

## Sommario

<b>QCL Libraries</b> .....	<b>3</b>
----------------------------	----------



## QCL Libraries

### Rules of using a function

#### FUNCTIONS FOR OPERATIONS ON DIGITAL INPUTS

<a href="#">IR10EdgeInp</a>	Rising edge and descent detection of an input or a flag
<a href="#">IR10Edge</a>	Rising edge detection of a digital signal with verification time
<a href="#">IR10EdgTmInp</a>	Rising edge and descent detection of an input o a flag with time reset capture flags fronts
<a href="#">VC10ChronVar</a>	Measurement timing of activation of an input or variable

#### FUNCTIONS FOR OPERATIONS ON DIGITAL OUTPUTS

<a href="#">OA10BlinkOut</a>	Blink management of an output or a flag
<a href="#">OA10SetTmOut</a>	Activation management of an output for a settable time
<a href="#">OT11PidReg</a>	Digital output modulation to check a temperature control process through P.I.D. system
<a href="#">OT21PidReg</a>	Modulation of two digital outputs for a generic process control through PID + FF adjustment
<a href="#">OT30PidReg</a>	Generic PID + FF controller

#### FUNCTIONS FOR OPERATIONS ON VARIABLES

<a href="#">VC10CollVal</a>	Comparison of a value towards to other two
<a href="#">VC10Copy</a>	Copy a value from A to B or from B to A
<a href="#">VC10HistVar</a>	A variable hysteresis towards other two values
<a href="#">VC10ChronVar</a>	Measuring switching time of a variable or input
<a href="#">VC10Calendar</a>	Calculating the day of the week (monday-etc) starting from a date
<a href="#">VC12FndXPnt</a>	Calculating the x-coordinate of a point on a line
<a href="#">VC12FndYPnt</a>	Calculation of the ordinate of a point on a line
<a href="#">VT10OnChVar</a>	Report of the variation of a variable
<a href="#">VT10OnChTVar</a>	Report of the variation of a variable with reset flag time of exchange value
<a href="#">VC10DivRound</a>	Smoothing a variable to a number of decimal places can be set
<a href="#">VC10LPFilter</a>	First-order low-pass digital filter (RC filter) for dimension data WORD
<a href="#">VC21LPFilter</a>	First-order low-pass digital filter (RC filter) for maximum size data +/- 999999 with preload option
<a href="#">VC10MkTime</a>	Gives the number of elapsed seconds from 00:00 of 1 January 1970 until the introduced date.
<a href="#">VC20MkTime</a>	Gives the number of elapsed seconds from 00:00 of 1 January 1970 until the introduced date.
<a href="#">VC11Hdr</a>	Viewing the location of a device with Hdr system <b>OBSELETE</b>
<a href="#">VC12Hdr</a>	Viewing the location of a device with Hdr system
<a href="#">VC11HdrJoint</a>	Viewing the location of Joint a device with Hdr system
<a href="#">VC10Granularity</a>	Displaying a value with granularity and threshold filter
<a href="#">VR10WrdtoLng</a>	Conversion: Converts two Word variables in a Long variables
<a href="#">VR10LngtoWrd</a>	Conversion: Turns a Long variable in two Word variables
<a href="#">VR10UbyToBin</a>	Conversion: Unsigned Byte → Binary
<a href="#">VR10SByToBin</a>	Conversion: Signed Byte → Binary
<a href="#">VR10UwrToBin</a>	Conversion: Unsigned Word → Binary
<a href="#">VR10SwrToBin</a>	Conversion: Signed Word → Binary
<a href="#">VR10SLnToBin</a>	Conversion: Signed Long → Binary
<a href="#">VR10BinToUBy</a>	Conversion: Binary → Unsigned Byte
<a href="#">VR10BinToSBy</a>	Conversion: Binary → Signed Byte
<a href="#">VR10BinToUWr</a>	Conversion: Binary → Unsigned Word
<a href="#">VR10BinToSWr</a>	Conversion: Binary → Signed Word
<a href="#">VR10BinToSLn</a>	Conversion: Binary → Signed Long
<a href="#">VR10UbyToAsc</a>	Conversion: Unsigned Byte → Ascii
<a href="#">VR10SbyToAsc</a>	Conversion: Signed Byte → Ascii
<a href="#">VR10UwrToAsc</a>	Conversion: Unsigned word → Ascii
<a href="#">VR10SwrToAsc</a>	Conversion: Signed Word → Ascii
<a href="#">VR10SLnToAsc</a>	Conversion: Signed Long → Ascii
<a href="#">VR10AscToUBy</a>	Conversion: Ascii → Unsigned Byte
<a href="#">VR10AscToSBy</a>	Conversion: Ascii → Signed Byte
<a href="#">VR10AscToUWr</a>	Conversion: Ascii → Unsigned Word
<a href="#">VR10AscToSWr</a>	Conversion: Ascii → Signed Word
<a href="#">VR10AscToSLn</a>	Conversion: Ascii → Signed Long
<a href="#">VR10ToSingle</a>	Copy a long integer encoded IEEE754 value in a single variable.

