Library/DLL is specifically dedicated for systems integrators and programmers who need to develop custom applications.

Library/DLL

Features

- **Included as standard in applicom® and Direct-Link™ PC interface card**

- **Independent from the product and the protocol used**

- **NO license required, NO extra cost**

- **Available for Windows 32-bit and other operating systems**

Description

Woodhead provides with its applicom® products, free of charge, a function library also called API (**A**pplication **P**rogram **I**nterface). Those API functions are standardized and independent of the operating system and the protocol used.

This library is available under Windows XP/NT4/2000 and other non-Windows operating systems as Linux, VxWorks, QNX, etc.

The API library is delivered as **DLL for Windows 32-bit environments** and can be used with all Windows based programming languages:
- Microsoft Visual Basic, Visual Basic. NET, Visual C/C++/C#/J#
- Borland Delphi, Borland C/C++
- National Instruments LabWindows/CVI, LabView/VI
- PCSoft Windev

Operating modes

1. Wait mode

This mode is used to manage **synchronous** exchange. When functions run in this mode, the application program will wait until:
- The function is transferred to the protocol task of the aimed channel
- The formatting related to the protocol is done and transmission is performed on the network
- The target device response is received and processed
- The formatting of the variables returned is done with an exchange status

Advantage: making the application upgradeable synchronously regarding to the variable status change in the device

Read functions: *Readpackbit, Readpackbyte, Readword, etc.*
Write functions: *Writepackbit, Writepackbyte, Writeword, etc.*
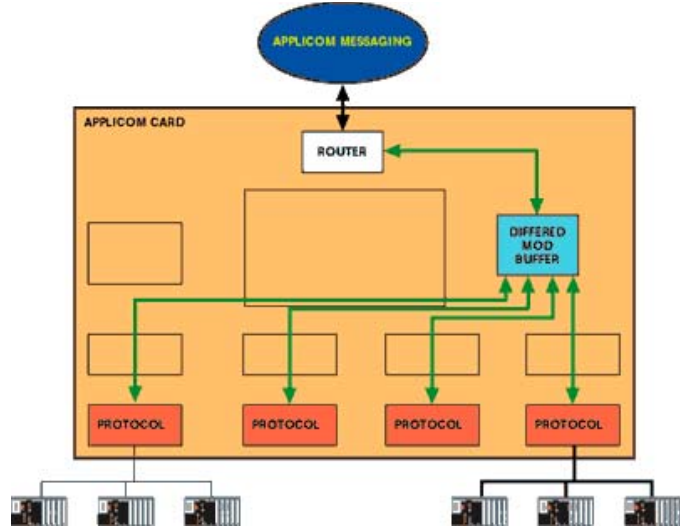

Woodhead
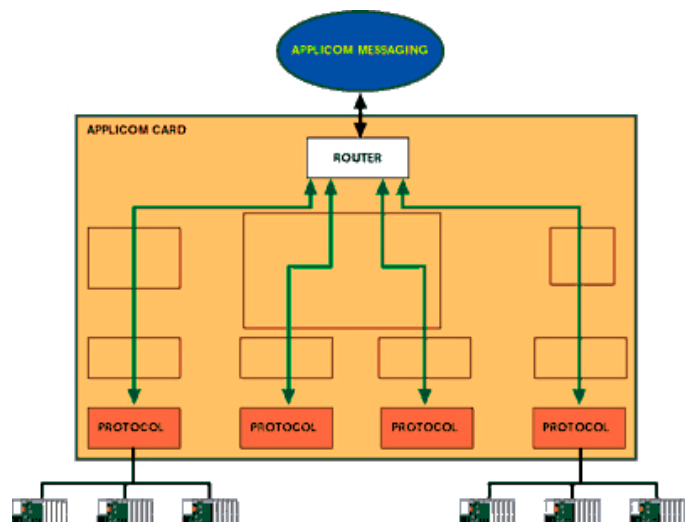
2. Differed mode

This mode is used to manage **asynchronous** exchange.

Advantage: not blocking the application program during data exchanges with the device.

The application program sends a request to the server that, after freeing the task from the application, resends this request to the remote device. The response from the device is therefore temporarily stored in a buffer on the card, until the application comes to empty this buffer. Each card can store a maximum of **64 deferred requests** being processed.

Data retrieval by the application is made using the **Transdif** function.

Read functions: *Readdifbit, Readdifbyte, Readdifword, etc.*
Write functions: *Writedifbit, Writedifbyte, Writedifword, etc.*

3. Cyclic mode

This mode defines data exchanges (in read/write) between the Woodhead internal Data-Base and the remote devices (PLC, sensor, activator...). The reading cyclic functions perform the reading and the data storage of the remote device into the Data-Base. The writing cyclic functions copy the data of the Data-Base in a buffer area of the remote device.

