



## Priore SmartCard ActiveX Control 1.4

Copyright © 1998/2004 Danilo Priore. All rights reserved.

<http://www.prioregroup.com>

### **Description**

The Smartcard Control is able to read and write numeric and alphanumeric data from GemClubMemo or other Smartcard compatible APDU commands.

### **File Name**

SCARD.OCX or SCARD.DLL for OLE/DLL ActiveX

### **Object Type**

SCard

### **Remarks**

This ActiveX OLE/DLL component is able to read and write numeric and alphanumeric data from GemClubMemo type's Smartcard, produced by [www.gemplus.com](http://www.gemplus.com), or others Smartcard compatible APDU commands.

The GemClubMemo is kind of smartcard have a very tiny memory but as well these have a very cheap price! Although everything, this kind of card is the best solution for a solid "protection" circuit, even because each card has onboard three different passwords (commonly named "Card Secret Code) and a protection that will put off-line the card after three unsuccessful authentication attempts for each password.

# Index

Technical Support	3
Requirements	3
<b>Properties</b>	<b>Pag</b>
ATR	4
ErrorDescription	
ErrorNumber	6
ReaderName	
State	
StatusWord	
Interval	
Connected	
<b>Methods</b>	
Connect	12
Disconnect	13
ReadCard	14
ReadCardAll	15
SendCSC	16
WriteCard	17
WriteCardAll	18
ShowReader	19
APDUCommand	20
<b>Events</b>	
CardIn	21
CardOut	22
Errors	23
<b>Classes</b>	
Readers	24
Cards	25
GemClubMemo	26
SmartCard	27
<b>HOW TO</b>	
Read from card	28
Write to card	29
Set new CSC	30
Check card when inserted or removed	33
Change secure access	34
Change the reader device	35
Send APDU commands	36
Write and Read numeric value	37
GemClubMemo Structure	38
License & Distribution	39

## Technical Support

For Technical support of this Control you may contact the author Danilo Priore thru e-mail at support@prioregroup.com. You may also find information of this Control bug fixes, new releases, discussion forum and FAQs thru the WEB at <http://www.prioregroup.com>

Please feel free to make any suggestion for this Control improvements

NOTE: Before requesting support thru e-mail please visit the Priore forums and try to search the forum for possible answers.

## Requirements

This Control for Visual Basic(tm) requires Visual Basic 6.0(tm) version or higher and Microsoft(tm) Smartcard System Library. The ActiveX(tm) controls can be used with any development environment that supports its. The 32-bit ActiveX(tm) control requires Windows 95/98(tm) or NT/2000(tm) or later.

**Copyright © 1998/2004 Danilo Priore. All rights reserved.**

This ActiveX Controls are trademarks of Danilo Priore. Microsoft Windows, Windows NT and Visual Basic are trademarks or registered trademarks of Microsoft Corporation. All other products and services mentioned are trademark or copyright of their respective owners

Information in this document is subject to change without notice. No part of this document can be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Danilo Priore.

## ATR Property

### **Description**

Return the ATR string of card.

### **Visual Basic**

*[form.]control*.**ATR**

### **Remarks**

The property **ATR** return a string to describe the ATR (Answer to Reset) of card.

### **Data Type**

String

### **Example**

MsgBox SCard1.**ATR**

## ErrorDescription Property

### Description

Return the description of error.

### Visual Basic

*[form.]control*.**ErrorDescription**

### Remarks

The property **ErrorDescription** return a string to describe the error.

### Data Type

String

### Example

MsgBox SCard1.**ErrorDescription**

## ErrorNumber Property

### **Description**

Return the number of error.

### **Visual Basic**

*[form.]control*.**ErrorNumber**

### **Remarks**

The property **ErrorNumber** return a code to describe the error.

### **Data Type**

Long

### **Example**

MsgBox SCard1.**ErrorNumber**

## ReaderName Property

### Description

Return the name of reader to installed on you system.

### Visual Basic

*[form.]control*.**ReaderName**

### Remarks

The property **ReaderName** return a string to describe the name of reader to installed on you system.

### Data Type

String

### Example

MsgBox SCard1.**ReaderName**

## State Property

### Description

Return the state of reader/card.

### Visual Basic

[*form.*]control.**State**

### Remarks

The property **State** return a state of reader/card.

