



Summary

1. Brief description.....	2
2. Installation	2
3. DLL Usage.....	3
3.1 Usage from VB-VBA	4
3.2 Usage from C++	6
4. Input array description	7
5. Results array description	9
6. Price calculation	13
7. Error codes.....	14

1. Brief description

In order to select a CONDAIR humidifier, a DLL is available for use with any programming language that supports standard Windows DLLs. By entering the required input data, the user obtains the most suitable humidifier system from the DLL.

Please note that only a minimal check is performed on the validity of the values passed to DLL; this could lead to unpredictable results if wrong data are specified. It is left to the user to assure the data consistency.

2. Installation

The DLL is distributed as a zip file. Unzip the file in the working directory of your application.

The package consists of three files

File name	Description
CONDAIRDLL.DLL	DLL implementation file.
libxl.dll	
CONDAIR	DLL Database

Table 1: distribution files

DLLs are written using Microsoft Visual C++ 2017. User needs to distribute the VC++ 2017 runtime also. Runtime can be downloaded from ALTEC site:

ALTEC site:

www.altecsoftware.com and login as

username ALTEC

password ALTEC

download and install VCRedist2015-x86.exe

DLLs are compiled as ANSI.

Be sure to check if new updates are available for VC runtime on Microsoft web site.

Inside zip file, a test program called Condairstest.exe is included to verify the correct usage of DLL.

3. DLL Usage

CONDAIRDLL.dll exports three main functions:

CondairSteam (double aInputData[INP_DATA_STEAM])

GetRisCondairSteam (variant[RIS_DATA_STEAM])

GetRisCondairESCO (Variant [RIS_DATA_STEAM])

INP_DATA_STEAM: 30

RIS_DATA_STEAM: 100

The selection method is the same for both type of humidifier(Steam or ESCO), while the method to get results varies. The first position of aInputData [INP_DATA_STEAM] _ indicates the type of humidifier to calculate (see section 4).

To avoid DLL location problems, check that CONDAIRDLL.DLL, and all support files, are placed in the same directory of your executable file.

The DLL also exports two additional methods for easier usage from .NET languages

GetRisCondairSteam1 (variant[RIS_DATA_STEAM], short pos)

GetRisCondairESCO1 (Variant [RIS_DATA_STEAM], short pos)

where pos is the result index that the caller wants to retrieve data for. See TestCondair example in vb.NET for further into

3.1 Usage from VB-VBA

Function declaration

```
Declare Function CondairSteam Lib "C:\\CONDAIR\\CondairDLL.dll" (ByRef p1 As Double) As Long
Declare Function GetRisCondairSteam Lib "C:\\CONDAIR\\CondairDLL.dll" (ByRef p1 As VariantType) As Long
Declare Function GetRisCondairESCO Lib "C:\\CONDAIR\\CondairDLL.dll" (ByRef p1 As VariantType) As Long
```

```
Const INP_DATA_STEAM = 30
Const RIS_DATA_STEAM = 100
```

where "C:\\CONDAIR\\CondairDLL.dll" is the absolute path of CONDAIRDLL.DLL

Usage

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim aInputData(INP_DATA_STEAM) As Double
    Dim aResult(RIS_DATA_STEAM) As VariantType
    Dim ris As Long

    aInputData(0) = 0
    aInputData(1) = 700
    aInputData(2) = 500
    aInputData(3) = 600
    aInputData(4) = 1.013
    aInputData(5) = 5000
    aInputData(6) = 0
    aInputData(7) = 20
    aInputData(8) = 30
    aInputData(9) = 20
    aInputData(10) = 50
    aInputData(11) = 10
    aInputData(12) = 0
    aInputData(13) = 0
    aInputData(14) = 0
    aInputData(15) = 0
    aInputData(16) = 0
    aInputData(17) = 4
    aInputData(18) = 0
    aInputData(19) = 0
    aInputData(20) = 0
    aInputData(21) = 0
    aInputData(22) = 0
    aInputData(23) = 0

    ris = CondairSteam(aInputData(0))
    If (ris < 0) Then
        MessageBox.Show("Error number" + ris)
    Exit Sub
```

```
End If
aResult(0) = 0 ' Retrieve first results
GetRisCondairSteam(aResult(0))

