I^{2}C Software Drivers

These drivers are based on the pseudo-code driver included in the I^{2}C FAQ published by Vincent Himpe<vincent.himpe@ping.be>. The FAQ can be downloaded from Vincent's homepage *which now appears to be closed*.

Standard Philips Copyright notice

' Purchase of Philips I^{2}C components conveys a licence under the Philips I^{2}C patent to use the components in the I^{2}C system provided the system conforms to the I^{2}C specifications defined by Philips '
The Philips patents have now expired.

This page has been translated to Ukrainian by Adrian Pantilimonu.

Background Late in 1996, I was asked to build a device which would demonstrate functions similar to a software security block, or "dongle", but without the usual cryptographic features. The end-result employed a 512-byte non-volatile memory chip, attached to a parallel port. Since the NVRAM used the I^{2}C bus, I had to learn how to drive it. Here Vincent's FAQ (above) proved invaluable.
The initial version was developed for 80X86 platforms, communicating via the standard parallel port. Basic read and write functions are provided. The schematic consists mainly of mechanism to switch access to either the dongle or a printer. The only I^{2}C specific part is the buffer U2C, which simulates an open-collector driver for the SDA line. (An actual open- collector transistor could have been used - inverting the sense of the D2 input - but the spare gate was available). Due to the uni-directional nature of a (basic) parallel port, separate lines - D2 and PE - are used for output & input.

Subsequently, I ported the software to the Zilog Z80 CPU, to run the "dongle" from my P112 board, which has a PC-like parallel port.
A subsequent, unrelated project led to my porting the low-level I^{2}C drivers to the Motorola 68705 CPU.

I am publishing these designs as freeware, under the GNU General Public License.

☞	"Dongle" schematic
☞	80X86 Version (36kB)
☞	Z80 Version (36kB)
☞	MC68HC705C8 Version (4kB)
☞	GNU General Public License

Note Most EEPROMs using the I^{2}C bus take several milliseconds to execute a write command. During this time they ignore the bus completely. This condition can be recognised, by the device failing to give an Acknowledge to a Select command. The "dongle" referred to above used the Ramtron NVRAMs, which are based on ferro-electric technology, and write in nanoseconds. Hence they do not "disappear" from the bus, and there is no need to poll them for completion. Consequently the above code does not include this feature, which should be taken into account in supporting "conventional" EEPROMs.

Back to Home Page