### Data Type

Long

### Example

```
if SCard1.State = STATE_ABSENT Then
    MsgBox "SmartCard non present"
Else
    MsgBox "SmartCard present"
End If
```

## StatusWord Property

### Description

Return the status of last command.

### Visual Basic

[*form.*]control.**StatusWord**

### Remarks

The property **StatusWord** return a code to describe the state of last command.

### Data Type

Long

### Example

```
SCard1.Write 16, "ABCD"
```

```
If SCard1.StatusWord = SWORD_COMMAND_SUCCESS Then
```

```
    MsgBox "Write command, ok"
```

```
Else
```

```
    MsgBox "Error to execute write command"
```

```
End If
```

## Interval Property

### Description

Return or set the interval time, in milliseconds, to check state of card.

### Visual Basic

*[form.]control.Interval = m-seconds*

### Remarks

The property **Interval** set then interval time, in milliseconds, to check state of card. When this property set > 0 to generate the CardIn and CardOut events.

### Data Type

Long > 0 < 65536

### Example

SCard1.**Interval** = 2000 ' 2 seconds

## Connected Property

### Description

Return the connection state.

### Visual Basic

*[form.]control.Connected*

### Remarks

The **Connected** property, return the state of the connection to smartcard.

### Example

```
If Not SCard1.Connected Then SCard1.Connect  
SCard1.WriteCard 16, "ABCD"  
SCard1.Disconnect
```

## Connect Method

### **Description**

Connect to smartcard for read/write.

### **Visual Basic**

[*form.*]control.**Connect**

### **Remarks**

The **Connect** method, connect to smartcard for read/write or send CSC code.

### **Example**

SCard1.**Connect**

SCard1.WriteCard 16, "ABCD"

SCard1.Disconnect

## Disconnect Method

### **Description**

Disconnect to smartcard.

### **Visual Basic**

*[form.]control*.**Disconnect**

### **Remarks**

The **Disconnect** method, disconnect to smartcard for terminate operations.

### **Example**

SCard1.Connect

SCard1.WriteCard 16, "ABCD"

SCard1.**Disconnect**

## ReadCard Method

### Description

Read a data from card.

### Visual Basic

```
variable = [form.]control.ReadCard(start-address, [end-address], [CSCType], [CSC], [CSCHex])
```

### Remarks

The **ReadCard** method, read the data characters from smartcard.

### Data Type

String

### Example

```
SCard1.Connect
```

```
MsgBox SCard1.ReadCard(16)
```

```
SCard1.Disconnect
```

## ReadCardAll Method

### Description

Read a data from card.

### Visual Basic

```
variable = [form.]control.ReadCardAll([CSCType],[CSC],[CSCHex])
```

### Remarks

The **ReadCardAll** method, read long string from smartcard.

### Data Type

String

### Example

```
SCard1.Connect  
MsgBox SCard1.ReadCardAll  
SCard1.Disconnect
```

## SendCSC Method

### Description

Send CSC to card

### Visual Basic

[*form.*]control.**SendCSC** *type-csc,csc-string,[CSCHex]*

### Remarks

The **SendCSC** method, to send CSC, 4 char Card Secret Code, to enables read/write card.

### Example

```
Dim sCSC As String
sCSC = Chr$(&hAA) & Chr$(&hAA) & Chr$(&hAA) & Chr$(&hAA)
SCard1.Connect
SCard1.SendCSC CSC_ADMIN, sCSC
' or SCard1.SendCSC CSC_ADMIN, "AAAAAAA", True
MsgBox SCard1.ReadCard(16)
SCard1.Disconnect
```

## SetCSC Method

### Description

Set new CSC for specific area of card.

### Visual Basic

[*form.*]control.**SetCSC** *type-csc,new-csc,[CSCHex]*

### Remarks

The **SetCSC** method, set/change the CSC (Card Secret Code) fro specific area of card.

### Example

```
Dim def_CSC As String
def_CSC = Chr(&hAA) & Chr(&hAA) & Chr(&hAA) & Chr(&hAA)
SCard1.Connect
SCard1.SendCSC CSC_ADMIN, def_CSC
SCard1.SetCSC CSC_ADMIN, "1234"
' or SCard1.SetCSC CSC_ADMIN, "1234", True
SCard1.Disconnect
```

## WriteCard Method

### **Description**

Write a data to card.

