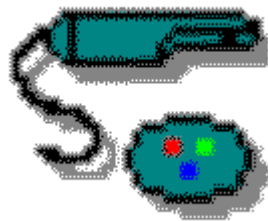


SIRONA SIUCOM



SOFTWARE DEVELOPER'S KIT

Version 3.4

Status: Released

Document history

| Version | Issue of | Status Draft/Released | Author | Dept. | Reason for revision |
|---------|------------|--------------------------|----------|-------|---|
| 3.2.2 | 29.08.2001 | Draft | Reinke | RSWE | Draft based on SIUCOM 3.2.2 |
| 3.3.0 | 05.12.2001 | Draft | Sonderm. | GEE 4 | Device.cfg: Description of control options available on dental devices. Added online help removed. Update to SIUCOM 3.3 |
| 3.3.1 | 17.12.2001 | Draft | Reinke | RSWE | + Import file formats, update of screenshots |
| 3.3.2 | 11.02.2002 | Draft | Schrön | GEE 4 | New document structure, supplements to import functions, correction of function codes in Protocol 4 |
| 3.4 | 22.02.2002 | Released | Schrön | GEE 4 | |
| | | | | | |
| | | | | | |

Contents

| | |
|--|-----------|
| History..... | 2 |
| Contents..... | 3 |
| Document history..... | 2 |
| 1 Introduction | 5 |
| 1.1 Purpose of the document | 5 |
| 1.2 Definitions and abbreviations | 5 |
| 1.3 Reference documents..... | 5 |
| 2 SIUCOM system overview | 5 |
| 2.1 Product description | 5 |
| 2.2 System structure..... | 6 |
| 3 SDK setup..... | 8 |
| 3.1.1 Installation..... | 8 |
| 3.1.2 Demo and test mode | 8 |
| 4 Integration of applications | 12 |
| 4.1 Development environment | 12 |
| 4.2 Response to SIUCOM actions | 12 |
| 4.2.1 ACTIVATE..... | 12 |
| 4.2.2 STOP..... | 12 |
| 4.2.3 TOGGLE | 12 |
| 4.2.4 MSG | 12 |
| 4.2.5 Suppression of multiple starts..... | 12 |
| 4.2.6 Definition of window class | 12 |
| 4.3 Interfacing of Windows programs (WinSample.exe)..... | 13 |
| 4.3.1 Overview..... | 13 |
| 4.3.2 Suppression of multiple starts..... | 13 |
| 4.3.3 Defining the window class name | 13 |
| 4.4 Interfacing of Windows dialogs (DlgSample.exe)..... | 15 |
| 4.4.1 Overview..... | 15 |
| 4.4.2 Suppression of multiple starts..... | 15 |
| 4.4.3 Defining the window class name | 15 |
| 5 Support | 17 |
| 6 Appendix..... | 18 |

| | | |
|------------|-----------------------------------|-----------|
| 6.1 | SIUCOM configuration | 18 |
| 6.1.1 | SIUCOM.INI..... | 18 |
| 6.1.2 | Messages.cfg | 18 |
| 6.1.3 | <DEVICE>.cfg..... | 19 |
| 6.1.4 | Import file formats | 26 |
| 6.1.5 | SIUCOM help..... | 27 |

Illustrations

| | | |
|----------|--|----|
| Figure 1 | SIUCOM components and interactions | 6 |
| Figure 2 | SIUCOM tray icon..... | 8 |
| Figure 3 | SIUCOM pop-up menu | 9 |
| Figure 4 | SIUCOM configuration dialog | 9 |
| Figure 5 | WINSAMPLE application window..... | 10 |
| Figure 6 | SIUCOM command configuration..... | 11 |
| Figure 7 | WinSample application | 13 |
| Figure 8 | WinSample application | 15 |
| Figure 9 | SIUCOM Import dialog | 26 |

1 Introduction

1.1 Purpose of the document

This document provides a functional overview of the Siucom interface software and describes the guidelines for integrating Windows programs in a SIRONA SIUCOM installation. All of the aspects relevant for integrating any Windows target application in the SIUCOM communication scenario are presented here based on the examples of two applications.

1.2 Definitions and abbreviations

| | |
|-----------------------------|---|
| CJ | Chair junction (chair connection box PCB) |
| PCB | Printed Circuit Board |
| HW | Hardware |
| Sirocam | Sirona camera system |
| SW | Software |
| 3 rd party appl. | Any 32-bit Windows application |
| Target application | To be integrated in the SIUCOM communication scenario |
| MFC | Microsoft Foundation Classes |

1.3 Reference documents

-

2 SIUCOM system overview

2.1 Product description

The PC software for SIVISION makes it possible to control PC programs from Sirona dental treatment centers in conjunction with the SIVISION option with PC connection.

The PC functions can be operated via the user interface of the treatment center (function keys) or the unit foot switch.

This is achieved through communication of the dental unit with the PC via the serial port.

The function codes generated in the dental unit for certain control actions are assigned to the desired PC functions in the SIUCOM program. Third-party applications can thus be started, terminated and controlled via WM_COMMAND messages.

2.2 System structure

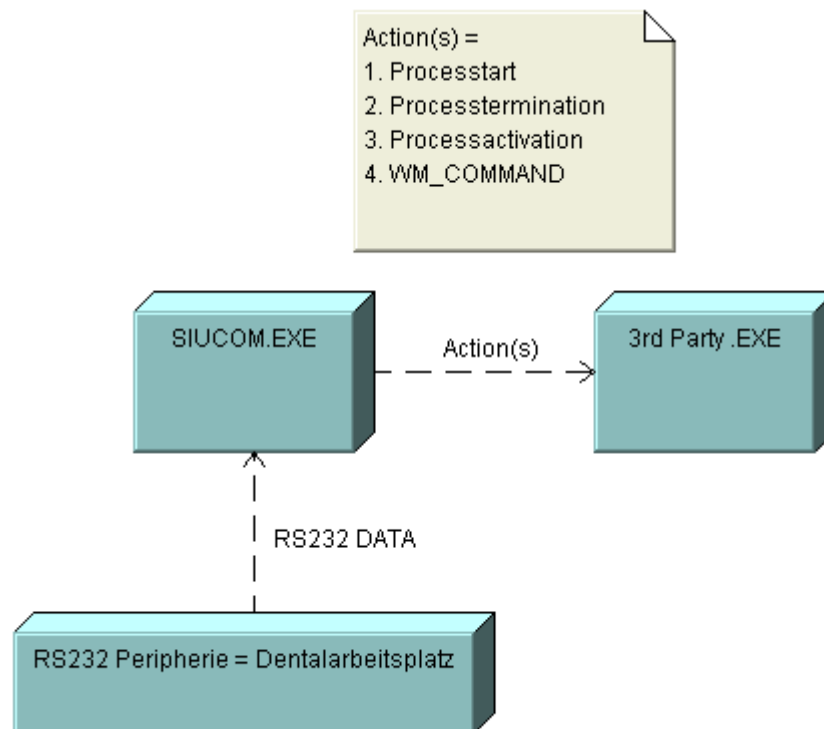


Figure 1 SIUCOM components and interactions

The RS232 interface in the treatment center is controlled in the connection box, e.g. on PCB CJ of the C2+. The RS232 interface at the PC end manages the siucom.exe program. The commands received there are interpreted and forwarded onto any desired third-party applications. The flexible and open-ended assignment of RS232 commands to PC applications is performed in the siucom.ini configuration file.

One of the following actions can be defined within SIUCOM for each input event (serial event):

| Type of action | Description | Remarks |
|----------------|--|--|
| ACTIVATE | Starts any Windows application | The target application is configured by indicating the path and file name |
| STOP | Terminates a running Windows application | Identification occurs via the window class configured for the target application |
| TOGGLE | If the application concerned is running in the foreground, it will be terminated. If the application is running in the background or not running at all, it will be brought to the foreground or started. | |
| MSG | Sends a definable WM_COMMAND message to the configured target application | |

The functional scope of the Siucom interface thus makes it possible to open and close any Windows applications or to switch between individual applications. In addition, individual functions of Windows applications can be controlled. Possible applications include e.g. the following control functions for video applications: Open or close video overlay when removing or depositing camera, foot-switch controlled change-over between live and still image, Save, Full Frame, Zoom In, Zoom Out, Tile, Select Image, Rotate Image, Copy and Paste Image, Print, etc.

