows the program to continue. Selecting NO will display the screen above giving instructions on moving the robot to the pointer.

The last screen of the setup procedure will appear once acknowledgment has been received that the robot was indeed at the pointer.

| Program Waiting for Data! |  |
| :--- | :--- | :--- |
| The BullsEye setup is now complete |  |
| Your TCP has been set to the following: |  |
| X=123 $\quad Y=0 \quad Z=350$ <br> Press PROCEED to exit setup. <br>  |  |

The above screen shows the values of the set up tool center point of this tool set up. It is known as the Day- 1 tcp. An entry will also be filed in the user error log of the robot controller:

> BullsEye setup done
> cp data for SetupNo: 1
> $\mathrm{X}=123 \quad \mathrm{Y}=0 \quad \mathrm{Z}=350$

Values saved as Day1 tcp

## Possible Error Messages Incurred During Setup:

Invalid setup number is used
Change setup number in instruction
Valid setup numbers are normally 1-3. Change instruction and re-execute.
The beam could not be found
Check beam and/or change startpoint
This comes up if the beam could not be found. Check beam function and/or change the start point.

The TCP estimate function failed
Check if enough wire is fed out
Comes up if a search failed during TCP estimation.
Torch alignment could not be done
Check beam and/or wire stick-out
Comes up if a search failed during gun alignment.
Tool measurement function failed
Check beam and/or wire stick-out
Comes up if a search failed during measure TCP operation.
WARNING !
Face plate pointing straight down, tool $X$ axis will parallel with beam $X$. Check with the manual on how to correct this problem.
This screen will appear if the system detects that the welding gun z axis is parallel with the robot mounting flange z axis. In this situation the TCP x axis direction can not calculated. There are two possible ways to cope with this problem.

1 Change the angle of the welding gun.
2 Set BullsEye_Param variable "ForceTcpQuat" to either -1 or 1. This will force the x axis of the TCP to be 90 degrees from the beam direction. Set to 1 , x will go in the direction opposite the tilt motion.

## 5 Operation

### 5.1 Basic Operation

When the CheckTCP routine is called by the robot, the BullsEye ${ }^{\circledR}$ will measure the TCP. A system with EasyArc installed will have menus which allow the operator to easily execute the procedure call and handle movements from the system's safe position and the approach point to the BullsEye ${ }^{\circledR}$ unit. In a system without EasyArc CheckTCP can be executed directly. In either event, one of the following will happen:

1 If the TCP is very close to the Day-1 TCP values, no update is made, and the robot returns to production.

2 If the TCP is within a predetermined allowed error ( $\mathrm{X}, \mathrm{Y}, \& \mathrm{Z}$ combined) of its Day-1 set up TCP, BullsEye ${ }^{\circledR}$ will automatically update it and allow the robot to return to production. The allowed error will be set at 5 mm , unless the optional set up parameter MaxFrDay1 is set to some other value. If the program running in manual teach mode the following options become available in this situation:

| Program Waiting for Data! |  |  |
| :---: | :---: | :---: |
| Tcp does not match "Present" definition: |  |  |
| $\mathrm{Present}_{\mathrm{x}=2}$ TCP - Day1_TCP SetupNo: $\mathrm{Y}=2$ |  |  |
|  |  |  |
| Day1Tcp: tcp when setup was made |  |  |
| Present: tcp with present values |  |  |
| Move to pointer before updating TCP? |  |  |
| Select which tcp to use at pointer or NO to continue with tcp update. |  |  |
|  |  |  |
| Day1Tcp | Present | NO |

If the tool is known to be bent, it is recommended that the torch be bent back to the original position. By selecting Day1Tcp the robot will move to the reference pointer defined during the set up. Selecting NO allows the robot to continue with updating the TCP, much as it would do in automatic mode. Selecting either Day1Tcp or Present TCP allows the robot to move to the pointer with that respective tool active. The following screen will appear:

| Program Waiting for Data! |
| :--- |
| The gun should now be on the pointer. |
| If not, it may need to be manually |
| bent back to the pointer. |
| Select which tcp to use at pointer or PROCEED to continue. |
| Day1Tcp/Present NO |

