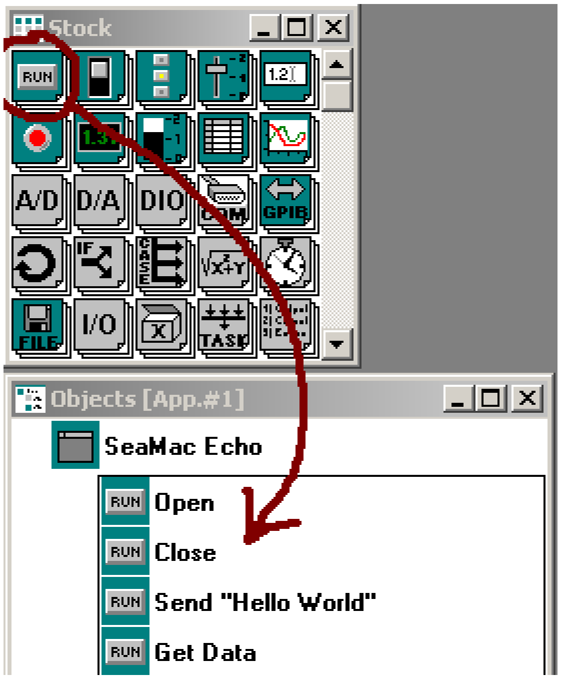
SEALEVEL
SYSTEMS INCORPORATED

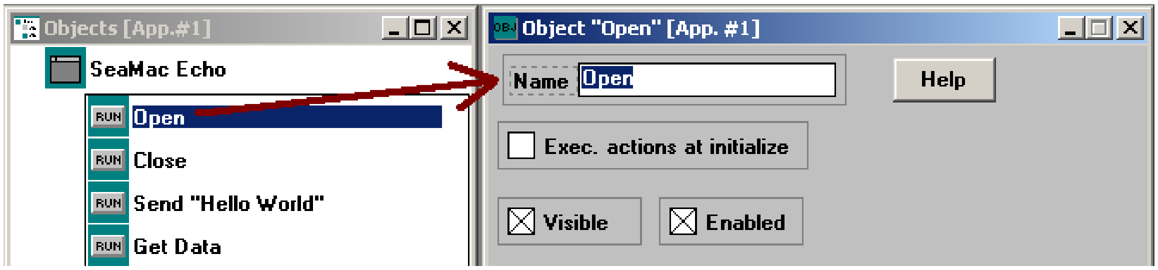
TestPoint with SeaMAC:

It is very easy to use any SeaMAC card with TestPoint. Install the card as per the directions in SeaMAC.hlp or your manual, and use the “SeaMAC Drivers” icon in the control panel to find out what port the card is installed as.

The method of attaching external APIs, (or any arbitrary .DLL,) to TestPoint is given in chapter 23 of the manual. For each call into the DLL, a separate "CODE" icon is brought into your list of objects. The "DLL filename" is SEAMAC32.DLL, and the "Subroutine Name" is any of the calls listed in the API. (Do not include the parentheses in the "Subroutine Name.")

Lets get started. First drag four “Pushbutton” objects to the objects list and name them as shown in the examples.





