



Руководство по работе с элементом управления ActiveX
CamMonitor.ocx

1. Введение	3
2. Установка CamMonitor.ocx	3
3. Параметры CamMonitor.ocx	3
4. Методы CamMonitor.ocx	7
5. События CamMonitor.ocx	9

Введение

На странице:

- [Назначение документа](#)
- [Общее описание CamMonitor.ocx](#)
- [Требования к разработчику](#)

Назначение документа

Настоящий документ содержит сведения, необходимые для использования ActiveX компонента CamMonitor.ocx. В документе описаны:

1. Установка CamMonitor.ocx.
2. Параметры CamMonitor.ocx.
3. Методы CamMonitor.ocx.
4. События CamMonitor.ocx.

Общее описание CamMonitor.ocx

CamMonitor.ocx представляет собой ActiveX компонент, который является полным аналогом интерфейсного объекта **Монитор видеонаблюдения**. Он позволяет управлять камерами, просматривать архив и т.д.

Компонент CamMonitor.ocx поддерживает работу в демо-режиме.

Требования к разработчику

Для использования CamMonitor.ocx требуется:

1. знание любого языка программирования, поддерживающего использование компонентов Component Object Model (COM);
2. знание основ программирования в Win32;
3. наличие среды разработки, поддерживающей работу с ocx-файлами.

Установка CamMonitor.ocx

Установка CamMonitor.ocx осуществляется при помощи файла CamMonitorInstaller.exe, который располагается в папке <Директория установки *Интеллект*>\Redist.

При установке на 32-битную систему файл cammonitor.ocx помещается в папку WINDOWS\system32, при установке на 64-битную систему – в папку WINDOWS\SysWOW64. Также осуществляется его стандартная регистрация как компонента ActiveX.

CamMonitorInstaller.exe выполняет установку требуемых файлов для всех пользователей системы.

Помимо самой библиотеки устанавливается также пакет драйверов CodecPack и утилита ITV VideoPlayer, которая работает с использованием компонента CamMonitor.ocx и позволяет просматривать видеоархив по выбранной камере. Данная утилита по умолчанию устанавливается в папку C:\Program Files\ITV VideoPlayer\. Интерфейс утилиты схож с интерфейсом Converter.exe, однако в ней отсутствуют некоторые функции Converter.exe, не связанные с просмотром архива. Данную утилиту можно использовать для проверки корректности установки CamMonitor.ocx.

Параметры CamMonitor.osx

На странице:

- [CamMenuOptions](#)
- [CamMenuProcessingOptions](#)
- [CamButtonsOptions](#)
- [MainPanelOptions](#)
- [KeysOptions](#)
- [OverlayMode](#)
- [Пример использования параметров](#)

В этом разделе перечислены параметры, позволяющие настроить режим работы компонента CamMonitor: задать отображаемые элементы интерфейса, а также режим использования оверлея.

Все параметры представляют собой целое число типа long.

Значения параметров для настройки элементов интерфейса, перечисленные в таблицах, сформированы таким образом, чтобы в двоичном представлении числа была только одна единица. Чтобы задать значение параметра, необходимо при помощи операции исключающего ИЛИ (XOR) объединить требуемые значения параметров, получив таким образом некое число, разряды которого в двоичном представлении говорят о том, какие элементы интерфейса следует отображать, а какие нет. См. также [Пример использования параметров](#).

Параметр OverlayMode отличается от прочих: он принимает значения от 0 до 2, и его значение задает режим использования оверлея.

CamMenuOptions

CamMenuOptions : long

Позволяет настроить функциональное меню камеры.

Допускается установка одного или нескольких флагов.

Возможные значения:

Значение	Описание
#define MENU_ENABLE_OPTION 0x00000001	Отображать меню
#define MENU_ARM_ENABLE_OPTION 0x00000002	Отображать пункт меню Поставить на охрану
#define MENU_REC_ENABLE_OPTION 0x00000004	Отображать пункт меню Начать запись
#define MENU_CAMS_ENABLE_OPTION 0x00000008	Отображать пункт меню Камера
#define MENU_TITLES_ENABLE_OPTION 0x00000010	Отображать пункт меню Отображение титров
#define MENU_PROCESSING_ENABLE_OPTION 0x00000020	Отображать пункт меню Обработка

#define MENU_EXPORT_ENABLE_OPTION 0x00000040	Отображать пункт меню Экспорт
--	--------------------------------------

CamMenuProcessingOptions

CamMenuProcessingOptions : long

Позволяет настроить меню **Обработка** в функциональном меню камеры.

Допускается установка одного или нескольких флагов.

Возможные значения:

Значение	Описание
#define MENU_PROCESSING_DEINTERLACE_ENABLE_OPTION 0x00000001	Отображать пункт меню Деинтерлейсинг
#define MENU_PROCESSING_ZOOM_ENABLE_OPTION 0x00000002	Отображать пункт меню Увеличение
#define MENU_PROCESSING_CONTRAST_ENABLE_OPTION 0x00000004	Отображать пункт меню Контрастирование
#define MENU_PROCESSING_MASK_ENABLE_OPTION 0x00000008	Отображать пункт меню Маска детектора

CamButtonsOptions

CamButtonsOptions : long

Позволяет настроить отображение кнопок компонента CamMonitor.

Допускается установка одного или нескольких флагов.

Возможные значения:

Значение	Описание
#define BUTTON_ARCH_ENABLE_OPTION 0x00000001	Отображать кнопку входа в архив
#define BUTTON_TIME_ENABLE_OPTION 0x00000002	Отображать время
#define BUTTON_NAME_ENABLE_OPTION 0x00000004	Отображать название камеры
#define BUTTON_MENU_ENABLE_OPTION 0x00000008	Отображать кнопку вызова функционального меню
#define BUTTON_RAYS_ENABLE_OPTION 0x00000010	Не используется
#define BUTTON_MICS_ENABLE_OPTION 0x00000020	Не используется

MainPanelOptions

MainPanelOptions : long

Позволяет настроить отображение панели инструментов CamMonitor.

