

# Autodesk® Scaleform®

## CLIK Buttons 시작하기

이문서는 스케일폼 CLIK 의 버튼 컴포넌트를 어떻게 하면 빠르게 사용할 수 있을지에 대하여 설명한다.

집필: Matthew Doyle

버전: 2.0

최종편집: 2010 년 8 월 19 일

## Copyright Notice

### Autodesk® Scaleform® 4.4

© 2014 Autodesk, Inc. All rights reserved. Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries: 123D, 3ds Max, Algor, Alias, AliasStudio, ATC, AutoCAD LT, AutoCAD, Autodesk, the Autodesk logo, Autodesk 123D, Autodesk CAM 360, Autodesk Homestyler, Autodesk Inventor, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSketch, AutoSnap, AutoTrack, Backburner, Backdraft, Beast, BIM 360, Burn, Buzzsaw, CADmep, CAiCE, CAMduct, CFdesign, Civil 3D, Cleaner, Combustion, Communication Specification, Configurator 360™, Constructware, Content Explorer, Creative Bridge, Dancing Baby (image), DesignCenter, DesignKids, DesignStudio, Discreet, DWF, DWG, DWG (design/logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DWGX, DXF, Ecotect, ESTmep, Evolver, FABmep, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, ForceEffect, FormIt, Freewheel, Fusion 360, Glue, Green Building Studio, Heidi, Homestyler, HumanIK, i-drop, ImageModeler, Incinerator, Inferno, InfraWorks, InfraWorks 360, Instructables, Instructables (stylized robot design/logo), Inventor, Inventor HSM, Inventor LT, Kynapse, Kynogon, LandXplorer, Lustre, MatchMover, Maya, Maya LT, Mechanical Desktop, MIMI, Mockup 360, Moldflow Plastics Advisers, Moldflow Plastics Insight, Moldflow, Moondust, MotionBuilder, Movimento, MPA (design/logo), MPA, MPI (design/logo), MPX (design/logo), MPX, Mudbox, Navisworks, ObjectARX, ObjectDBX, Opticore, Pipeplus, Pixlr, Pixlr-automatic, Productstream, Publisher 360, RasterDWG, RealdWG, ReCap, ReCap 360, Remote, Revit LT, Revit, RiverCAD, Robot, Scaleform, Showcase, Showcase 360 ShowMotion, Sim 360, SketchBook, Smoke, Socialcam, Softimage, Sparks, SteeringWheels, Stitcher, Stone, StormNET, TinkerBox, ToolClip, Topobase, Toxik, TrustedDWG, T-Splines, ViewCube, Visual LISP, Visual, VRED, Wire, Wiretap, WiretapCentral, XSI.

All other brand names, product names or trademarks belong to their respective holders.

## **Disclaimer**

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. "AS IS." AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

### 연락처:

문서	CLIK Buttons 시작하기
주소	Autodesk Scaleform Corporation 6305 Ivy Lane, Suite 310 Greenbelt, MD 20770, USA
웹사이트	<a href="http://www.scaleform.com">www.scaleform.com</a>
이메일	<a href="mailto:info@scaleform.com">info@scaleform.com</a>
직통전화	(301) 446-3200
팩스	(301) 446-3199

# 목차

<b>1. 개요 .....</b>	<b>1</b>
<b>2. 버튼 .....</b>	<b>2</b>
2.1 시작하기 .....	2
2.2 버튼 레이어 .....	9
2.3 버튼 상태.....	11
2.3.1 Up.....	7
2.3.2 Over.....	7
2.3.3 Down.....	7
2.3.4 Disabled.....	8
2.4 버튼 전달인자.....	9
2.4.1 인스펙터블(Inspectables).....	9
<b>3. 토플 버튼.....</b>	<b>10</b>
3.1 버튼 상태 전환.....	19
3.1.1 Selected_Up.....	20
3.1.2 Selected_Over.....	20
3.1.3 Selected_Down .....	20
3.1.4 Selected_Disabled.....	21
<b>4. 애니메이션 버튼 .....</b>	<b>22</b>
4.1 애니메이션 버튼 전이.....	22
4.1.1 Over .....	24
4.1.2 Down.....	25
4.1.3 Release .....	25
4.1.4 Out .....	17

4.2 키보드 전이 .....	17
4.2.1 포커스 지시자 .....	17
4.2.2 KB_Down.....	18
4.2.3 KB_Release.....	19
<b>5. 애니메이션 토글 버튼 .....</b>	<b>29</b>
5.1 애니메이션 토글 버튼 전이 .....	29
5.1.1 Selected_Selecting.....	31
5.1.2 Selected_Release.....	32
5.1.3 Selected_Out.....	32
5.1.4 Selecting .....	33
5.2 키보드 전이 .....	33
5. 2.1 Selected_KB_Down .....	33
5.2.2 Selected_KB_Release.....	33
5.2.3 Selected_KB_Selecting .....	34
5.2.4 KB_Selecting .....	34
<b>6. 결론 .....</b>	<b>35</b>

## 1. 개요

스케일풀의 CLIK은 개발자로 하여금 게임용의 빠른 전면 메뉴 시제품을 개발하게 해주며, 최소한의 시간과 노력으로 최종 제품까지 완성시킬 수 있다. 또한 개발자로 하여금 디자인에 대한 빠른 반복 테스트도 가능하게 해준다. 개발자 키에는 버튼, 슬라이더, 진행 바, 드롭다운 메뉴, 텍스트 영역, 스크롤 리스트 등을 비롯한 15개의 개별 위젯과 컴포넌트를 포함하고 있다. 이 문서는 버튼에 초점을 맞출 것인데, 이는 이 컴포넌트가 CLIK 전체의 핵심이기 때문이다. 중요한 4 가지 타입의 버튼이 있는데, Button, ToggleButton, AnimatedButton, AnimatedToggleButton이다. 뒤의 3 가지는 일반 Button의 변종이다. 이들 컴포넌트와 변종들을 사용하는 방법을 차근차근 익히는 것이 CLIK 컴포넌트 학습의 첫걸음이다.

먼저 "[CLIK 시작하기](#)" 문서를 읽어둘 것을 추천한다. 여기서 설명하는 상당수의 개념은 이미 그 문서에서 소개된 것들이다.

주의 할 점은 플래시에서 같은 작업 물을 만드는데 여러 가지의 방법이 있다는 것인데, 개발자들은 이 튜토리얼의 접근 방식보다 좋은 대안이 될 수 있는 작업흐름을 찾아서 적용할 수도 있다.