The functionality of the Siucom interface can be configured according to the customer's requirements in a setting dialog.

Starting with Version 3.3, the control of 3rd party software is supported in Siucom by two import functions:

1. Message import, i.e. functions from 3rd party software are imported into the message.cfg (message picklist). These functions must be made available by the 3rd party software in the appropriate format, i.e. in an *.msd file.
2. Configuration import, i.e. functions with previously assigned events are copied to siucom.ini (the Siucom configuration file). These configuration sets also must be made available by the 3rd party software in the prescribed format, i.e. in an *.ind file.

Furthermore, it is of course also possible to directly write into the message.cfg or siucom.ini file when installing 3rd party software. Ideally, both of these options should be supported by the 3rd party software, since they are required by the user depending on the order of installation.

3 SDK setup

A Siucom SDK is available for demonstrating the functionality of Siucom or supporting tests of Siucom's ability to control one's own Windows applications. The Siucom SDK contains the Siucom application along with the functions for controlling 3rd party software including test possibilities. Serial communication with dental treatment centers is not possible.

3.1.1 Installation

For installation, start the SETUP.EXE program on the installation storage medium. The Setup program then installs the following components:

- SIUCOM TESTSYSTEM: All of the programs and configuration files required for the demo mode.
- WINSAMPLE: Source coding for the standard Windows sample
- DLGSAMPLE: Source coding for the dialog sample

You can also select individual components in order to carry out a selective installation. The default target folder is PROGRAM FILES\SIUCOM SDK, where the following subfolders and files will be created:

| | |
|-----------------------|--|
| \SYSTEM | System folder for the test system |
| \SYSTEM\SIUCOM.INI | SIUCOM configuration file |
| \SYSTEM\SIUCOM.EXE | The SIUCOM application |
| \SYSTEM\SIUCOM.HLP | The SIUCOM online help function |
| \SYSTEM\MESSAGES.CFG | Message configuration file (actions) |
| \SYSTEM\WINSAMPLE.EXE | Windows sample application |
| \SYSTEM\WINSAMPLE.CFG | Windows sample configuration file (events) |
| \SYSTEM\DLGSAMPLE.EXE | Sample dialog program |
| \SYSTEM\DLGSAMPLE.CFG | Dialog sample configuration file (events) |
| \WINSAMPLE | Sample source codes |
| \DLGSAMPLE | Sample source codes |

As soon as the installation has been completed, the SIUCOM.EXE can be started immediately for demonstration purposes. You can start the SIUCOM program either directly from the Setup program or via the Windows Start menu.

3.1.2 Demo and test mode

The following steps provide a functional overview of the SIUCOM SDK:

3.1.2.1 Starting SIUCOM

- Select the entry SIUCOM SDK->SIUCOM from the Start menu. SIUCOM.EXE is then started and appears as an icon in the system tray:



Figure 2 SIUCOM tray icon

- Right-click the SIUCOM tray icon and select SETTINGS from the pop-up menu which then appears:

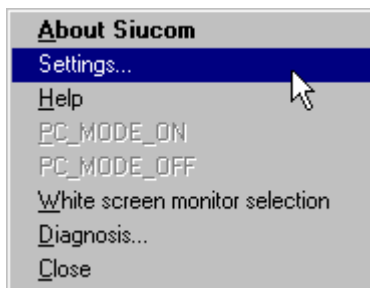


Figure 3 SIUCOM pop-up menu

- The SIUCOM setting dialog with the preset WINSAMPLE configuration appears:

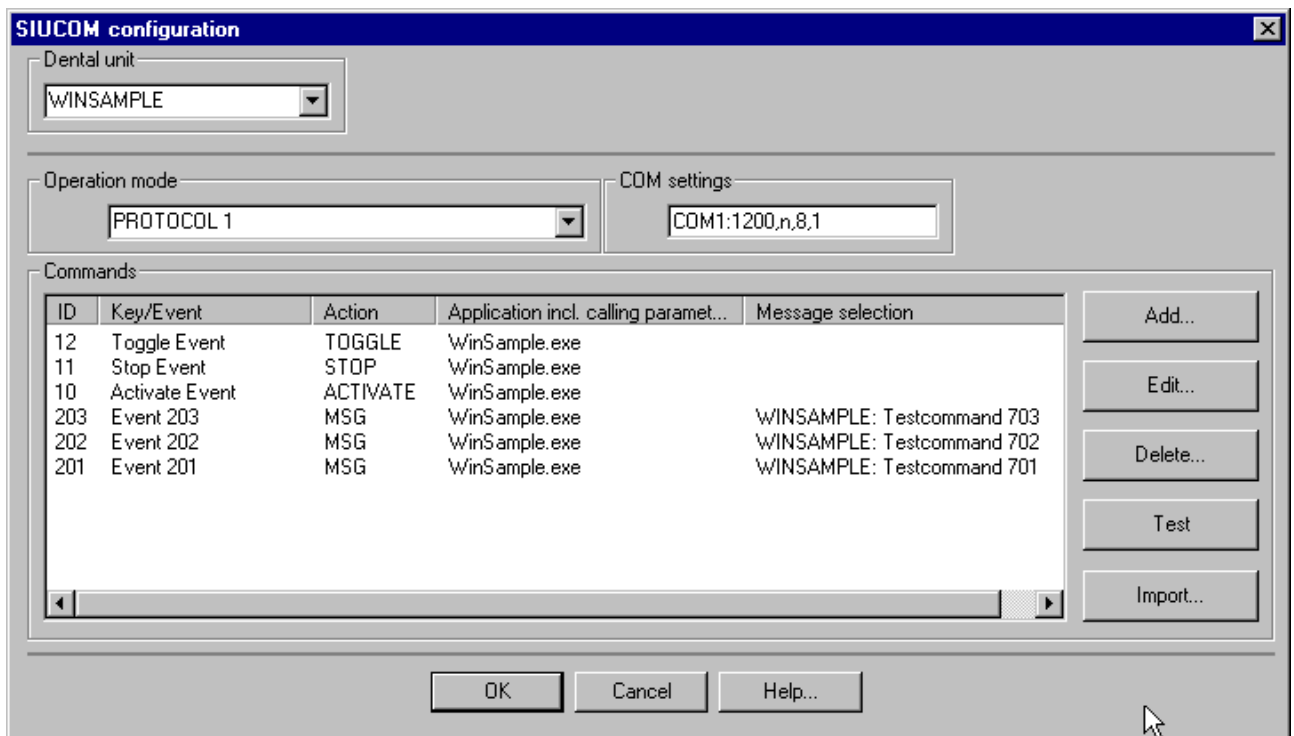


Figure 4 SIUCOM configuration dialog

3.1.2.2 Activating the WinSample program

- Select ACTIVATE from the command list and press the TEST button. The WINSAMPLE sample program appears.

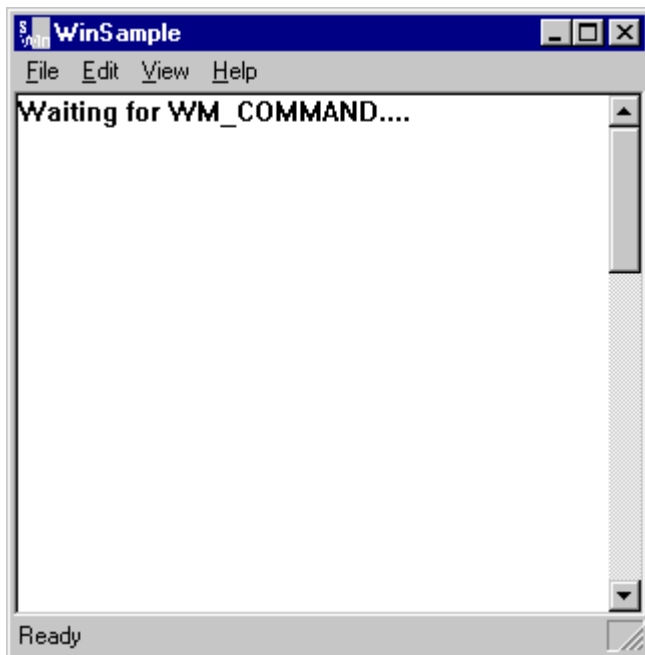
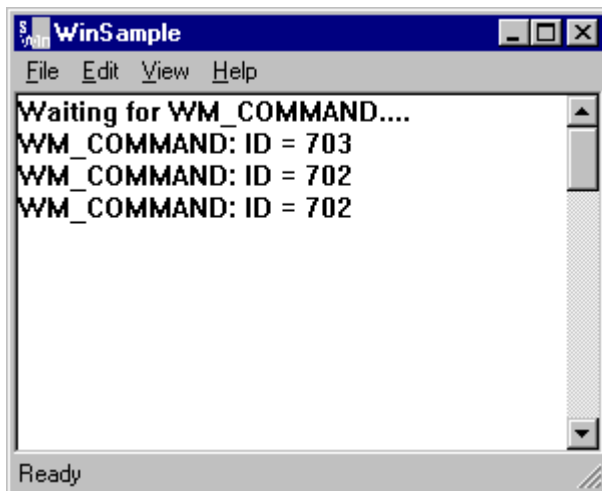


