



02

Callback Programming

- 01 Display Callback
- 02 Reshape Callback
- 03 Keyboard Callback
- 04 Mouse Callback
- 05 Menu Callback
- 06 Idle Callback & Double Buffering
- 07 Timer Callback

COMPUTER GRAPHICS





01

Section

Display Callback



Display callback이 자동으로 호출되는 경우

- ① 처음 Window를 열 때
- ② Window의 위치를 이동시킬 때
- ③ Window의 크기를 변경할 때
- ④ 앞 Window에 가려서 안 보이던 뒤 Window가 활성화되어 앞으로 나타날 때
- ⑤ glutPostRedisplay 함수에 의해 Event Queue에 Flag가 게시될 때



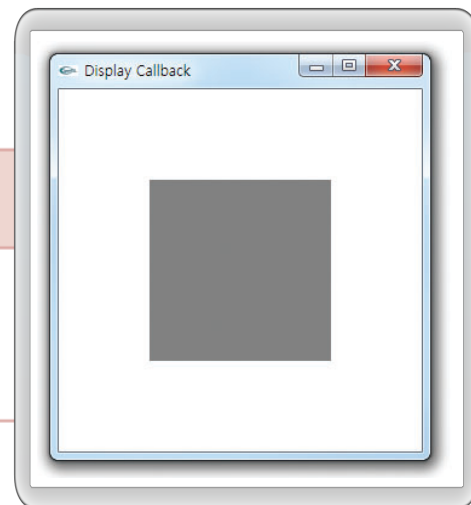
glutDisplayFunc 함수의 Prototype

```
void glutDisplayFunc ( void (*func)(void) );
```

Parameters

Help

func // 호출할 새로운 Display Callback 함수





02

Section

Reshape Callback



Reshape callback이 자동으로 호출되는 경우

- ① 처음 Window를 열 때
- ② Window의 위치를 이동시킬 때
- ③ Window의 크기를 변경할 때

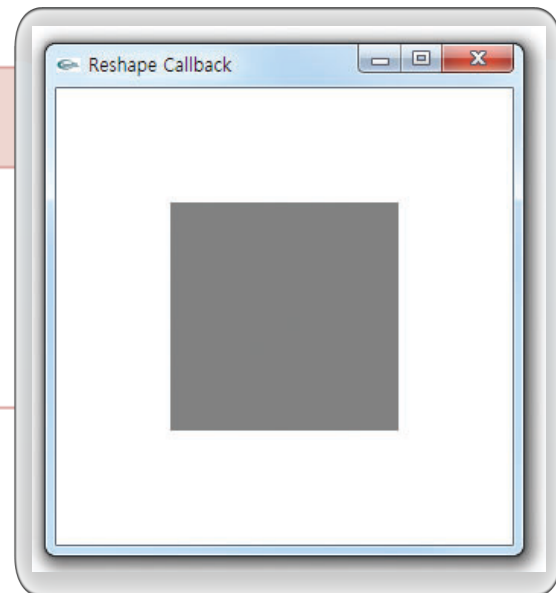


glutReshapeFunc 함수의 Prototype

```
void glutReshapeFunc ( void(*func)( int width, int height ) );
```

Parameters
Help

func // 호출할 새로운 Reshape Callback 함수
width // Window의 넓이 (단위 : Pixel)
height // Window의 높이 (단위 : Pixel)





Keyboard Callback



glutKeyboardFunc 함수의 Prototype

```
void glutKeyboardFunc ( void(*func)(unsigned char key, int x, int y) );
```

Parameters

Help

func // 호출할 새로운 Keyboard Callback 함수
key // Keyboard로부터 입력되는 ASCII 문자
x // Mouse의 GLUT 좌표계의 X축 위치
y // Mouse의 GLUT 좌표계의 Y축 위치



glutSpecialFunc 함수의 Prototype

```
void glutSpecialFunc ( void(*func)(int key, int x, int y) );
```

Parameters

Help

func // 호출할 새로운 Entry Callback 함수
key // Special Key를 위한 GLUT_KEY_* 의 상수
x // Mouse의 GLUT 좌표계의 X축 위치
y // Mouse의 GLUT 좌표계의 Y축 위치