## 2. 버튼(Button)

버튼은 슬라이더, 진행바, 스크롤 바 등을 포함해서 CLIK 컴포넌트의 가장 기본을 구성하는 핵심 컴포넌트(혹은 위젯)이다. 다른 많은 컴포넌트들이 기본 버튼 컴포넌트로부터 기능을 상속받으며, 대부분의 유저 인터페이스(UI)도 버튼의 기능을 확장하는 것이다. 핵심 버튼 컴포넌트는 애니메이션이 되지 않는 단순 버튼이다. 버튼은 핵심적인 행동만 처리하는데, 이들은 마우스 지나가기(mouseover), 마우스 클릭(mouse click), 키보드 눌림(keyboard presses), 게임 컨트롤러 눌림(game controller presses)이다.

핵심 버튼 컴포넌트의 가능한 사용 예는 다음과 같다.

- 메뉴 버튼
  - OK/취소/적용
  - 화면 선택
- 게임 콘솔에서 컨트롤 버튼
- (미니게임 버튼)

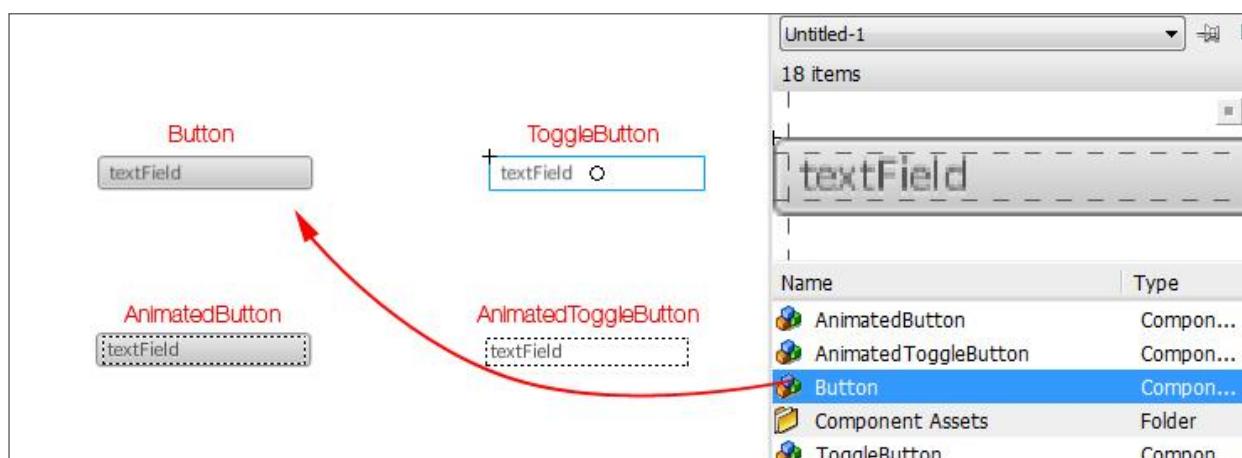
이들 버튼 컴포넌트와 기타 변종들은 다음 장에 나오는 동일한 Button.as 액션스크립트 클래스를 사용한다. 이 코드에는 버튼이 작동하는 필요한 모든 것들이 담겨있다. 버튼을 사용하기 위해서 반드시 이 스크립트를 읽어볼 필요는 없지만, 가능하다면 한번쯤 일독(一讀)하기를 권한다.

### 2.1 시작하기

1. 시작하려면 Adobe® Flash® 로 *C:/Program Files/Scaleform/GFx SDK 4.4/Resources/AS2/CLIK/components/CLIK\_Components.fla* 경로에 있는 CLIK\_Components.fla 파일을 연다.

Mac에서의 이 컴포넌트 파일은 Scaleform 4.4 이 설치된 경로의 *scaleform\_gfx\_4.4\_macos/Resources/AS2/CLIK/components* 안에서 찾을 수 있다.

2. CTRL 키를 누른 채로 마우스 좌측 버튼으로 다음 파일들을 라이브러리 창에서 클릭한다.
  - a. *Button*
  - b. *ToggleButton*
  - c. *AnimatedButton*
  - d. *AnimatedToggleButton*
3. CTRL 키를 빼고, *Button*에서 마우스 우측 버튼을 누른다. 이때 4개의 버튼들이 선택되어 있어야 한다.
4. *Copy*선택
5. CTRL+N 키를 누르거나 File→New를 선택해서 새로운 문서를 만든다.
6. 새로운 플레이시 문서의 라이브러리 창의 빈 부분에 마우스 우측버튼 클릭
7. Paste를 선택해서 라이브러리 창에 4개의 버튼 변종 붙여넣기
8. 라이브러리 창에서 마우스 좌측 버튼으로 *Button*선택
9. 라이브러리 창의 버튼 컴포넌트를 스테이지로 끌어놓기
10. 8-9의 과정을 다른 3개의 변종 버튼에도 반복적용(*ToggleButton*, *AnimatedButton*, *AnimatedToggleButton*)



*AnimatedToggleButton)*

## 2.2 버튼 레이어

기본 버튼 컴포넌트와 3 가지 변종들(ToggleButton, AnimatedButton, AnimatedToggleButton)은 5 개의 레이어로 구성되어 있다. 버튼 컴포넌트의 레이어를 보려면 스테이지에 있는 버튼을 더블 클릭한다. 일단 이번 장에서 설명하는 것은 표준 버튼이므로 다른 버튼을 더블 클릭하지 않도록 주의한다.

이들 레이어는

- *actions*: 액션 스크립트 문을 놓는데 사용
- *labels*: 버튼의 상태를 나타내는 시각 신호로 사용됨(2.3 장에서 설명)
- *textField*: 실시간으로 버튼에 출력될 텍스트
- *button*: 버튼의 다양한 상태에 대하여 사용될 그래픽 이미지

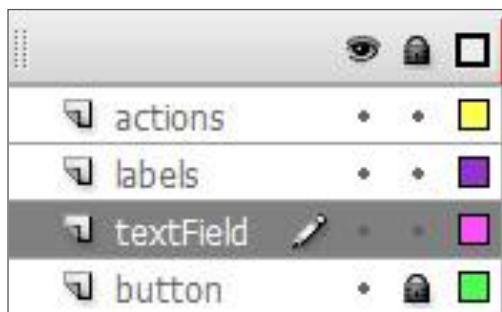


그림 2: 버튼 레이어

이들 레이어가 반드시 필요한 것은 아니다. 위치를 바꾸거나, 이름을 바꾸거나, 합치거나, 삭제할 수도 있다. 버튼은 UI 아티스트가 선택한 레이어로 구성할 수 있는 것이다. 이런 구성으로 하는 것을 추천하는 것이다. 하지만, *actions* 레이어의 액션스크립트와, *labels* 레이어의 keyframes 가 적절하게 설정되어 있지 않으면 버튼이 정상적으로 작동하지 않을 것이다. 마찬가지로, 상태에 따른 그래픽 설정이 타임라인의 적절한 위치에 *labels* 키프레임으로 설정되어 있어야만 한다. 따라서, UI 아티스트가 버튼 컴포넌트가 어떻게 작동하는지에 대한 보다 나은 이해를 하기 전까지는

수정하지 않고 그냥 사용할 것을 권장한다.

