



Weld your way.

## Carola requirements

Requirements for the  
Carola program code

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>OEE – Overall equipment effectiveness</b>	<b>3</b>
2.1	Determining the actual and OK quantities	3
2.1.1	Master program	3
2.1.2	Component program	5
2.1.3	Simple component program	6
2.1.4	Complex component program	7
<b>3</b>	<b>Instructions</b>	<b>9</b>

## 1 Introduction

For some C-Gate functions, it is necessary for information to be passed to C-Gate from the Carola program. Please refer to the relevant sections for the necessary blocks and instructions for the appropriate functionality.

If you use the `ReadVar` data structure or `OPCUAVAR` command in another context, ensure that the areas used do not overlap. Please note in addition that the index used in the Carola program is not the same index as on the web interface. The values are passed to C-Gate by the `ReadVar` data structure. In the controller, the first index is 1 and the last is 100. In C-Gate, the first index is 0 and the last one 99. A value that is stored in the controller under index 50 can be accessed in C-Gate under index 49.

Please note that the Carola code examples explained below might already be stored in full or in part in the master program. Please compare the sections and adapt them if necessary. A duplication of functions and variables is detected as an error by the Carola interpreter.

Note: Changes to the `QIROX WriteVar` variables can influence the Carola program flow.

**Note:** Please use Adobe Acrobat Reader to open this file if you want to copy the Carola programme code.

## 2 OEE – Overall equipment effectiveness

### 2.1 Determining the actual and OK quantities

The following program sections must be stored in the master and component programs respectively to determine the actual and OK quantities. It is important that the following procedures have exclusive use of index range 94 to 100, and it can not be used for other purposes. The range might be extended in the future.

#### 2.1.1 Master program

The first lines are necessary so that variables can be imported from other programs and written. They are placed immediately after the `RESTART` instruction.

```
RESTART
! VARIABLE DECLARATION
VAR SEL, ERG
STRING VAR COMPSTAT, COMPQUAL
PUBLIC STRING VAR COMPTYPE, COMPVARI, COMPSERI, COMPSTEP, COMPSEAM
```

This block defines a procedure that passes to C-Gate the information that a component is starting. The variables used below must first be set in the component program.

```
PUBLIC PROC COMPSTART
COMPSTAT := ''
OPCUAVAR(0,96, COMPSTAT, 'COMPSTATUS')
WAITM(100)

STRHND(2, COMPTYPE, COMPTYPE, ERG)
IF ERG=0 THEN COMPTYPE:='NODATA'
```

```

STRHND (0,COMPTYPE,'NODATA',ERG)
IF ERG=1 THEN BEGIN
LOPCUA1:
WRITE ('DO NOT WRITE VARIABLE >COMPTYPE< YET.')
WAITM(500)
PAUSE
JUMP LOPCUA1
END

COMPSTAT := 'START'
COMPQUAL := 'IN_PROGRESS'
COMPSEAM := ''
OPCUAVAR(0,100,COMPTYPE,'COMPTYPE')
OPCUAVAR(0,99,COMPVARI,'COMPVARIANT')
OPCUAVAR(0,98,COMPSEAM,'COMPSEAM')
OPCUAVAR(0,97,COMPSTEP,'COMPSTEPNAME')
OPCUAVAR(0,96,COMPSTAT,'COMPSTATUS')
OPCUAVAR(0,95,COMPQUAL,'COMPQUALITY')
OPCUAVAR(0,94,COMPSEAM,'COMPSEAMNAME')
ENDP

```

This block defines a procedure that passes to C-Gate the information that a component has ended.

```

! END OF COMPONENT RECORDING
PUBLIC PROC COMPEND
COMPSEAM := ''
COMPSTAT := 'END'
OPCUAVAR(0,94,COMPSEAM,'COMPSEAMNAME')
OPCUAVAR(0,96,COMPSTAT,'COMPSTATUS')
WAITM(100)

COMPTYPE := ''
COMPVARI := ''
COMPSEAM := ''
COMPSTEP := ''
COMPSTAT := ''
COMPQUAL := ''
COMPSEAM := ''
OPCUAVAR(0,100,COMPTYPE,'COMPTYPE')
OPCUAVAR(0,99,COMPVARI,'COMPVARIANT')
OPCUAVAR(0,98,COMPSEAM,'COMPSEAM')
OPCUAVAR(0,97,COMPSTEP,'COMPSTEPNAME')
OPCUAVAR(0,96,COMPSTAT,'COMPSTATUS')
OPCUAVAR(0,95,COMPQUAL,'COMPQUALITY')
OPCUAVAR(0,94,COMPSEAM,'COMPSEAMNAME')
WAITM(100)