It is possible to change between day 1 TCP and present TCP. Present TCP is the TCP that was calculated the last time the GetNewTcpData was executed. In teach mode, the following result screen will appear:

| Program Waiting for Data! |  |  |
| :--- | :--- | :--- |
|  |  |  |
| Difference from Day 1 | $\mathrm{x}(\mathrm{mm})$ | 1 |
|  | $\mathrm{Y}(\mathrm{mm})$ | 1 |
| TCP was updated by | $\mathrm{Z}(\mathrm{mm})$ | 2 |
|  | $\mathrm{x}(\mathrm{mm})$ | 0.5 |
|  | $\mathrm{Y}(\mathrm{mm})$ | 0.1 |
|  | $\mathrm{Z}(\mathrm{mm})$ | 0.2 |
|  |  |  |
|  |  | PROCEED |

In all modes the following message is recorded in the robot's user error log:
Tcp updated 0.1 mm
Present_TCP - Day1_TCP SetupNo 1
$\mathrm{X}=0.1 \quad \mathrm{Y}=0.3 \mathrm{Z}=0.5$
Tcp was updated by
$\mathrm{X}=0.1 \quad \mathrm{Y}=0.3 \mathrm{Z}=0.5$
3 If the TCP is off by more than the allowed limit from its original position, updating will stop and the following screen will be displayed:

```
Program Waiting for Data!
The system has detected that the TCP has
changed more than that allowed from Day 1
Max. error from day 1: }\quad5\textrm{mm
TCP-Day 1: }\quad\textrm{x}=5\textrm{Y}=3\textrm{Z}=
(It is recommended that the torch be
manually adjusted at the pointer).
Do you want to manually adjust?
Yes No
```

This is a warning that a large change in TCP has occurred since setup. It is recommended that the torch be manually adjusted back to the day-1 setup position. If the TCP is allowed to drift away from its original values too far, it is very likely that parts of the robot program will be out of reach for the robot or that portions of the robot will collide with the fixture.

- If No is selected, the TCP will be re-measured and the values of the TCP will be updated to the new position.
- If Yes is chosen, the robot will move to the original position that was programmed during setup for the pointer located at the bottom or the BullsEye ${ }^{\circledR}$ opening, and the following screen will be displayed:

```
Program Waiting for Data!
The gun needs to be manually bent
back to the pointer.
GetNewTcpData will be re-executed
when PROCEED is pressed.
```

You can now bend the torch until it is aligned with the pointer. Pressing PROCEED will re-start the program. BullsEye ${ }^{\circledR}$ will then re-measure and the values of the TCP will be updated to the new position.
Once the update has been performed the opportunity arises to change the Day-1 TCP values to the Present values:

| Program Waiting for Data! |
| :--- |
| TCP has been updated. |
| Day1 TCP is now different from present |
| TCP by more than the allowed amount. |
| (If Day1 TCP is not updated, this screen |
| will come up every time). |
| Do you want to reset the Day1 tcp to |
| present values? |

If a search error occurs during gun alignment, the following screen will appear:

| Program Waiting for Data! |
| :--- |
| Torch alignment could not be done. |
| Check beam and/or wire stick-out. |
| Check if gun is bent too much. |
| POINTER: Robot will move to pointer |
|  |

It is recommended to move to the pointer and check to see how far the tool is bent.