## 2.3 버튼 상태

각각의 CLIK 컴포넌트는 상태(state), 혹은 변이(transition, 애니메이션) 되는 버튼일 경우에 해당하며 나중에 자세히 설명할 예정)들로 구성되어 있다. 상태는 사용자 상호작용이나 코드에 의해서 설정할 수 있다. 이러한 상태는 시각적 연출도 포함하는데, 이러한 연출로 컴포넌트의 상태간 차이를 보여주게 된다. 표준 버튼은 4 가지 상태만 갖는다. 각 상태의 위치는 버튼 타임라인의 Label 레이어에서 키프레임으로 나타낸다. 이들과 연관된 그래픽 데이터는 같은 위치의 State 와 Border 에 위치하게 된다.

### 핵심 버튼 상태

- *up*
- *over*
- *down*
- *disabled*

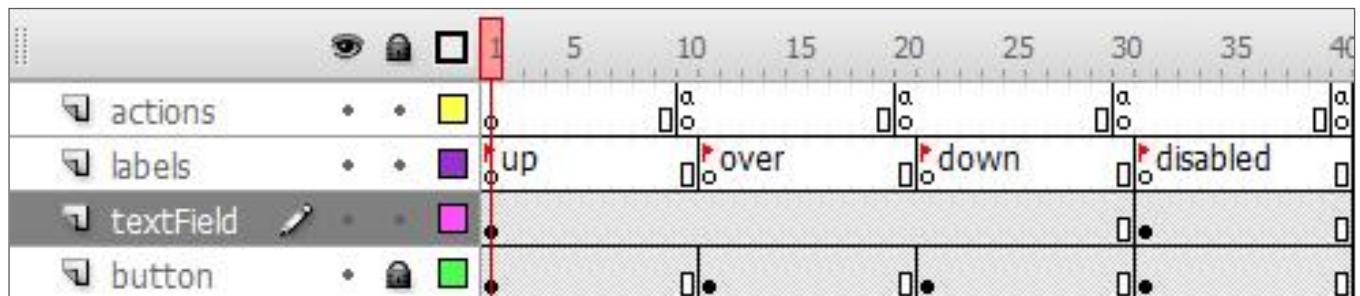


그림 3 : Label 레이어에 상태에 따라 깔끔하게 명명된 버튼 타임라인



그림 4: 버튼 상태의 그래픽 와이어프레임 연출

### **2.3.1 Up**

Up 상태란 버튼이 정상적으로 “눌고”있는 상태다. 사용자가 버튼과 아무런 상호작용을 일으키고 있지 않은 상태로서, 기본적인 상태라고 할 수 있다.

**발생:** 버튼의 기본 상태. 아무런 상호작용 발생 없음.

**사용:** 버튼을 정상 상태로 표현

### **2.3.2 Over**

over 상태는 버튼이 하이라이트 된 상태다. 사용자가 마우스 커서를 버튼 위에 놨을 때 발생한다. 커서가 버튼에서 벗어나면 up 상태로 돌아간다.

**발생:** 사용자가 마우스 커서를 버튼 위로 옮겼을 때, 혹은 마우스 입력, 키보드 화살표키, 게임 컨트롤러 방향 패드 등에 의해서 포커스를 가졌을 때(포커스는 2.3.1.1 에서 설명),

**사용:** 마우스가 버튼 위에 있거나 포커스를 가진 상태로 표현

### **2.3.3 Down**

Down 상태는 사용자가 마우스 우측버튼, 키보드의 엔터 키, 혹은 적절한 게임 컨트롤러로 버튼을 눌렀을 때 발생하는 상태다. 버튼 컴포넌트는 마우스, 키보드, 게임 컨트롤러 버튼이 눌려있는 상태에서는 계속 down 상태로 있다. 눌림 상태가 해제되면 over 상태로 돌아간다.

기본적으로 버튼 위에서 마우스 버튼이 눌리면 포커스를 받는다. 일단 포커스를 받게 되면 버튼이 “눌고”있는 상태에서도 up 상태로 돌아가지 않고 over 상태에 있게 된다. 게다가 마우스가 더 이상 버튼 위에 있지 않아도 그 상태를 유지한다. 즉, 다른 컴포넌트로 포커스를 이동시키지 않는 한 버튼의 기본 상태가 over 로 유지된다는 것이다. 사용자가 마우스 커서를 버튼이 눌린 상태에서 버튼의 위치를 벗어나게 되면 버튼은 over 상태로 돌아가고 여전히 포커스를 갖고 있게 된다.

**발생:** 버튼 컴포넌트 위에서 마우스 좌측 버튼, 키보드의 엔터 키, 적절한 컨트롤러 버튼을 눌렀을 때

**사용:** 버튼이 클릭되었음을 표현

### 2.3.3.1 포커스(Focus)

포커스를 가진 컴포넌트는 모든 키보드와 컨트롤러 이벤트의 타겟이 된다. 이 말은 버튼이 포커스를 갖게 되면 엔터 키를 눌렀을 때 오직 그 버튼에만 영향을 미친다는 것이다. 포커스는 마우스 좌측버튼이나 키보드의 화살표를 사용해서 주변의 컴포넌트로 이동시킬 수 있다. 게임 컨트롤러의 방향 패드가 키보드의 화살표키에 맵핑되어 있다면 이것도 동일하게 포커스를 이동시킨다. 또한, 포커스는 액션스크립트 코드에 의해서 이동할 수도 있다.

### 2.3.4 Disabled

disabled 상태는 버튼이 사용불가인 상태를 나타낸다. 사용불가 된 버튼은 누를 수도, 포커스를 받을 수도 없으며 mouseover 이벤트에 반응하지도 않는다. 버튼은 *Component Inspector* *Parameters* 탭에서 사용불가 상태로 만들거나, 액션스크립트의 다음 코드를 사용해도 똑같이 사용불가 상태로 만들 수 있다.

```
buttonInstanceName.disabled = true;
```

**발생:** 액션스크립트나 *Parameters* 탭에서 설정

**사용:** 현재 사용불가임을 표현

## 2.4 버튼 전달인자

버튼 컴포넌트는 전달인자 리스트를 가지고 있고, 이들은 *Parameters* 탭과 *Component Inspector*에서 찾을 수 있다. 이 값들은 UI 아티스트로 하여금 코드에 대한 수정 없이 기본적인 버튼 속성을 설정할 수 있도록 한 것이다. 이 값들은 다음과 같다.