```

```
ENDP
```

This block defines a function that passes to C-Gate the information that a new seam is starting. The name of the seam must be written to the `COMPSEAM` variable before the procedure is called.

```
! SEAM START
PUBLIC PROC SEAMSTRT
OPCUAVAR(0,94, COMPSEAM, 'COMPSEAMNAME')
WAITM(100)
ENDP
```

This block defines a function that passes to C-Gate the information that the current seam has ended.

```
! SEAM END
PUBLIC PROC SEAMEND
COMPSEAM := ''
OPCUAVAR(0,94, COMPSEAM, 'COMPSEAMNAME')
WAITM(100)
ENDP
```

This block defines a function that asks on the teach pendant whether the component is OK or not. The check is based on a visual inspection by the worker, and blocks subsequent program flow until something is entered. Wording `IS THE COMPONENT OK=` can be changed to how you want it.

```
! QUALITY OF COMPONENT RECORDING
PUBLIC PROC COMPQMAN
WREAD ('IS THE COMPONENT OK=1 OR NOK=0 ?', SEL )
IF SEL=1 THEN COMPQUAL := 'OK' ELSE COMPQUAL := 'NOT_OK'
OPCUAVAR(0,95,COMPQUAL, 'COMPQUALITY')
WAITM(100)
ENDP
```

This block defines a function that automatically (without prompting) sets the component quality to OK. This function can be used if you do not want to record the quality of the OEE dynamically, but still want to have the overall OEE metric calculated correctly.

```
! QUALITY OF COMPONENT RECORDING
PUBLIC PROC COMPQAUT
COMPQUAL := 'OK'
OPCUAVAR(0,95,COMPQUAL, 'COMPQUALITY')
WAITM(100)
ENDP
```

### 2.1.2 Component program

Please import the variables and procedures from the master program.

- **COMPTYPE** : Type of component.
- **COMPVARI** : Variant of the component.
- **COMPSEAM** : Serial number of the component. If no value is set, C-Gate automatically creates a serial number based on a timestamp.
- **COMPSTEP** : Name of the production step. The value is important if the component is processed at several stations.
- **COMPSEAM** : Name of the seam. A seam can consist of several web segments.

### 2.1.3 Simple component program

Please import the variables and procedures from the master program.

#### RESTART

```
! IMPORT OF VARIABLES AND PROCEDURES
EXTERNAL STRING VAR COMPTYPE,COMPVARI,COMPSEAM FROM MASTER
EXTERNAL PROC COMPSTART,COMPEND,COMPQMAN,COMPQAUT,SEAMSTRT,SEAMEND FROM MASTER
```

#### MAIN

In the program flow, set at least the mandatory fields and, if necessary, also the optional fields. As soon as all fields are set, call the procedure **COMPSTART**. If you have purchased the C-Gate.QM module, you can also define seams and structure the component protocol more precisely. To do this, use the procedures **SEAMSTRT** and **SEAMEND**. As soon as the part is finished and you want to set part quality from the Carola program, call either **COMPQAUT** or **COMPQMAN** to transfer the quality data. If you have purchased the C-Gate.QM module, the part quality is automatically determined using QIROX SD monitoring. Calling the procedure **COMPQAUT** or **COMPQMAN** has no effect on the component log.

To complete the component, call the procedure **COMPEND**. The order of the calls is important.