If a search error occurs during TCP measurement, the following screen will appear:

| Program Waiting for Data! |  |  |
| :--- | :--- | :--- |
| Tool measurement function failed. |  |  |
| Check beam and/or wire stick-out. |  |  |
| Check if gun is bent too much. |  |  |
| POINTER: Robot will move to pointer |  |  |
|  |  |  |
|  | POINTER | RETRY |

It is recommended to move to the pointer and check to see how far the tool is bent. When POINTER has been selected in the above two screens, the following message appears:

| Program Waiting for Data! |
| :--- |
| The gun should now be in the pointer. |
| If not, it may need to be manually |
| bent back to the pointer. |
| GetNewTcpData will be re-executed |
| when PROCEED is pressed. |
|  |

4 If the BullsEye ${ }^{\circledR}$ has moved on the floor or the robot mechanics have been altered (e.g. the robot has been damaged in a collision, or the calibration of the robot has been changed.), the following screen will display:

| Program Waiting for Data! |
| :--- |
| The system has detected that the beam |
| position is more than 2 mm from where it |
| was day 1. |
| Either the robot is out of sync. or |
| the beam has moved. |
| Please check calibration of the robot |
| (the TCP might not be correct). |

Either the robot is out of synchronization or the BullsEye ${ }^{\circledR}$ unit has moved. Press PROCEED to continue. Note that the TCP measurements will still be valid in most cases. If necessary, you should update the revolution counters of the robot and run the BullsEye ${ }^{\circledR}$ program again. When the robot is properly synchronized and the mechanics are set-up correctly, this error will not appear.

## 6 Advanced Setup Information

NOTE! The information in this section is typically used only by ABB technicians setting up a system with unique properties.

### 6.1 Global routines in the BullsEye.sys module: <br> IsTCPOK Function for TCP quick check

Function that performs a quick check of the tool. The function returns TRUE if the tool is within specified tolerances

```
Example
    PROC CheckTcp()
        IF NOT IsTCPOK(1) THEN
            GetNewTCPData tWeldGun,1;
        ENDIF
        ERROR
            Standard BullsEye error handler
    ENDPROC
```


## Returned Value

Data type: bool
TRUE if the tool is within specified parameters otherwise FALSE.

## Arguments IsTCPOK (SetupNo [\Wobj])

## SetupNo <br> Data type: num

Specifies setup no 1, 2, or 3 ( 1 is default). Up to 3 different TCP setups can be done on the same system. The same setup no need to be used both during setup and check.

## [\Wobj]

Data type: wobjdata
The parameter is normally used if the robot is moved by a robot carrier and the bullseye is mounted to the carrier.
If a work object with a reference to the track can be used, the Wobj parameter only need to included during setup.
If a gantry system is used with no gantry work object, a work object needs to be generated with the instruction GetGantryFrame.

A work object is required for both setup and check.

## GetNewTCPData:

The instruction returns TCP values calculated by the BullsEye ${ }^{\circledR}$ routine. The new TCP values are returned in the INOUT variable YourTool.

## Arguments

## GetNewTCPData YourTool SetupNo [\DoSetup] [\MaxQChkErr] [ $\mathrm{MaxFrDay1]} \mathrm{[ } \mathrm{\ GasCupAlign]} \mathrm{[ } \mathrm{\ GasCupMinDia]}$ [\GasCupOffs][\DistGasCup] [InitMove] [SSliceGap] [ ForceTcpQuat] [\WireScanLngth] [\CupScanLngth] [\Wobj]

Inout parameter for the tool to be updated.

## SetupNo

Data type: num
Specifies setup no 1,2 , or 3 ( 1 is default). Up to 3 different TCP setups can be done on the same system. The same setup no need to be used both during setup and check.

## [\DoSetup] <br> Data type: switch

If selected, setup of the BullsEye ${ }^{\circledR}$ will be performed. Parameters below except for the Wobj parameter will only be used during setup.

## [ $M$ MaxQChkErr]

Data type: num
Min 0.1 Max 2 Default 0.3 (unit mm)
Limit for IsTCPOK function. If the TCP if found to have changed more than this parameter the IsTCPOK function will return FALSE.

## [ $\mathrm{MaxFrDay1]}$ <br> Data type: num

Min 2 Max 10 Default 5 (unit mm)
Max allowed change of the tep compared with day 1 tcp.

