

사용설명서 ver. 1.0K

PLENUE D Flash UCI (User Creative Interface) 제작지침

1. Flash UCI (User Creative Interface) 개요

UCI 는 사용자가 Flash 를 이용해 자유롭게 음악 UI 를 디자인하고 제어할 수 있도록 만들어졌습니다.

임베디드와 Flash 라는 한계로 인해 기기의 모든 기능을 UCI 에서 제어할 수는 없지만, 최대한 UCI 개발자가 자유롭게 개발할 수 있도록 구현되어 있습니다. 각 기능 제어에 필요한 FS Command 들은 간단하고 명료하게 구현되어 있기 때문에 기존 Flash ActionScript 개발자들은 PLENUE D 의 '입력장치'와 그에 따른 'FS Command' 처리만 기존 Flash ActionScript 에서 추가적으로 이해한다면 큰 어려움 없이 UCI 개발을 할 수 있습니다.

2. Flash Engine Specification

- 1) Flash Version : Flash Player 7
- 2) ActionScript Version : ActionScript 2.0
- 3) Display Resolution : 240 * 320
- 4) Frame Rate : 24fps
- 5) Sound : UI 에서는 미지원
- 6) Font : Vector Font 지원. 임베디드 기계 특성으로 인해 ActionScript 에서 폰트를 지정해도 “시스템 폰트” 또는 “사용자 폰트” 로만 출력
- 7) Rotation : TextField 를 제외하고는 제한 없음
- 8) 최대 메모리 크기 : 15 MB

3. 소프트웨어 구조

PLENUE D 의 Flash 는 UCI 를 구현하기 위해 '그림 3 - 1' 과 같이 4 계층으로 구현되어 있습니다.

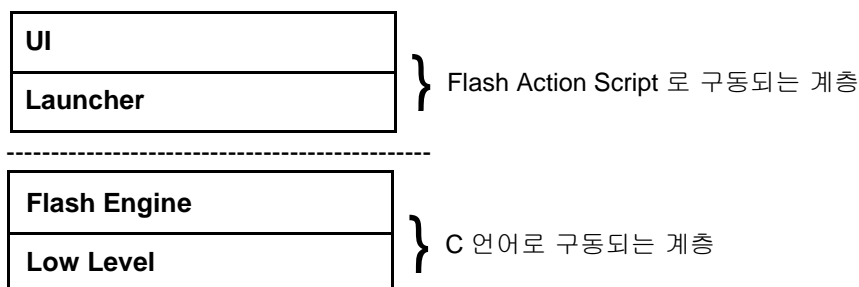


그림 3 - 1

- Low Level 계층 : Low Level Device (CPU, LCD, Codec 등) 제어
- Flash Engine 계층 : Flash 콘텐츠를 구동하며, Low Level 계층 이벤트 등을 Flash

컨텐츠로 전달

- Launcher 계층 : Flash 에서 'Level0' 에 위치하여 UI 를 담당하는 Flash 컨텐츠를 Load, Unload 처리
- UI 계층 : Flash 'Level 1'에 위치하며 일반 UI 를 담당

이 구조는 일반동작에서만 적용되며, 'USB', '펌웨어 업그레이드', '시작 화면', '종료화면' 등에서는 'Low Level 계층' 에서 직접 화면을 제어하기 때문에 UCI 가 적용되지 않습니다. 또한 LCD Off 한 상태에서는 구조적인 이유로 'Low Level 계층' 만 동작하도록 구현되어 있습니다.

Launcher 계층과 UI 계층을 담당하는 Flash 컨텐츠 파일들은 펌웨어에 포함되어 가상으로 실행됩니다. UCI 를 실행시키려면, 기기의 특정 폴더('System\Flash UI') 에 UCI 파일을 복사하면 UCI 가 기존 UI 대신 작동하게 됩니다.

1) Low Level 계층

CPU, LCD, Codec 등 Low Level Device 를 제어하는 가장 하위 계층입니다. Flash Script 에서는 FS Command 를 통하여 미리 서로 정의되어 있는 Command 를 통해서만 Low Level Device 제어 가능합니다. FS Command 는 기본적으로 Synchronous Command 이며, 소수의 Command 만 Asynchronous 하게 동작합니다. Asynchronous FS Command FS Command 설명 시 별도로 언급되며, 언급이 없는 FS Command 는 Synchronous 로 생각하고 사용하면 됩니다.

2) Flash Engine 계층

Flash 컨텐츠를 해석하여 컨텐츠에 있는 ActionScript 에 따라 Flash 컨텐츠를 구동합니다. 그리고 ActionScript 에 따라서 FS Command 를 호출하여 Low Level Device 를 제어 하거나, Low Level Device 에서 발생한 Key, Touch 등의 Event 를 ActionScript 로 전달합니다.