ShowResult(aResult(0))

End Sub
```

3.2 Usage from C++

```
#define INP_DATA_STEAM          30
#define RIS_DATA_STEAM         100

extern "C" LONG FAR WINAPI EXPORT CondairSteam(double vInp[INP_DATA_STEAM]);
extern "C" LONG EXPORT GetRisCondairSteam(VARIANT vRis[RIS_DATA_STEAM]);

void CCondairView::OnCalcolo()
{
    double aInp[INP_DATA_STEAM];
    VARIANT aRis[RIS_DATA_STEAM];

    // Collect data from input mask
    GetData(vInp);

    // Check for errors
    long nRis = CondairSteam (aInp);
    if (nRis < 0)
    {
        TCHAR p[20];
        _stprintf(p, _T("Error number %d"), nRis);
        AfxMessageBox(p);
        return;
    }
    aRis[0].iVal = 0
    GetRisCondairSteam(aRis);

    // Show results
    ShowResults(aRis);
}
```

Use the standard LoadLibrary method to get a pointer to the methods exported from the DLL

Warning: positions 0,1,5,6,7,13,16,19,22,31,32,33,34,39,40,41,42,47,48,49,50,54,55,56 and 57 of the output array contains BSTR values, so in order to convert them to CString (in Visual C++) is possible to use the class CString passing *aRis[n].bstrVal* to the constructor (see VARIANT structure declaration):

N.B.: All BSTR values are allocated by the DLL and must be freed by the caller