- *disabled*: 사용불가 상태로 하려면 true로 설정
- *disableFocus*: 포커스를 못 받게 하려면 true로 설정
- *disableConstraints*: 레이블의 카운터 스케일링을 막으려면 true로 설정
- *label*: 여기에 텍스트를 입력하면 실행 시에 버튼에 출력됨
- *toggle*: 버튼이 토플(toggle) on/off 가능하게 하려면 true로 설정
- *visible*: 실행 시 버튼을 감추려면 false

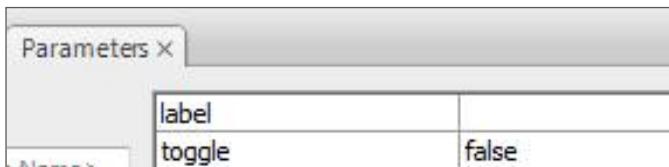


그림 5: CS 의 **Parameters** 탭에 있는 버튼 컴포넌트 전달인자

### 2.4.1 인스펙터블(Inspectables)

전달인자는 Button.as 파일을 수정해서 추가할 수 있다. 이렇게 하려면 inspectable이라는 문장을 멤버 변수 위쪽이나 get/set 함수들 위에 추가하면 된다. 예를 들어서 다음 inspectable 문은 버튼의 레이블 설정 위쪽에 추가되었기 때문에 UI 아티스트가 코드를 수정할 필요 없이, 플래시의 *Parameters* 탭에서 레이블 값을 설정할 수 있다.

```
[Inspectable(name="label", defaultValue="")]
```

Button.as 의 toggle 변수 바로 위에 다음 inspectable 문장이 있기 때문에 UI 아티스트가 *Parameters* 탭에서 true, false 를 설정할 수 있다.

```
[Inspectable(defaultValue="false")]
```

### 3. 토글 버튼(Toggle Button)

토글버튼은 체크박스 스타일의 기능을 표준 버튼에 추가한 것이다. 이렇게 하면 선택과 선택되지 않은 상태를 가지는 버튼을 만들 수 있다. 토글버튼은 매번 사용자가 클릭할 때마다 선택에서 비선택으로 전환된다(on에서 off로). 또한 액션스크립트에서 전환하는 것도 가능하다.

:토글 버튼은 다음과 같은 사용예가 있을 수 있다.

- 메뉴버튼
  - 블루 이펙트 켜기/끄기
  - 오디오 켜기/끄기
  - 고급 조명 켜기/그기
  - 자동저장 켜기/끄기
- on/off 상태가 필요한 게임콘솔 컨트롤
- on/off 상태가 필요한 미니게임 버튼

토글버튼은 표준 버튼과 동일한 액션스크립트 클래스를 사용한다. 단지 차이점은 toggle 속성이 true라는 것뿐이다. 액션스크립트 클래스가 토글버튼을 상속형태로 지원하기 때문에 제공되는 토글버튼 컴포넌트를 사용하지 않고 표준 버튼의 타임라인에서 추가적인 상태 프레임을 제공해서 사용해도 된다. 하지만, 이미 만들어진 토글버튼을 사용하는 것이 시간이 절약될 것이다. 이것은 애니메이션 되는 버튼, 애니메이션 되는 토글 버튼도 마찬가지로 적용된다.

토글 버튼이 적절하게 작동하게 하려면 CS4의 *Component Inspector Parameters* 탭(CS3는 그냥 *Parameters* 탭)의 toggle 값을 true로 설정한다. 어떤 CLIK 버튼 컴포넌트든지 toggle 값을 설정하면 토글 버튼처럼 작동하게 할 수 있다.



그림 6: toggle 인자 값을 true로 설정

### 3.1 버튼 상태 전환

표준 버튼의 상태인 *up*, *over*, *down*, *disabled* 외에 토글버튼은 다음과 같은 특별한 상태가 더 있다.

- *selected\_up*
- *selected\_over*
- *selected\_down*
- *selected\_disabled*

이들 특별한 4 가지 상태는 토글 버튼이 on(혹은 선택된) 상태를 나타내는데 사용된다. 원래 상태인 상태인 *up*, *over*, *down*, *disabled*은 "off"(즉 비 선택된) 상태를 나타낸다.

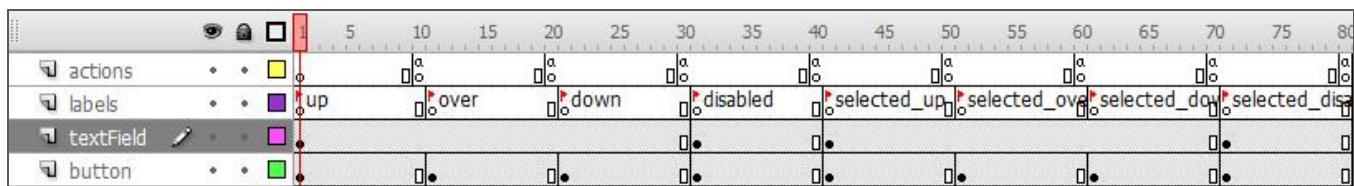


그림 8: 타임라인에 상태가 깔끔하게 레이블로 정리된 토글버튼



그림 7: 새로운 토글 버튼 상태의 그래픽 와이어 프레임 연출

### **3.1.1 Selected\_Up**

*Selected\_up* 은 포커스 없는 토글 버튼이 on 혹은 선택된 상태에서 “놀고” 있음을 나타낸다. 이 상태는 특히 선택되었음을 나타내는 그래픽적인 연출을 해줘야 하는데, 일반적으로 굵은 경계선이나 색깔을 바꾸는 등의 방법을 사용한다.

**발생:** 현재 off 이거나 비선택된 상태에서 토글 버튼을 클릭하여 선택된 혹은 on 되면 상태로 *selected\_over* 바뀜. 또한, 사용자가 마우스 클릭, 커서키, 게임 컨트롤러 등으로 포커스를 이동할 경우에는 *selected\_up* 상태가 됨.

**사용:** 포커스 없는 토글 버튼을 선택된 혹은 on 상태로 표현

### **3.1.2 Selected\_Over**

*Selected\_over* 는 사용자가 마우스 커서를 토글된 버튼위로 이동하거나 화살표 키, 게임 컨트롤러를 사용하여 포커스가 없는 “on”, “선택된” 상태의 토글 버튼으로 이동하면 발생한다. 마우스가 버튼을 벗어나거나 키보드, 게임 컨트롤러가 버튼을 벗어나면 *selected\_up* 상태로 돌아간다.