Keyboard Callback



glutSpecialFunc 함수에서 사용 가능한 특수 key들에 대한 정의

Function Keys		Directional Keys	
#define GLUT_KEY_F1	1	#define GLUT_KEY_LEFT	100
#define GLUT_KEY_F2	2	#define GLUT_KEY_UP	101
#define GLUT_KEY_F3	3	#define GLUT_KEY_RIGHT	102
#define GLUT_KEY_F4	4	#define GLUT_KEY_DOWN	103
#define GLUT_KEY_F5	5	#define GLUT_KEY_PAGE_UP	104
#define GLUT_KEY_F6	6	#define GLUT_KEY_PAGE_DOWN	105
#define GLUT_KEY_F7	7	#define GLUT_KEY_HOME	106
#define GLUT_KEY_F8	8	#define GLUT_KEY_END	107
#define GLUT_KEY_F9	9	#define GLUT_KEY_INSERT	108
#define GLUT_KEY_F10	10		
#define GLUT_KEY_F11	11		
#define GLUT_KEY_F12	12		



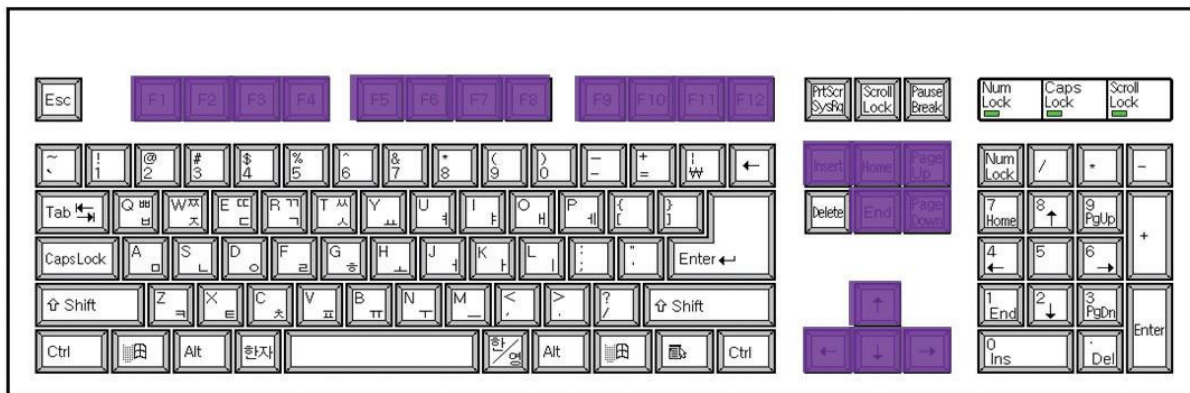
03

Section

Keyboard Callback

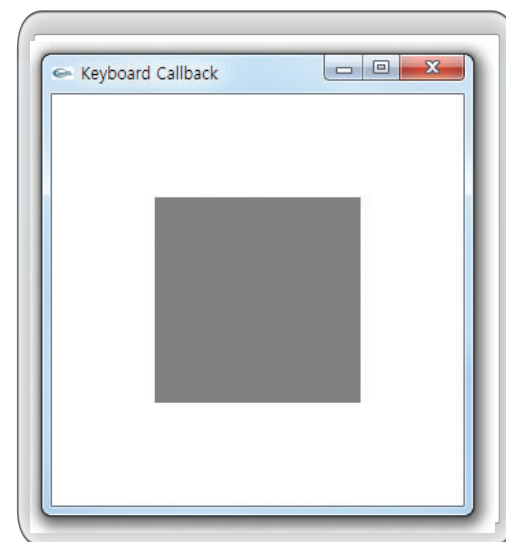


GlutSpecialFunc 함수에서 Event로 취급하는 keyboard의 key들



keyboard callback 구현 결과

코드 2-3





03

Section

Keyboard Callback



ctrl Alt, Shift 등과 함께 문자 key나 함수 key를 사용하는 경우

코드 2-4



Ctrl, Alt, Shift 등과 함께 문자키나 함수키를 사용하는 경우

```
if ( (glutGetModifiers() == GLUT_ACTIVE_CTRL) && (key == 'C') ) {  
    ...  
} else if ( (glutGetModifiers() == GLUT_ACTIVE_SHIFT) && (key == 'A') ) {  
    ...  
} else if ( (glutGetModifiers() == GLUT_ACTIVE_ALT) && (key == 'B') ) {  
    ...  
}
```