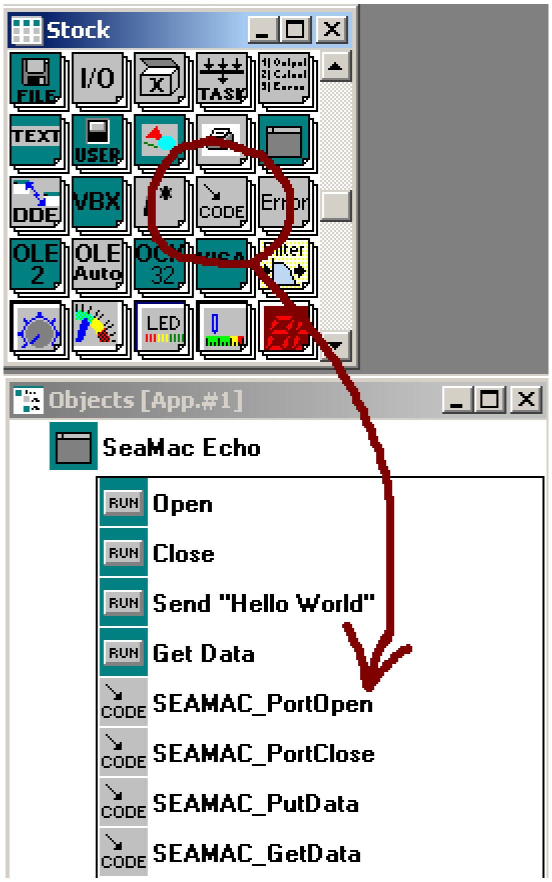
Name Open

☐ Exec. actions at initialize

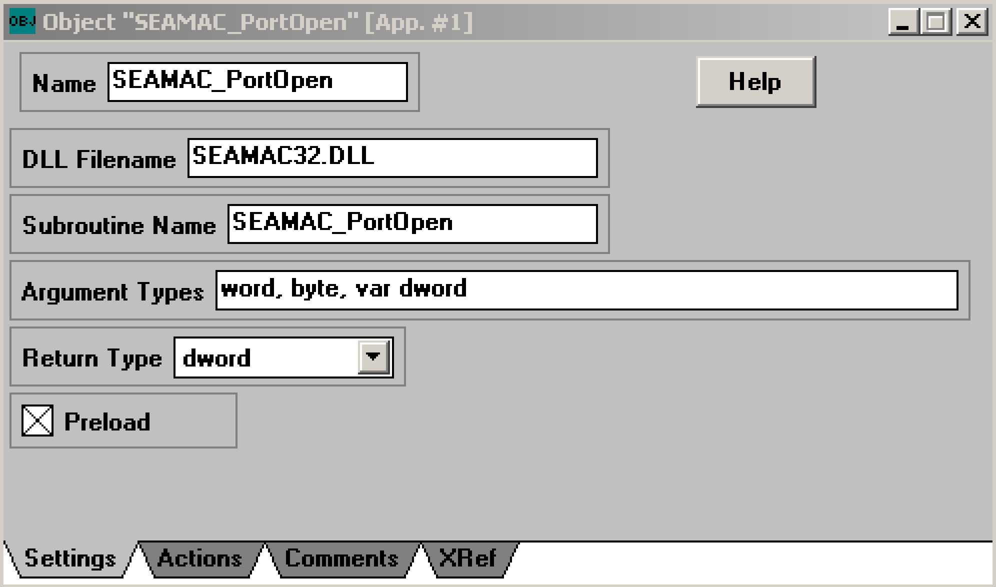
☒ Visible

☒ Enabled

Now that we have the Buttons on the form lets add the code that makes it all work! Drag four “Code” objects to the object list.



Double clicking on one of the code objects will bring up the object property dialog box. Open all the code property dialog boxes and set up as per the examples below.



Object "SEAMAC_PortOpen" [App. #1]

Name: SEAMAC_PortOpen

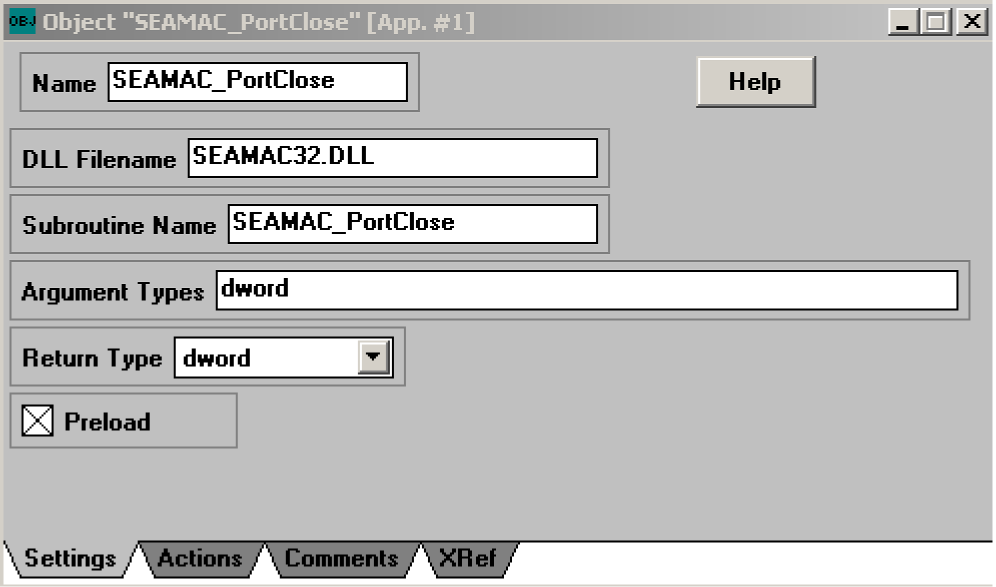
DLL Filename: SEAMAC32.DLL

Subroutine Name: SEAMAC_PortOpen

Argument Types: word, byte, var dword

Return Type: dword

☒ Preload



Object "SEAMAC_PortClose" [App. #1]

