

### **\_\_QUERYPOINTER**

This operator is not prescribed by the IEC 61131-3 standard.

At runtime \_\_QUERYPOINTER is assigning an interface reference to a untyped pointer. The operator returns a result with type BOOL. TRUE implies, that the conversion has been successfully executed.

**Note:** For compatibility reasons the definition of the intended interface reference must be an extension of the base interface \_\_SYSTEM.IQueryInterface and the compiler version must be  $\geq 3.3.0.20$ .

#### **Syntax:**

**\_\_QUERYPOINTER** (<ITF\_Source>, < Pointer\_Dest>)

For the first operand the operator requires a interface reference or a function block instance of the intended type and for the second operand a untyped pointer. After execution of \_\_QUERYPOINTER the Pointer\_Dest holds the address of the reference to the intended interface. In this case the conversion is successful and the result of the operator returns TRUE. In all other cases the operator returns FALSE. Pointer\_Dest is untyped and can be cast to any type. The actual type has to be ensured by the programmer. For example the interface could provide a method returning a type code. Precondition for an explicit conversion is that the ITF\_Source is an extension of the interface \_\_System.IQueryInterface. This interface is provided implicitly and needs no library.

**Example:**

```
INTERFACE ItfBase EXTENDS __System.IQueryInterface
METHOD mbase : BOOL
END_METHOD

INTERFACE ItfDerived EXTENDS ItfBase
METHOD mderived1 : BOOL
END_METHOD

FUNCTION_BLOCK FBVariante IMPLEMENTS ITFDerived

PROGRAMM POU
VAR
itfderived : ItfDerived;
insV : FBVariante;
xResult, xTest : BOOL;
pVar: POINTER TO DWORD;
END_VAR
itfderived := insV;
xResult := __QUERYPOINTER(itfderived, pVar);
IF xResult THEN
xTest := pVar.mderived();
END_IF
```