04

Section

Mouse Callback



glutMouseFunc 함수의 Prototype

```
void glutMouseFunc ( void ( *func )( int button, int state, int x, int y ) );
```

Parameters

Help

func // 호출할 새로운 Mouse Callback 함수

button // Mouse의 Button을 선택
 // GLUT_LEFT_BUTTON
 // GLUT_RIGHT_BUTTON
 // GLUT_MIDDLE_BUTTON

state // Mouse의 Click 여부 즉, GLUT_DOWN, GLUT_UP

x // Mouse의 GLUT 좌표계의 X축 위치

y // Mouse의 GLUT 좌표계의 Y축 위치



GlutMouseFunc 함수에서 사용 가능한 Mouse Button들에 대한 정의

Mouse Buttons		Mouse Button State	
#define GLUT_LEFT_BUTTON	0	#define GLUT_DOWN	0
#define GLUT_MIDDLE_BUTTON	1	#define GLUT_UP	1
#define GLUT_RIGHT_BUTTON	2		



04

Section

Mouse Callback



glutMotionFunc 함수의 Prototype

```
void glutMotionFunc ( void ( *func )( int x, int y ) );
```

Parameters

Help

func // 호출할 새로운 Motion Callback 함수

x // Mouse의 GLUT 좌표계의 X축 위치

y // Mouse의 GLUT 좌표계의 Y축 위치



GlutPassiveMotionFunc 함수의 Prototype

```
void glutPassiveMotionFunc ( void ( *func )( int x, int y ) );
```

Parameters

Help

func // 호출할 새로운 Passive Motion Callback 함수

x // Mouse의 X축 위치

y // Mouse의 Y축 위치



glutEntryFunc 함수의 Prototype

```
void glutEntryFunc ( void ( *func )( int state ) );
```

Parameters

Help

func // 호출할 새로운 Entry Callback 함수

state // Mouse Pointer가 Window 안에 있는지 밖으로 나갔는지를 감지

// GLUT_ENTERED, GLUT_LEFT



04

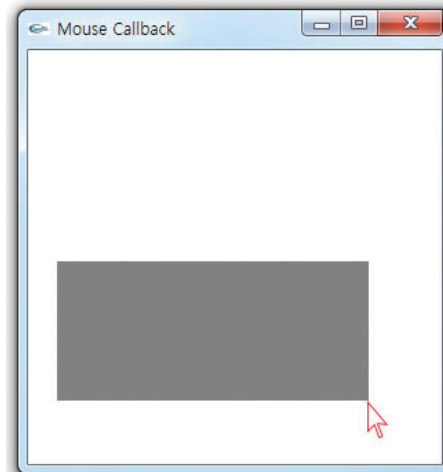
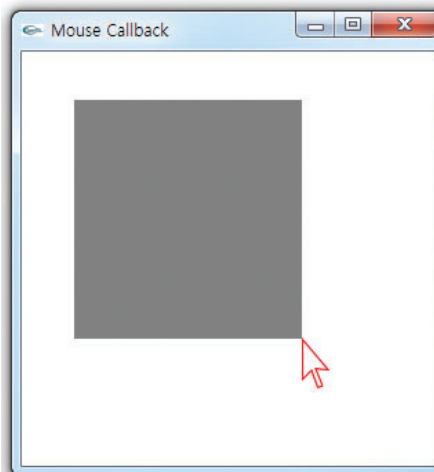
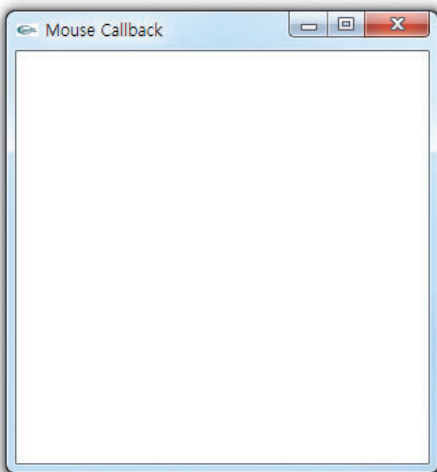
Section

Mouse Callback



Mouse callback 구현 결과

코드 2-5





04

Section

Mouse Callback



glutMouseWheelFunc 함수의 Prototype

```
void glutMouseWheelFunc ( void ( *func )( int wheel, int direction, int x, int y ) );
```