### **Visual Basic**

*[form.]control*.**WriteCard** *address,data-string,[CSCType],[CSC],[CSCHex]*

### **Remarks**

The **WriteCard** method, write string to smartcard.

### **Example**

SCard1.Connect

SCard1.**WriteCard** 16, "ABCD"

SCard1.Disconnect

## WriteCardAll Method

### **Description**

Write a data to card.

### **Visual Basic**

*[form.]control*.**WriteCard** *data-string*,*[CSCType]*,*[CSC]*,*[CSCHex]*

### **Remarks**

The **WriteCardAll** method, write long string to smartcard.

### **Example**

SCard1.Connect

SCard1.**WriteCardAll** "http://www.prioregroup.com"

SCard1.Disconnect

## ShowReadersList Method

### Description

Show readers list installed.

### Visual Basic

*variable* = [*form.*]control.**ShowReadersList**

### Remarks

The **ShowReadersList** method, show readers installed on system and return the name of reader selected.

### Data Type

String

### Example

```
SCard1.ReaderName = SCard1.ShowReadersList
```

## APDUCommand Method

### Description

Send APDU command to card.

### Visual Basic

[*form.*]control.**APDUCommand**(CLA, INS, P1, P3, Le, SendData(), RecvData())

### Remarks

The **APDUCommand** method, send the APDU command to smartcard, additional include more data to send and retrieve the response data.

### Data Type

Long

### Example

```
Dim Send(0) As Byte ' 0 no more data to send
Dim Recv() As Byte ' no default size
Dim i As Integer
Dim s As String
```

```
SCard1.Connect
SCard1.APDUCommand &h80, &hBE, &h0, &h10, &h4, Send(), Recv()
SCard1.Disconnect
s = vbNullString
For i = 0 To Ubound(Recv)
    s = s & Hex$(Recv(i)) & " "
Next
MsgBox s
```

## CardIn Event

### **Description**

Fired when card is present.

### **Remarks**

The event **CardIn** fired one time when card is present and property Interval is set  $> 0$ .

## CardOut Event

### **Description**

Fired when card is not present.

### **Remarks**

The event **CardOut** fired one time when card not is present and property Interval is set  $> 0$ .

## Errors Event

### Description

Fired when error is occurred.

### Visual Basic

[*form.*]control.**Errors**(IErrNum As Long, sErrDesc As String)

### Remarks

The event **Errors** fired when error is occurred or smartcard is removed.

## Readers Class

### Description

Return a collection of readers.

### Visual Basic

Set Rdr = SmartCardCtl.**Readers**

### Remarks

The Class **Readers** return a collection of readers installed on computer.

### Data Type

Class

### Example

```
Dim Rdr As New SmartCardCtl.Readers
For i = 1 To Rdr.Count
    Debug.Print Rdr.Item(i)
Next
```

## Cards Class

### **Description**

Return a collection of the cards.

### **Visual Basic**

Set Crd = SmartCardCtl.**Cards**

### **Remarks**

The Class **Cards** return a collection of cards installed on computer.

### **Data Type**

Class

### **Example**

```
Dim Crd As New SmartCardCtl.Card  
For i = 1 To Crd.Count  
    Debug.Print Crd.Item(i)  
Next
```

## GemClubMemo Class

### Description

Class Module for GemClubMemo smartcard type.

### Visual Basic

Set GCM = SmartCardCtl.**GemClubMemo**

### Remarks

This Class Module to able to send commands to GemClubMemo smartcard type.

### Data Type

Class

### Example

```
Dim hCard As Long
Dim SC As New SmartCardCtl.SmartCard
Dim GCM As New SmartCardCtl.GemClubMemo
If SC.Connect = 0 Then
    hCard = SC.CardHandle
    Debug.Print GCM.ReadCard(hCard, &h10)
    SC.Disconnect
End If
```

## SmartCard Class

### Description

Class Module for generic smartcard type.

### Visual Basic

Set SC = SmartCardCtl.**SmartCard**

### Remarks

This Class Module to able to send commands to generic smartcard type.