Name: SEAMAC_PortClose

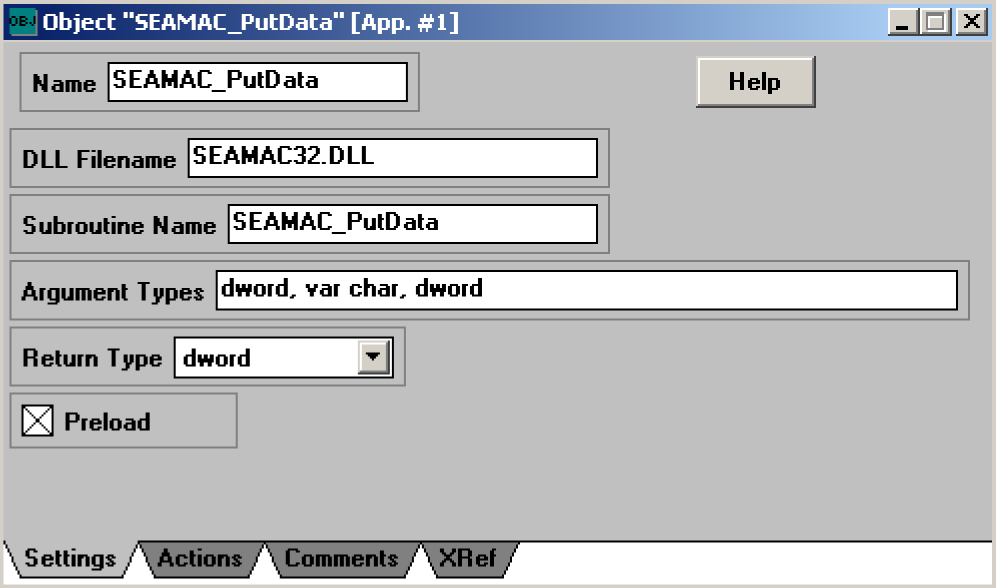
DLL Filename: SEAMAC32.DLL

Subroutine Name: SEAMAC_PortClose

Argument Types: dword

Return Type: dword

☒ Preload



Object "SEAMAC_PutData" [App. #1]

Name: SEAMAC_PutData

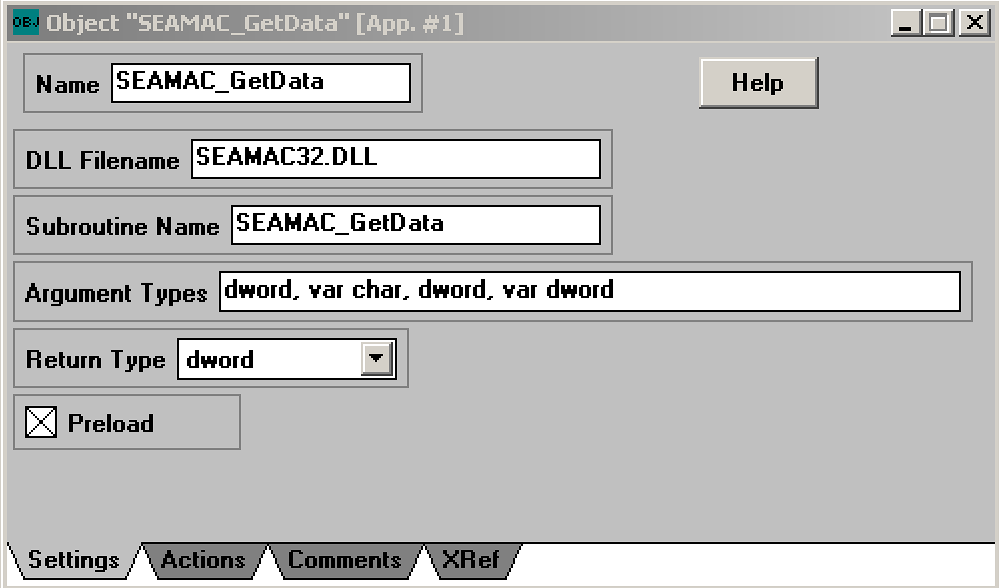
DLL Filename: SEAMAC32.DLL

Subroutine Name: SEAMAC_PutData

Argument Types: dword, var char, dword

Return Type: dword

☒ Preload



Object "SEAMAC_GetData" [App. #1]