Parameters
Help

func // 호출할 새로운 Mouse Wheel Callback 함수
wheel // Mouse Wheel Number, Default : 0
direction // +1은 전방, -1은 후방
x // Mouse의 GLUT 좌표계의 X축 좌표
y // Mouse의 GLUT 좌표계의 Y축 좌표



freeglut Download

<http://freeglut.sourceforge.net/>



04

Section

Mouse Callback



Freeglut Library 설치

Classification	Folder Location	Files
Header File (* .h)	C:\Program Files\Microsoft Visual Studio\VC98\include\gl C:\Program Files\Microsoft Visual Studio 8\VC\include\gl C:\Program Files\Microsoft Visual Studio 9.0\VC\include\gl C:\Program Files\Microsoft Visual Studio 10.0\VC\include\gl C:\Program Files\Microsoft Visual Studio 11.0\VC\include\gl C:\Program Files\Microsoft Visual Studio 12.0\VC\include\gl	freeglut.h freeglut_ext.h freeglut_std.h
Library File (* .lib)	C:\Program Files\Microsoft Visual Studio\VC98\lib C:\Program Files\Microsoft Visual Studio 8\VC\lib C:\Program Files\Microsoft Visual Studio 9.0\VC\lib C:\Program Files\Microsoft Visual Studio 10.0\VC\lib C:\Program Files\Microsoft Visual Studio 11.0\VC\lib C:\Program Files\Microsoft Visual Studio 12.0\VC\lib	freeglut.lib
Dynamic Link Library File (* .dll)	C:\Windows\system32 (32 bit의 경우) C:\Windows\SysWOW64 (64 bit의 경우)	freeglut.dll



04

Section

Mouse Callback



Mouse wheel callback

코드 2-6



Mouse Wheel Callback

```
#include <freeglut.h>

.....

void MyMouseWheel(int Button_Number, int Direction, int X, int Y) {
    if (Direction > 0) {
        TopLeftX   += 1.0;
        TopLeftY   += 1.0;
        BottomRightX += 1.0;
        BottomRightY += 1.0;
    } else {
        TopLeftX   -= 1.0;
        TopLeftY   -= 1.0;
        BottomRightX -= 1.0;
        BottomRightY -= 1.0;
    }

    glutPostRedisplay( );
}

.....

int main(int argc, char** argv) {
    ...
    glutMouseWheelFunc(MyMouseWheel);
    ...
}
```



Menu Callback



glutCreateMenu 함수의 Prototype

```
void glutCreateMenu ( void( *func )( int value ) );
```

Parameters

func // 호출할 새로운 Menu Callback 함수

Help

value // Entry 값



glutSetMenu 함수의 Prototype

```
void glutSetMenu ( int id );
```

Parameters

id // Menu를 생성하기 위한 Menu의 식별자

Help



glutAddMenuEntry 함수의 Prototype

```
void glutAddMenuEntry ( char *name, int value );
```

Parameters

name // 메뉴 항목(Menu Entry)에 보여주기 위한 ASCII 문자열

Help

value // Menu Entry가 선택되어질 때 Menu의 Callback 함수를 돌려주기
// 위한 정수 값



Menu Callback



glutAttachMenu 함수의 Prototype

```
void glutAttachMenu ( int button ); ↔ glutDetachMenu ( int button );
```

Parameters

Help

button // Menu를 붙이기 위한 Button



glutAddSubMenu 함수의 Prototype

```
void glutAddSubMenu ( char *name, int menu );
```

