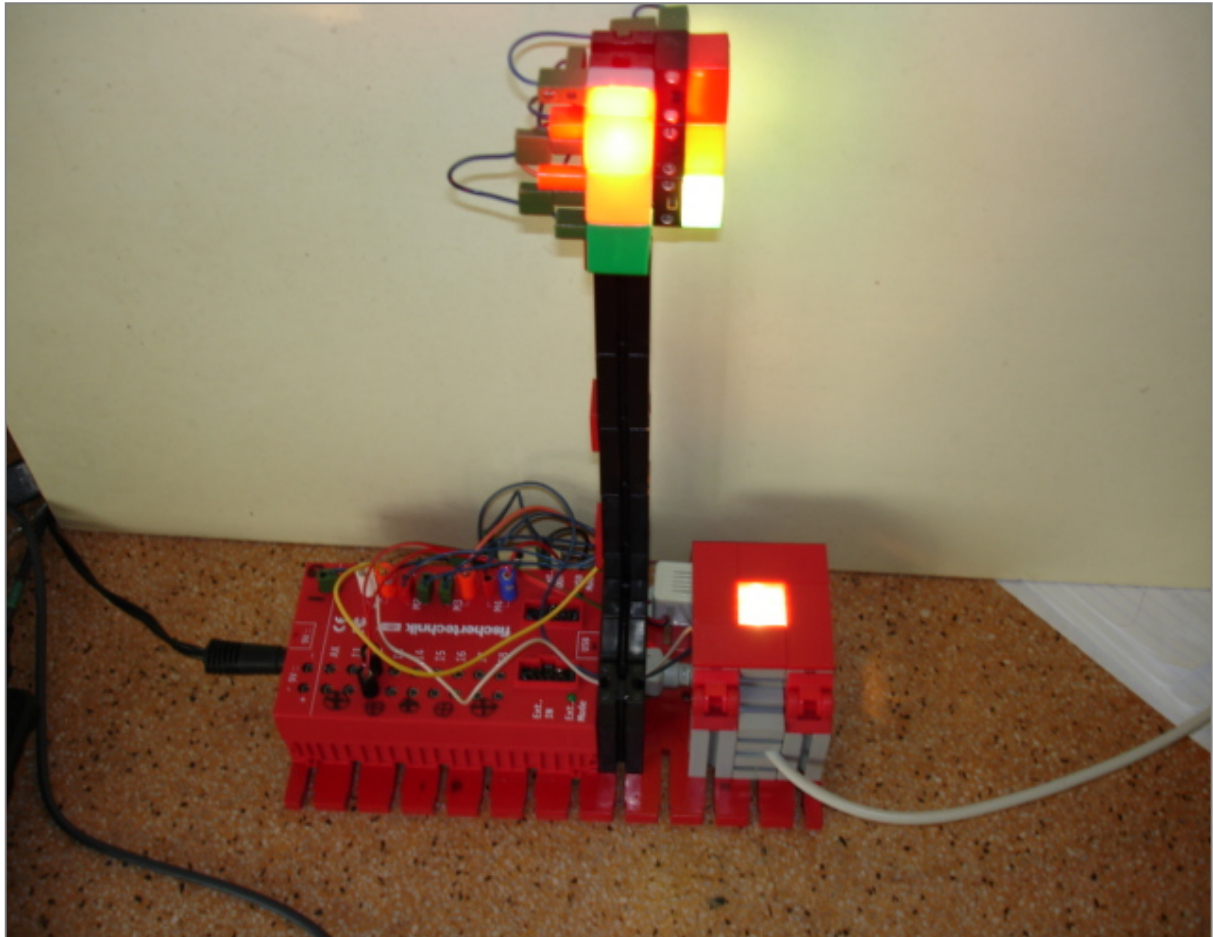
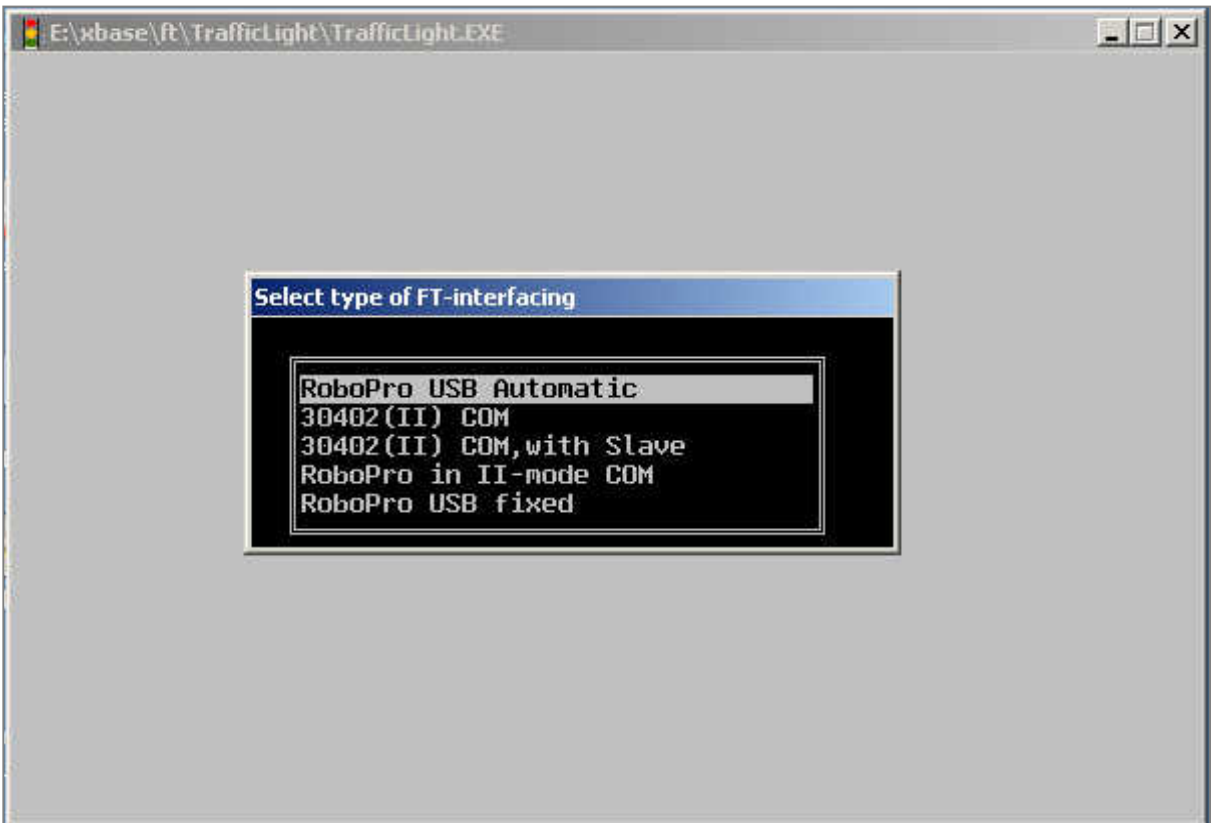
