REAL TIME OPERATING SYSTEM PROGRAMMING-II: Real time Linux, Windows CE and OSEK

Lesson-10:
Win32 APIs and Creating Windows

1. Win32

Win32

- Win32 has large number of APIs in a PC.
- A subset is required for handheld devices and small screen size systems.
- WCE uses a subset of Win32 APIs

Windows for the graphic user interfaces (GUIs)

- Screen or touch screen interaction with a user.
- GUIs facilitate interaction and inputs from user after graphic screen displays of menus, buttons, dialog boxes, text fields, labels, check box and radio buttons and others.

Application Window

- Displays the messages in central region, title, command, tool and status bars
- Displays commands (buttons) so that a stylus tap (or mouse click) sends the selected command using menu and buttons.
- Displays icons for maximizing, minimizing and closing at right hand side top corner.
- Windows also show icon for Help (to help the user) and a ? sign icon (to show more buttons on a tap there).

Application Window **in WCE**

- WCE has single line controls for command, tool and status bars.
- A stylus tap or mouse click sends the menu choice to the application.
- WCE has new format for the Windows controls (command, menu, toolbar bars) and new controls (data, time, calendar) and organizer (for example, task-to-do)

2. Win32 API Example

WINAPI WinMain ( )

- int WINAPI WinMain (HINSTANCE hPresentinstance; HINSTANCE hPreviousinstance, LPWSTR lpCommandline, int iCommandshow) {MessageBox (NULL, TEXT (“Welcome”), TEXT (“WelcMsg”), MB_DEFBUTTON1, MB_DEFBUTTON2, MB_ICONQUESTION);
- return 0; } /*

WINAPI WinMain ( )…

- /* After third argument the last argument(s) is one or more among the series of flags which can be used for showing the buttons or icons in MessageBox Windows bar.
- The buttons and icons must be those as provided for in the procedure MessageBox. */

WINAPI WinMain ( )…

- Presentinstance

a parameter to identify the present instance,

- previnstance

a parameter to identify the Previuosinstance (WCE always assumes it = 0)

WINAPI WinMain ( )…

- Commandline is a Unicode string

–to specify the functions of the program

Arguments:

- Handle

– for file

- DCB

a long pointer to device control block (DCB), defines 32-bits for DCB length, baud rate, binary flag, parity flag, and 24 other flags

WINAPI WinMain ( )…

- Commandshow

an integer to specify *state* of the program, which defines a configuration of main window.

- The state parameter is passed from parent application to new application.
- The state configuration in a personal computer can be the one which shows minimize, maximized or normal icons.

WCE WINAPI WinMain ( )…

- Only three states and configuration of WCE Windows is as per variables show without activate (SW_SHOWNOACTIVATE), show hidden (SW_HIDE) and show normal (SW_SHOW).
- Default value of Commandshow is used as per the value for the main window show command

WINAPI WinMain ( )…

- MessageBox creates a window over the main window.
- The window shows messages in the box until window is closed.

Message Box

- Shows (a) no other Windows because first argument is NULL
- Shows (b) text message Welcome in the Unicode-message window (at center) and text Unicode-message caption (title) WelcMsg at left corner in the command-cum-tool-status bar,

Message Box

- Shows (c) buttons as per definitions MB_DEFBUTTON1, MB_DEFBUTTON2 in the middle of command bar and icon of ?
- Shows(d) at the end of bar, a sign X icon creates at right corner in the bar.
- The X enables the closing the window by the user on tapping on the touch screen or mouse click.

WINAPI WinMain ( )…

- MessageBox is used here in place of printf otherwise a driver console.dll needs to be added to enable printing on console (screen).

Handle

- INSTANCE is a Handle object.
- In present case, a handle is a reference to an interface, which handles a Window instance

3. Creating Windows

Windows procedures to create its own Windows

- CreateWindowEx

– to create main window.

- MainWndProc,

– to create application window

- WM_Paint

– to draw the window background and put text within it at specified position after first creating a client rectangle

Drawing on Screen

- WCE does not support full Win32 graphics API and different mapping modes in Windows.
- WCE does not support coordinate transformations. A text is written using DrawText procedure.
- WCE always sets device context in MM_TEXT mapping mode.
- Windows application does not write directly to the screen. It requests a Handle.

Handle for drawing and displaying device context

- Device context specifies the application Windows.
- Windows sends the pixels to screen using the device context.
- A device context is a tool, which Windows use to managing the access to the display and printer.
- Two attributes of device context are colours for background and foreground.
-

Handle for drawing and displaying device context.

- Text alignment attributes of device context are left, right, top, center, bottom, no update and update of current point of device context, and baseline alignment.
- Font of the displayed text from the device context can be specified. Font can also be created for an application as alternative to WCE default fonts

Summary

We learnt

- WCW Win32 subset provides for drawing Windows
- Subset of Win32 APIs in WCE provision for inputs from keys, touch screen or mouse, communication with serial port, Bluetooth, IrDA, WiFi, networking, device to device socket and communication Functions

End of Lesson-10 of chapter 12 on
Win32 APIs and Creating Windows