## [\GasCupAlign] <br> Data type: num

Min 10 Max 100 Default 50 (unit mm)
Defines the distance between the lower and higher search location when doing the align function. If a short gas cup is used this parameter needs to be lowered.

## [GasCupMinDia]

Data type: num
Min 3 Max 15 Default 6 (unit mm)
Defines the minimum diameter for the gas cup. The system is using this parameter when the bottom of the gas cup is detected. If a small gas cup or a thick welding wire is used this parameter needs to be changed.

## [\GasCupOffs]

## Data type: num

Min 3 Max 10 Default 6 (unit mm)
Defines the forward movement between the center of the wire and the vertical search location for the gas cup. If a thick wire is used this parameter might need to be changed.

## [\DistGasCup]

## Data type: num

Min 3 Max 15 Default 6 (unit mm)
Defines the vertical distance between the gas cup and the wire search location.

## [ InitMove]

Data type: num
Min 100 Max 500 Default 300 (unit mm)
Defines the vertical distance between the beam and the approach location.

## [ SliceGap]

Data type: num
Min 1 Max 15 Default 8 (unit mm)
Defines the vertical movement between each gas cup scan.
[ Force TcpQuat]
Data type: num
Min-1 Max 1 Default 0
This parameter can be used to force the orientation of the tool x axis to 90 from the beam axis. Used if the tool is mounted straight out from the robot face plate or a specific orientation of the tool is required.

## [WireScanLngth]

## Data type: num

Min 8 Max 15 Default 8 (unit mm)
Defines the search length for the wire. Might need to be increased if a thick wire is used.

## [\CupScanLngth]

## Data type: num

Min 30 Max 100 Default 50 (unit mm)
Defines the length of each gas cup scan. Initial scans are $60 \%$ longer.

## [WWobj] <br> Data type: wobjdata

The parameter is normally used if the robot is moved by a robot carrier and the BullsEye ${ }^{\circledR}$ is mounted to the carrier.

If a work object with a reference to the track can be used, the Wobj parameter only need to included during setup.

If a gantry system is used with no gantry work object, a work object needs to be generated with the instruction GetGantryFrame. A work object is required for both setup and check.

BullsEye® 7
Advanced Setup Information

## 7 User’s Guide

### 7.1 Program execution

This procedure is designed to give access to a number of parameters that affect how the BullsEye program runs. The parameters can be changed to adapt the program to tools that do not resemble a standard MIG gun (a plasma cutter, for example). When the procedure call is made GetNewTCPData without any of it's optional arguments attached, the program runs with default values designed to work with most standard MIG guns. Optional parameters, with the exception of the Wobj parameter, are only checked when the DoSetup flag is selected.

## Backward execution

Not supported

### 7.2 Fault management

The following code needs to be pasted in to the error handler of the procedure with the GetNewTCPData instruction.

## BullsEye error handler

```
TEST BullsEye_Error(ERRNO)
CASE1:
        !
        Stop;
        ! Move the gun to the beam
        ! Restart the program
        !
        BlsEye_Param{1,12}:=1;
        RETRY;
CASE2:
        !
        Stop;
        ! Move the gun/tcp to pointer
        ! Restart the program
        !
        BLSEye_Param{1,12}:=2;
        RETRY;
CASE3:
    BlsEye_Param{1, 12}:=3;
    RETRY
CASE4:
    RETRY
CASE5:
    TRYNEXT;
ENDTEST
```


## Example

## Normal setup

Stationary robot with the bullseye mounted to the floor.
The SetupBullsEye procedure is used when the BullsEye ${ }^{\circledR}$ needs to be setup. After execution, the tWeldGun variables will have new values. The tWeldGun variable can be replaced with your tcp name if different.

```
PROC SetupBullsEye()
    GetNewTCPDatatWeldGun,1\DoSetup
ERROR
        Standard BullsEye error handler
ENDPROC
```