Figure 5 WINSAMPLE application window

3.1.2.3 Sending messages

- Select any line with an action of the MSG type from the SIUCOM command list and press the TEST button again. Repeat this procedure as often as necessary.



3.1.2.4 Adding new actions

You want to switch on and off the WINSAMPLE status bar from SIUCOM:

- Select the ADD button in the SIUCOM configuration dialog.
- The command configuration dialog opens.
- Select the following under KEY/EVENT: *New Event 204*
- Select the following under Action: MSG
- Select the following under Message selection: *WINSAMPLE: Statusbar ON/OFF*
- Confirm the new configuration with OK.

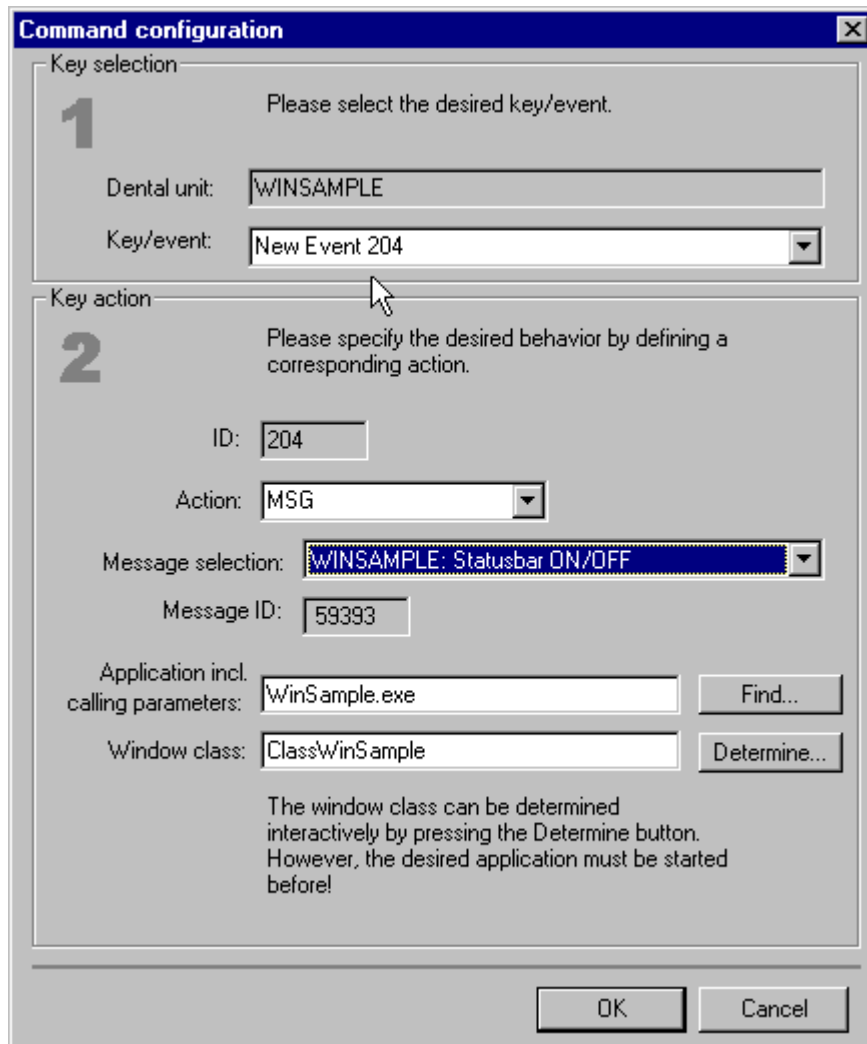
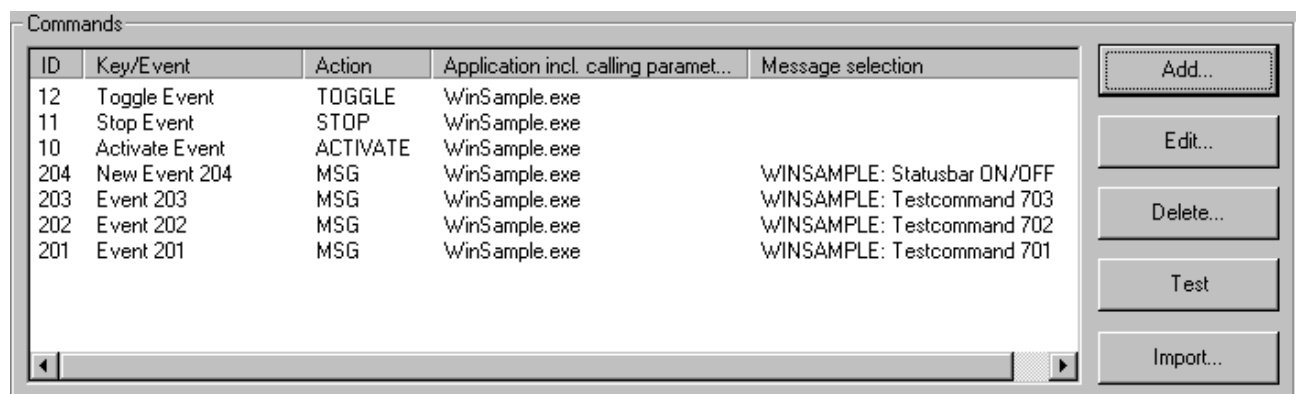


Figure 6 SIUCOM command configuration

- A new entry appears on the SIUCOM command list:



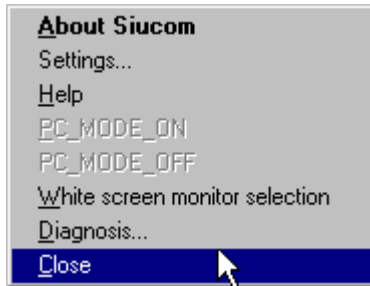
- Select this entry and press the TEST button repeatedly. The WINSAMPLE status bar is switched on and off accordingly.

3.1.2.5 Terminating the WinSample program

- Select STOP from the SIUCOM command list and press the TEST button. The WINSAMPLE program is terminated.

3.1.2.6 Terminating SIUCOM

- Close the SIUCOM setting dialog by clicking OK or CANCEL
- Select CLOSE from the SIUCOM pop-up menu.



4 Integration of applications

4.1 Development environment

The following sample source codes were developed using Microsoft Visual C++ Version 6. In principle, all Windows program development environments can be used.

4.2 Response to SIUCOM actions

All potential interactions of SIUCOM.EXE with target applications are described in the following.

4.2.1 ACTIVATE

SIUCOM.EXE tries to execute a process start for the defined target application. At this point, the target application only has to suppress a multiple start (see below).

4.2.2 STOP

SIUCOM.EXE tries to terminate the process of the target application. The latter should not require any interactive actions during process termination. Otherwise you would have to leave your operating environment (dental treatment center) and possibly perform the interactive steps from the PC.

4.2.3 TOGGLE

No special activities required.

4.2.4 MSG

The WM_COMMAND handler of the target application must respond to correspondingly defined SIUCOM MSG actions.

4.2.5 Suppression of multiple starts

Multiple instancing of a target application should be prevented during program start within the application, since SIUCOM cannot clearly differentiate between multiple instances.

