```c
#ifndef __FREEGLUT_EXT_H__
#define __FREEGLUT_EXT_H__

/*
 * freeglut_ext.h
 *
 * The non-GLUT-compatible extensions to the freeglut library include file
 *
 * Copyright (c) 1999-2000 Pawel W. Olszta. All Rights Reserved.
 * Written by Pawel W. Olszta, <olszta@sourceforge.net>
 * Creation date: Thu Dec 2 1999
 *
 * Permission is hereby granted, free of charge, to any person obtaining a
 * copy of this software and associated documentation files (the "Software"),
 * to deal in the Software without restriction, including without limitation
 * the rights to use, copy, modify, merge, publish, distribute, sublicense,
 * and/or sell copies of the Software, and to permit persons to whom the
 * Software is furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included
 * in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
 * OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.  IN NO EVENT SHALL
 * PAWEL W. OLSZTA BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
 * IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
 * CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

#ifdef __cplusplus
    extern "C" {
#endif

/*
 * Additional GLUT Key definitions for the Special key function
 */
#define GLUT_KEY_NUM_LOCK        0x006D
#define GLUT_KEY_BEGIN           0x006E
#define GLUT_KEY_DELETE          0x006F
#define GLUT_KEY_SHIFT_L         0x0070
#define GLUT_KEY_SHIFT_R         0x0071
#define GLUT_KEY_CTRL_L          0x0072
#define GLUT_KEY_CTRL_R          0x0073
#define GLUT_KEY_ALT_L           0x0074
#define GLUT_KEY_ALT_R           0x0075

/*
```

```
 * GLUT API Extension macro definitions -- behaviour when the user clicks on an "x" to clos
e a window
 */
#define GLUT_ACTION_EXIT                    0
#define GLUT_ACTION_GLUTMAINLOOP_RETURNS        1
#define GLUT_ACTION_CONTINUE_EXECUTION          2

/*
 * Create a new rendering context when the user opens a new window?
 */
#define GLUT_CREATE_NEW_CONTEXT             0
#define GLUT_USE_CURRENT_CONTEXT            1

/*
 * Direct/Indirect rendering context options (has meaning only in Unix/X11)
 */
#define GLUT_FORCE_INDIRECT_CONTEXT         0
#define GLUT_ALLOW_DIRECT_CONTEXT           1
#define GLUT_TRY_DIRECT_CONTEXT             2
#define GLUT_FORCE_DIRECT_CONTEXT           3

/*
 * GLUT API Extension macro definitions -- the glutGet parameters
 */
#define  GLUT_INIT_STATE                0x007C

#define  GLUT_ACTION_ON_WINDOW_CLOSE        0x01F9

#define  GLUT_WINDOW_BORDER_WIDTH           0x01FA
#define  GLUT_WINDOW_BORDER_HEIGHT          0x01FB
#define  GLUT_WINDOW_HEADER_HEIGHT          0x01FB  /* Docs say it should always
have been GLUT_WINDOW_BORDER_HEIGHT, keep this for backward compatibility */

#define  GLUT_VERSION                   0x01FC

#define  GLUT_RENDERING_CONTEXT         0x01FD
#define  GLUT_DIRECT_RENDERING          0x01FE

#define  GLUT_FULL_SCREEN               0x01FF

#define  GLUT_SKIP_STALE_MOTION_EVENTS      0x0204

/*
 * New tokens for glutInitDisplayMode.
 * Only one GLUT_AUXn bit may be used at a time.
 * Value 0x0400 is defined in OpenGLUT.
 */
#define  GLUT_AUX                   0x1000

#define  GLUT_AUX1                      0x1000
#define  GLUT_AUX2                      0x2000
#define  GLUT_AUX3                      0x4000
#define  GLUT_AUX4                      0x8000
```

```c
/*
 * Context-related flags, see freeglut_state.c
 */
#define  GLUT_INIT_MAJOR_VERSION        0x0200
#define  GLUT_INIT_MINOR_VERSION        0x0201
#define  GLUT_INIT_FLAGS            0x0202
#define  GLUT_INIT_PROFILE           0x0203


/*
 * Flags for glutInitContextFlags, see freeglut_init.c
 */
#define  GLUT_DEBUG              0x0001
#define  GLUT_FORWARD_COMPATIBLE        0x0002



/*
 * Flags for glutInitContextProfile, see freeglut_init.c
 */
#define GLUT_CORE_PROFILE            0x0001
#define  GLUT_COMPATIBILITY_PROFILE       0x0002

/*
 * Process loop function, see freeglut_main.c
 */
FGAPI void    FGAPIENTRY glutMainLoopEvent( void );
FGAPI void    FGAPIENTRY glutLeaveMainLoop( void );
FGAPI void    FGAPIENTRY glutExit       ( void );


/*
 * Window management functions, see freeglut_window.c
 */
FGAPI void    FGAPIENTRY glutFullScreenToggle( void );
FGAPI void    FGAPIENTRY glutLeaveFullScreen( void );


/*
 * Window-specific callback functions, see freeglut_callbacks.c