#### ENCODER CONTROL FUNCTIONS

<a href="#">DT11BreakEnc</a>	Encoder breakage control ANALOG axes
------------------------------	--------------------------------------

DT21BreakEnc	Encoder breakage control ON/OFF axes
--------------	--------------------------------------

## GENERAL FUNCTIONS FOR POSITIONERS

DA11Ramp	Generic ramp generator
DA10AnOpos	Analog output management for ON/OFF axis with ramps

## CIRCULAR BUFFER MANAGEMENT FUNCTIONS

BC10InitBuf	Init of a circular buffer (with internal buffer management variables to the array)
BC10PushBuf	Inserting of a value (push) in a circular buffer (with internal buffer management variables to the array)
BC10PopBuf	Extracting of a value (pop) from a circular buffer (with internal buffer management variables to the array)
BC21InitBuf	Init of a buffer FIFO type (circular)
BC21PushBuf	Inserting of a value (push) in a buffer FIFO type
BC21PopBuf	Extracting of a value (pop) from a buffer FIFO type
BC21Inspect	Acquisition of a value into the buffer
BC21Elements	Acquisition of the number of elements in the buffer
BC22Inspect	Acquisition of a value into the buffer



## FUNCTIONS WITH GENERAL OPERATIONS ON ARRAYS

AC10AvergArr	Calculating the arithmetic mean of the elements of an array
AC10CtrlArr	Analysis of the elements in an array
AC10SortUpAr	Sorting in ascending on the elements of an array
AC10SortDwAr	Sorting in descending order of the elements of an array
AC10FdMaxArr	Extracting the maximum value in an array
AC10FdMinArr	Extracting the minimum value in an array
AC11ResetArr	Full reset of an array
BC10ArrFifo	Managing a FIFO buffer (first input-first output)

## FUNCTIONS FOR MODBUS PROTOCOL

DW13Modbus	Modbus SLAVE protocol: managing data exchange with MODBUS devices
DW11SerModMa	Modbus MASTER simulated protocol through SERCOM device
DW14SerModSI	Modbus SLAVE simulated protocol through SERCOM device
DU10MbRetry	Implementation of the SEND command of MODBUS devices with check and manage any programmed attempts

## FUNCTIONS FOR VECTOR IMAGE (QPAINT)

VI10InitBuffer	Buffer Initialize
VI10BeginDrawBuffer	Prepares the Buffer to add the drawing operations and returns the previous error code
VI10DrawBuffer	Draws the contents of the Buffer
VI10WaitBufferReadyUsingEND	Waits for the buffer is ready for new operations coming out of the special task with END
VI10WaitBufferReadyUsingWAIT VI11WaitBufferReadyUsingWAIT	Waits for the buffer is ready for new operations coming out of the special task with WAIT
VI10GetErrorCode	Returns the current error code
VI10ClrErrorCode	Clears the current error code
VI10GetUnusedBufferSize	Retrieves the Buffer size used
VI10AddNop	Adds to the NOP command at the Buffer (no operation)
VI10AddCls	Adds to the CLS command at the Buffer (clear the Vector Image area)
VI10SetLayer	Adds to the SET_LAYER command at the Buffer (sets the active layer)
VI10AddPen	Adds to the PEN command at the Buffer (sets the color drawing)
VI10AddSet	Adds to the SET command at the Buffer (sets the current coordinates)
VI10AddPoint	Adds to the POINT command at the Buffer (draws a point)
VI10AddLine	Adds to the LINE command at the Buffer (draws a line)
VI10AddRect	Adds to the RECT command at the Buffer (draw a rectangle)
VI10AddCircle	Adds to the CIRCLE command at the Buffer (draw a circle)
VI10AddArc1	Adds to the ARC1 command at the Buffer (draws an arc of type 1)
VI10AddArc2	Adds to the ARC2 command at the Buffer (draws an arc of type 2)
VI10AddArc3	Adds to the ARC3 command at the Buffer (draws an arc of type 3)
VI10AddArcBetweenAngles	Added at the buffer the command for the draw of an arc from one corner A to corner B
VI10SetBackground	Adds to the background command at the Buffer (sets the background color)
VI10AddMoveArea	Adds to the MOVEAREA command at the Buffer (move the contents of the Vector Image)
VI10UnsetLayer	Adds to the UNSET_LAYER command at the Buffer (disable the layer indicated)
VI10LoadImage	Adds to the LOAD_IMAGE command at the Buffer (load the image attached to an <i>Image</i> object)

<a href="#">VI10GetDimension</a>	Adds to the GET_DIMENSION command at the Buffer (reads the VectorImage object size)
<a href="#">VI10ExtractDimension</a>	Extracts the dimensions of the VectorImage from the Buffer after the execution of the command <a href="#">VI10GetDimension</a>