### Data Type

Class

### Example

```
Dim hCard As Long
Dim SC As New SmartCardCtl.SmartCard
Dim GCM As New SmartCardCtl.GemClubMemo
If SC.Connect = 0 Then
    hCard = SC.CardHandle
    Debug.Print GCM.ReadCard(hCard, &h10)
    SC.Disconnect
End If
```

## HOWTO: Read from card

For your read data from card, you may use two of the following way:

### Read a string of 4 characters length, this a standard size for data

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True
    s = .ReadCard(&h10) ' read 4 charaters string on 0x10h address
    .Disconnect ' disconnect from card
End With
```

### Read a string of >4 characters length, but less or equal 64 characters length

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True
    s = .ReadCard(&h10, &h1F) ' read string on start 0x10h address to 0x1F address
    .Disconnect ' disconnect from card
End With
```

### Read a string of >4 characters length, but less or equal 128 characters length

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True
    s = .ReadCardAll ' read string
    .Disconnect ' disconnect from card
End With
```

## HOWTO: Write to card

For your write data to card, you may use this following way:

### **Write a string of >0 characters length, but less or equal 64 characters length**

With SCard1

```
.Connect ' connect to card  
.SendCSC CSC_ADMIN, "AAAAAAA", True  
.WriteCard &h10, "ABCD" ' write ABCD string on start 0x10h address  
.Disconnect ' disconnect from card
```

End With

### **Write a string of >0 characters length, but less or equal 128 characters length**

With SCard1

```
.Connect ' connect to card  
.SendCSC CSC_ADMIN, "AAAAAAA", True  
.WriteCardAll "ABCD" ' write ABCD string  
.Disconnect ' disconnect from card
```

End With

## HOWTO: Set new CSC

For your set or change the new CSC (Card Secret Code), you may use tree of the following way:

### Set CSC for Admin Area

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAA", True ' send the default CSC in hex
    .SetCSC CSC_ADMIN, "1234" ' set the new CSC
    .Disconnect ' disconnect from card
End With
```

### Set CSC for User Area 1

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "11111111", True ' send the default CSC in hex
    .SetCSC CSC_USER1, "5678" ' write new CSC for User Area 1
    .Disconnect ' disconnect from card
End With
```

### Set CSC for User Area 2

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "22222222", True ' send the default CSC in hex
    .SetCSC CSC_USER2, "9012" ' write new CSC for User Area 2
    .Disconnect ' disconnect from card
End With
```

## HOWTO: Order ActiveX Control

For your convenience, you may order by credit card in any of the following ways:

World Wide Web Credit Card Interface (Safe Transactions)

Online: <http://www.prioregroup.com>

Fax Ordering

You may fax your completed order form at (+39)-06-233-226-259  
24 hours a day, 7 days a week.

Postal Ordering

Orders sent via postal delivery should be sent to our P.O. Box. Please include the complete payment in the form of a personal check, money order, or cashier's check. Do not send cash. Payments made by personal check will be held until the check clears. Payments made with cashier's checks will be immediately fulfilled.

## HOWTO: Check card when inserted or removed

For your to know the card when inserted or removed, you may use two of the following way:

### **Check the state property**

```
With SCard1
    Do While .State <> STATE_SPECIFIC
        MsgBox "Please, Insert the Smartcard on Reader."
    Loop
End With
```

### **Activate events**

```
' write follow code on Form Load
SCard1.Interval = 2000 ' every 2 seconds

' on CardIn event
Private Sub SCard1_CardIn()
    ' write code here for read or write card
    ' when card is inserted
End Sub

' on CardOut event
Private Sub SCard1_CardOut()
    MsgBox "Attention, the Smartcard is removed from Reader."
End Sub
```

## HOWTO: Change Secure Access

For your change secure access for read and write data from card, you may use four of the following way:

### Free read and protected write (standard)

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True ' send CSC Admin Area in hex
    .WriteCard &h05, Chr(&h0) & Chr(&h0) & Chr(&h0) & Chr(&h0)
    .Disconnect ' disconnect from card
End With
```

### Protected read and write

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True ' send CSC Admin Area in hex
    .WriteCard &h05, Chr(&hAA) & Chr(&h0) & Chr(&h0) & Chr(&h0)
    .Disconnect ' disconnect from card
End With
```

### Free read and no write allow

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True ' send CSC Admin Area in hex
    .WriteCard &h05, Chr(&h55) & Chr(&h0) & Chr(&h0) & Chr(&h0)
    .Disconnect ' disconnect from card
End With
```

### Protected read and no write allow

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "AAAAAAA", True ' send CSC Admin Area in hex
    .WriteCard &h05, Chr(&hFF) & Chr(&h0) & Chr(&h0) & Chr(&h0)
    .Disconnect ' disconnect from card
End With
```