**발생:** 사용자가 “on”이나 “선택된” 상태의 토글버튼 위로 마우스를 이동하거나 화살표 키, 게임 컨트롤러로 포커스를 버튼으로 이동했을 때

**사용:** 포커스를 가지고 “on” 혹은 “선택된” 상태의 토글버튼을 표현

### **3.1.3 Selected\_Down**

*Selected\_down* 은 사용자가 “on”, “선택된” 상태의 토글버튼을 마우스 좌측버튼, 엔터 키, 적절한 게임 컨트롤러로 눌렸을 때 발생한다. 마우스나 키보드, 컨트롤러가 계속 눌려진 상태라면 *Selected\_down* 상태를 유지하고 있게 된다. 토글버튼을 누른다는 것은 선택된 상태를 바꾼다는 것이므로 버튼은 *over* 상태로 바뀔 것이다. 즉, “off”, “비 선택된” 상태를 말한다.

"off" 표준버튼과 마찬가지로 선택 된 토글버튼을 누르면 포커스를 그쪽으로 바꾼다. 버튼이 on 상태였으면 *selected\_over*가 되고, off 였으면 *over*로 바뀐다. 마우스 버튼이 눌린 채로 커서가 버튼 컴포넌트에서 벗어나고, 그 이후에 마우스 버튼이 풀어지게 되면 버튼 컴포넌트는 on 상태였을 경우에는 *selected\_over* 가 되고, off 였을 때는 *over* 가 된다.

**발생:** 사용자가 토글버튼에 포커스가 있는 상태에서 그 위에 커서를 갖다 놓고 마우스 좌측버튼을 누르거나 엔터키, 게임 컨트롤러 버튼을 누른다.

**사용:** 포커스가 있는 토글버튼이 on 되어 있는 상태 표현

### 3.1.4 Selected\_Disabled

*selected\_disabled* 상태는 표준 버튼의 *disabled* 상태와 비슷하다. 사용불가 된 버튼을 on 상태로 표현한다. 사용불가 된 토글버튼은 누를 수도, 포커스를 받을 수도, 마우스가 위에 와도 아무런 반응을 보울 수도 없다. 토글 버튼은 *Parameters* 탭이나 액션스크립트의 다음 코드를 사용해서 사용불가로 만들 수 있다.

```
buttonInstanceName.disabled = true;
```

**발생:** 액션스크립트나 *Parameters* 탭에서 설정

**사용:** on 된 상태로 사용불가인 버튼 표현

## 4. 애니메이션 버튼

애니메이션 버튼은 기본적으로 표준 버튼 컴포넌트와 똑같다. 하지만, 이 버튼은 상태간에 애니메이션 되며 전이될 수 있다(up에서 over, over에서 down 등). 애니메이션 버튼을 사용하는 것은 버튼의 상태들을 타임라인 기반 애니메이션으로 변이 시키는 것으로, 표준버튼이 간단하게 각 버튼 상태를 애니메이션 효과 없이 구현한 것과 구별된다.

애니메이션 버튼 컴포넌트는 다음과 같은 사용예가 있을 수 있다.

- 애니메이션 변이가 있는 메뉴버튼
  - OK/취소/적용
  - 화면 선택
- 애니메이션 변이가 되는 게임콘솔 내의 제어버튼
- 애니메이션 변이가 되는 미니게임 버튼

표준 버튼과 애니메이션 버튼의 유일한 실질적 차이는 비쥬얼이다. 디자이너가 버튼의 상태와 상태간에 애니메이션 효과를 주길 원한다면 애니메이션 버튼이 적당한 선택이다.

애니메이션 버튼에는 추가적인 코드가 필요 없다. CLIK 버튼 클래스는 이 문서에서 설명하는 모든 변종들의 기능을 제공한다.

### 4.1 애니메이션 버튼 전이

애니메이션 버튼은 상태를 나타내는 키프레임을 나타내는데 있어서 다른 용어를 사용한다. 상태 자체를 나타내는 것이 아니라 각 키프레임은 두 가지 상태의 애니메이션 전이를 나타낸다. Up, down, over는 여전히 상태로서의 의미를 갖고 있지만, 애니메이션 버튼이 유사한 상태로 재순환되는 이름을 나타내기 위한 목적으로 사용된다.

*up, down, over, selected\_up, selected\_down, selected\_over* 를 버튼이 전이되는 상태라고 생각하자. 이렇게 하면 견고하게 애니메이션 되는 버튼 컴포넌트에서 수없이 많은 다른 변이들의 복잡함을 이해하는데 도움이 된다.

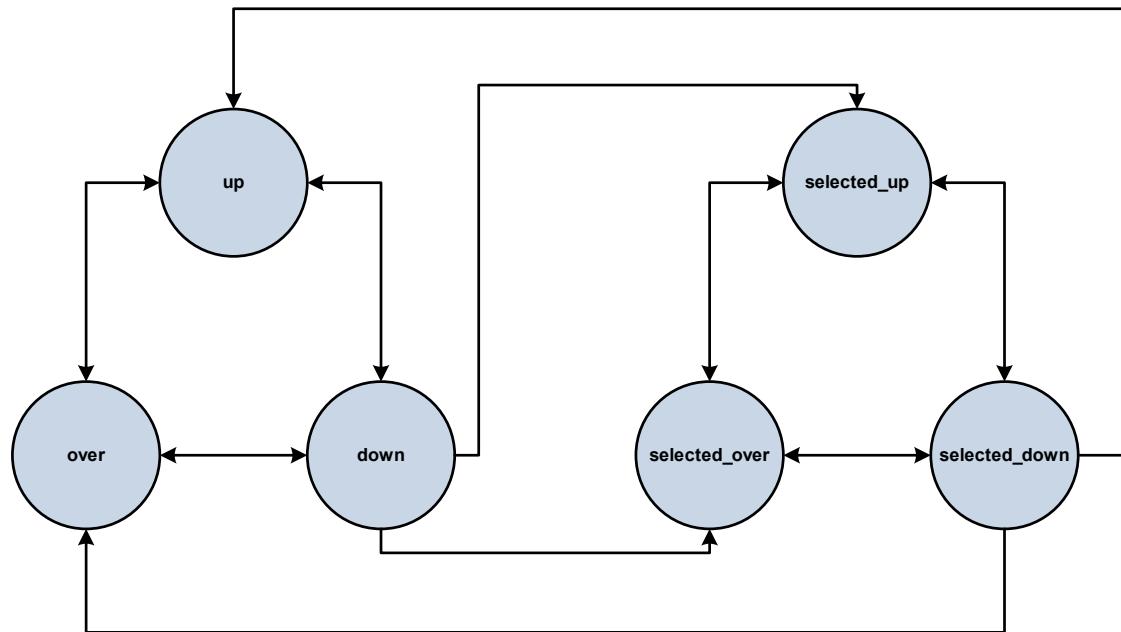


그림 9: 애니메이션 버튼 상태 변이

상태간에 고전적인 애니메이션을 하려면 마지막 키프레임을 최종 상태의 모습으로 설정한다. 그리고 나서 첫번째 프레임과 마지막 프레임의 중간 프레임(tween)을 생성한다. 예를 들어서, over 전이라면, 첫번째 애니메이션 키프레임은 up 상태에 가까운 그래픽일 것이다. 그리고 마지막 키프레임은 실제 over 상태를 나타내는 그래픽이 될 것이다.

