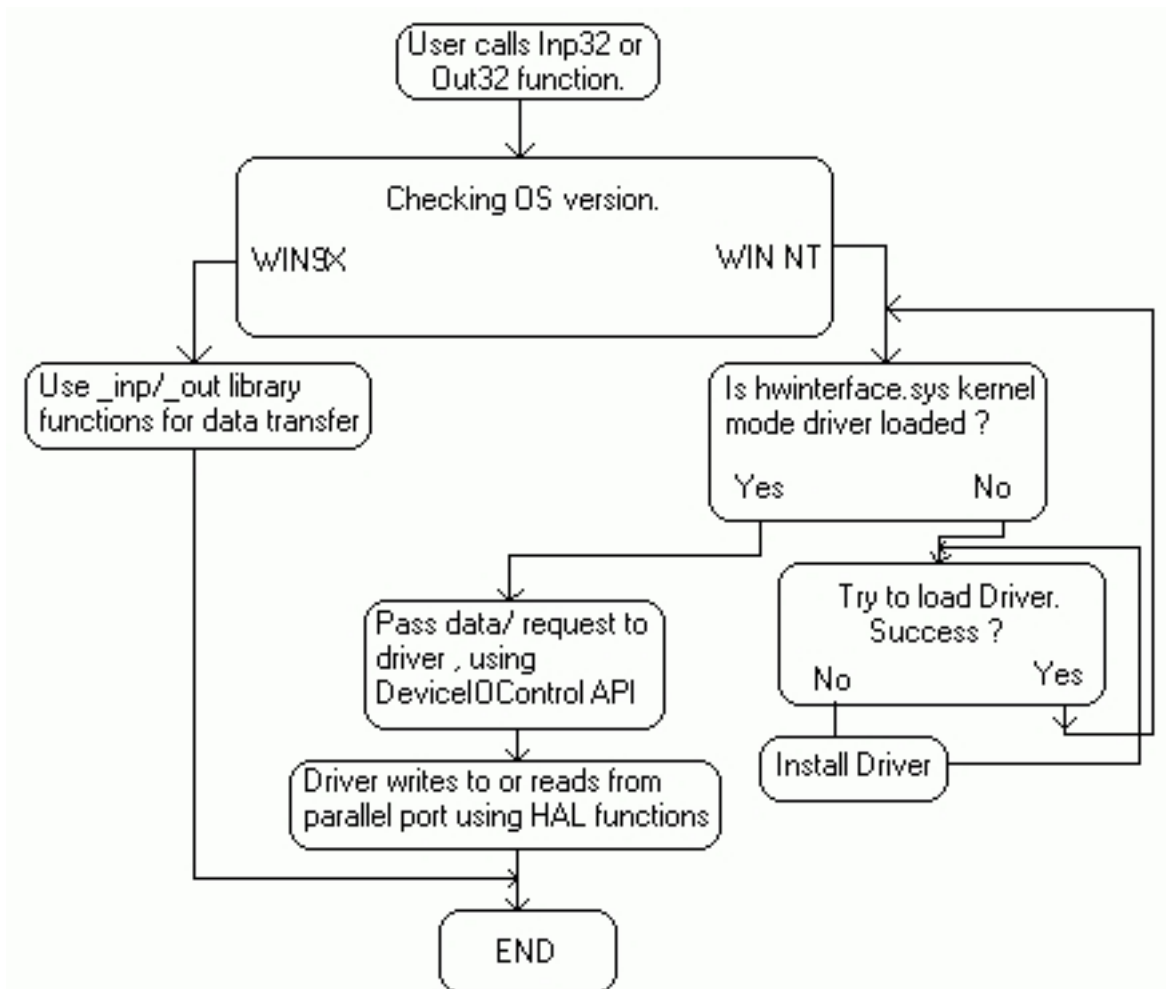


Tried the Inpout32.dll..? then Learn how Inpout32.dll does the things. This brief tutorial explains about the working of Inpout32.dll in simple steps, with the help of a flow chart. This could help you much if you want to modify the Inpout32 dll source code

If you don't know what is Inpout32.dll, please [read it here](#) and then continue.

How it works

The outstanding feature of Inpout32.dll is , it can work with all the windows versions without any modification in user code or the DLL itself. This tutorial describes how it is achieved, what programming methods used, what are the APIs used, etc.... The Dll will check the operating system version when functions are called, and if the operating system is WIN9X, the DLL will use `_inp()` and `_outp` functions for reading/writing the parallel port. On the other hand, if the operating system is WIN NT, 2000 or XP, it will install a kernel mode driver and talk to parallel port through that driver. The user code will not be aware of the OS version on which it is running. This DLL can be used in WIN NT clone operating systems as if it is WIN9X. The flow chart of the program is given below.



The flowchart illustrates the execution path of Input32.dll. It starts with 'User calls Inp32 or Out32 function.', followed by 'Checking OS version.' which branches into 'WIN9X' and 'WIN NT'. The 'WIN9X' path leads to 'Use _inp/_out library functions for data transfer' and then 'END'. The 'WIN NT' path leads to 'Is hwinterface.sys kernel mode driver loaded?'. If 'Yes', it goes to 'Pass data/ request to driver, using DeviceIOControl API', then 'Driver writes to or reads from parallel port using HAL functions', and finally 'END'. If 'No', it goes to 'Try to load Driver. Success?'. If 'Yes', it goes to 'Install Driver' and then loops back to 'Is hwinterface.sys kernel mode driver loaded?'. If 'No', it goes to 'Install Driver' and then loops back to 'Try to load Driver. Success?'.