Parameters

Help

name // Menu 항목에서 Sub Menu로 보여주기 위한 ASCII 문자열

menu // Sub Menu 항목으로 나열하기 위한 Menu의 식별자



05

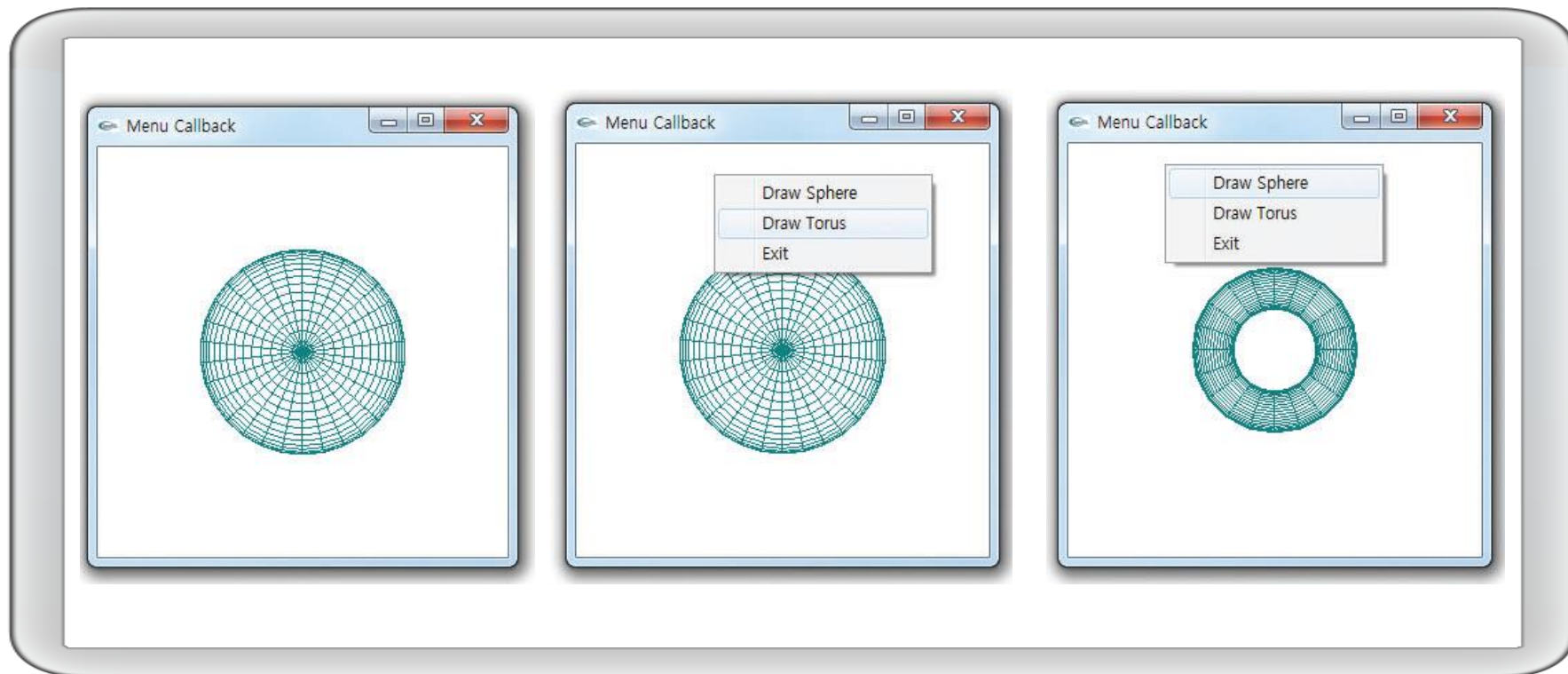
Section

Menu Callback



Menu callback 함수를 이용한 주 메뉴의 구현 결과

코드 2-7





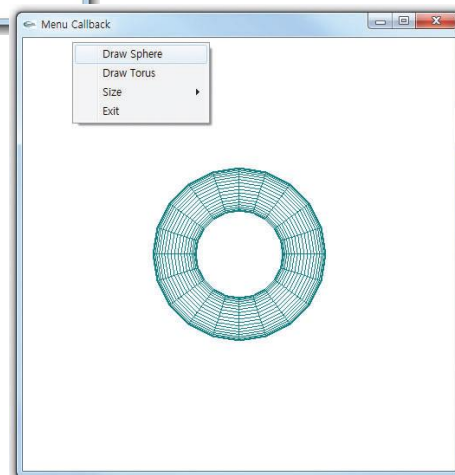
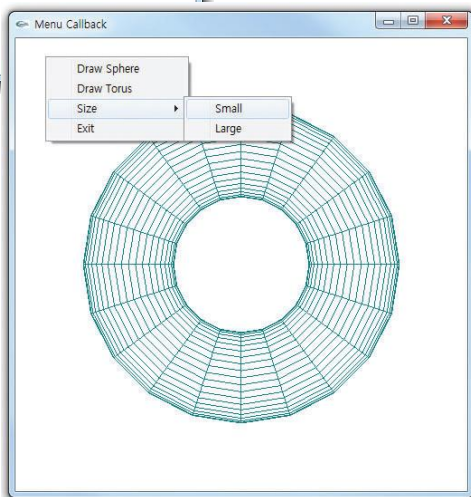
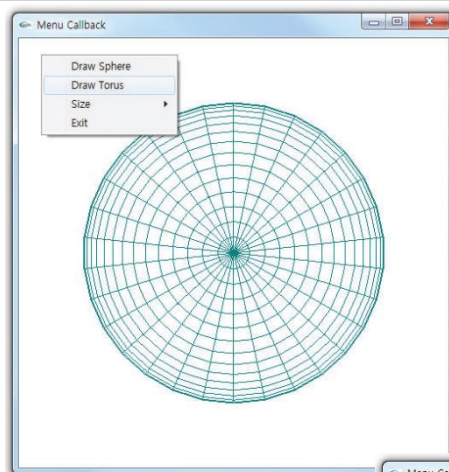
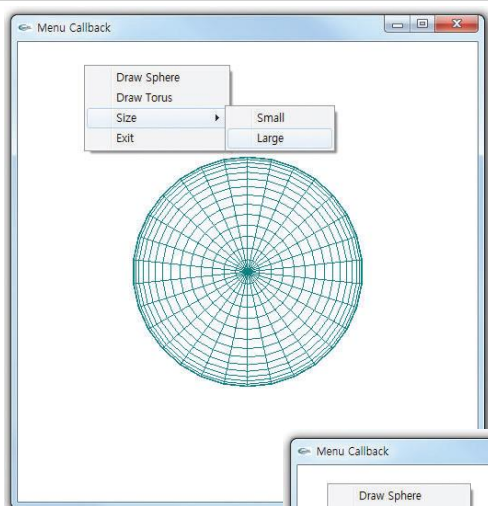
05

Section

Menu Callback



Menu callback 함수를 이용한 시브 메뉴의 구현 결과



코드 2-8





Menu Callback



서브 메뉴의 위치에 따른 구현 결과 비교

소스 코드에서 서브 메뉴의 위치	구현 결과
<pre>glutAddMenuEntry("Draw Sphere", 1); glutAddMenuEntry("Draw Torus", 2); glutAddSubMenu("Size", MySizeID); glutAddMenuEntry("Exit", 3);</pre>	<pre>Draw Sphere Draw Torus Size Exit</pre>