표준 버튼 전이인 *up, over, down, disabled* 외에 애니메이션 버튼은 다음과 같은 특별한 전이도 포함한다.

- *release*
- *out*
- *kb\_down*
- *kb\_release*

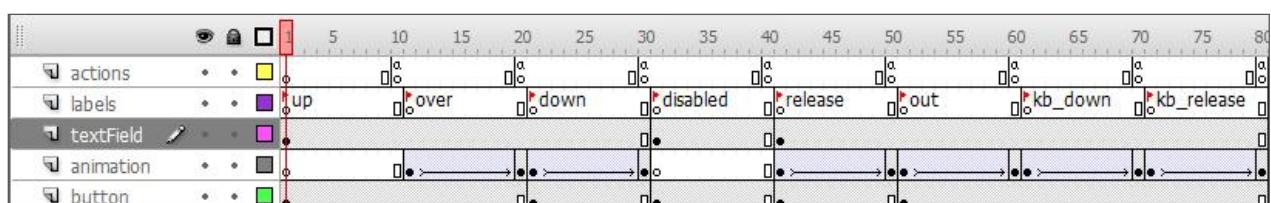


그림 10: 타임라인에 변이가 깔끔하게 레이블로 정리된 애니메이션 버튼

애니메이션 버튼의 over 와 down 상태는 표준버튼에 비해서 새로운 기능을 가지고 있다. 정적 이미지 상태가 아니라, 정적 상태를 나타낼 때 애니메이션 변이를 사용한다.

#### 4.1.1 Over



그림 11: 새로운 애니메이션 버튼 상태의 그래픽 와이어 프레임 표현

over 는 애니메이션 버튼에서 살짝 다르게 기능한다. 버튼의 up 상태에서 over 상태로 전이하는 것을 애니메이션 전이로 출력하는데 사용된다.

**발생:** 사용자가 마우스 커서를 버튼위로 옮기거나 화살표 키나 게임 컨트롤러를 사용해서 버튼으로 포커스를 옮길 때

**사용:** 마우스 커서가 버튼 위에 있거나 포커스를 가지고 있을 때를 표현

#### 4.1.2 Down

Down 은 over 상태에서 down 으로 애니메이션 전이되는 것을 출력할 때 사용된다.

**발생:** 버튼 위에 있을 때 사용자가 마우스 좌측버튼, 키보드, 게임컨트롤러를 조작하여 버튼을 눌렀을 때

**사용:** 클릭된 버튼 표현

### 4.1.3 Release

*Release* 란. 애니메이션 버튼과 애니메이션 토글 버튼에만 있는 독특한 상태다. down 상태에서 over 상태로 버튼을 애니메이션 전이 시키는 것을 출력할 때 사용된다. 이 변이(over 상태)의 마지막 프레임은 포커스를 가진 상태를 보여주는데 사용된다.

**발생:** 사용자가 마우스 좌측버튼이나 엔터키, 게임 컨트롤러 버튼을 클릭했다가 뗐을 때. 혹은 버튼 위에 있는 상태에서 버튼을 누른 다음 커서를 버튼에서 벗어나게 했을 때.

**사용:** 마우스 좌측버튼, 엔터키, 동등한 컨트롤러 버튼이 떼어진 (포커스를 가진) 애니메이션 버튼 표현

### 4.1.4 Out

*Out* 은 over 상태에서 up 상태로 애니메이션 전이되는 것을 출력하는 용도로 사용됨.

**발생:** 사용자가 버튼으로부터 마우스를 벗어나게 할 때

**사용:** 마우스 커서가 버튼 경계를 벗어난 포커스 없는 버튼 표현

## 4.2 키보드 전이

키보드 전이는 하나의 상태에서 다른 상태로 애니메이션 전이될 때 마우스가 아니라 키보드나 게임 컨트롤러를 사용해서 전이되거나 분할된 포커스 지시자 무비 클립이 사용될 때 사용된다. 포커스 지시자가 사용 되면 키보드 용으로 2 개의 추가적인 변이가 존재한다. 이는 kb\_down 과 kb\_release 며, 4.2.2 와 4.2.3 에서 설명할 것이다.

### 4.2.1 포커스 지시자

기본적으로 버튼 컴포넌트는 release 변이의 마지막 프레임을 포커스 상태를 출력하는데 사용한다. 하지만, 키보드나 게임 컨트롤러를 입력기로 사용할 때는 포커스 상태를 나타내기 위해서 별도의

그래픽을 지시자로 사용하는 것이 좋을 때가 있다. CLIK 버튼은 내부적으로 이러한 기능을 지원하는데, 버튼 무비 클립에 포커스 지시자 무비 클립이 내부적으로 포함되어 있으면 된다. 포커스 지시자 무비 클립의 인스턴스 명은 반드시 focusIndicator 여야 하며, hide 와 show 라는 2 개의 키프레임 명으로 순서대로 구성되어 있어야 한다. *show*프레임은 포커스 상태를 나타내므로 경계면을 빛나게 하는 등의 처리를 하고, *hide*프레임은 포커스가 없는 상태를 나타내므로 추가적인 그래픽 처리를 안하면 될 것이다. 포커스 지시자 무비 클립은 다음과 같은 단계를 따른다.