The CheckTcp procedure is used when the tcp needs to be checked. If the tcp is OK the IsTCPOK will return TRUE. The tWeldGun variable can be replaced with your tcp name if different.

```
PROC CheckTcp()
IF NOT IsTCPOK(1) THEN
    GetNewTCPDatatWeldGun,1;
ENDIF
ERROR
    Standard BullsEye error handler
ENDPROC
```


## Robot on a track system

The bullseye is mounted to the moving part of the track. With this setup, the tcp can be checked at any location along the track.

```
PERS wobjdata obTrack:=
[FALSE,FALSE,"TRACK",[[0,0,0],[1,0,0,0]],[[0,0,0],[1,0,0,0]]];
PROC SetupBullsEye()
    GetNewTCPDatatWeldGun,1\DoSetup\WObj:=obTrack
ERROR
    Standard BullsEye error handler
ENDPROC
```

The CheckTcp procedure is used when the tcp needs to be checked. If the tcp is OK the IsTCPOK will return TRUE. The tWeldGun variable can be replaced with your tcp name if different.

```
PROC CheckTcp()
IF NOT IsTCPOK(1) THEN
    GetNewTCPDatatWeldGun,1;
ENDIF
ERROR
    Standard BullsEye error handler
ENDPROC
```


## Robot on a gantry system

The BullsEye ${ }^{\circledR}$ is mounted to a moving part of the gantry system. With a gantry system there is not a coordinate system that is referenced to the base of the robot. In this case a work object has to be created every time the BullsEye ${ }^{\circledR}$ program is executed.
The object frame is used by the BullsEye ${ }^{\circledR}$ routine and has to be set to 0 .

```
PERS wobjdata obGantry:=
[FALSE,TRUE,"",[[1000, 1000, 1000], [1,0,0,0]],[[0,0,0],[1,0,0,0]]];
CONST jointtarget jByFrame:=*;
PROC SetupBullsEye()
    ! Program a point where the tool0 z axis is aligned with world z
    GetGantryFrame jByFrame, obGantry.uframe,v100;
    GetNewTCPDatatWeldGun,1\DoSetup\WObj:=obGantry
ERROR
    Standard BullsEye error handler
ENDPROC
```

The CheckTcp procedure is used when the tcp needs to be checked. If the tcp is OK the IsTCPOK will return TRUE. The tWeldGun variable can be replaced with your tcp name if different.

```
PROC CheckTcp()
IF NOT IsTCPOK(1\WObj:=obGantry)THEN
    ! Program a point where the tool0 z axis is aligned with world z
    GetGantryFrame jByFrame, obGantry.uframe,v100;
    GetNewTCPDatatWeldGun,1\WObj:=obGantry;
ENDIF
ERROR
    Standard BullsEye error handler
ENDPROC
```


## GetGantryFrame

The GetGantryFrame instruction is basically an encapsulated MoveAbsJ instruction "MoveAbsJ ToJointPos,Speed,fine,tool0" that returns the frame of tool0 when the location is reached.

The procedure can be used to calculate a frame based on a robot joint move. This is needed when the robot is mounted on a gantry system and the Bullseye moves with the gantry.

## Arguments

## GetGantryFrame ToJointPos Frame [\ExtAxes] Speed [\Ax7] [\Ax8] [\Ax9] [\Ax10] [\nAx11] [\Ax12]

## ToJointPos (To Joint Position) <br> Data type: jointtarget

The destination absolute joint position of the robot and external axes. It is defined as a named position or stored directly in the instruction (marked with an * in the instruction).

## Frame

Data type: pose
INOUT parameter that returns the location of tool0 as a frame when the programmed location is reached. The bulleye routine can then be executed in the calculated frame.
[ ExtAxes]
Data type: extjoint
INOUT parameter that returns the present location of all external axes.

## Speed

## Data type: speeddata

The speed data that applies to movements. Speed data defines the velocity of the tool centre point, the tool reorientation and external axes.

