OMRON

myKnowledge

RSS FEED

Main > Product Type > Automation Systems > PLCs

Select by category Go Search

Print E-mail English

# Passing An Array Of Data To A Function Block

## Objective

This article describes how to pass an array of data (in this case a number of Integers) to a Function Block for processing. It assumes the use of a CJ1/CS1 (V4 CPU) or CJ2 PLC.

## Arrays and defining them

Data passed using an array is a way to pass more than one piece of information, but with just one reference point.

Creating a variable as an array allows one name to be defined for the variable, but accessing the data with a 'subscript' value to 'point' to the data required. For example, the 10 pieces of data held in an array 'ArrayData_1' are defined with a physical address to store the data in the PLC memory (for example DM).

| Array Variable Access | PLC Memory | Value |
|---|---|---|
| ArrayData_1[0] | D00000 | 1234 |
| ArrayData_1[1] | D00001 | 0029 |
| ArrayData_1[2] | D00002 | 1939 |
| ... | ... | ... |
| ArrayData_1[8] | D00008 | 3652 |
| ArrayData_1[9] | D00009 | 3331 |

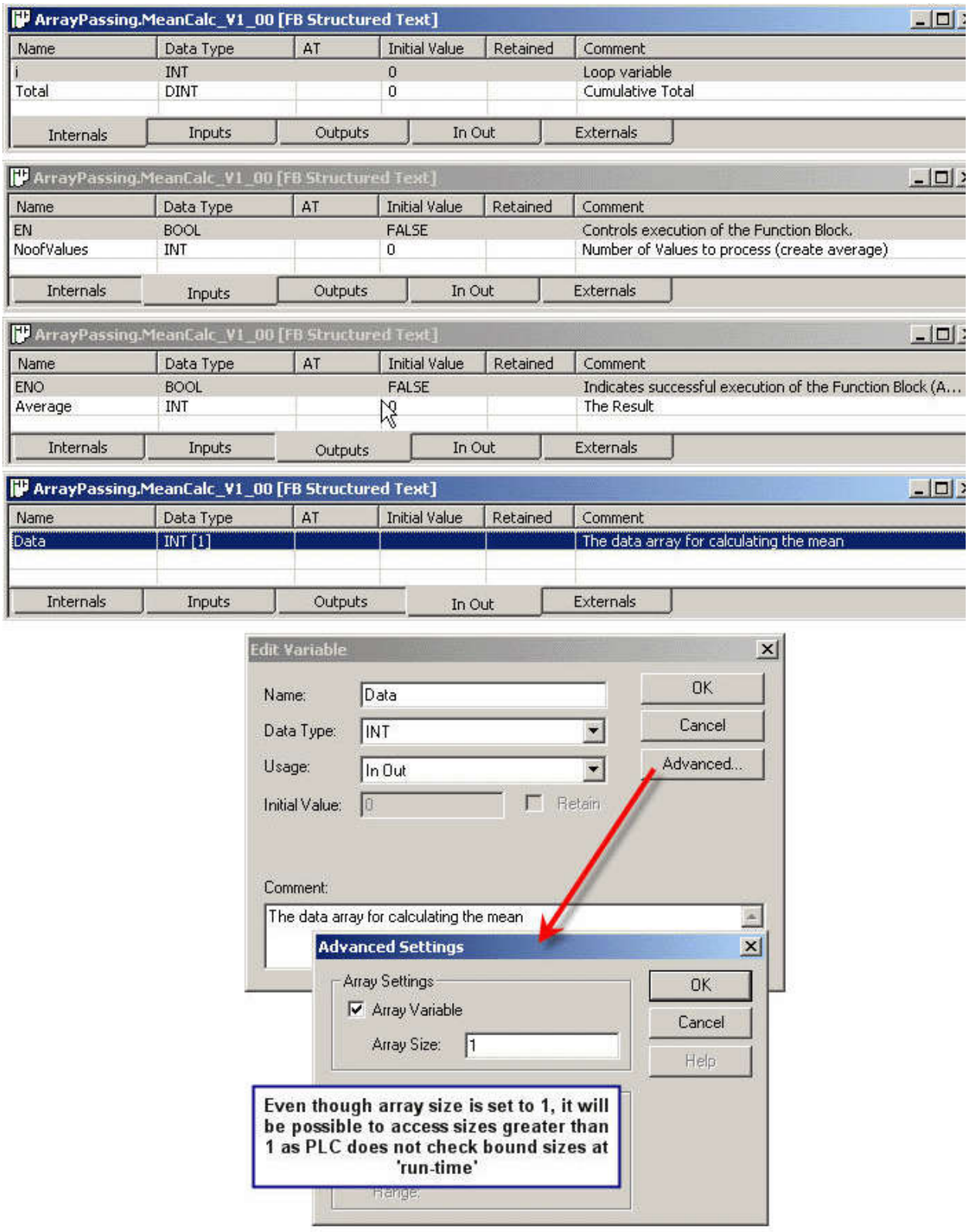
Therefore accessing variable ArrayData_1[0] (subscript 0) will result in the value contained in D0000 to be returned. Accessing ArrayData_1[8] (subscript 8) will result in the value 3652 (held in D00008).