## HOWTO: Change the Reader device

For your change the reader with one of multiple readers installed on computer, you may use the two following way:

### **From default dialogbox**

```
With SCard1
    sRdr = .ShowReadersList
    if Len(sRdr) > 0 Then .ReaderName = sRdr
End With
```

### **Without dialogbox (select last reader)**

```
Dim cRdr As New SmartCardCtl.Readers
SCard1.ReaderName = cRdr.Item(cRdr.Count)
```

## HOWTO: Send APDU Command

For your send generic APDU command to card, you may use two of the following way:

### Send APDU command without more data

```
Dim Send(0) As Byte ' 0 no more data to send
Dim Recv() As Byte ' no default size
Dim i As Integer
Dim s As String

With SCard1
    .Connect
    ' read data from card
    .APDUCommand &h80, &hBE, &h0, &h10, &h4, Send(), Recv()
    .Disconnect
End With

s = vbNullString
For i = 0 To Ubound(Recv)
    s = s & Hex$(Recv(i)) & " "
Next
MsgBox s
```

### Send APDU command with more data

```
Dim Send(3) As Byte ' 4 more data to send
Dim Recv() As Byte ' no default size
Dim i As Integer
Dim s As String

Send(0) = 65 ' A
Send(1) = 66 ' B
Send(2) = 67 ' C
Send(3) = 68 ' D

With SCard1
    .Connect
    ' write data ti card
    .APDUCommand &h80, &hDE, &h0, &h10, &h4, Send(), Recv()
    .Disconnect
End With

s = vbNullString
For i = 0 To Ubound(Recv)
    s = s & Hex$(Recv(i)) & " "
Next
MsgBox s
```

## HOWTO: Write and Read a numeric value

For your write and read numeric value, you may use two of the following way:

### **Write a numeric value no more than 4294967295 (0xFFFFFFFF)**

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "11111111", True
    .WriteCard &h10, Hex$(number) ' write numeric value in hex
    .Disconnect ' disconnect from card
End With
```

### **Read a numeric value no more than 4294967295 (0xFFFFFFFF)**