- 1.버튼의 타임라인에 새로운 레이어 생성
- 2.레이어 명은 FocusIndicator
- 3.레이어의 전체 프레임 수를 버튼 타임라인의 마지막 프레임으로 확장
- 4.*FocusIndicator* 레이어에 새로운 무비클립 생성(키보드 포커스 상태)
  - a. 예를 들어, 내부가 빈 상자를 그리고 경계선을 붉은색으로 처리
  - b.스테이지의 사각형을 우측버튼으로 클릭해서 *Convert to Symbol*.선택
  - c. 무비클립의 이름을 정하고 OK 클릭
- 5.*Properties* 패널에서 인스턴스 명을 focusIndicator 로
- 6.*focusIndicator* 무비클립을 더블클릭해서 타임라인으로 들어감
- 7.*Layer1* 타임라인의 frame 2 에서 두번째 키프레임 추가
- 8.*Properties* 패널에서 keyframe 1 을 선택하고 hide 로 명명
- 9.액션 패널(F9)을 열어서 frame 1 에 다음 코드 삽입 : `stop();`
10. keyframe 2 을 선택하고 show 로 명명
11. 액션스크립트 패널에서 frame 2 에 다음 코드 삽입 : `stop();`
12. *show*프레임에서 붉은색 경계선이 나오고, *hide*에서 없어지는 것을 확인

#### 4.2.2 KB\_Down

키보드 이벤트는 포커스가 있을 때만 버튼에게 전달된다. 분할된 포커스 지시자가 사용되지 않는다면 버튼의 포커스 상태는 over 상태(*release* 전이의 마지막 프레임)로 돌아가 것이다. 분할

포커스 지시가 사용된다면 포커스가 있는 상태에서도 버튼이 up 상태일 것이므로 up에서 down으로의 전이가 필요하다.

*Kb\_down*은 up 상태에서 down으로 전이되는 것을 표현할 때 사용된다.

**발생:** 사용자가 엔터 키나 적절한 게임 컨트롤러로 포커스 된 버튼을 눌렀을 때. 단 이 버튼은 분할 포커스 지시자 무비 클립이 있음

**사용:** 분할 포커스 지시자가 있는 버튼이 눌렸을 때를 표현

#### 4.2.3 KB\_Release

*Kb\_release*는 down에서 up으로 애니메이션 전이되는 것을 출력할 때 사용됨.

**발생:** 사용자가 키보드나 게임 컨트롤러에서 버튼을 떼었는데, 이 버튼이 포커스가 있으며 분할 포커스 지시자 무비 클립을 가진 버튼인 경우. 또한, 분할 포커스 지시자가 사용되었으며 사용자가 마우스를 끌어서 벗어났을 때, 혹은 화살표 키나 게임 컨트롤러를 (엔터 키를 떼기 전에) 다른 버튼으로 옮겨갔을 때

**사용:** 분할 포커스 지시자를 가진 포커스 된 버튼이 떼어졌을 때를 표현

## 5. 애니메이션 토글 버튼

애니메이션 토글 버튼은 토글버튼과 애니메이션 버튼을 합성한 혼혈 컴포넌트다. 애니메이션 토글 버튼은 타이라인에 기반해서 다양한 상태가 애니메이션 변이 되는데 반해서, 표준 토글 버튼은 애니메이션 효과 없이 단순히 상태만을 출력한다.

이 버튼 타입은 4 개 중에서 가장 종합적이다. 일단 앞서 설명한 3 가지 타입의 기능을 모두 포함하고 있을 뿐만 아니라, 자신만의 특징적인 변이도 가지고 있다. 앞서 언급한 애니메이션 버튼처럼 재순환되는 *selected\_up*, *selected\_down*, *selected\_over* 변이를 가지고 있다.

애니메이션 토글 버튼 컴포넌트는 다음과 같은 사용예가 있을 수 있다.

- 애니메이션 변이가 있으며 on, off 토글 되는 메뉴 버튼
  - 블루 비쥬얼 효과 켜기/끄기
  - 오디오 켜기/끄기
  - 고급 조명 켜기/끄기
  - 자동저장 켜기/끄기
- on/off 상태가 필요한 게임콘솔 컨트롤
- on/off 상태가 필요한 미니게임 버튼

### 5.1 애니메이션 토글 버튼 전이

애니메이션 토글 버튼은 앞서 애니메이션 버튼에서 키프레임에 대하여 설명했던 것과 동일한 용어를 사용한다. 각 키 프레임은 두가지 상태 간의 애니메이션 전이를 나타낸다.

애니메이션 토글 버튼은 표준 버튼, 토글 버튼, 애니메이션 버튼 들의 상태 외에 다음과 같은 8 가지의 특별한 상태를 추가적으로 포함한다.

- *selected\_selecting*
- *selected\_release*
- *selected\_out*
- *selecting*
- *selected\_kb\_down*
- *selected\_kb\_release*
- *selected\_kb\_selecting*
- *kb\_selecting*