Please refer to the two sample projects for a concrete solution.

4.2.6 Definition of window class

While searching for a target application, SIUCOM runs a window search based on the known window class name.

If the search for a class name is not successful, SIUCOM initiates a second attempt based on the window caption. In this way it is easy e.g. to integrate Windows dialogs in the SIUCOM communication scenario.

Please refer to the two sample projects for concrete solutions.

4.3 Interfacing of Windows programs (WinSample.exe)

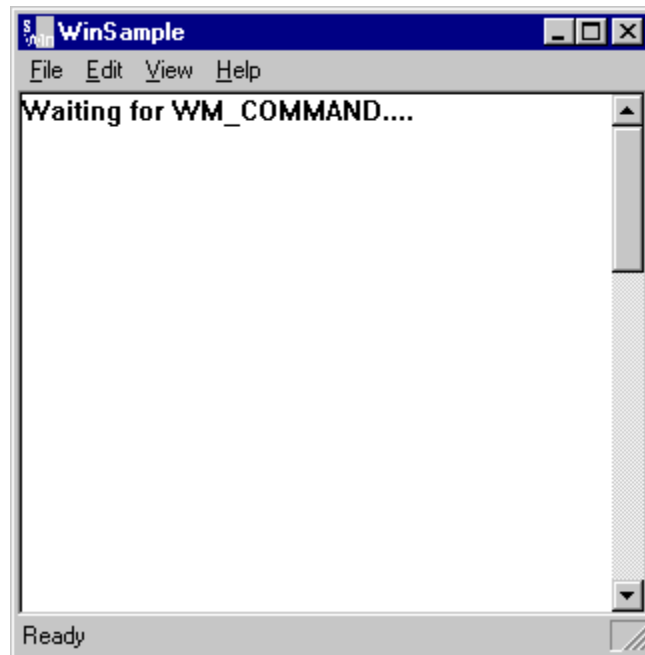


Figure 7 WinSample application

4.3.1 Overview

WinSample is a WIN32 application featuring a CScrollView view class. All incoming WM_COMMAND messages are displayed together with the current message ID in the ScrollView.

4.3.2 Suppression of multiple starts

This check takes place in CWinSampleApp::InitInstance() directly during application start. If the application is already running, the old application will be moved to the foreground and the renewed program start will be suppressed.

```
BOOL CWinSampleApp::InitInstance()
{
    // Standard initialization
    // If you are not using these features and wish to reduce the size
    // of your final executable, you should remove from the following
    // the specific initialization routines you do not need.

    // If a previous instance of the application is already running,
    // then activate it and return FALSE from InitInstance to
    // end the execution of this instance.

    // Determine if another window with our class name exists...
    CWnd *pWndPrev;
    pWndPrev = CWnd::FindWindow(_T(WINSAMPLE_CLASS), NULL);
    if (pWndPrev)
    {
        // Yes, there is an instance running, bring it to front and exit.
        pWndPrev->SetForegroundWindow();
        return FALSE;
    }
}
```

4.3.3 Defining the window class name

The application-specific definition of the window class name is performed in 2 steps under MFC applications:

1. Registration of new class name in CWinSampleApp::InitInstance()
2. Activation of class in CMainFrame::PreCreateWindow()

The window class is defined as follows in the example:

```
#define WINSAMPLE_CLASS "ClassWinSample"
```

The corresponding command configurations must be made under this class name within SIUCOM.

```
CWinSampleApp::InitInstance()
...
// Register your unique class name that you wish to use
WNDCLASS wndcls;

memset(&wndcls, 0, sizeof(WNDCLASS)); // start with NULL
                                       // defaults

wndcls.style = CS_DBLCLKS | CS_HREDRAW | CS_VREDRAW;
wndcls.lpfnWndProc = ::DefWindowProc;
wndcls.hInstance = AfxGetInstanceHandle();
wndcls.hIcon = LoadIcon(IDR_MAINFRAME); // or load a different
                                       // icon
wndcls.hCursor = LoadCursor( IDC_ARROW );
wndcls.hbrBackground = (HBRUSH) (COLOR_WINDOW + 1);
wndcls.lpszMenuName = NULL;

// Specify your own class name for using FindWindow later
wndcls.lpszClassName = _T(WINSAMPLE_CLASS);

// Register the new class and exit if it fails
if(!AfxRegisterClass(&wndcls))
{
    TRACE("Class Registration Failed\n");
    return FALSE;
}
bClassRegisteredG1 = TRUE;
...

////////////////////////////////////
//
// MFC PreCreateWindow
//
// Return      : MFC standard
//
// Remarks: In CSiucomApp::InitInstance() we prepared a window subclassing.
//          Use new classname WINSAMPLE_CLASS = "ClassWinSample" now
//
////////////////////////////////////
BOOL CMainFrame::PreCreateWindow(CREATESTRUCT& cs)
{
    if( !CFrameWnd::PreCreateWindow(cs) )
        return FALSE;

    cs.dwExStyle &= ~WS_EX_CLIENTEDGE;
    cs.lpszClass = _T(WINSAMPLE_CLASS);

    return TRUE;
}
```

4.4 Interfacing of Windows dialogs (DlgSample.exe)

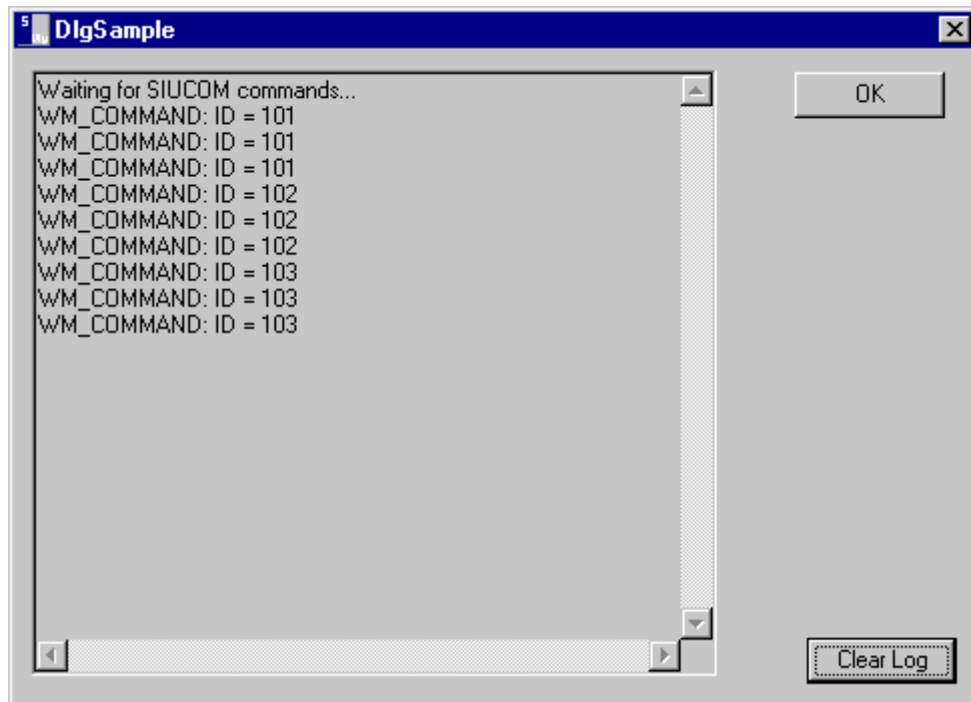


Figure 8 DlgSample application

4.4.1 Overview

DlgSample shows how to use an MFC-based dialog program in connection with SIUCOM. Incoming WM_COMMAND messages are output in a dialog-internal edit box.

Decisive difference from a window application:

The dialog window class name is constant, i.e. it cannot be used for unique identification of a target application. For this reason, SIUCOM also uses the window caption (Captiontext) to identify the target application!