```
With SCard1
    .Connect ' connect to card
    .SendCSC CSC_ADMIN, "11111111", True
    s = .ReadCard(&h10) ' read numeric value
    .Disconnect ' disconnect from card
    MsgBox CLng("&h" & s) ' show the value
End With
```



# END-USER LICENSE AGREEMENT FOR DANILO PRIORE SOFTWARE

IMPORTANT-READ CAREFULLY: This End User License Agreement (this "**EULA**") contains the terms and conditions regarding your use of the SOFTWARE (as defined below). This EULA contains material limitations to your rights in that regard. You should read this EULA carefully and treat it as valuable property.

## I. THIS EULA.

1. **Software Covered by this EULA.** This EULA governs your use of the Danilo Priore software product(s) ("DPS") enclosed or otherwise accompanied herewith (individually and collectively, the "**SOFTWARE**"). The term "SOFTWARE" includes, to the extent provided by DPS: 1) any revisions, updates and/or upgrades thereto; 2) any data, image or executable files, databases, data engines, computer software, or similar items customarily used or distributed with computer software products; 3) anything in any form whatsoever intended to be used with or in conjunction with the SOFTWARE; and 4) any associated media, documentation (including physical, electronic and on-line) and printed materials (the "**Documentation**").

2. **This EULA is a Legally Binding Agreement Between You and DPS.** If you are acting as an agent of a company or another legal person, such as an officer or other employee acting for your employer, then "you" and "your" mean your principal, the entity or other legal person for whom you are acting. However, importantly, even if you are acting as an agent for another, you may still be personally liable for violation of federal and State laws, such as copyright infringement.

**This EULA is a legally binding agreement between you and DPS.** You intend to be legally bound to this EULA to the same extent as if DPS and you physically signed this EULA. By installing, copying, or otherwise using the SOFTWARE, you agree to be bound by the terms and conditions contained in this EULA. If you do not agree to all of the terms and conditions contained in this EULA, you may not install or use the SOFTWARE. If, for whatever reason, installation has begun or has been completed, you should cancel installation or un-install the SOFTWARE, as the case may be. (You may click on the "exit" button or its equivalent to immediately abort installation.) If you do not agree to all of these terms and conditions, then you must promptly return the SOFTWARE to the place of business from which you obtained it in accordance with any return policies of such place of business. Return policies may vary between or among resellers, and you must comply with your particular reseller's return policies as agreed at the point of purchase. If the place of business from which you purchased the SOFTWARE does not honor a complete refund for a period of thirty (30) days from the date of proof of purchase, then you may return the SOFTWARE directly to DPS for a period of thirty (30) days from the date of your purchase. To return the product directly to DPS, you must obtain a DPS Return Authorization Number by contacting DPS, and you must forward all items purchased, including the proof of purchase, directly to DPS. The return must be postage-prepaid, and post-marked within thirty (30) days from the proof of purchase, time being of the essence. The return option to DPS is only available to the original purchaser of an unopened factory packaged item.

## II. YOUR LICENSE TO DEVELOP AND TO DISTRIBUTE.

As provided in more detail below, this EULA grants you two licenses: 1) a license to use the SOFTWARE to develop other software products (the "**Development License**"); and 2) a license to use and/or distribute the Developed Software (the "**Distribution License**"). Both of these licenses (individually and collectively, the "**Licenses**") are explained and defined in more detail below.

1. Definitions. The following terms have the respective meanings as used in this EULA:

"**Network Server**" means a computer with one or more computer central processing units (CPU's) that operates for the purpose of serving other computers logically or physically connected to it, including, but not limited to, other computers connected to it on an internal network, intranet or the Internet. "**Web Server**" means a type of Network Server that serves other computers more particularly connected to it over an intranet or the Internet.

"**Developed Software**" means those computer software products that are developed by or through the use of the SOFTWARE. "**Developed Web Server Software**" means those Developed Software products that reside logically or physically on at least one Web Server and are operated (meaning the computer software instruction set is carried out) by the Web Server's central processing unit(s) (CPU). "**Developed Legacy Software**" means those Developed Software products that are not Developed Web Server Software, including, for example, stand-alone applications and applications accessed by a file server only.

"**Redistributable Files**" means the SOFTWARE files or other portions of the SOFTWARE that are provided by DPS and are identified as such in the Documentation for distribution by you with the Developed Software. "**Developer**" means a human being or any other automated device using the SOFTWARE in accordance with the terms and conditions of this EULA.

"**Developer Seat License**" means that each Developer using or otherwise accessing the programmatic interface or the SOFTWARE must obtain the right to do so by purchasing a separate End User License.

"**Source Code**" shall mean computer software code or programs in human readable format, such as a printed listing of such a program written in a high-level computer language. The term "Source Code" includes, but is not limited to, documents and materials in support of the development effort of the SOFTWARE, such as flow charts, pseudo code and program notes.