```
CString humTechnology(aRis[1].bstrVal);
AfxMessageBox(CString("Humidifier technology: ") + humTechnology);
```

Note: Arrays in C/C++ are 0-based (first position is position number 0).

4. Input array description

Each position of the input array specified in the parameter on CondairSteam function contains value related to a variable used by the software according to the following table:

Pos.	Meaning	Acceptable values
0	Humidifier serie	0: EC 1: EL 2: RS 3: GS 4: ESCO
1	Available width of duct	mm
2	Available height of duct	mm
3	Length of duct	mm
4	Atmospheric pressure relative to the altitude above the sea level	bar
5	Air volume flow	m ³ /h
6	Voltage	0: 400V/3 1: 230V/3 2: 230V/1
7	Inlet air temperature	°C
8	Inlet air humidity	%
9	Inlet air humidity	°C
10	Outlet air humidity	%
11	Humidifier load	Kg/h
12	Regulation	Valid only for RS series 0: Normal regulation 1: Fine regulation
13	RFI card accessories	0: No – 1: Yes
14	Aux. supply accessories	0: No – 1: Yes
15	Mounting rail	0: No – 1: Yes
16	Water drain cooling	0: No – 1: Yes
17	Steam hoses length	Meters
18	Number of “roll” steam tube	Pcs (taken into consideration if > 0)
19	Number of “roll” condensing tube	Pcs (taken into consideration if > 0)

Pos.	Meaning	Accettable values
20	Distributor type	Available only with EC/EL/RS/GS series. 0: DV distributor 1: Optisorp distributor
21	Material type	Available only with ESCO valve 0: Cast iron 1: INOX
22	Actuator type	Available only with ESCO valve 0: Electric 1: Pneumatic
23	Calculation type	Available only with ESCO valve 0: Automatic 1: One nozzle 2: Two nozzles 3: Three nozzles

Table 2. Input array description.

5. Results array description

GetRisCondairSteam is the function that return result when a EC/EL/RS/GS humidifier series is request. Function has a parameter that contains results of the selection based on values specified on input array.

The position 0 in the array must contains the index of the results the user wants to retrieve values for before the call.

Position 0 will be replaced with value described in the following table after the call

Meaning of each position of array is given in Table 3.

Pos.	Meaning	Unit measure
0	Series	
1	Technology	
2	Model	
3	Sprayed water flow	kg/h
4	Qty of steam producers	
5	Voltage	
6	Absorption (<u>in case of more than one steam producers, contains a string with values of each steam producers</u>)	Kw
7	Rated current (<u>in case of more than one steam producers, contains a string with values of each steam producers</u>)	A
8	Width (<u>in case of more than one steam producers, width shall be multiplied by quantities of steam producer</u>)	mm
9	Height	mm
10	Depth	mm
11	Weight	Kg
12	Humidifier price	€
13	RFI card description	
14	RFI qty	

Pos.	Meaning	Unit measure
15	RFI total price	€
16	Aux. supply description	
17	Aux. supply qty	
18	Aux. supply total price	€
19	Mounting rail description	
20	Mounting rail qty	
21	Mounting rail total price	€
22	Drain water cooling description	
23	Drain water cooling qty	
24	Drain water cooling total price	€
25	Distributor number	
26	Distributor type	
27	Distributor pipes length	mm
28	Distributor supply	
29	Distributor flanges	
30	Distributor total price	€
31	Steam tube code	
32	Steam tube material	
33	Steam tube um	
34	Steam tube diameter Ext/Int	
35	Steam tube min	
36	Steam tube max	
37	Steam tube step	
38	Steam tube total price	€
39	Condensing tube code	
40	Condensing tube material	
41	Condensing tube um	
42	Condensing tube diameter Ext/Int	
43	Condensing tube min	

Pos.	Meaning	Unit measure
44	Condensing tube max	
45	Condensing tube step	
46	Condensing tube total price	€
47	“Roll” Steam tube code (1)	
48	“Roll” Steam tube material (1)	
49	“Roll” Steam tube um (1)	
50	“Roll” Steam tube diameter Ext/Int (1)	
51	“Roll” Steam tube min (1)	
52	“Roll” Steam tube max (1)	
53	“Roll” Steam tube total price (1)	€
54	“Roll” Condensing tube code (2)	
55	“Roll” Condensing tube material (2)	
56	“Roll” Condensing tube um (2)	
57	“Roll” Condensing tube diameter Ext/Int (2)	
58	“Roll” Condensing tube min (2)	
59	“Roll” Condensing tube max (2)	
60	“Roll” Condensing tube total price (2)	€
61	Total price of solution	€

Table 3. Output array description of GetRisCondairSteam function.

(1) return a value if vInp[18] > 0

(2) return a value if vInp[19] > 0

GetRisCondairESCO is the function that return result when an ESCO valve is request.

Function has a parameter that contains results of the selection based on values specified on input array.

The position 0 in the array must contains the index of the results the user wants to retrieve values for before the call.

Position 0 will be replaced with value described in the following table after the call

Meaning of each position of array is given in Table 4.

Pos.	Meaning	Unit measure
0	Valve series	
1	Model description	
2	Model code	
3	Material description	
4	Material code	
5	Sprayed water flow	kg/h
6	Actuator type	
7	Actuator code	
8	Actuator model	
9	Price (3)	€
10	Distributor series	
11	Distributor model	
12	Distributor material	
13	Distributor connection	mm
14	Distributor nozzle number	
15	Distributor flow	kg/h
16	Distributor price (3)	€
17	Fittings series	
18	Fittings model	
19	Fittings material	
20	Fittings connection	

Pos.	Meaning	Unit measure
21	Fittings flow	kg/h
22	Fittings interaxis	
23	Fittings height	mm
24	Fittings price (3)	€
25	Absorption length IMPORTANT – This value MUST be verified with CONDAIR.	mm
26	Total price of solution (3)	€

Table 4. Output array description of GetRisCondairESCO function.

(3) Prices provided are official list price without any customer discount.

6. Price calculation

DLL returns humidifier, distributor and all other component prices. If available, and selected, also accessories prices are provided.

Prices are expressed without any customer's discount.

7. Error codes

Error code	Error description
0	No error
-1	Selected voltage not valid for series
-2	Regulation type not valid for selected series
-3	Width or height of duct less than zero
-4	Temperature of humidity out of range
-5	Input temperature less than zero or grater that 100°C
-6	Computed air speed less than zero or air flow less than zero
-7	Distributor not found
-8	Database file not found
-9/-10/-11	General problem reading data from configuration file
-12	Steam tube length less than zero
-13	Steam tube not found (check maximum meters in input)
-14	Condensing tube not found
-15	Fittings not found
-16	“Roll” steam tube not found (check maximum meters in input)
-17	“Roll” condensing tube not found

Table 4: Error codes.