The Ax7 to Ax12 parameter is used to force one or multiple axes to a specific location. If not used, no external axes will move when executed.

## Tool orientation

The BullsEye program always defines the tool z-axis as straight out from the torch gas nozzle. The tool x -axis will be oriented in such as way that it is parallel with tool 0 z -axis. The positive direction will be towards the robot face plate.
In most cases the orientation will be the same as doing a manual 5-point calibration. With the exception of when a tool has a negative x component (robot is overhead and the tool is pointing down). In this case using the manual method, the x -axis will come out 180 degrees opposite of the bullseye method.

## Recommendation

Always define the tool using the bullseye or use the 6-point method when the robot is mounted overhead.

### 7.2.1 Description of the $\mathbf{5}$ and $\mathbf{6}$ point method

- 5-point TCP\&Z

Four approach points are used to define the TCP and one elongator point is used to define the z direction of the tool. The x and y directions will be as close as possible to the corresponding axes in the wrist coordinate system. Using the 5-point method, the TCP and the tool's $z$ direction are defined. The x and y directions are set automatically by the robot.

- 6-point TCP\&ZX

Four approach points are used to define the TCP, one elongator point is used to define the z direction and one elongator point is used to define the x direction of the tool.

## 8 Maintenance

The BullsEye ${ }^{\circledR}$ is shipped complete and requires very little maintenance aside from keeping the unit clean. Refer to Section 4.2 for wiring information. Replacements parts may be obtained from After Sales at +46584 81666. Be prepared to provide complete information on the part(s) you need.

BullsEye® 7
Maintenance

## 9 Reservdelsförteckning/Spare parts list

Reservdelar beställs genom ABB Flexible Automation AB. Vid beställning var vänlig uppge typ och tillverkningsnummer samt benämningar och beställningsnummer enligt reservdelsförteckningen.

Rätt till ändring av specifikationer utan avisering förbehålles.
Spare parts are to be ordered from ABB Flexible Automation AB. Kindly indicate type of unit, serial number, denominations and ordering number according to the spare parts list.

Rights to reserved to alter specifikations without notice.

## Reservdelsförteckning/Spare parts list

Reservdelslistan innehåller all information som behövs för beställning av speciella delar till modulen för mätning av TCP. Var vänlig kontrollera att de uppgifter som lämnas är korrekta och motsvarar den önskade reservdelen. En specialtång är absolut nödvändig för montering av den fiberoptiska ledaren.
The spare parts list contains all information required for ordering special parts of the TCP gauging unit. Please ensure that you give us the precise description of the part which you require. A pair of special-purpose pliers is absolutely essential for fitting the fibre-optic cable (Item 2).

| Positionsnummer Position number | Antal Qty. | Beställningsnummer Ordering number | Benämning | Denomination | Anmärkningar Remarks |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | 501 527-880 | TCP-mätning, kompl. | TCP gauging unit, compl. | 0.746.346.000 |
| 1 | 1 | 0.746.335.025+6 | Mätspets med hylsa | Gauging prod with sleeve |  |
| 2 | 1 | 0.746.346.011 | Ljusledare, inkl. specialverktyg | Fibre-optic, incl. special tool |  |
| 3 | 1 | 0.746.346-010 | Optoelektronisk sensor | Optoelectronic sensor |  |



| Positionsnummer Position number | Antal Qty. | Beställningsnummer Ordering number | Benämning | Denomination | Anmärkningar Remarks |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | $\begin{gathered} 550880-001 \\ 2004414 \\ 2001430 \\ 706-034-011 \\ 706-034-013 \end{gathered}$ | BullsEye, kompl. <br> BullsEye-ram <br> Kabeladapter, skarv <br> Uttag, hona <br> Kabel | BullsEye, compl. <br> BullsEye frame <br> Adapter, cable <br> Receptacle <br> Cable | 12 |

550880A01


BullsEye®7
Reservdelsförteckning/Spare parts list