Therefore it becomes easy to access lots of data associated with one variable and its array subscript. Using a variable to modify the subscript value makes for a very powerful tool to iteratively access data. For example ArrayData_1[ i ], when i=1 will lead to the value 0029.

## Worked Example: Find the Mean (average) of a number of values.

This function block calculates the average of a number of values passed as an array to the Function Block. The output or result from the Function Block is the Mean (average). The number of points to create the average is also passed.

1. Create a CX-Programmer Project and Function Block (in this case Structured Text).
Assign the following variables.




Note: The InOut parameter type allows passing of arrays - It is not possible to pass arrays using either Input or Output parameter types.

2. Now add the FB code using array data passed via the InOut Parameter type.

ArrayPassing.MeanCalc_V1_00 [FB Structured Text]

| Name | Data Type | AT | Initial Value | Retained | Comment |
|---|---|---|---|---|---|
| Data | INT [1] | | | | The data array for calculating the mean |

Internals | Inputs | Outputs | In Out | Externals

```
(* Calculate the mean of a number of values                          *)
(* This uses integer maths - to show the workings                    *)

(* Even though Data[1] array is only one array in size,              *)
(* it is possible to pass a variable to the array offset to access multiple data  *)
(* For example if i=10, then Data[ i ] will provide the 11th offset from the      *)
(* start of Data[0]                                                  *)

(*Initialise data *)
Total := 0;

(* Find total sum of data *)
FOR i := 0 TO (NoofValues - 1) DO
    Total := Total + INT_TO_DINT(Data[ i ]);
END_FOR;

(* Calculate Average *)
Average := DINT_TO_INT(Total / INT_TO_DINT(NoofValues));
```

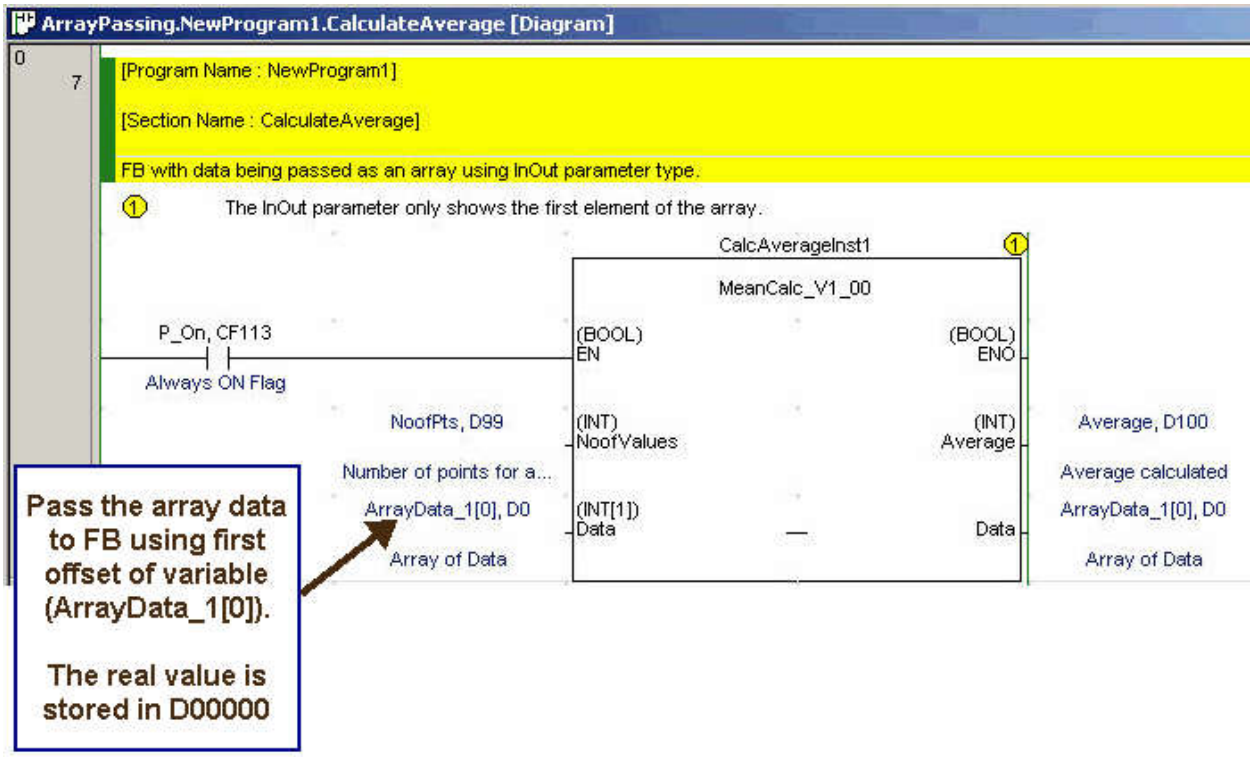
**Access the data passed as an array of data by using the '[ ]' operator. The variable inside ( i ) defines the offset from the start of the data being passed.**