Name: SEAMAC_GetData

DLL Filename: SEAMAC32.DLL

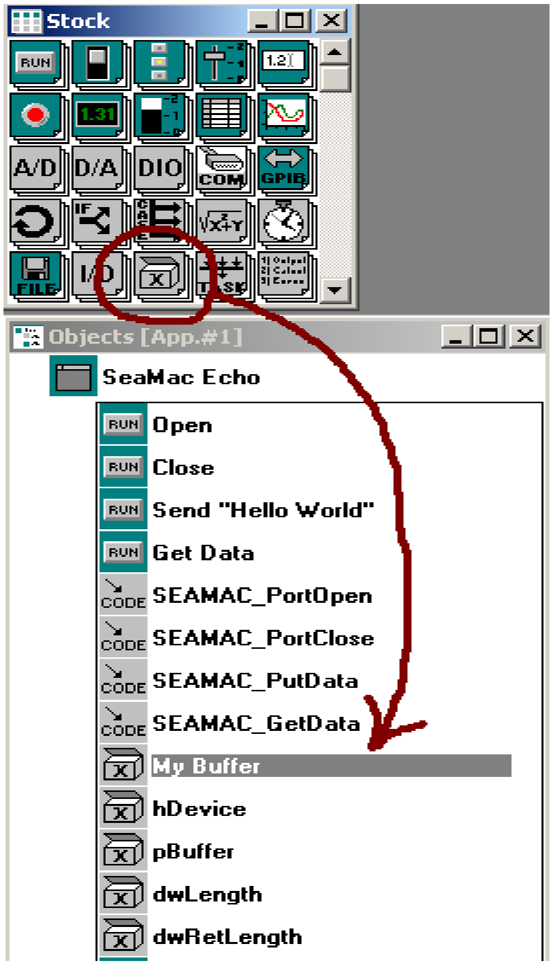
Subroutine Name: SEAMAC_GetData

Argument Types: dword, var char, dword, var dword

Return Type: dword

☒ Preload

Now that the code properties are setup correctly we can setup a couple of “Containers” to store data. Drag five “Container” objects to the object list and name them as per example.

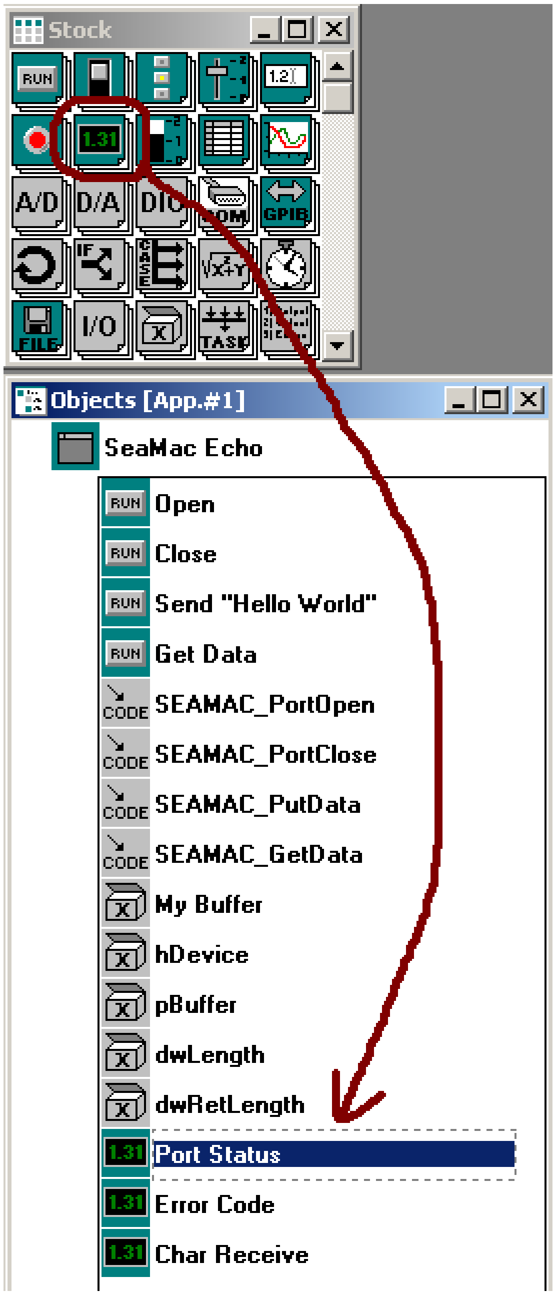


Objects [App.#1]

