Instructions of CedarX Recorder

Author	XSJ
Date Created	2011-6-29
Current	1.00

Contents

1 Summary

The document illustrates the procedure to use CedarX Recorder Framework through the APP of Recorder. You can refer to the testing program CedarXRecorder Framework under cedarx/test/recorder

2 API
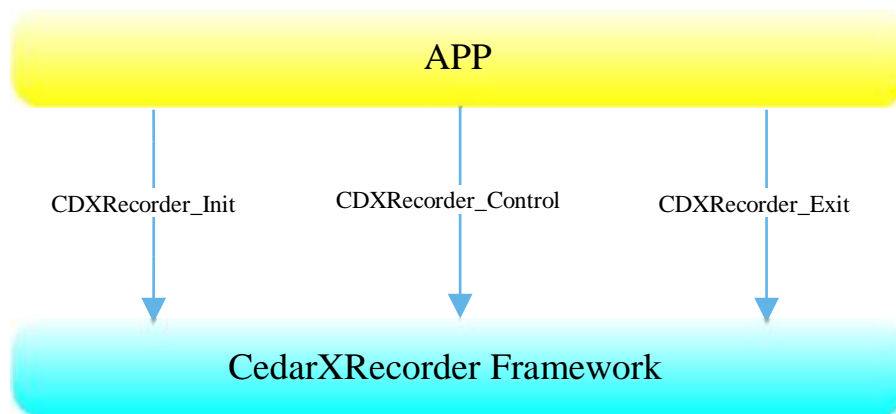
CedarXRecorder Framework provides only 3 API port for the upper APP：

extern int CDXRecorder_Init();

extern int CDXRecorder_Control(int cmd, unsigned int para0, unsigned int para1);

extern int CDXRecorder_Exit();

CDXRecorder_Init finishes some initial preparation；

CDXRecorder_Exit finishes clean-up work as releasing resources before exit；

Some concrete functions are realized by command set up by CDXRecorder_Control，such as transfer of status， set-up of parameter and etc.

Pay attention to several command of App

Command	Instruction
CDX_CMD_PREPARE	Create and connect component
CDX_CMD_START	Finishing creating component and setting –up parameter, record by this command.

CDX_CMD_PAUSE	Pause recording by this command，send CDX_CMD_START again, continue recording
CDX_CMD_STOP	Stop recording.
CDX_CMD_GETSTATE	Get the current running status
CDX_CMD_SET_SAVE_FILE	App open file，set up this file to corresponding component to save recording file
CDX_CMD_SET_VIDEO_INFO	Set up parameter of video
CDX_CMD_SET_PREVIEW_INFO	Set up parameter of preview

3 steps

3.1 Step1 Initialization

Use CDXRecorder_Init();

```
CDXRecorder_Init();
```

3.2 Step2 Create CedarX

Send　CDX_CMD_PREPARE：

```
CDXRecorder_Control(CDX_CMD_PREPARE, 0, 0)
```

3.3 Step3 Set-up of parameter

3.3.1 Set the file pointer

When App open a file, it send CDX_CMD_SET_SAVE_FILE and　transfer file pointer to CedarX：

```
    FILE * save_file_mp4 = NULL;
```

```
    save_file_mp4 = fopen("rec.mp4", "wb");

    if (save_file_mp4 == 0)

    {

        LOGE("open save file failed");

        return -1;

    }


    // set file handle to CDX_Recorder render component

    ret = CDXRecorder_Control(CDX_CMD_SET_SAVE_FILE, (unsigned int)save_file_mp4, 0);

    if(ret != 0)

    {

        LOGE("CedarXRecorder::prepare, CDX_CMD_SET_SAVE_FILE failed\n");

        return ret;

    }
```

Attention:the create and close of file are both completed by APP. Close file pointer before exit。

```
    if (save_file_mp4 != 0)

    {

        fclose(save_file_mp4);

        save_file_mp4 = 0;

    }
```

3.3.2 Set-up parameter of sample video

Set up parameter of video by CDX_CMD_SET_VIDEO_INFO，as video frame(640*480），frame rate（30 frame /s），bit rate, etc.

VIDEOINFO_t vInfo;

```
    // set video size and FrameRate to CDX_Recorder
```

```
memset((void *)&vInfo, 0, sizeof(VIDEOINFO_t));


vInfo.width = 640;

vInfo.height = 480;

vInfo.frameRate = 30;

vInfo.bitRate = 1024*1024;

ret = CDXRecorder_Control(CDX_CMD_SET_VIDEO_INFO, (unsigned int)&vInfo, 0);

if(ret != 0)

{

    LOGE("CedarXRecorder::prepare, CDX_CMD_SET_VIDEO_INFO failed\n");

    return ret;

}
```

Attention： current sensor only support two resolution:640*480 and 320*240；bigger the bit rate, clearer，but take more space；

Small the bit rate，worse the image，but take less space（choose a optimum after testing）。

3.3.3 set up parameter of preview

Set up coordinate, width and height of parameter by CDX_CMD_SET_PREVIEW_INFO:

```
PREVIEWINFO_t preview_info;

// set preview info to CDX_Recorder

memset(&preview_info, 0, sizeof(PREVIEWINFO_t));

preview_info.width = 640;

preview_info.height = 480;

preview_info.left = (800 - preview_info.width) / 2;

preview_info.top = (480 - preview_info.height) / 2;

ret = CDXRecorder_Control(CDX_CMD_SET_PREVIEW_INFO, (unsigned int)&preview_info, 0);

if(ret != 0)
```

```
{
    LOGE("CedarXRecorder::prepare, CDX_CMD_SET_PREVIEW_INFO failed\n");

    return ret;

}
```

3.4 Step4 Start recording

After components are created, parameter are set up, record via CDX_CMD_START.

```
CDXRecorder_Control(CDX_CMD_START, 0, 0);
```

Pause in the process，use CDX_CMD_PAUSE. For instance, check whether the status is pause. If yes, send

CDX_CMD_START，it continue srecording. If not , send CDX_CMD_PAUSE, it pauses.

```
if(CDXRecorder_Control(CDX_CMD_GETSTATE, 0, 0) == CDX_STATE_PAUSE)

    CDXRecorder_Control(CDX_CMD_START, 0, 0);

else

    CDXRecorder_Control(CDX_CMD_PAUSE, 0, 0);
```

3.5 Step5 Stop recording

Stop recording via CDX_CMD_STOP

```
CDXRecorder_Control(CDX_CMD_STOP, 0, 0);
```

3.6 Step6 Exit

When completing recording，use CDXRecorder_Exit to exit CedarX。

```
CDXRecorder_Exit();
```