**2. Your Development License.** You are hereby granted a limited, royalty-free, non-exclusive right to use the SOFTWARE to design, develop, and test Developed Software, on the express condition that, and only for so long as, you fully comply with all terms and conditions of this EULA.

The SOFTWARE is licensed to you on a Developer Seat License basis.

The Developer Seat License means that you may perform a single install of the SOFTWARE for use in designing, testing and creating Developed Software by a single Developer on a single computer with a single set of input devices, so long as such computer is used only by one Developer. Conversely, you may not install or use the SOFTWARE on a computer that is a network server or a computer at which the SOFTWARE is used by more than one Developer. You may not network the SOFTWARE or any component part of it, where it is or may be used by more than one Developer unless you purchase an additional Development License for each Developer. You must purchase another separate license to the SOFTWARE in order to add additional developer seats, whether the additional developers are accessing the SOFTWARE in a stand-alone environment or on a computer network.

In all cases, you may not use DPS 's name, logo, or trademarks to market your Developed Software without the express written consent of DPS; (b) you must include the following DPS copyright notice in your Developed Software documentation and/or in the "About Box" of your Developed Software, and wherever the copyright/rights notice is located in the Developed Software ("Portions Copyright © Danilo Priore, LLC 1998-2004. All Rights Reserved."); (c) agree to indemnify, hold harmless, and defend DPS, its suppliers and resellers, from and against any claims or lawsuits, including attorney's fees that may arise from the use or distribution of your Developed Software; (d) you may use the SOFTWARE only to create Developed Software that is significantly different than the SOFTWARE.

### **3. Your Distribution License.**

**License to Distribute Developed Software.** Subject to the terms and conditions in this EULA, you are granted the license to use and to distribute Developed Software on a royalty-free basis, provided that the Developed Software incorporates the SOFTWARE as an integral part of the Developed Software in machine-language compiled format (customarily an ".exe", or ".dll", etc.). You may not distribute, bundle, wrap or subclass the SOFTWARE as Developed Software which, when used in a "designtime" development environment, exposes the programmatic interface of the SOFTWARE. You may distribute, on a royalty-free basis, Redistributable Files with Developed Software only.

**4. Specific Product Limitations.** Notwithstanding anything in this EULA to the contrary, if the license you have purchased is for any of the following products, then the following additional limitations will apply:

**a. Priore FingerPrint for .NET.** Priore FingerPrint for .NET includes at least: 1) tree dynamic link library (pb.dll, pbsdk.dll and pbwiz.dll), 2) the Source Code of the Database Sample Application. The pb.dll, pbsdk.dll and pbwiz.dll dynamic link library is subject to the general terms and restrictions set forth in PreciseBiometrics™ SDK; this dynamic link library may be distributed, free of royalties, only in conjunction with the Developed Software.

Subject to the terms and conditions in this EULA, DPS hereby grants you the right to use the Sample Source Code. You are hereby also granted the right to modify such Sample Source Code and to create derivative works that are based on the licensed Source Code. You may distribute the derivative works that you develop, solely in object code format and exclusively in conjunction with and/or as a part of the Developed Software. You are expressly not granted the right to distribute, disclose or otherwise make available to any third party the licensed Source Code, any modified version, derivative work, or any portion thereof, in source code format.

DPS shall retain all right, title and interest in and to the licensed Source Code, and all DPS updates, modifications or enhancements thereof. Nothing herein shall be deemed to transfer any ownership or title rights in and to the licensed Source Code from DPS to you.

THE SAMPLE SOURCE CODE IS LICENSED TO YOU AS IS. DPS DOES NOT AND SHALL NOT PROVIDE YOU WITH ANY TECHNICAL SUPPORT FOR YOUR SOURCE CODE LICENSE.

**5. Updates/Upgrades.** Subject to the terms and conditions of this EULA, the Licenses are perpetual. Updates and upgrades to the SOFTWARE may be provided by DPS from time-to-time, and, if so provided by DPS, are provided upon the terms and conditions offered at that time by DPS in its sole discretion. DPS may provide updates and upgrades to the SOFTWARE for free or for any charge, at any time or never, and through its chosen manner of access and distribution, all in DPS 's sole and complete discretion.

**6. Serial Number.** Within the packaging of the SOFTWARE, a unique serial number (the "**Serial Number**") is included, which allows for the registration of the SOFTWARE. The Serial Number is subject to the restrictions set forth in this EULA and may not be disclosed or distributed either with your Developed Software or in any other way. The disclosure or distribution of the Serial Number shall constitute a breach of this EULA, the effect of which shall be the automatic termination and revocation of all the rights granted herein.

**7. Evaluation Copy; SHAREWARE.** If you are using an "evaluation copy" or similar version, specifically designated as such by DPS on its website or otherwise, then the Licenses are limited as follows: a) you are granted a license to use the SOFTWARE for a period of thirty (30) days counted from the day of installation (the "**Evaluation Period**"); b) upon completion of the Evaluation Period, you shall either i) delete the SOFTWARE from the computer containing the installation, or you may ii) contact DPS or one of its authorized dealers to purchase a license of the SOFTWARE, which is subject to the terms and limitations contained herein; and c) any Developed Software may not be distributed or used for any commercial purpose.

### **III. INTELLECTUAL PROPERTY.