Flash Engine 에는 Flash 컨텐츠를 구동하기 위해 15MB(펌웨어 버전에 따라 다를 수 있습니다) 를 할당하고 있으며, 컨텐츠 메모리가 15MB 이상을 사용할 경우 외부에 Dynamic Memory 를 자동으로 할당하여 사용합니다. 다만, Dynamic Memory 는 음악 재생에서도 사용하기 때문에 가능하면 Flash 컨텐츠를 15MB 이내로 구현하기를 권장합니다.

3) Launcher 계층

Launcher 계층은 'launcher.swf' 로 구현되어 있으며, UI 를 담당하는 Flash 컨텐츠 파일을 Load, Unload 하는 역할을 합니다. 이 파일에는 Global 함수로 LoadSWF() 함수가 구현되어 있는데, 이 함수를 통해 개발자들은 'Level 1' 에 원하는 Flash 컨텐츠를 올릴 수 있습니다.

LoadSWF() 함수를 실행하면 함수 내부에서 'unloadMovieNum(1)' 함수가 먼저 호출되어 이전에 LoadMovie 되었던 Flash 컨텐츠를 UnloadMovie 합니다. 그 이후 입력된 ItemNum 값에 따라서 Flash 컨텐츠가 실행됩니다.

- 함수 원형

`_global.LoadSWF = function(itemNumber : Number) : Void`

- 입력 값

itemNum : 실행할 Flash 콘텐츠의 Index. 입력값은 아래와 같습니다. 아래 Global 변수는 'launcher.swf' 에 정의되어 있습니다.

`_global.MODE_MUSIC : '0'` 값을 갖고 있는 Number 형 상수이며, 'music.swf' 파일을 실행

`_global.MODE_SETTING : '14'` 값을 갖고 있는 Number 형 상수이며, 'setting.swf' 파일을 실행

`_global.MODE_BROWSER : '15'` 값을 갖고 있는 Number 형 상수이며, 'browser_total.swf' 파일을 실행

- 반환 값

Void

4) UI (User Interface) 계층

실제 사용자가 기기를 사용하는데 필요한 UI 가 구현되어 있는 계층입니다. UCI 는 이 계층에서 구현됩니다.

4. Flash ActionScript 구조

Flash ActionScript 는 Launcher 계층에서 사용자가 선택한 Flash UI 파일을 실행하여 UI 계층이 동작하도록 되어 있습니다.

UI를 구현하다 보면 "Low Level 계층"을 직접 제어할 필요는 없지만, 저장에 필요한 Parameter 가 있습니다. 이런 경우 아래와 같은 방법으로 ActionScript 에서 필요한 Parameter 를 저장할 수 있습니다. 여기서는 개요만 언급합니다. 자세한 내용은 FS Command 문서를 참조하시기 바랍니다.

- 각 Mode 별 할당되어 있는 저장 공간을 사용하는 방법 : 'GetEtcUIConfig', 'SetEtcUIConfig' FS Command 로 구현되어 있습니다.

- ActionScript 개발자에게 개방된 저장 공간을 사용하는 방법 : 'EtcUsrGetNumber', 'EtcUsrSetNumber', 'EtcUsrGetString', 'EtcUsrSetString' FS Command 로 구현되어 있습니다. 다만, 사용자가 수시로 UCI 를 바꿀 수 있기 때문에 이 FS Command 를 사용하기 위해선 별도의 간단한 Header 를 붙여 고유한 정보를 확인해야 합니다. 또한, 사용자가 한 개발자가 만든 UCI 를 사용하는 것이 아닌, 여러 버전의 UCI 를 사용할 경우, 해당 값을 보장할 수 없습니다. 따라서 가급적 Mode 별로 할당된 저장 공간을 사용하는 것을 권장합니다.

5. 입력 장치

1) 상단 Key

PLENUE D 의 상단 Key 는 'Power' Key 가 배치되어 있습니다. Key 동작은 아래와 같습니다.

- 'Power' Key : Low Level 계층에서 처리하기 때문에 Key Event 가 발생되지 않습니다.

2) 우측 Key

PLENUE D 의 우측에는 '+', '-', '재생/일시정지', '이전', '다음' Key 가 배치되어 있습니다. Key 동작 은 반복적으로 이벤트를 발생하도록 구현되어 있습니다. 키를 누르고 있으면, Key Down Event 가 반복적으로 발생합니다. 반대로 사용자가 누르고 있던 키를 떼면 Key Up Event 가 발생합니다. Flash 엔진에서 ActionScript 로 전달되는 Key Code 값은 아래와 같습니다.