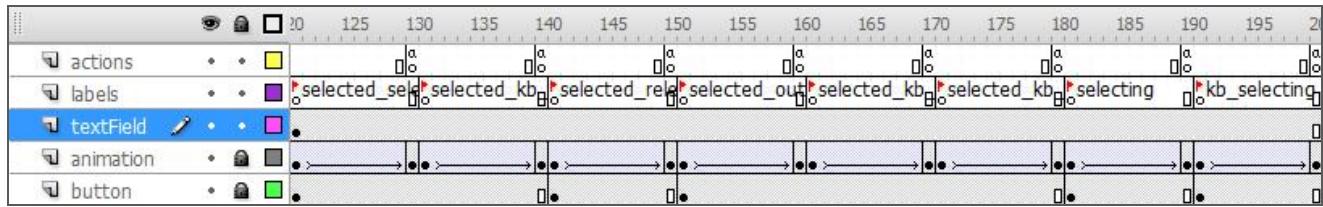


그림 12: 타임라인에 새로운 전이가 깔끔하게 정리된 애니메이션 토클 버튼

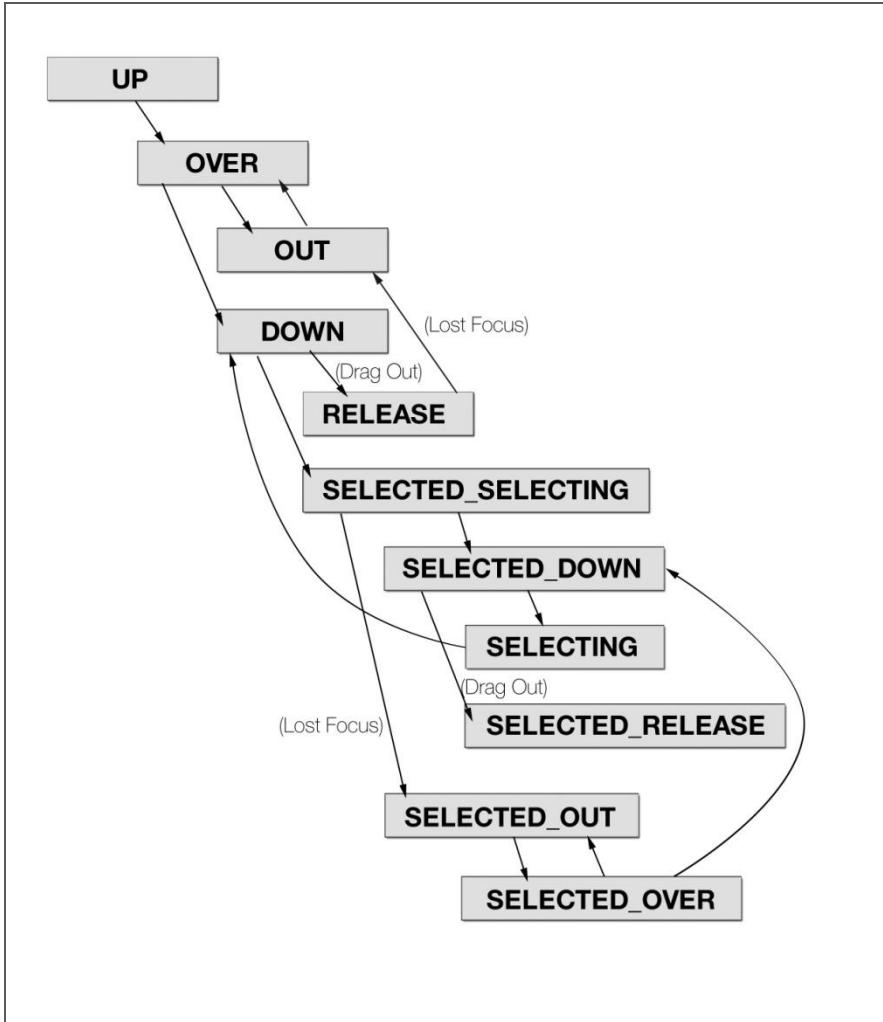


그림 13: 애니메이션 토글 버튼 전이(키보드 전이는 불포함)

### 5.1.1 Selected\_Selecting

*Selected\_selecting* 은 *down* 상태에서 “selected”나 “on”으로 포커스 된 버튼 상태(*selected over*)를 나타낼 때 사용됨. “unselected”에서 “selected”로의 토글 전이임.

**발생:** 사용자가 현재 on 이나 selected 가 아닌 애니메이션 토글 버튼을 클릭 했을 때. 버튼은 on 이나 selected 로 바뀜

**사용:** 포커스를 가지고 있으나 포커스 지시자는 없는 selected 로 토글된 상태를 나타내는데 사용됨.

### **5.1.2 Selected\_Release**

*selected\_down(버튼이 눌렸고 현재 선택된 상태)에서 "selected"된 포커스 상태의 버튼(selected\_over)을 애니메이션 전이할 때 사용*

**발생:** 사용자가 selected 버튼을 클릭하고나서 마우스 커서가 버튼에서 벗어났을 때. "selected"에서 "unselected"로 취소됨

**사용:** "selected"에 포커스를 가진 상태에서 취소된 상태를 나타냄

### **5.1.3 Selected\_Out**

*selected\_down(포커스를 갖고 selected 된 상태) 상태에서 포커스가 없는 selected 된 상태로 전이할 때 사용됨(selected\_up)*

**발생:** 사용자가 좌측 마우스 버튼으로 다른 컴포넌트를 클릭하거나 화살표 키, 게임 컨트롤러 등으로 다른 컴포넌트를 선택했을 때. selected 되었으나 포커스는 없는 버튼상태

**사용:** 포커스 없으나 selected 된 상태 표현

### **5.1.4 Selecting**

*selected\_down에서 over 상태로 전이 할 때 사용됨. selected에서 unselected로 상태 변화.*

**발생:** 현재 selected 된 버튼을 클릭했을 때. 버튼은 unselected로 바뀜

**사용:** unselected 상태를 나타내지만 포커스를 가지고 있음을 표현

## **5.2 키보드 전이**

애니메이션 토글 버튼은 애니메이션 버튼에서 설명한 것과 동일한 설정이 필요하다

### **5.2.1 Selected\_KB\_Down**

*selected\_kb\_down* 는 *selected\_up* 에서 *selected\_down* 으로의 전이를 나타낼 때 사용된다.

**발생:** selected 되고, 포커스를 가졌으며, 분할 포커스 지시자 무비클립을 가진 버튼이 키보드의 엔터 키나 게임 컨트롤러의 적절한 키가 눌렸을 때

**사용:** 엔터 키나 게임 컨트롤러 버튼이 눌려서 selected 된 애니메이션 토글 버튼 표현

### **5.2.2 Selected\_KB\_Release**

*selected\_down*에서 포커스가 있는 *selected\_up* 상태의 버튼으로 애니메이션 전이 될 때 사용.

**발생:** selected 되고, 포커스를 가졌으며, 분할 포커스 지시자 무비클립을 가진 버튼이 키보드의 엔터 키나 게임 컨트롤러의 적절한 키가 눌렸을 때, 혹은 마우스 좌측 버튼으로 눌렸을 때, 그리고나서 사용자가 마우스 커서를 이동해서 버튼에서 벗어 났을 때, 혹은 화살표 키나 게임 컨트롤러로 다른 버튼을 선택 했을 때

**사용:** selected 되고 포커스를 가진 애니메이션 토글 버튼이 떼어진 것을 표현

### **5.2.3 Selected\_KB\_Selecting**

down에서 *selected\_up*으로 애니메이션 전이 될 때 사용. *unselected*에서 *selected*로 상태 변화

**발생:** 사용자가 엔터키나 게임 컨트롤러 버튼으로 *unselected*로 바꿀 때.

**사용:** selected 되고 포커스가 있으며, 분할 포커스 지시자 무비 클립을 가지고 있음.

### **5.2.4 KB\_Selecting**

*selected\_down*에서 up으로 애니메이션 전이 될 때 사용. *selected*에서 *unselected*로 상태 변화

**발생:** 사용자가 엔터키나 게임 컨트롤러 버튼으로 *selected*로 바꿀 때.

**사용:** *unselected* 되고 포커스가 있으며, 분할 포커스 지시자 무비 클립을 가지고 있음.

## 6. 결론

스케일폼 CLIK 버튼 변종들은 UI 아티스트에게 보다 나은 유연성을 제공하기 위해서 디자인 되었다. 다양한 각 버튼 변종들의 변이와 상태를 익히는 것은 성공적인 CLIK 사용에서 크리티컬 한다. 각 변종들을 언제 사용할지 알게 되면 많은 시간과 노력을 절약해 줄 것이다.

- *Button*: 기반 버튼 변종
- *Toggle Button*: on/off 기능이 추가된 버튼
- *Animated Button*: 애니메이션 되는 버튼
- *Animated Toggle Button*: 토글버튼과 애니메이션 버튼 기능의 조합