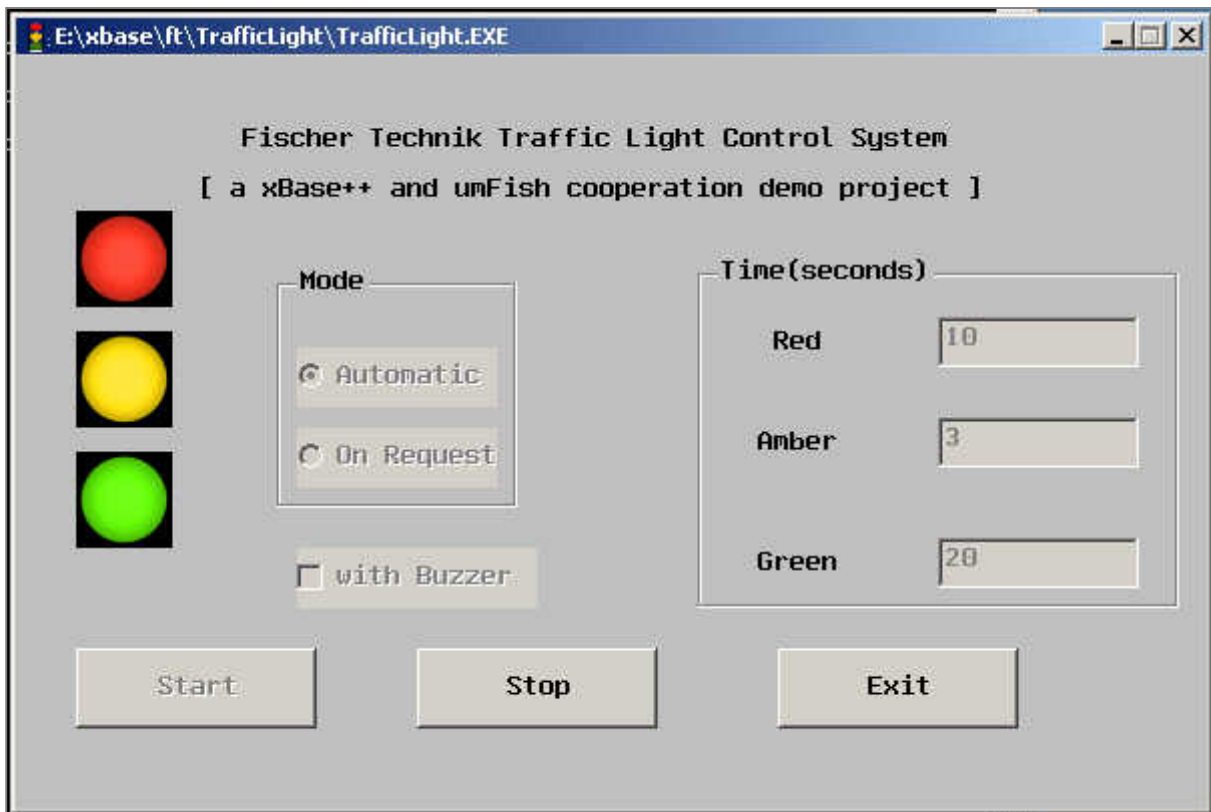


