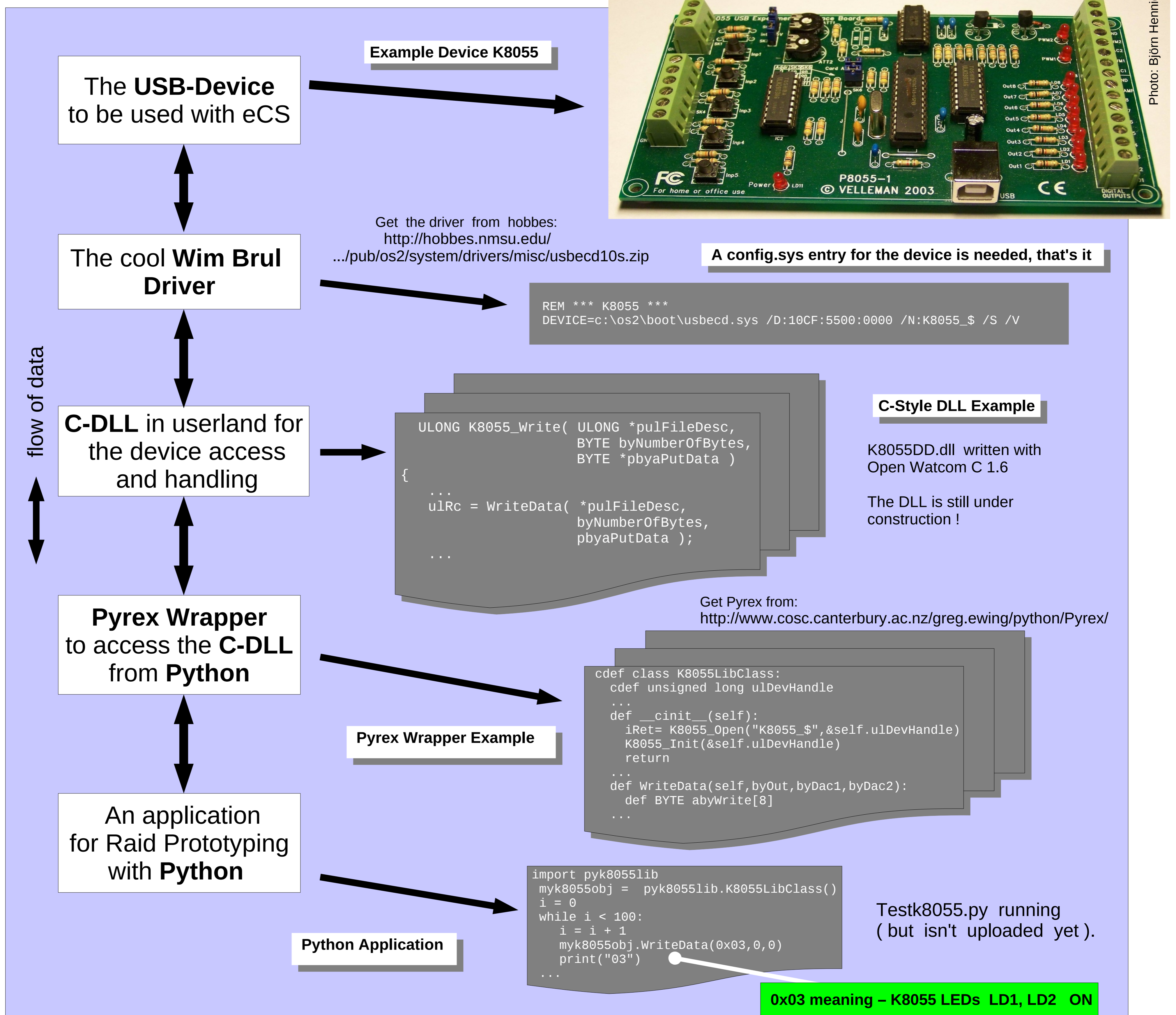# K8055 with eCS, USB accessed from Python / Pyrex