## FUNCTIONS FOR MANAGING THERMOCOUPLES

<a href="#">IR10CJRead</a>	Cold-junction reading
<a href="#">IR10HJRead</a>	Warm-junction reading
<a href="#">VC10TCoupleB</a>	Temperature calculation for thermocouple type B
<a href="#">VC10TCoupleJ</a>	Temperature calculation for thermocouple type J
<a href="#">VC10TCoupleK</a>	Temperature calculation for thermocouple type K
<a href="#">VC10TCoupleN</a>	Temperature calculation for thermocouple type N
<a href="#">VC10TCoupleT</a>	Temperature calculation for thermocouple type T
<a href="#">IR11PTCRead</a>	Reading the PT100 resistance from the 3 points board
<a href="#">IR20PTCRead</a>	Reading the PT100 resistance from the 3 points board with the differential reading mode
<a href="#">VC10PTC100</a>	Calculation of temperature for PT100

## FUNCTIONS FOR MANAGING PROGRAMMABLE OUTPUTS

<a href="#">OP10Init</a>	Programmable outputs, initializing
<a href="#">OP10isOutOn</a>	Programmable outputs, tests whether active output status
<a href="#">OP10isParOk</a>	Programmable outputs, check correct value
<a href="#">OP10Manage</a>	Programmable outputs, data processing
<a href="#">OP10ResOut</a>	Programmable outputs, reset output status
<a href="#">OP10ResRet</a>	Programmable outputs, Reset output restraint
<a href="#">OP10SetOut</a>	Programmable outputs, set output status

## FUNCTIONS FOR CALCULATING CHECKSUM

<a href="#">VC10Cr32Init</a>	Calculation CRC, initialization
<a href="#">VC10Cr32Beg</a>	Calculation CRC, Beginning of the calculation
<a href="#">VC10Cr32Calc</a>	Calculation CRC, Conclusion of the calculation
<a href="#">VC10Cr32Udt</a>	Calculation CRC, Update calculation

## FUNCTIONS FOR SYNCROMOVE

### GENERAL FUNCTIONS FOR WORKING WITH CAMMING

<a href="#">DW22WrCam</a>	Writing to sectors cam (40 sectors) (CAMMING2,CAMMING3,CAMMING4)
<a href="#">DW31WrCam</a>	Writing to sectors cam (128 sectors) (CAMMING3,CAMMING4)
<a href="#">DC11SpaceCam</a>	Calculations for research space master set with minimum slave space
<a href="#">DC10VelCam</a>	Calculations for construction of the sectors of acceleration, constant speed and deceleration of a electronic cam, setting the Master and Slave and their speed

### GEARING FUNCTIONS

<a href="#">DC10ElGear</a>	Calculations for managing the Master/Slave gearing
<a href="#">DC10ChGear</a>	Exchange sync ratio calculations "on the fly" in a Master/Slave gearing (without ramps)
<a href="#">DC10ChVelRat</a>	Calculation to set and/or change the Slave/Master speed ratio of a gearing dynamically with flights of softening when changing speed

### FUNCTIONS FOR FLY CUT

<a href="#">DC21FlyCut</a>	Calculations for linear fly cut with machine productivity optimization
<a href="#">DC30FlyCut</a>	Calculations for linear fly cut with fixed Slave space
<a href="#">DW22WrCam</a>	Writing to sectors cam (40 sectors) (CAMMING2,CAMMING3,CAMMING4)
<a href="#">DW31WrCam</a>	Writing to sectors cam (128 sectors) (CAMMING3,CAMMING4)
<a href="#">DW10ChLenght</a>	Writing to sectors cam to change the linear length fly cut (CAM01)
<a href="#">DW22ChLenght</a>	Writing to sectors cam to change the linear length fly cut (CAMMING, CAMMING2, CAMMING3)
<a href="#">DC10DoubFlyC</a>	Calculation for taking a sample piece during the fly cut execution
<a href="#">DC10DinHFlyC</a>	Calculation for length change on the fly of the workpiece during the linear fly cut execution (typically cut defect on material)

### FUNCTIONS FOR WIRE-GUIDES

<a href="#">DC10Winding</a>	Calculation for the cam building for the wire-guides management
-----------------------------	---

### FUNCTIONS FOR CIRCULAR FLY CUT

<a href="#">DC12RotCut</a>	Managing of a circular fly cut with single or multi-blade cutting cylinder (from 1 to 6 blades)
<a href="#">DC22RotCut</a>	Managing of a circular fly cut with single or multi-blade cutting cylinder (from 1 to 6 blades) stopping to Home of the Slave and manual cutting no-synchronized command

## SYSTEM FUNCTIONS

<a href="#">SY10InitializeCriticalSection</a>	Managing critical section initialization
<a href="#">SY10EnterCriticalSection</a>	Entering critical section management
<a href="#">SY10LeaveCriticalSection</a>	Exit critical section management

## STRING FUNCTIONS

<a href="#">ST10StrStr</a>	String search in substring
<a href="#">ST10StrCpy</a>	String copy
<a href="#">ST10StrLen</a>	String length
<a href="#">ST10StrNCpy</a>	Copy characters from string
<a href="#">ST10atoi</a>	Convert string to integer
<a href="#">ST10StrCat</a>	Concatenate strings

Documento generato automaticamente da **Qem Wiki** - <https://wiki.qem.it/>

Il contenuto wiki è costantemente aggiornato dal team di sviluppo, è quindi possibile che la versione online contenga informazioni più recenti di questo documento.