# the TrafficLight Demo Application

Demo Application uses linkable object modules for **xBase++**  
(<http://members.chello.nl/r.nootboom2>)  
and (<http://www.ftcomputing.de/xbasee.htm>)  
(by *r.nootboom2@chello.nl*)

*for win32*





## Description of the TrafficLight demo xBase++-umFish project

The *TrafficLight.prg* is an application that controls a traffic-light at a crossroad intersection . There is a trafficlight in the West-direction and one in the North-direction. Assumed is that the East-to-West road is a busy important one and the South-to-North road is a less used quiet one. The application has 2 working modes : 1) Automatic : the light sequence is processed automatically and continuously and an 2) OnRequest-mode : the light in the North-direction changes only to green if a request is made by pushing a button on the model, the traffic lights process then exactly one cycle of the light-phases(stages). That a request has been done is spotted by a white light inside the push button on the ft-model so a person can see that the request is already being processed.

A nice detail is that an optional buzzer(build in the model)-sound goes off when the North-direction-lights goes green ,this is an extra help for the visually handicapped.

**Technical notes by the xBase++ program** : In the *Init-AppSys()* procedure a GUI-*XbpDialog()*-window is created. The *main()*-routine consists only of the initialization-call for the interface (*FtRInit()*), the creation of the dialog-objects(xBase-parts) and contains the first main-eventloop/handler. The activate-codeblock of the Start-pushbutton starts the actual TrafficLight process(*StartTrafficLight()*) . The *StartTrafficLight()*-routine is in fact also a 2nd-eventloop-handler that walks through all the TrafficLightPhases contained in the array : *LightPhases\_* created by the *CreatePhases()* routine.

All readable and clarified constant names are in the header file *TrafficLight.ch*.

This model is controlled with an USB RoboPro IO-Extension module which can be controlled by *umFish40.dll* via the xBase++ *ftroboii.obj(.prg)* wrapper module. But the model can also be made with the Intelligent Interface,the RoboPro or the older parallel interface.

All xBase++ umFish drivers can be found in the package : ***ftxbase.zip***.

All sources of the TrafficLight-demo are also in ***ftxbase.zip***

**<http://members.chello.nl/r.nooteboom2>**

**email : [r.nooteboom2@chello.nl](mailto:r.nooteboom2@chello.nl)**