Please note that between the call of **COMPEND** and the restart of a component with **COMPSTART** up to one second must pass, because due to a data synchronization in the controller otherwise the end signal can be lost. It is planned that this limitation will be removed in one of the upcoming software versions for the robot controller.

```
! SETTING THE COMPONENT INFORMATION
! MANDATORY FIELD
COMPTYPE := 'CONTROL CABINET'

! OPTIONAL
! COMPVARI := 'XL'

! OPTIONAL
! COMPSEAM := '1234567890'

! OPTIONAL
! COMPSTEP := 'FLOOR'

! START OF THE COMPONENT
```

```

CALL COMPSTART

! *****
! COMPONENT PROGRAM
! *****

COMPSEAM := 'FILLET WELD 1'
CALL SEAMSTRT

! *****
! PROGRAM FILLET WELD 1. FOR EXAMPLE, VARIOUS GC MOVEMENTS.
! *****

COMPSEAM := 'FILLET WELD 2'
CALL SEAMSTRT

! *****
! PROGRAM FILLET WELD 2. FOR EXAMPLE, VARIOUS GC MOVEMENTS.
! *****

CALL SEAMEND

! DETERMINE THE QUALITY OF THE COMPONENT IN THE CAROLA PROGRAM
! CALL BEFORE COMPEND
CALL COMPQMAN
! OR
! CALL COMPQAUT

! END OF THE COMPONENT
! CALL FOR COMPQMAN OR COMPQAUT
CALL COMPEND

! END OF THE COMPONENT PROGRAM
END

```

#### 2.1.4 Complex component program

The following Carola program creates a part with two manufacturing steps. The manufacturing steps can either be welded sequentially on one robot or in parallel on several. The prerequisite for this is that the steps are combined in a protocol:

- Type, variant and serial number must be set the same on all robots.
- The corresponding part template must have the correct minimum number of production steps set. In this case, two production steps are welded, so the number must be set to at least the value two.

```

RESTART
EXTERNAL STRING VAR COMPTYPE,COMPVARI,COMPSEMI,COMPSTEP,COMPSEAM FROM MASTER
EXTERNAL PROC COMPSTART,COMPEND,COMPQMAN,COMPQAUT,SEAMSTRT,SEAMEND FROM MASTER
MAIN

! *****
! PRODUCTION STEP 1
! *****

COMPTYPE := 'BOILER'
COMPVARI := '80KW'
COMPSEMI := '0042012342'
COMPSTEP := 'FLOOR'

CALL COMPSTART

COMPSEAM := 'FILLET WELD 1'
CALL SEAMSTRT
! PROGRAM FILLET WELD 1. FOR EXAMPLE, VARIOUS GC MOVEMENTS.

COMPSEAM := 'FILLET WELD 2'
CALL SEAMSTRT
! PROGRAM FILLET WELD 2. FOR EXAMPLE, VARIOUS GC MOVEMENTS.

CALL SEAMEND
CALL COMPQAUT
CALL COMPEND

! *****
! PRODUCTION STEP 2
! *****

COMPTYPE := 'BOILER'
COMPVARI := '80KW'
COMPSEMI := '0042012342'
COMPSTEP := 'COVER'

CALL COMPSTART

COMPSEAM := 'FILLET WELD 3'
CALL SEAMSTRT
! PROGRAM FILLET WELD 3. FOR EXAMPLE, VARIOUS GC MOVEMENTS.

COMPSEAM := 'FILLET WELD 4'

```



```
CALL SEAMSTRT
```

```
! PROGRAM FILLET WELD 4. FOR EXAMPLE, VARIOUS GC MOVEMENTS.
```

```
CALL SEAMEND
```

```
CALL COMPQAUT
```

```
CALL COMPEND
```

```
END
```

### 3 Instructions

If you are or have been a PDM or UMS customer and one of the two options is activated in the controller configuration, the following note must be imperatively be observed! If one of the two options is activated without the robot controller being connected to the corresponding PC application, this will lead to gaps in the data stream that may last for up to 5 seconds and falsify the component protocol significantly. Please either connect the robot controller to the PC application or have the corresponding option removed from the configuration.

If you are working on the same component with multiple robots and would like to merge all data in one component log in the C-Gate.QM module, please contact CLOOS Service for more information. Otherwise please leave the `COMPSERI` field empty or else undesired side effects can occur in the OEE calculation.