by **Björn Hennig** and **Uwe Hinz**
b.hennig@gmx.de   uhdrelb@t-online.de
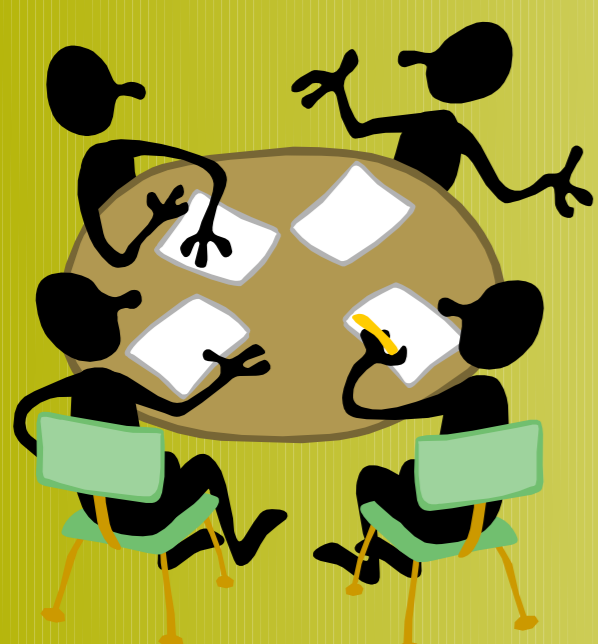
Motivation

There is a lot of USB-Devices on the market or under development. They will never have any driver for eCS. It's a bad situation. This poster will show a way to get out of that deficiency. Here is a possible method:

1st   A generic USB-Driver that works with eCS has to be found (Wim Brul's driver 'usbecd.sys' is appropriate) ,

2nd   A DLL is to be progammed, that interfaces the generic driver and the particular USB-Device ,

3rd   In order to make work with a high level programming language effective, a suitable Wrapper is needed !

---

flow of data

The **USB-Device** to be used with eCS

Example Device K8055

Photo: Björn Hennig

The cool **Wim Brul Driver**

Get the driver from hobbes:
http://hobbes.nmsu.edu/
.../pub/os2/system/drivers/misc/usbecd10s.zip

A config.sys entry for the device is needed, that's it

```
REM *** K8055 ***
DEVICE=c:\os2\boot\usbecd.sys /D:10CF:5500:0000 /N:K8055_$ /S /V
```

C-DLL in userland for the device access and handling

C-Style DLL Example

```
ULONG K8055_Write( ULONG *pulFileDesc,
                   BYTE byNumberOfBytes,
                   BYTE *pbyaPutData )
{
   ...
   ulRc = WriteData( *pulFileDesc,
                     byNumberOfBytes,
                     pbyaPutData );
   ...
```

K8055DD.dll written with Open Watcom C 1.6

The DLL is still under construction !

Pyrex Wrapper to access the **C-DLL** from **Python**

Get Pyrex from:
http://www.cosc.canterbury.ac.nz/greg.ewing/python/Pyrex/

Pyrex Wrapper Example

```
cdef class K8055LibClass:
   cdef unsigned long ulDevHandle
   ...
   def __cinit__(self):
     iRet= K8055_Open("K8055_$",&self.ulDevHandle)
     K8055_Init(&self.ulDevHandle)
     return
   ...
   def WriteData(self,byOut,byDac1,byDac2):
     def BYTE abyWrite[8]
   ...
```

An application for Raid Prototyping with **Python**

Python Application

```
import pyk8055lib
myk8055obj =  pyk8055lib.K8055LibClass()
i = 0
while i < 100:
    i = i + 1
    myk8055obj.WriteData(0x03,0,0)
    print("03")
...
```

Testk8055.py running
(but isn't uploaded yet).

0x03 meaning – K8055 LEDs LD1, LD2   ON

Outlook

As soon as Python is used to create applications, programming becomes easy. Easier than today!   A nice GUI with Qt4, wxWidgets or wxPython will appear as up to date. eCS-Users who could not benefit from their USB devices in the past, can create their own software solutions for classless USB-Devices for eCS in future.

And what is hot right now ?

New Devices on the workbench:

- **meM-PIO** → **Work started**

- **Meilhaus RedLab** → **Device purchased**

- **TinyCAN** → **Already soldered**

- **Octopus USB** → **Just an idea**