<pre>glutAddMenuEntry("Draw Sphere", 1); glutAddSubMenu("Size", MySizeID); glutAddMenuEntry("Draw Torus", 2); glutAddMenuEntry("Exit", 3);</pre>	<pre>Draw Sphere Size Draw Torus Exit</pre>
<pre>glutAddMenuEntry("Draw Sphere", 1); glutAddMenuEntry("Draw Torus", 2); glutAddMenuEntry("Exit", 3); glutAddSubMenu("Size", MySizeID);</pre>	<pre>Draw Sphere Draw Torus Exit Size</pre>



06

Section

Idle Callback & Double Buffering



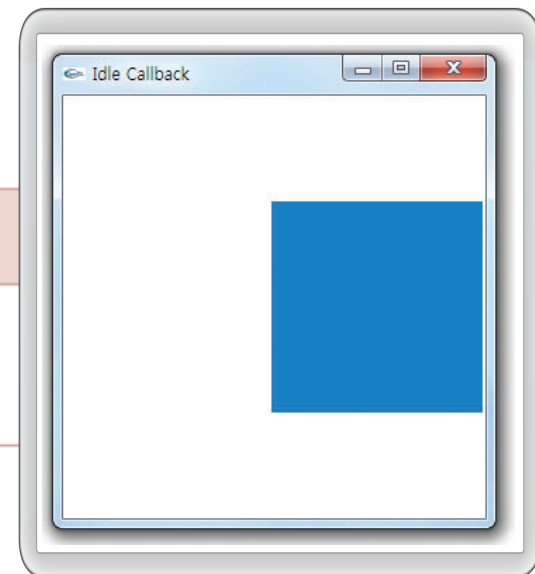
glutIdleFunc 함수의 Prototype

```
void glutIdleFunc ( void( *func )( void ) );
```