4.4.2 Suppression of multiple starts

Basically, the procedure is the same as for a window-based Windows program. During the initialization of the application class, a search is made for an identical instance (via caption text) and the latter is moved to the foreground, if found.

```
BOOL CDlgSampleApp::InitInstance()
{
    ...
    CWnd *pWndPrev;
    CString strCaption;
    strCaption.LoadString(IDS_DLGCAPTION);

    // we are looking for a running dlg instance containing the same
    // captiontext !
    pWndPrev = CWnd::FindWindow(NULL, (LPCSTR) strCaption);
    if (pWndPrev)
    {
        pWndPrev->SetForegroundWindow();
        return FALSE;
    }
    ...
}
```

4.4.3 Defining the window class name

As already mentioned before, it is not possible to identify dialogs by their window class names. As an alternative, they are assigned using the static dialog window caption texts.

The SIUCOM command configuration allows you to enter a target dialog window caption text in the window class text field.

```
BOOL CDlgSampleDlg::OnInitDialog()
{
    ...
    // Set Captiontext if not already defined in the global project
    // settings. This is a good place if You want to distinguish
    // between several dialogwindows with different captions. That's
    // the way how SCREEN.EXE handles the white and black image cases
    CString strTemp;
    strTemp.LoadString(IDS_DLGCAPTION);
    SetWindowText((LPCSTR) strTemp);
    ...
}
```


5 Support

Integration support is available under the address Siucom@Sirona.de.

6 Appendix

6.1 SIUCOM configuration

6.1.1 SIUCOM.INI

```
' (Version 1.0b)
[Configuration]
.....
' valid device configuration
' a configuration file <Device>.cfg must be available for this purpose !
' the detailed configuration can be found in the corresponding section [<Device>]
.....
Device=WINSAMPLE

[DLGSAMPLE]
.....
' the following entries are made for each device configuration:
'
' Protocol = 1|2|3|4                = selected RS232 application protocol
'
' COM      = <COMn:baudrate,parity,databits,stopbits> ' n = COM Port 1|2|3|4
'
' a list of the trigger/action database with the following format then follows:
' <RS232 event> = <action>, <application name>, <window class>, <event>, opt.<Msg. ID>,
' opt. <message description>
'
'          <action>                = serial input event, a selection from <device>.cfg
'          <application name>      = name of target application to be controlled
'          <window class>          = name of window class to be controlled
'          <event>                 = text description of input event, also from <Device>.cfg
'
' and in case <action> == MSG:
'          <Msg. ID>                = message ID to be sent to a target application via WM_COMMAND
'          <message description>    = text description to be sent to a target application via WM_COMMAND
'                                   Both entries are retrieved from Messages.cfg and must be
'                                   configured there accordingly.
'
' .....
Protocol=1
COM=COM1:1200,n,8,1
10=ACTIVATE,DlgSample.exe,DlgSample,Activate Event
11=STOP,DlgSample.exe,DlgSample,Stop Event
12=TOGGLE,DlgSample,DlgSample,Toggle Event
101=MSG,DlgSample.exe,DlgSample,Event 101,701,DLGSAMPLE: Testcommand 701
102=MSG,DlgSample.exe,DlgSample,Event 102,702,DLGSAMPLE: Testcommand 702
103=MSG,DlgSample.exe,DlgSample,Event 103,703,DLGSAMPLE: Testcommand 703

[WINSAMPLE]
Protocol=1
COM=COM1:1200,n,8,1
10=ACTIVATE,WinSample.exe,ClassWinSample,Activate Event
11=STOP,WinSample.exe,ClassWinSample,Stop Event
12=TOGGLE,WinSample.exe,ClassWinSample,Toggle Event
201=MSG,WinSample.exe,ClassWinSample,Event 201,701,WINSAMPLE: Testcommand 701
202=MSG,WinSample.exe,ClassWinSample,Event 202,702,WINSAMPLE: Testcommand 702
203=MSG,WinSample.exe,ClassWinSample,Event 203,703,WINSAMPLE: Testcommand 703
```

6.1.2 Messages.cfg

Contains all selectable messages for the configuration case ACTION == MSG in the following format:

1st line: Identification/comment

Then one line for each selection:

<target application>,<window class>,<message description>,<messages = Msg. ID>

Example:

```
SIRONA SIUCOM MESSAGE CONFIGURATION (Version 0.2)
DlgSample.exe,DlgSample,DLGSAMPLE: Testcommand 701, 701
DlgSample.exe,DlgSample,DLGSAMPLE: Testcommand 702, 702
DlgSample.exe,DlgSample,DLGSAMPLE: Testcommand 703, 703
WinSample.exe,ClassWinSample,WINSAMPLE: Testcommand 701, 701
WinSample.exe,ClassWinSample,WINSAMPLE: Testcommand 702, 702
WinSample.exe,ClassWinSample,WINSAMPLE: Testcommand 703, 703
WinSample.exe,ClassWinSample,WINSAMPLE: Statusbar ON/OFF, 59393
```

6.1.3 <DEVICE>.cfg

Contains all selectable input events in the following format:

1st line: Identification/comment

Then one line for each selection:

<RS232 event value>,<text description>

The text description may e.g. specify the corresponding key on the dental treatment center.

In the SDK example WinSample.cfg, the keys of the imaginary sample device are labeled with the messages which are triggered with the factory setting of "siucom.ini":

```
SIRONA SIUCOM CONFIGURATION (Version 0.1)
10, Activate Event
11, Stop Event
12, Toggle Event
201,Event 201
202,Event 202
203,Event 203
204,New Event 204
```

In the following, all PC-relevant user actions and user interfaces as well as their function codes sent via the serial interface are listed for all Sirona dental units capable of being interfaced with a PC:

The unit-specific function codes are stored in the *.cfg files bearing the same name located in the Siucom setup folder. Their contents depend on the hardware and software properties of the corresponding dental units. For this reason, these files may be created and/or edited only by Sirona.

The user selects the unit which is connected to the PC in the Siucom configuration.

The correspondingly available keys and events as well as their function codes are listed below:

6.1.3.1 C1

This device selection is valid for the following Sirona dental units:

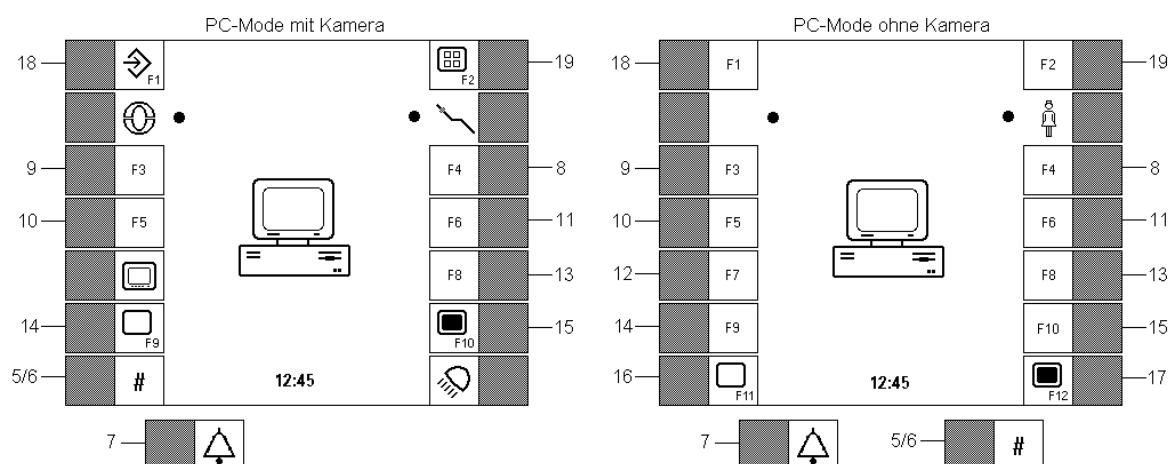
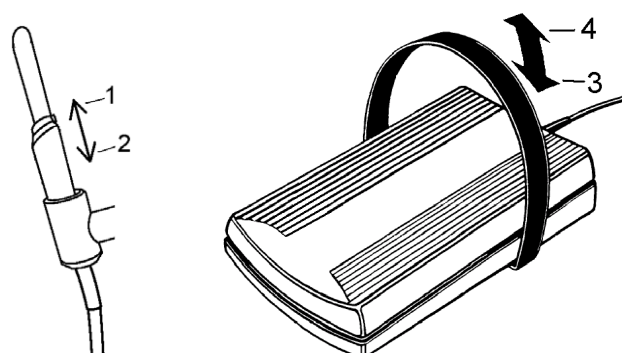
- C1⁺ with camera in the dentist element
- C1⁺ with second monitor function without camera

| No. | Key / Event | Code (decimal) |
|-----|--|----------------|
| 1 | Camera removed from instrument holder (=camera on)) ² | 60001 |
| 2 | Camera returned to instrument holder (=camera off)) ² | 60002 |
| 3 | Unit foot switch actuated (=foot switch active)) ^{2 3} | 60004 |
| 4 | Unit foot switch released (=foot switch inactive)) ^{2 3} | 60003 |
| 5 | # key active | 136 |
| 6 | # key inactive | 144 |
| 7 | Call key | 152 |
| 8 | Key F4 | 160 |
| 9 | Key F3 | 168 |
| 10 | Key F5 | 176 |
| 11 | Key F6 | 184 |
| 12 | Key F7) ⁴ | 192 |
| 13 | Key F8 | 200 |
| 14 | Key F9 | 208 |
| 15 | Key F10 | 216 |
| 16 | Key F11) ⁴ | 224 |
| 17 | Key F12) ⁴ | 232 |
| 18 | Key F1 | 240 |
| 19 | Key F2 | 248 |

)² = Key/Event is available with Protocol=4 only if a camera is integrated.