**

**1. Copyright.** You agree that all right, title, and interest in and to the SOFTWARE (including, but not limited to, any images, photographs, animations, video, audio, music, text, and "applets" incorporated into the SOFTWARE), and any copies of the SOFTWARE, and any copyrights and other intellectual properties therein or related thereto are owned exclusively by DPS, except to the limited extent that DPS may be the rightful license holder of certain third-party technologies incorporated into the SOFTWARE. The SOFTWARE is protected by copyright laws and international treaty provisions. The SOFTWARE is licensed to you, not sold to you. DPS reserves all rights not otherwise expressly and specifically granted to you in this EULA.

**2. Backups.** You may either: (a) copy the SOFTWARE solely for backup or archival purposes; or (b) install the SOFTWARE on a single computer, provided you keep the original solely for backup or archival purposes. Notwithstanding the foregoing, you may not copy the Documentation.

**3. General Limitations.** You may not reverse engineer, decompile, or disassemble the SOFTWARE, except and only to the extent that applicable law expressly permits such activity notwithstanding this limitation.

**4. Software Transfers.** You may not rent or lease the SOFTWARE. You may transfer the SOFTWARE to another computer, provided that it is completely removed from the computer from which it was transferred. You may permanently transfer all of your rights under the EULA, provided that you retain no copies, that you transfer all the SOFTWARE (including all component parts, the media and printed materials, any dates, upgrades, this EULA and, if applicable, the Certificate of Authenticity), and that the recipient agrees to the terms and conditions of this EULA as provided herein. If the SOFTWARE is an update or upgrade, any transfer must include all prior versions of the SOFTWARE.

**5. Termination.** Without prejudice to any other rights it may have, DPS may terminate this EULA and the Licenses if you fail to comply with the terms and conditions contained herein. In such an event, you must destroy all copies of the SOFTWARE and all of its component parts.

### **IV. WARRANTIES AND REMEDIES.**

**1. Limited Warranty.** DPS warrants that the original media, if any, are free from defects for ninety (90) days from the date of delivery of the SOFTWARE. **EXCEPT AS OTHERWISE PROVIDED IN THE PRECEDING SENTENCE, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, DPS EXPRESSLY DISCLAIMS ANY WARRANTY FOR THE SOFTWARE, DOCUMENTATION AND ANYTHING ELSE PROVIDED BY DPS HEREBY AND DPS PROVIDES THE SAME IN "AS IS" CONDITION WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE AND DOCUMENTATION REMAINS WITH YOU. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS WHICH VARY FROM STATE TO STATE.**

**2. Limited Remedy.** DPS 's entire liability and your exclusive remedy under this EULA shall be, at DPS 's sole option, either (a) return of the price paid for the SOFTWARE; (b) repair the SOFTWARE through updates distributed online or otherwise in DPS's discretion; or (c) replace the SOFTWARE with SOFTWARE that substantially performs as described in the SOFTWARE documentation, provided that you return the SOFTWARE in the same manner as provided in Section I.2 for return of the SOFTWARE for non-acceptance of this EULA. Any media for any repaired or replacement SOFTWARE will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. **TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL DPS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFIT, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF DPS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES/JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES IN CERTAIN CASES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.**

## **V. MISCELLANEOUS.**

**1. This is the Entire Agreement.** This EULA (including any addendum or amendment to this EULA included with the SOFTWARE) is the final, complete and exclusive statement of the entire agreement between you and DPS relating to the SOFTWARE. This EULA supersedes any prior and contemporaneous proposals, purchase orders, advertisements, and all other communications in relation to the subject matter of this EULA, whether oral or written. No terms or conditions, other than those contained in this EULA, and no other understanding or agreement which in any way modifies these terms and conditions, shall be binding upon the parties unless entered into in writing executed between the parties, or by other non-oral manner of agreement whereby the parties objectively and definitively act in a manner to be bound (such as by continuing with an installation of the SOFTWARE, "clicking-through" a questionnaire, etc.) Employees, agents and other representatives of DPS are not permitted to orally modify this EULA.

**2. You Indemnify DPS.** . You agree to indemnify, hold harmless, and defend DPS and its suppliers and resellers from and against any and all claims or lawsuits, including attorney's fees, which arise out of or result from your breach of any of the terms and conditions of this EULA.

**3. Interpretation of this EULA.** If for any reason a court of competent jurisdiction finds any provision of this EULA, or any portion thereof, to be unenforceable, that provision of this EULA will be enforced to the maximum extent permissible so as to effect the intent of the parties, and the remainder of this EULA will continue in full force and effect. Formatives of defined terms shall have the same meaning of the defined term. Failure by either party to enforce any provision of this EULA will not be deemed a waiver of future enforcement of that or any other provision. Except as otherwise required or superseded by law, this EULA is governed by the laws of the legal offices of Italy, without regard to its conflict of laws principles. The parties consent to the personal jurisdiction and venue of the legal offices of Italy, and agree that any legal proceedings arising out of this EULA shall be conducted solely in such legal offices. If the SOFTWARE was acquired outside the Italy, then local law may apply.