Допускается установка одного или нескольких флагов.

Возможные значения:

Значение	Описание
#define MAIN_PANEL_ENABLE_OPTION 0x00000001	Включает отображение панели

KeysOptions

KeysOptions : long

Позволяет настроить управление компонентом при помощи клавиатуры и мыши.

Допускается установка одного или нескольких флагов.

Возможные значения:

Значение	Описание
#define TELEMETRY_DISABLE_OPTION 0x00000001	Отключает возможность управления компонентом CamMonitor при помощи горячих клавиш, доступных для Монитора видеонаблюдения (см. Монитор видеонаблюдения).
#define TELEMETRY_DISABLE_OPTION 0x00000002	Отключает возможность управления телеметрией из компонента CamMonitor (см. Управление поворотными устройствами)

OverlayMode

OverlayMode : long

Задаёт режим отображения.

Возможные значения:

Значение	Описание
0	Не использовать Overlay
1	Overlay 1
2	Overlay 2

Пример использования параметров

```
DWORD options = CamMonitor1->CamMenuOptions;  
  
options = options^MENU_CAMS_ENABLE_OPTION^MENU_ARM_ENABLE_OPTION^MENU_REC_ENABLE_OPTION;  
  
CamMonitor1->CamMenuOptions = options;  
  
CamMonitor1->CamMenuProcessingOptions ^= MENU_PROCESSING_MASK_ENABLE_OPTION;
```

Методы CamMonitor.osx

На странице:

- [Connect](#)
- [ShowCam](#)
- [DoReactMonitor](#)
- [RemoveAllCams](#)
- [IsConnected](#)
- [GetCurIp](#)
- [SendRawMessage](#)
- [Disconnect](#)

Connect

Connect(BSTR **ip**, BSTR **login**, BSTR **password**, BSTR **arch_password**, long **param**) установка соединения с сервером.

- BSTR **ip** – IP адрес сервера.
- BSTR **login** – логин для соединения с сервером (может быть пустым).
- BSTR **password** – пароль на соединение с сервером (может быть пустым).
- BSTR **arch_password** – пароль для доступа к архиву (т.е. пароль администратора, может быть пустым).
- long **param** – роль, исполняемая сервером:
 - 0 – видеосервер;
 - 1 – оперативный архив;
 - 2 – видеоплюс.

Установка связи с сервером происходит **асинхронно**.

ShowCam

ShowCam(long **cam_id**, long **compress**, long **show**) выводит/скрывает камеру с экрана

- long **cam_id** – идентификатор(номер) камеры.
- long **compress** – уровень компрессии видео 0-5 (для локальной камеры =0).
- long **show** – флаг означающий действие: показать/скрыть камеру (1/0).

DoReactMonitor

DoReactMonitor(BSTR **react_string**) – управление поведением монитора/камер

- BSTR **react_string** – строковое представление реакции.

Пример формирования react_string:

```
react_string = "MONITOR||ARCH_FRAME_TIME|cam<3>,date<dd-mm-yy>,time<hh:mm:ss>";
CamMonitor1->DoReactMonitor(react_string);
```

Результат вызова функции с таким параметром: камера 3 будет переведена в режим архива, и архив будет позиционирован на дату «dd-mm-yy» и время «hh:mm:ss» (дату и время необходимо задавать только в таком формате).

"MONITOR|<id игнорируется >|ARCH_FRAME_TIME|..."



Примечание:

Опционально можно указать также точность при позиционировании до миллисекунд например:

```
DoReactMonitor("MONITOR||ARCH_FRAME_TIME|cam<3>,date<02-10-05>,time<12:12:22.345>
```

RemoveAllCams

RemoveAllCams() : long – удаление всех камер с экрана

IsConnected

IsConnected() : boolean - метод говорит о наличии/отсутствии связи с видеосервером.

GetCurIp

GetCurIp() : BSTR – возвращает IP адрес сервера, ранее указанный при вызове **Connect**.

SendRawMessage

SendRawMessage(BSTR **msg**) – дать видеосерверу команду на исполнение.

- BSTR **msg** – строковое представление команды.

Примеры вызова функции:

```
m_Cam.SendRawMessage("CAM|1|REC");
```

```
m_Cam.SendRawMessage("CAM|1|REC_STOP");
```

```
m_Cam.SendRawMessage("CAM|1|ARM");
```

```
m_Cam.SendRawMessage("CAM|1|DISARM");
```

Disconnect

Disconnect() – осуществляет отсоединение от видеосервера.

События CamMonitor.ocx

OnCamListChange (long **cam_id**, long **action**) - возникает при установлении связи с сервером или при изменении количества камер на сервере

- long **cam_id** – идентификатор камеры.
- long **action** – равен 1, если камера с **id** == **cam_id** существует, иначе **action** == 0.

Данное событие возникает столько раз, сколько камер на данном видеосервере. Признаком окончания вызовов **OnCamListChange** является отрицательное значение параметра **cam_id** (**cam_id** < 0).

Если на сервере находится, например, 3 камеры (1, 2, 3), то последовательно возникнут события:

```
CamListChange(1,1)
```

```
CamListChange(2,1)
```

```
CamListChange(3,1)
```

```
CamListChange(-1,1)
```

Пример:

Показать камеру с **cam_id** =2 с уровнем компрессии **compress** =1;

```
CamMonitor1CamListChange(long cam_id, long action)
{
  if(cam_id == -1)
  {
    CamMonitor1->ShowCam(2,1,1);
  }
}
```