3. The symbols and actual array data definition are defined in the symbol table (either Local or Global). Note the required number of elements are defined here (e.g. 10):

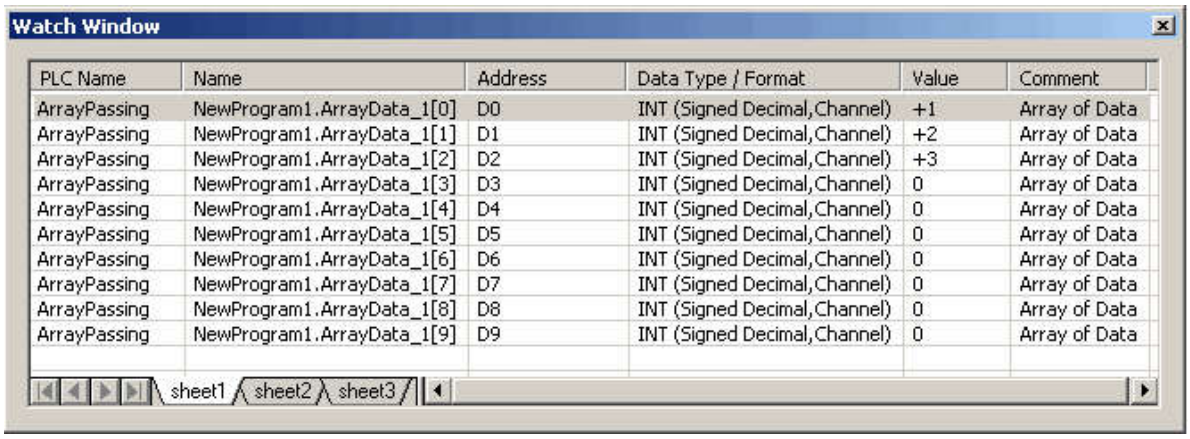
ArrayPassing.NewProgram1 [Symbols]

| Name | Data Type | Address / Value | Rack Location | Usage | Comment |
|---|---|---|---|---|---|
| ArrayData_1 | INT[10] | D0 | | Work | Array of Data |
| Average | INT | D100 | | Work | Average calculated |
| NoofPts | INT | D99 | | Work | Number of points for averaging |

4. Calling the Function Block (from ladder) requires passing actual array variable from the 'outside' world to the Function Block (hence allowing reuse of code).




5. Using the CX-Programmer 'Watch Window', it is possible to see the array information, the actual PLC address and the values...




The completed tutorial solutuion can be found as an attachment.

## Attachments

**ArrayPassing.zip** - Size: 5795

## Comments (View All Comments / Add Comment)

## Related Articles

No related articles found.

| | |
|---|---|
| Created | 2009-08-28 |
| Modified | 2009-09-07 |
| Views | 8552 |

Search Glossary

You are not logged in.