

I/O



[THE BLAZINGCORE SERIES]

WITH SUPERIOR NUMBER CRUNCHING ABILITIES AND PERIPHERAL HANDLING ON OUR CUSTOM EMBEDDED OS,
RAPID PROTOTYPING IS NOW EASY... AND BLAZING FAST.

I/O

The following table lists the commands available for addressing I/O ports, as well as methods for getting information from a particular I/O pin.

Operator	Description
GetADC	Returns converted digital values of an analog device attached to an ADC pin.
GetPin	Returns logical state of specified pin.
High	Set a specified pin's logical state to '1'.
Low	Set a specified pin's logical state to '0'.

Note: Using commands `GetADC`, `GetPin`, `High`, `Low` without referring to the IO Namespace is also allowed.

GETADC

Returns converted digital values of an analog device attached to an ADC pin. (*ADC Pins; 17 to 32*)

```
Integer variable = IO.GetADC(pin)  
Integer variable = GetADC(pin)
```

Code:

```
Dim I As Integer  
Public Sub Main()  
  
Do  
    I = IO.GetADC(17)  
    Debug.Print Cstr(I)  
    Delay(200)  
Loop  
  
End Sub
```

GETPIN

Returns logical state (High/Low) of specified pin.

```
Integer variable = IO.GetPin(pin)  
Integer variable = GetPin(pin)
```

Assuming that a Push Button Module is attached to the JST1 Port of the BCore, the following example code polls pin 1 at a rate of 5 times a second to detect if the push button has been pressed.

Code:

```
Dim I As Integer  
Public Sub Main()  
  
Do  
    I = IO.GetPin(1)  
    Debug.Print Cstr(I)  
    Delay(200)  
Loop  
  
End Sub
```

HIGH

Set a specified pin's logical state to '1'.

```
IO.High(pin)  
High(pin)
```

LOW

Set a specified pin's logical state to '0'.

```
IO.Low(pin)  
Low(pin)
```

Assuming that an LED Module is attached to the JST2 Port of the BCore, the following code is an example program that illustrates the use of the commands High and Low to toggle Pin 3 onboard the BCore Board, resulting in a blinking LED program.

Code:

```
Public Sub Main()  
Do  
    IO.High(3)  
    Delay(100)  
    IO.Low(3)  
    Delay(100)  
Loop  
End Sub
```

LATEST DOCUMENTATION

All of our documentations are constantly updated to provide accurate and/or new information that we feel would help you with developing with our products.

The latest documentation may be obtained from our website: <http://www.aiscube.com/main/downloads.html>

HOW YOU CAN HELP

You can help us to improve our documentations by emailing to us or posting a thread in our forum, reporting any mistakes/typos or errata that you might spot while reading our documentation.

Email: TechSupport@aiscube.com

Forum: <http://forum.aiscube.com/index.php>

DISCLAIMER

All information in this documentation is provided 'as-is' without any warranty of any kind.

The products produced by AIS Cube are meant for rapid prototyping and experimental usage; they are not intended nor designed for implementation in environments that constitute high risk activities.

AIS Cube shall assume no responsibility or liability for any indirect, specific, incidental or consequential damages arising out of the use of this documentation or product.

COPYRIGHT© 2009 - 2011 AIS CUBE. ALL RIGHTS RESERVED.

ALL PRODUCT AND CORPORATE NAMES APPEARING IN THIS DOCUMENTATION MAY OR MAY NOT BE REGISTERED TRADEMARKS OR COPYRIGHTS OF THEIR RESPECTIVE COMPANIES. AND ARE ONLY USED FOR IDENTIFICATION OR EXPLANATION FOR THE OWNER'S BENEFIT. WITH NO INTENT TO INFRINGE.

SONATA IDE AND BLAZINGCORE(BCORE) ARE TRADEMARKS OF AIS CUBE IN SINGAPORE AND/OR OTHER COUNTRIES. ALL IMAGES DEPICTING THE BLAZINGCORE OR ANY PART OF IT IS COPYRIGHTED.

ALL OTHER TRADEMARKS OR REGISTERED TRADEMARKS ARE THE PROPERTY OF THEIR RESPECTIVE OWNERS.