This mode increases your application performances: as the data-base gives you in real time the status of this buffer, the access speed to the data is significantly increased.
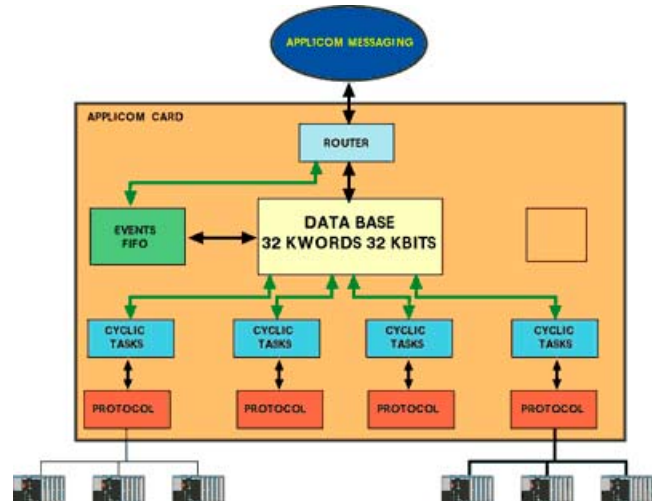
The cyclic mode could be implementing:
- by configuration using the configuration console
- dynamically by using dedicated functions of the API library

These cyclic functions can be permanent, activated or not by the application program. The frequency of the cyclic functions can be set up either in chained mode or in **multiples of 100ms**. The maximum number of cyclic functions is **255 per channel**. For each exchange, a communication status is accessible from the application. They also allow the user to make automatic data format modification (bit, byte, word, double word, floating). For example, the user can read words and store them in the bit table of the Data-Base.

For the *cyclic read* functions, the variables read periodically in the device are stored in the data base (**32 kbits/32 Kwords**), at addresses predefined when configuring the cyclic functions. For the *cyclic write* functions, variables located in the data base are written in the device targeted.

Access to variables in the device occurs using the functions *Transcyc* or *Transcycpack*. These functions are used to retrieve variables by calling the channel N and the cyclic function N.

Others library features

✓ **Data Base access**

The Data-Base may be accessed from the application program using API functions:

- Read functions: *Getbit, Getpackbit, Getword, Getpackword, etc.*
- Write functions: *Setbit, Setpackbit, Setword, Setpackword, etc.*

Using the FIFO events in Data Base - *Confdb*, *Getevent* - the status change of variables in the Data Base can be logged in an FIFO event (up to 4096 positions) using the *Confdb* function. These variables are listed and time stamped. The application accesses to this FIFO event by using the function *Getevent*.

✓ **Modem signal management**

Woodhead protocol also integrates modem signal management on Serial port of applicom® interface cards. Those functions - *Setmodem* and *Getmodem* – are used to access to the leased lines and to the switched telephone network (STN).

✓ **Redundancy implementation**

The library/DLL includes functions to manage interface cards redundancy by using the digital input and the digital output available on the card.

A digital output is available by using the *AppGetWatchDog* function. This digital output can be set as closed, opened or maintained closed for a defined period. In this last case, the contact closes as soon as the function is called and reopens after the time entered in parameter, unless the user program has called this function again before this time has elapsed. The *AppGetWatchDog* function also returns the current value of the digital input (0 or 1).
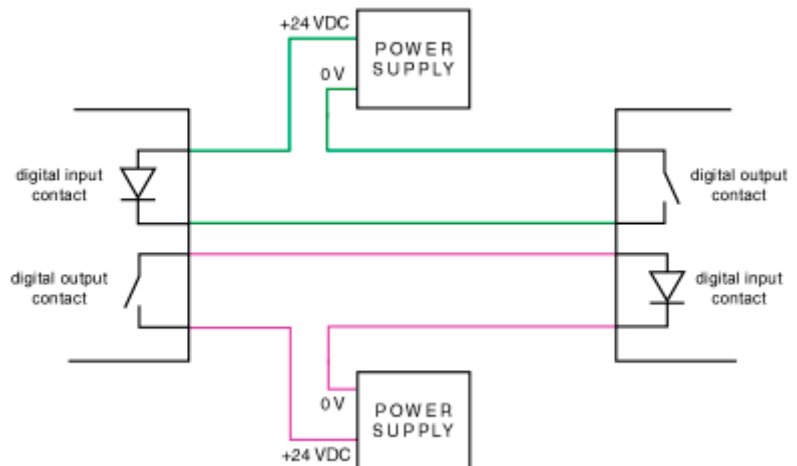
- Hot backup scenario between two interface cards -

BradCommunications™