Parameters

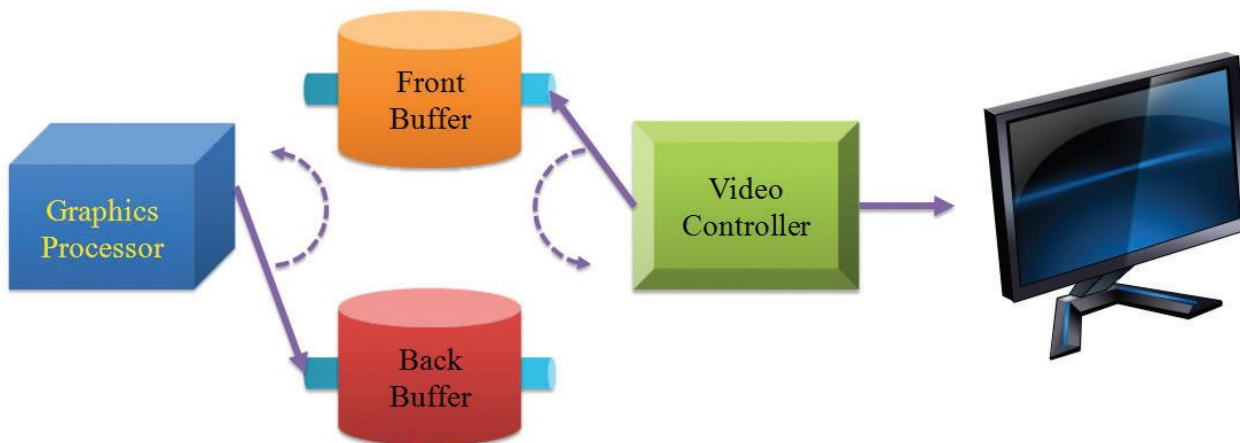
Help

func // 호출할 새로운 Idle Callback 함수



Double Buffering

코드 2-9





Timer Callback



glutTimerFunc 함수의 Prototype

```
void glutTimerFunc ( unsigned int msec, void (*func)( int value), int value );
```

Parameters
Help

msec // 호출할 새로운 Timer Callback 함수를 실행하기 위한 최소한의 시간
func // 호출할 새로운 Timer Callback 함수
value // Callback 함수가 실행되었을 때 Callback을 무시하기 위한 값
value // Callback 함수가 실행되었을 때 Callback을 무시하기 위한 값
// 혹은 Event가 발생할 때 Callback 함수에 넘겨주고 싶은 값



07

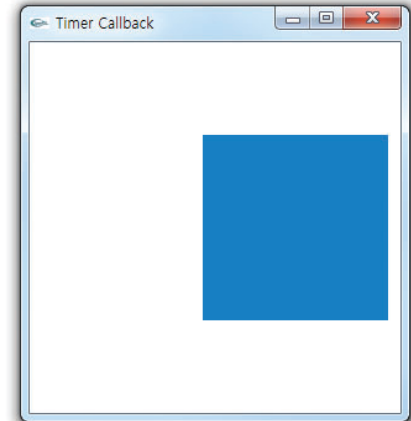
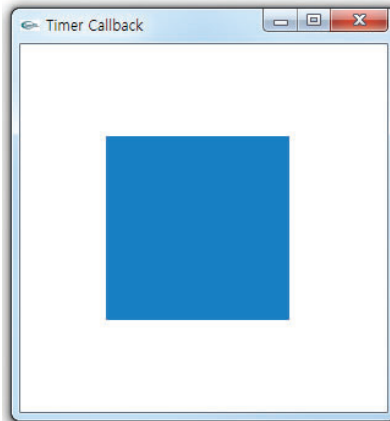
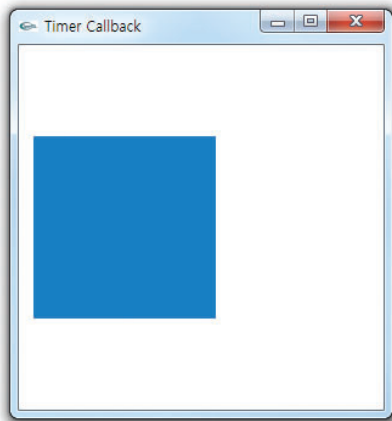
Section

Timer Callback



Timer callback 함수의 구현 결과

코드 2-10



코드 2-11