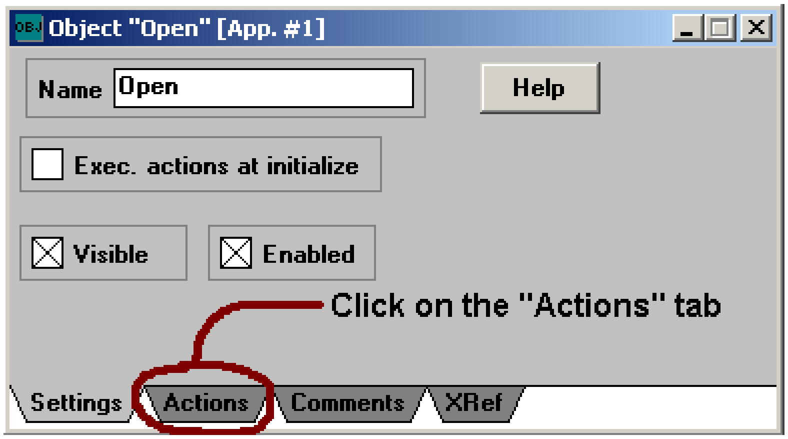
SeaMac Echo

- Open
- Close
- Send "Hello World"
- Get Data
- SEAMAC_PortOpen
- SEAMAC_PortClose
- SEAMAC_PutData
- SEAMAC_GetData
- My Buffer
- hDevice
- pBuffer
- dwLength
- dwRetLength

Lets add some “Display” objects so that we can view what is going on in our program. Drag three “Display” objects to the objects list and name as per example.



Lets get back to the "Pushbuttons". Double clicking on the pushbutton icons in the objects list will bring up their property dialog box. Click on the "Actions" tab and setup the four pushbuttons as per examples.



Object "Open" [App. #1]

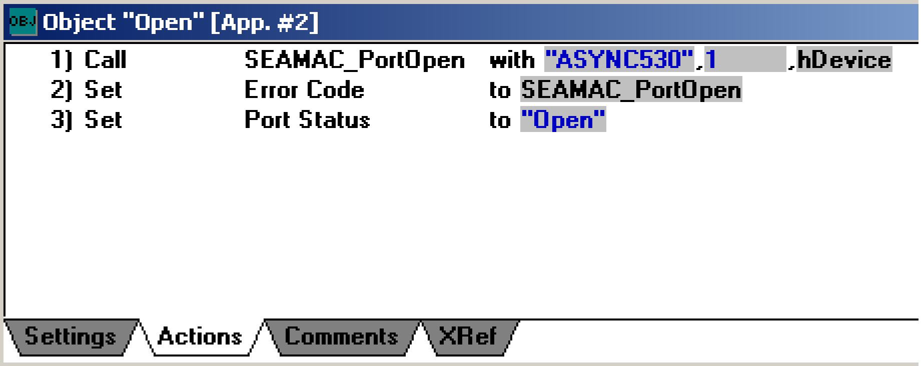
Name Open

☐ Exec. actions at initialize

☒ Visible ☒ Enabled

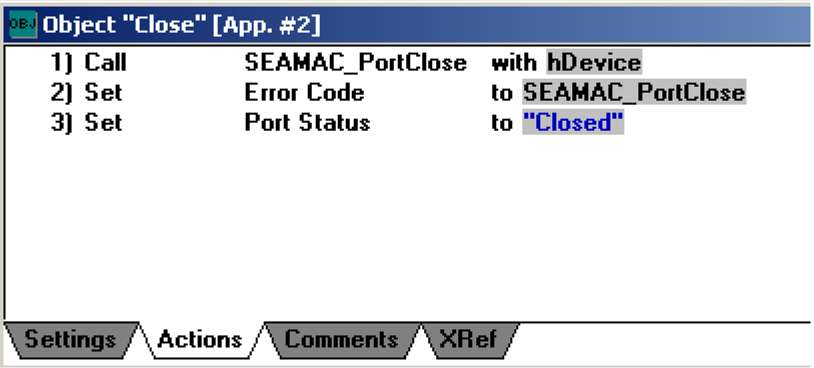
Click on the "Actions" tab

Here are the settings for each button.