)³ = Only with camera removed or active PC menu on the dentist panel

)⁴ = Available only in PC mode without camera (second monitor function)



6.1.3.2 C2

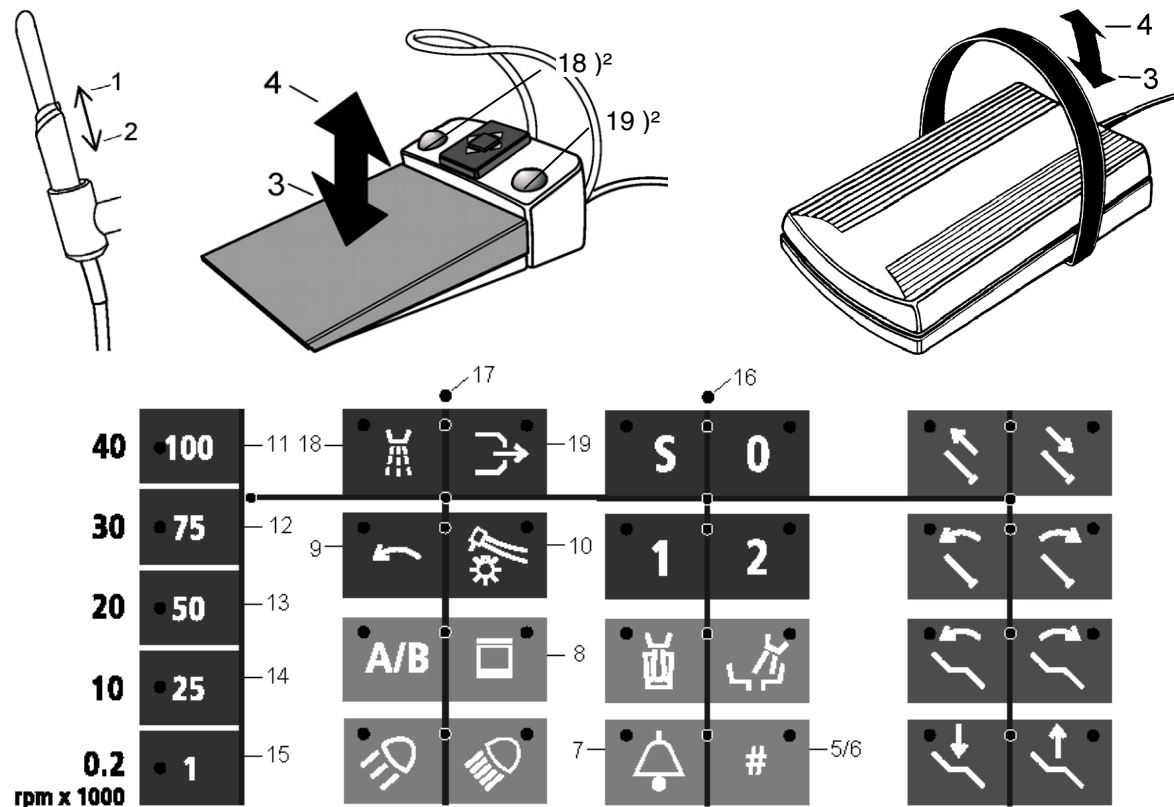
This device selection is valid for the following Sirona dental units:

- C2⁺ with camera in the dentist element
- C2⁺ with second monitor function without camera

| No. | Key / Event | Code (decimal) |
|-----|--|----------------|
| 1 | Camera removed from instrument holder (=camera on)) ² | 60001 |
| 2 | Camera returned to instrument holder (=camera off)) ² | 60002 |
| 3 | Unit foot switch actuated (=foot switch active)) ^{2 3} | 60004 |
| 4 | Unit foot switch released (=foot switch inactive)) ^{2 3} | 60003 |
| 5 | # key active | 136 |
| 6 | # key inactive | 144 |
| 7 | Call key | 152 |
| 8 | X-ray image viewer key | 160 |
| 9 | Counterclockwise rotation key | 168 |
| 10 | Instrument light key | 176 |
| 11 | Quick setting key 100 | 184 |
| 12 | Quick setting key 75 | 192 |
| 13 | Quick setting key 50 | 200 |
| 14 | Quick setting key 25 | 208 |
| 15 | Quick setting key 1 | 216 |
| 16 | Forward cursor from chair key block | 224 |
| 17 | Forward cursor from instrument key block | 232 |
| 18 | Spray key (C+ left foot switch button) | 240 |
| 19 | Chip blower key (C+ right foot switch button) | 248 |

)² = Key/Event is available with Protocol=4 only if a camera is integrated.

)³ = Only with camera removed



6.1.3.3 C3

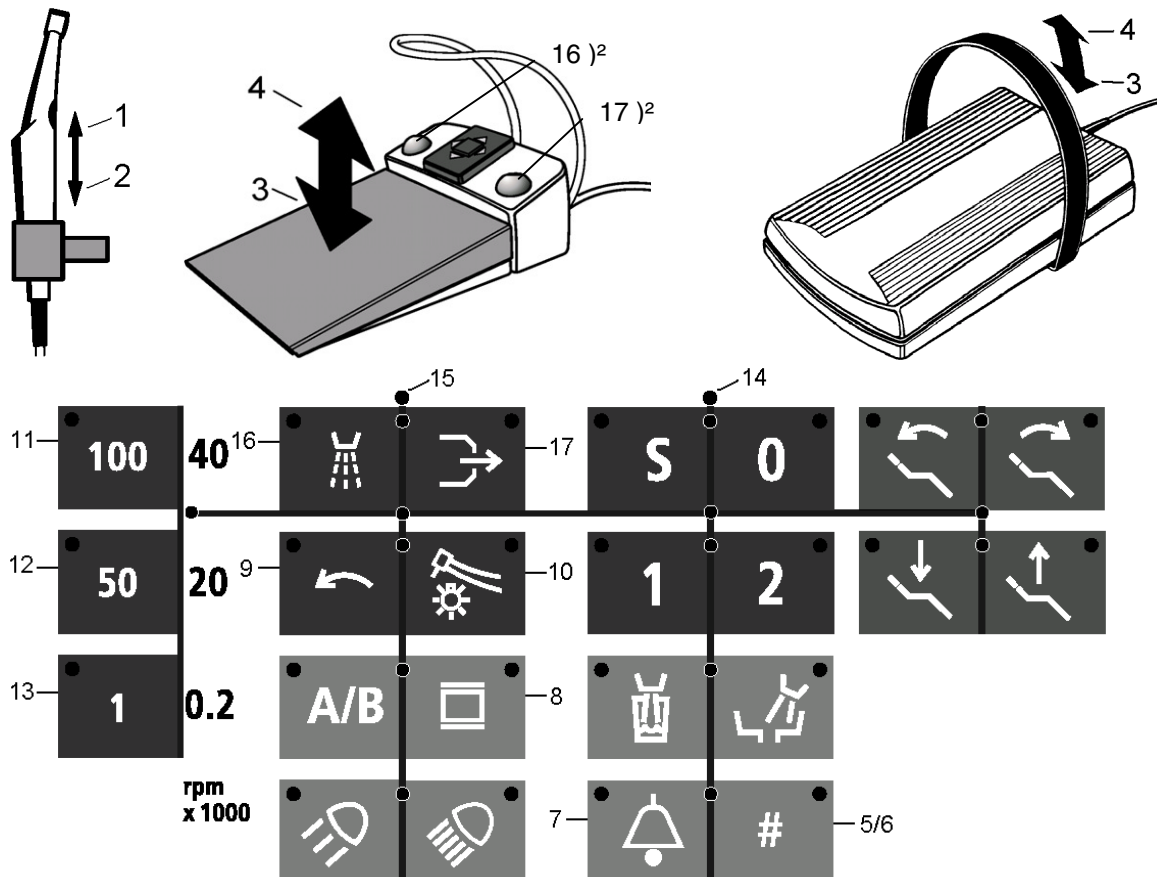
This device selection is valid for the following Sirona dental units:

- C3⁺ with camera in the dentist element
- C3⁺ with second monitor function without camera

| No. | Key / Event | Code (decimal) |
|-----|--|----------------|
| 1 | Camera removed from instrument holder (=camera on)) ² | 60001 |
| 2 | Camera returned to instrument holder (=camera off)) ² | 60002 |
| 3 | Unit foot switch actuated (=foot switch active)) ^{2 3} | 60004 |
| 4 | Unit foot switch released (=foot switch inactive)) ^{2 3} | 60003 |
| 5 | # key active | 136 |
| 6 | # key inactive | 144 |
| 7 | Call key | 152 |
| 8 | X-ray image viewer key | 160 |
| 9 | Counterclockwise rotation key | 168 |
| 10 | Instrument light key | 176 |
| 11 | Quick setting key 100 | 184 |
| 12 | Quick setting key 50 | 200 |
| 13 | Quick setting key 1 | 216 |
| 14 | Forward cursor from chair key block | 224 |
| 15 | Forward cursor from instrument key block | 232 |
| 16 | Spray key (C+ left foot switch button) | 240 |
| 17 | Chip blower key (C+ right foot switch button) | 248 |

)² = Key/Event is available with Protocol=4 only if a camera is integrated.

)³ = Only with camera removed



6.1.3.4 C4

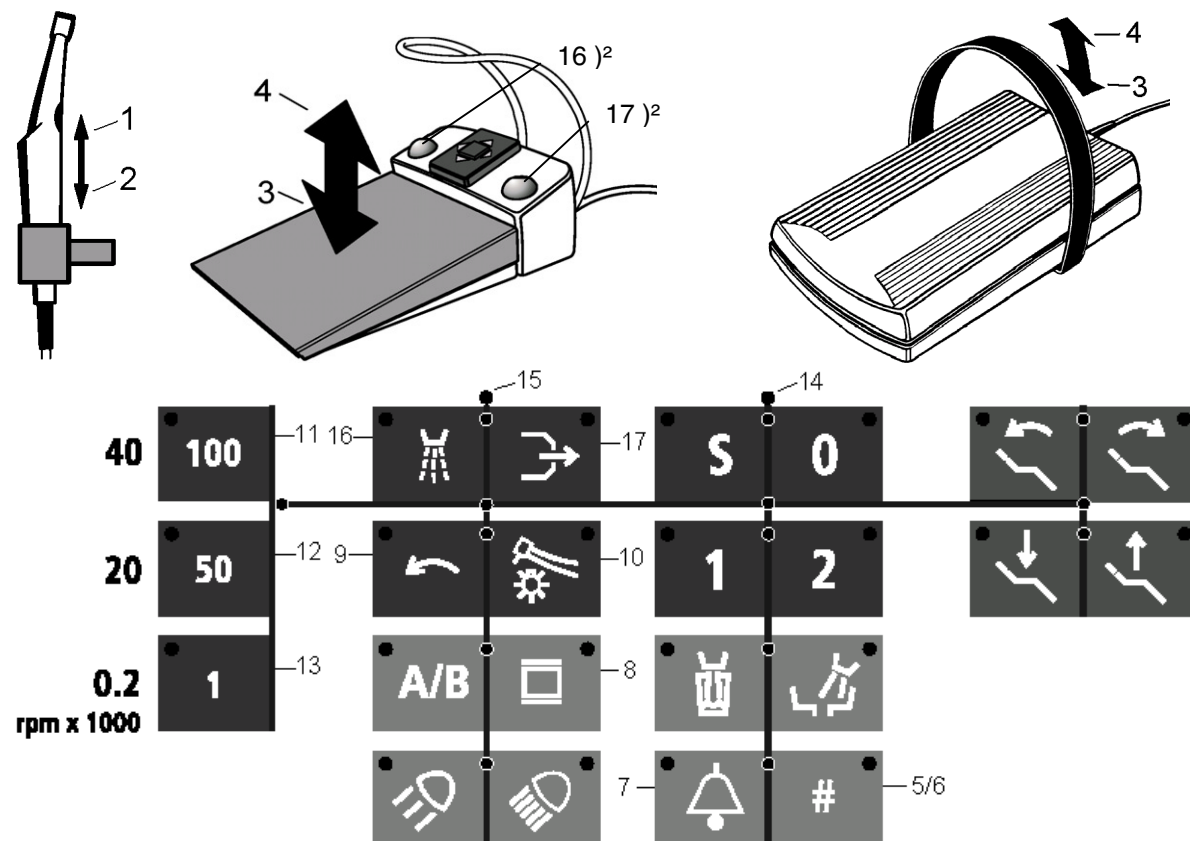
This device selection is valid for the following Sirona dental units:

- C4⁺ with camera in the dentist element
- C4⁺ with second monitor function without camera

| No. | Key / Event | Code (decimal) |
|-----|--|----------------|
| 1 | Camera removed from instrument holder (=camera on)) ² | 60001 |
| 2 | Camera returned to instrument holder (=camera off)) ² | 60002 |
| 3 | Unit foot switch actuated (=foot switch active)) ^{2 3} | 60004 |
| 4 | Unit foot switch released (=foot switch inactive)) ^{2 3} | 60003 |
| 5 | # key active | 136 |
| 6 | # key inactive | 144 |
| 7 | Call key | 152 |
| 8 | X-ray image viewer key | 160 |
| 9 | Counterclockwise rotation key | 168 |
| 10 | Instrument light key | 176 |
| 11 | Quick setting key 100 | 184 |
| 12 | Quick setting key 50 | 200 |
| 13 | Quick setting key 1 | 216 |
| 14 | Forward cursor from chair key block | 224 |
| 15 | Forward cursor from instrument key block | 232 |
| 16 | Spray key (C+ left foot switch button) | 240 |
| 17 | Chip blower key (C+ right foot switch button) | 248 |

)² = Key/Event is available with Protocol=4 only if a camera is integrated.

)³ = Only with camera removed



6.1.3.5 SIROCAM C DIRECT

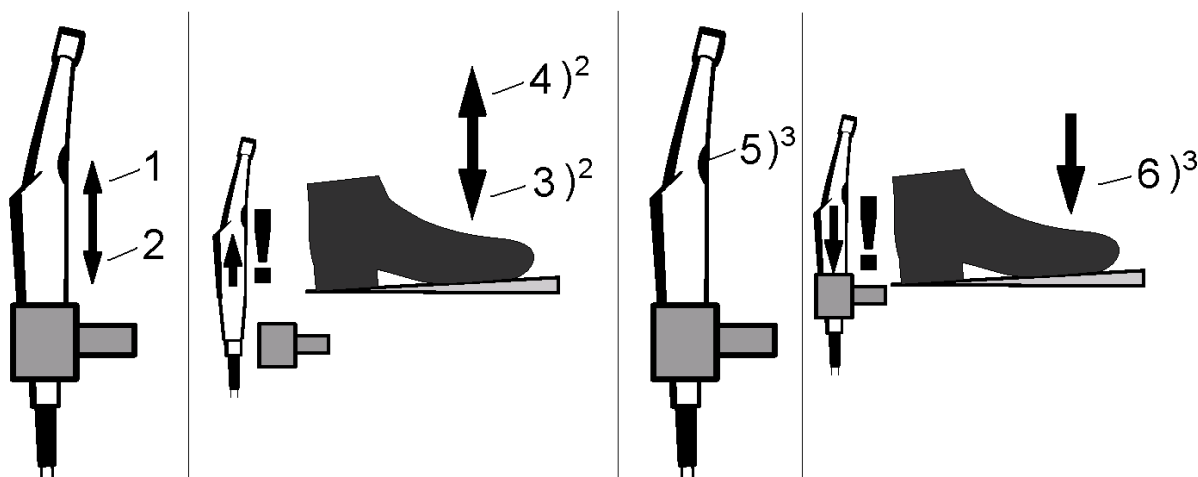
This device selection is valid for the following Sirona dental units:

- C1, C1⁺, C2, C2⁺, C3, C3⁺, C4, C4⁺, C6, C8 and ProFeel with "Sirocam C" camera at the monitor
- C3⁺ and C4⁺ with camera in the assistant element
- C6 and C8 with camera in the dentist element

| No. | Key / Event | Code (decimal) |
|-----|---|----------------|
| 1 | Camera removed from instrument holder (=camera on) | 60001 |
| 2 | Camera returned to instrument holder (=camera off)) ² | 60002 |
| 3 | Foot switch actuated (=foot switch active)) ^{2 3} | 60004 |
| 4 | Foot switch released (=foot switch inactive)) ² | 60003 |
| 5 | Camera button actuated) ³ | 160 |
| 6 | Foot switch actuated) ³ | 232 |

)² = Only with camera removed

)³ = Only with camera deposited.



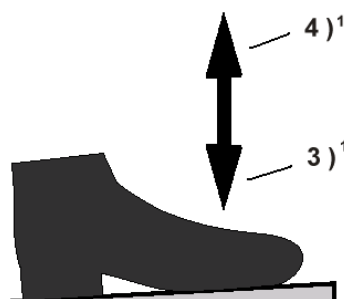
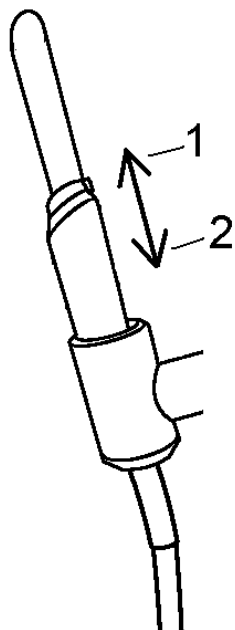
6.1.3.6 SIROCAM BOX

This device selection is valid for the following Sirona dental units:

- C1, C2, C3, C4 with Sivation 1
- Sirocam Box
- ProFeel with video application (integrated Sirocam Box)

| No. | Key / Event | Code (decimal) |
|-----|---|----------------|
| 1 | Camera removed from instrument holder (=camera on) | 60001 |
| 2 | Camera returned to instrument holder (=camera off) | 60002 |
| 3 | Foot switch actuated (=foot switch active)) ¹ | 60004 |
| 4 | Foot switch released (=foot switch inactive)) ¹ | 60003 |

)¹ = Only with camera removed



6.1.4 Import file formats

Starting with version 3.3, interactive import of third-party configurations can be performed with the help of the SIUCOM "Import" function.

The original factory configuration of both configuration sections can be restored at any time.

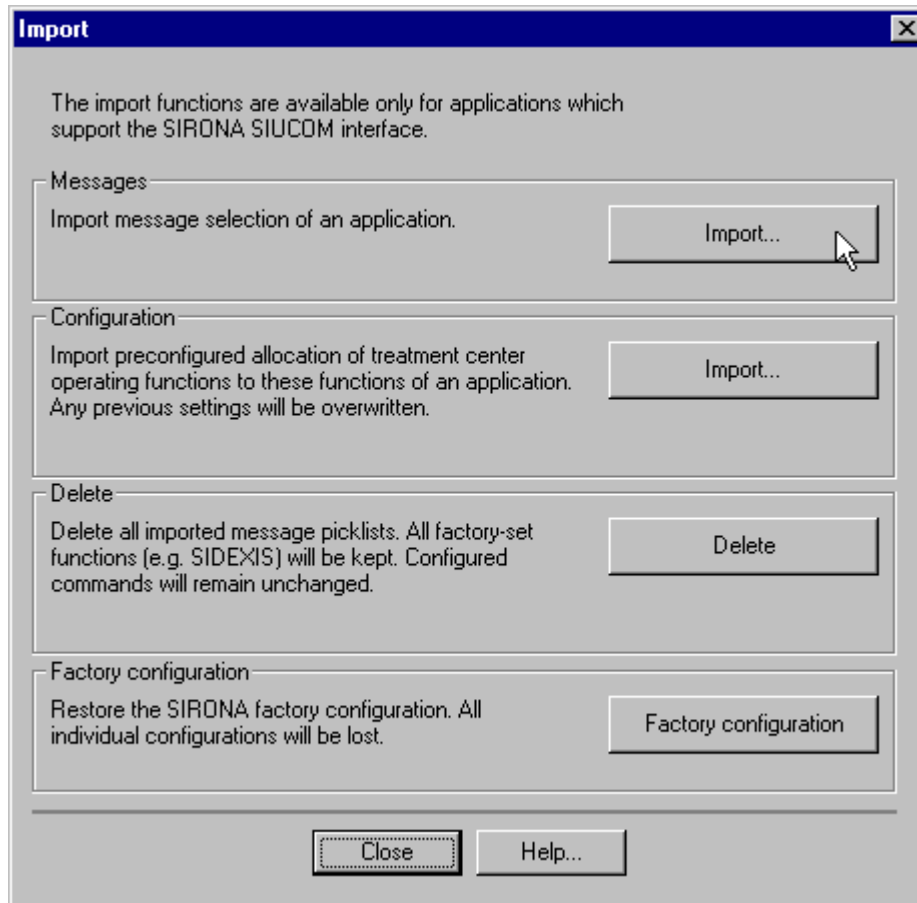


Figure 9 SIUCOM Import dialog

6.1.4.1 Importing message configurations (*.msd)

The contents of these files are transferred to the central MESSAGES.CFG. The first (comment) line is ignored, and all following lines must conform to the following format:

<target application>,<window class>,<message description>,<messages = Msg. ID>

Example:

```
TEST SIUCOM MESSAGE CONFIGURATION (Version test 0.0)
C:\WINDOWS\SNDREC32.EXE,SoundRec,Recorder: Stop,208
C:\WINDOWS\SNDREC32.EXE,SoundRec,Recorder: Replay,207
C:\WINDOWS\SNDREC32.EXE,SoundRec,Recorder: Begin,205
```

6.1.4.2 Importing configuration files (*.ind)

These configuration files are used to supplement/modify the central configuration database "SIUCOM.INI". The file structure corresponds to SIUCOM.INI. Existing sections are modified during the import. Previously existing key lines are overwritten, new ones are supplemented accordingly.

Format of configuration records:

<RS232 event> = <action>, <application name>, <window class>, <event>, opt.<Msg. ID>, opt.
<application name: message description>

The protocol definition defines the operating mode of Siucom. Protocol=4 must be set in order to activate 3rd party software.

```
[C2]
' This device should be modified/supplemented
Comment=camera in dentist element / second monitor function
Protocol=4

248=MSG,sidexis.exe,MdiXFrameClass,key F2,701,SIDEXIS: Tile
224=TOGGLE,screen.exe white,SIRONA WHITE SCREEN,key F11
232=TOGGLE,screen.exe black,SIRONA BLACK SCREEN,key F12

' This configuration line will be supplemented
161=TOGGLE,screen.exe white,SIRONA WHITE SCREEN,X-ray image viewer 2 key
```

The configurations of all devices from which the 3rd party software is to be controlled must be specified. Prepared files <application>.ind (in 4 languages) with all device sections and available control elements (RS232 events) are included in the SDK. The events used for the configuration import will be supplemented by the elements action, application name, window class and, if <action>=MSG, by MSG-ID and the functional description. Any key lines not required are deleted. Other device sections or key lines beyond those of the IND model are not possible.

6.1.5 SIUCOM help

When SIUCOM is started, the SIUCOM icon appears in the bottom right corner of the screen. Right-click this icon. The SIUCOM menu appears: You can obtain basic online help by selecting "Help".