- '+' Key : 38
- '-' Key : 40
- '재생/일시정지' Key : 35
- '이전' Key : 39
- '다음' Key : 37

3) Touch

Flash 에서는 Touch 이벤트를 처리하는 기능이 별도로 없기 때문에 PLENUE D 에서 터치가 눌리면 Flash Engine 계층에서는 Mouse Event 로 대체되어 이벤트를 발생시켜 처리합니다. Touch 동작 별 처리는 아래와 같습니다.

- 사용자가 LCD On 상태에서 Touch 를 할 경우 : Mouse Up Move Event 를 먼저 발생한 다음 Mouse Down Event 가 발생합니다. 이후부터는 한 Frame 마다 Mouse Down Move Event 가 발생합니다. 사용자가 터치한 위치는 Mouse X, Y 값을 통해서 알 수 있습니다.
- 사용자가 LCD On 상태에서 Touch 를 뗄 경우 : Mouse Up Event 가 발생합니다.

4) 가상 Key

LCD Off가 된 상태에서는 Flash Engine이 동작하지 않습니다. 따라서 LCD Off 상태에서 On 상태로 변경되면 Off된 상태의 GUI가 남아 있는 현상이 발생합니다. 또한 음악 GUI에서 재생 중 다음 곡으로 넘어갈 때 펌웨어 내부에서 처리되기 때문에 Flash에서는 다음 곡으로 넘어갔는지 여부를 알 수 없어 GUI는 이전 곡으로 남아있는 현상이 발생합니다. 이런 현상을 해결하기 위해서 Flash GUI를 업데이트하기 위한 목적으로 가상 Key인 'F12' Key를 정의하여 사용합니다. 따라서 Action Script에서는 'F12' Key가 발생하면 GUI를 갱신하면 됩니다.

6. 앨범아트의 Load

Movie Clip Load(loadMovie(), loadClip() 함수 등)를 통해 가상의 파일을 Load할 수 있습니다.

Tag 및 Folder DB를 사용하여 제어를 하고, 또한 파일 내부에 JPEG 파일이 존재하는 경우가 있어 Flash Spec에 없는 별도의 기능을 제공합니다.

1) 일반 앨범아트 Load

- '*.MU0' 확장자로 Load 하면 해당 파일의 앨범아트가 Load 됩니다.
'SetAudAlbumArtMCSize' FS Command를 통해 미리 설정된 Pixel 값보다 크다면, 가로 혹은 세로 중의 최대 크기가 해당 Pixel 값에 맞도록 축소하여 Load합니다.

예) loadClip("0.MU0", "MC_AlbumArt")

- 앞에 붙는 숫자는 앨범아트를 Load 할 파일의 인덱스 입니다. 즉 재생 목록에 10개의 파일이 있을 경우 Load 할 수 있는 인덱스는 '0 ~ 9'까지 입니다.
- 재생 목록의 총 파일 개수는 'GetEtcTotalPLNum' FS Command 를 통해서 알 수 있습니다.
- 음원 파일이 멀티 앨범아트를 지원할 경우, 'GetAudAlbumArtTotalNum'와 'SetAudAlbumArtIdx'를 통해 최대 6장까지 멀티 앨범아트를 Load 할 수 있습니다.
만약 파일에는 앨범아트가 3개 밖에 없는데도 4번째 앨범아트를 Load 할 경우, 펌웨어 내부에서 첫 번째 앨범아트를 Load 하도록 처리됩니다.

2) 블러 필터 앨범아트 Load

- '*.BL0' 확장자로 Load 하면 블러(Blur) 필터가 적용된 앨범아트가 Load 됩니다.
'SetAudAlbumArtBlurSize' FS Command 를 통해 미리 설정된 pixel 값으로 이미지를 Load 합니다.

예) loadClip("0.BL0", "MC_BlurAlbumArt")

- 앞에 붙는 숫자는 앨범아트를 Load 할 파일의 인덱스 입니다. 즉 재생 목록에 10개의 파일이 있을 경우 Load 할 수 있는 인덱스는 '0 ~ 9'까지 입니다.
- 재생 목록의 총 파일 개수는 'GetEtcTotalPLNum' FS Command 를 통해서 알 수 있습니다.
- 음원 파일이 멀티 앨범아트를 지원할 경우, 'GetAudAlbumArtTotalNum'와 'SetAudAlbumArtIndex'를 통해 최대 6장까지 멀티 앨범아트를 Load 할 수 있습니다.
만약 파일에는 앨범아트가 3개 밖에 없는데도 4번째 앨범아트를 Load 할 경우, 펌웨어 내부에서 첫 번째 앨범아트를 Load 하도록 처리됩니다.