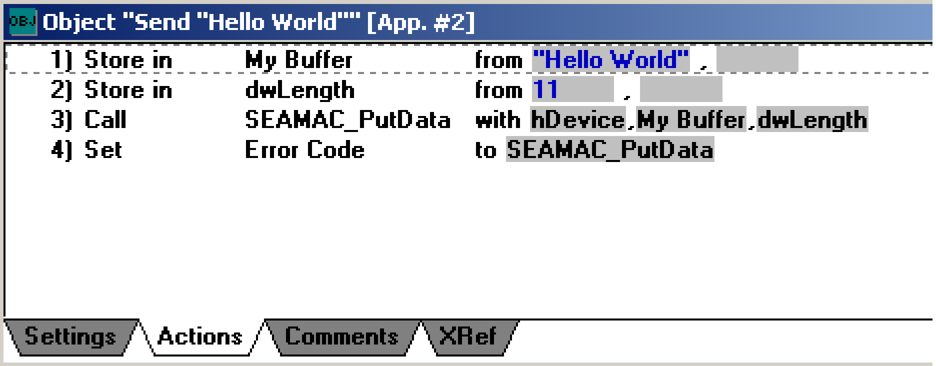
Object "Open" [App. #2]

```
1) Call     SEAMAC_PortOpen    with "ASYNC530",1     ,hDevice
2) Set      Error Code         to SEAMAC_PortOpen
3) Set      Port Status        to "Open"
```



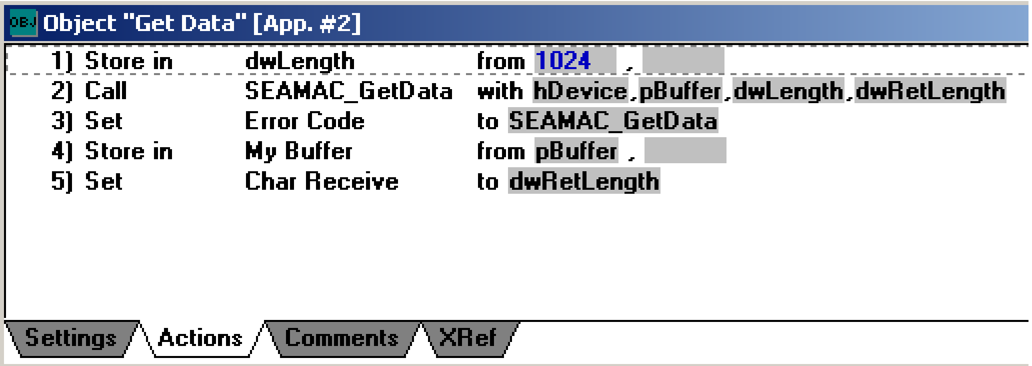
Object "Close" [App. #2]

```
1) Call     SEAMAC_PortClose   with hDevice
2) Set      Error Code         to SEAMAC_PortClose
3) Set      Port Status        to "Closed"
```



Object "Send "Hello World"" [App. #2]

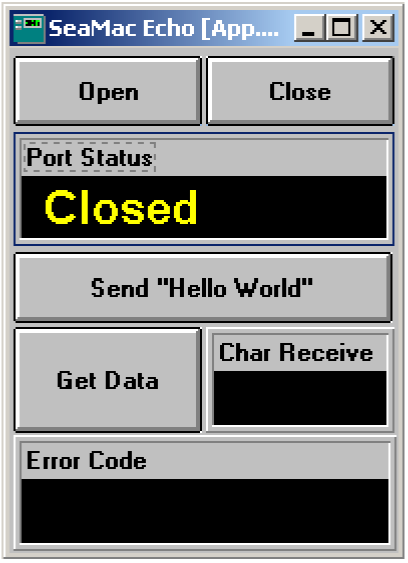
```
1) Store in   My Buffer        from "Hello World" ,
2) Store in   dwLength         from 11      ,
3) Call       SEAMAC_PutData   with hDevice,My Buffer,dwLength
4) Set        Error Code       to SEAMAC_PutData
```



Object "Get Data" [App. #2]

```
1) Store in    dwLength        from 1024 ,
2) Call        SEAMAC_GetData  with hDevice,pBuffer,dwLength,dwRetLength
3) Set         Error Code      to SEAMAC_GetData
4) Store in    My Buffer       from pBuffer ,
5) Set         Char Receive    to dwRetLength
```

Now that we have everything set up correctly, lets do some cosmetic cleanup. Here is an example of how ours turned out.



SeaMac Echo [App....

Port Status: Closed

The SeaMAC example distributed here, SeaMacEcho.TST, works with with the SeaMAC driver set to "Async" protocol. The "Get Data" function in the example assumes that an external loopback is in place, but is not necessary for running the driver.