 */
FGAPI void    FGAPIENTRY glutMouseWheelFunc( void (* callback)( int, int, int, int ) );
FGAPI void    FGAPIENTRY glutCloseFunc( void (* callback)( void ) );
FGAPI void    FGAPIENTRY glutWMCloseFunc( void (* callback)( void ) );
/* A. Donev: Also a destruction callback for menus */
FGAPI void    FGAPIENTRY glutMenuDestroyFunc( void (* callback)( void ) );


/*
 * State setting and retrieval functions, see freeglut_state.c
 */
FGAPI void    FGAPIENTRY glutSetOption ( GLenum option_flag, int value );
FGAPI int *   FGAPIENTRY glutGetModeValues(GLenum mode, int * size);
/* A.Donev: User-data manipulation */
FGAPI void*   FGAPIENTRY glutGetWindowData( void );
FGAPI void    FGAPIENTRY glutSetWindowData(void* data);
FGAPI void*   FGAPIENTRY glutGetMenuData( void );
FGAPI void    FGAPIENTRY glutSetMenuData(void* data);
```

```c
/*
 * Font stuff, see freeglut_font.c
 */
FGAPI int     FGAPIENTRY glutBitmapHeight( void* font );
FGAPI GLfloat FGAPIENTRY glutStrokeHeight( void* font );
FGAPI void    FGAPIENTRY glutBitmapString( void* font, const unsigned char *string );
FGAPI void    FGAPIENTRY glutStrokeString( void* font, const unsigned char *string );


/*
 * Geometry functions, see freeglut_geometry.c
 */
FGAPI void    FGAPIENTRY glutWireRhombicDodecahedron( void );
FGAPI void    FGAPIENTRY glutSolidRhombicDodecahedron( void );
FGAPI void    FGAPIENTRY glutWireSierpinskiSponge ( int num_levels, GLdouble offset
[3], GLdouble scale );
FGAPI void    FGAPIENTRY glutSolidSierpinskiSponge ( int num_levels, GLdouble offset
[3], GLdouble scale );
FGAPI void    FGAPIENTRY glutWireCylinder( GLdouble radius, GLdouble height, GLint s
lices, GLint stacks);
FGAPI void    FGAPIENTRY glutSolidCylinder( GLdouble radius, GLdouble height, GLint s
lices, GLint stacks);


/*
 * Extension functions, see freeglut_ext.c
 */
typedef void (*GLUTproc)();
FGAPI GLUTproc FGAPIENTRY glutGetProcAddress( const char *procName );


/*
 * Multi-touch/multi-pointer extensions
 */

#define GLUT_HAS_MULTI 1

FGAPI void FGAPIENTRY glutMultiEntryFunc( void (* callback)( int, int ) );
FGAPI void FGAPIENTRY glutMultiButtonFunc( void (* callback)( int, int, int, int, int ) );
FGAPI void FGAPIENTRY glutMultiMotionFunc( void (* callback)( int, int, int ) );
FGAPI void FGAPIENTRY glutMultiPassiveFunc( void (* callback)( int, int, int ) );


/*
 * Joystick functions, see freeglut_joystick.c
 */
/* USE OF THESE FUNCTIONS IS DEPRECATED !!!!! */
/* If you have a serious need for these functions in your application, please either
 * contact the "freeglut" developer community at freeglut-developer@lists.sourceforge.net,
 * switch to the OpenGLUT library, or else port your joystick functionality over to PLIB's
 * "js" library.
 */
int     glutJoystickGetNumAxes( int ident );
int     glutJoystickGetNumButtons( int ident );
int     glutJoystickNotWorking( int ident );
float   glutJoystickGetDeadBand( int ident, int axis );
void    glutJoystickSetDeadBand( int ident, int axis, float db );
float   glutJoystickGetSaturation( int ident, int axis );
```

```c
void    glutJoystickSetSaturation( int ident, int axis, float st );
void    glutJoystickSetMinRange( int ident, float *axes );
void    glutJoystickSetMaxRange( int ident, float *axes );
void    glutJoystickSetCenter( int ident, float *axes );
void    glutJoystickGetMinRange( int ident, float *axes );
void    glutJoystickGetMaxRange( int ident, float *axes );
void    glutJoystickGetCenter( int ident, float *axes );

/*
 * Initialization functions, see freeglut_init.c
 */
FGAPI void    FGAPIENTRY glutInitContextVersion( int majorVersion, int minorVersion );
FGAPI void    FGAPIENTRY glutInitContextFlags( int flags );
FGAPI void    FGAPIENTRY glutInitContextProfile( int profile );

/* to get the typedef for va_list */
#include <stdarg.h>

FGAPI void    FGAPIENTRY glutInitErrorFunc( void (* vError)( const char *fmt, va_list a
p ) );
FGAPI void    FGAPIENTRY glutInitWarningFunc( void (* vWarning)( const char *fmt, va
_list ap ) );

/*
 * GLUT API macro definitions -- the display mode definitions
 */
#define  GLUT_CAPTIONLESS           0x0400
#define  GLUT_BORDERLESS            0x0800
#define  GLUT_SRGB                  0x1000

#ifdef __cplusplus
    }
#endif

/*** END OF FILE ***/

#endif /* __FREEGLUT_EXT_H__ */
```