

## EasyAccess-Code für unsere Zeilenkameras via Python

In diesem Applikationsbeispiel zeigen wir, wie einfach es ist, unsere Zeilenkameras der Serie e9u-LSMD mit Python unter Windows zu verwenden. Hierzu verwenden wir einige Funktionen für den asynchronen Betrieb der Kameras aus der dynamischen Kamera-Programm-Bibliothek (DLL). Die entsprechenden DLL-Dateien für Ihr Betriebssystem finden Sie auf unserer Service-Seite unter <https://service.eureca.de/LSMD/standard/Windows/index.php>. Unter <https://service.eureca.de/LSMD/> finden Sie auch Informationen für den Betrieb unserer Kameras unter anderen Betriebssystemen bzw. Sprachen.

Die hier gezeigten Funktionen für die asynchrone Ansteuerung sind bereits leistungsfähig genug, um die Kamera in einem breiten Spektrum von Anwendungen einzusetzen. Tatsächlich wurden die meisten unserer Anwendungshinweise für unsere Zeilenkameras mit Hilfe dieser Funktionen erstellt.

Somit sind diese Funktionen ein idealer Ausgangspunkt, um sich mit der Arbeitsweise der Kameras vertraut zu machen. Sobald Sie diese getestet haben und mit der Kamerasteuerung vertraut sind, können Sie auch die anderen Routinen aus der DLL für Anwendungen verwenden, die eine komplexere Ansteuerung der Kamera oder anspruchsvollere Parameter erfordern.

## 1 Vorbereitungen

### 1.1 Python installieren

Überprüfen Sie Ihre Python-Installation, um sicherzustellen, dass sie auf dem neuesten Stand ist. Für ein notwendiges Update oder eine komplette Neuinstallation siehe <https://www.python.org/downloads/>.

Sie sollten dem Installationsassistenten erlauben, Python zum PATH Ihres Computers hinzuzufügen, damit Sie Python-Programme bequem von der Kommandozeile aus starten können. Möglicherweise müssen Sie Ihren PC nach der Installation neu starten, damit diese Änderung wirksam wird.

### 1.2 Bei Bedarf: ctypes installieren

Das Python-Paket ctypes wird benötigt, um die externen Kamera-DLLs zu verwenden und so Daten mit der Kamera auszutauschen.

Wenn dieses Paket nicht bereits in Ihrer Python-Installation enthalten ist, erhalten Sie eine entsprechende Fehlermeldung, wenn Sie den EasyAccess-Python-Code ausführen. In diesem Fall kann das Paket ctypes von der Kommandozeile aus mit dem folgenden Befehl installiert werden:

```
pip install ctypes
```

### 1.3 Installation der Kamera-DLL-Datei

Alle notwendigen Routinen sind in einer externen DLL namens `libe9u_LSMD_x64.dll` enthalten. Um eine kostenlose Kopie zu erhalten, kontaktieren Sie uns bitte über [info@eureca.de](mailto:info@eureca.de) oder laden Sie sich diese unter <https://service.eureca.de/LSMD/standard/Windows/index.php> herunter.

### 1.4 EasyAccess-Python-Code erhalten

Kopieren Sie den unten gezeigten EasyAccess-Code von <https://www.eureca.de/EasyAccess-de> und speichern ihn in eine neue Python-Datei mit z. B. dem Namen `EasyAccess.py`. Stellen Sie sicher, dass sich der Python-Code und die DLL `libe9u_LSMD_x64.dll` im selben Verzeichnis befinden!



## 2 Ausführen des *EasyAccess*-Codes

Schließen Sie die Kamera mit einem geeigneten USB-Kabel an einen Ihrer USB-Anschlüsse an. Vergewissern Sie sich, dass das Kabel wirklich ein Datenkabel und nicht nur ein USB-Stromkabel ist! Führen Sie den *EasyAccess*-Python-Code über die Befehlszeile aus. Sie sollten nun eine Ausgabe sehen, die Informationen über das Kameramodul sowie die Sensordaten enthält.

## 3 *EasyAccess* Python-Code

```
# EasyAccess.py V1.1
#
# Python tool as example code for line scan cameras of the e9u series by
#   EURECA Messtechnik GmbH
# - detects and starts an attached camera in asychrone mode
# - reads out the camera once and prints the recorded sensor data
#
# For details please refer to: www.eureca.de

# import library for handling external camera DLL
import ctypes

# basic values of the used linear sensor; e.g. for e9u-LSMD-TCD1304-STD:
#   3648 pixel
SENSOR_PIXELS = 3648
EXPOSURE_TIME = 10000 # given in micro seconds
FRAME_TIME = 10000 # given in micro seconds

# open external DLL for 64 bit systems
libe9u = ctypes.WinDLL('./libe9u_LSMD_x64.dll')

# define argument and return types for the used functions
libe9u.e9u_LSMD_search_for_camera.argtype = ctypes.c_uint
libe9u.e9u_LSMD_search_for_camera.restype = ctypes.c_int

libe9u.e9u_LSMD_start_camera_async.argtype = ctypes.c_uint
libe9u.e9u_LSMD_start_camera_async.restype = ctypes.c_int

libe9u.e9u_LSMD_set_times_us.argtypes = (ctypes.c_uint, ctypes.c_uint,
    ctypes.c_uint)
libe9u.e9u_LSMD_set_times_us.restype = ctypes.c_int

libe9u.e9u_LSMD_get_next_frame.argtype = ctypes.c_uint
libe9u.e9u_LSMD_get_next_frame.restype = ctypes.c_int

libe9u.e9u_LSMD_get_pixel_pointer.argtypes = (ctypes.c_uint, ctypes.c_uint)
libe9u.e9u_LSMD_get_pixel_pointer.restype = ctypes.POINTER(ctypes.c_int16)

# search for camera
libe9u.e9u_LSMD_search_for_camera(0)

# start the found device in asychrone mode
libe9u.e9u_LSMD_start_camera_async(0)

# setting integration time
```



```
libe9u.e9u_LSMD_set_times_us (0, EXPOSURE_TIME, FRAME_TIME)

# get an image and store it for readout
libe9u.e9u_LSMD_get_next_frame (0)

# getting the pointer to the array with the sensor data
pointer = libe9u.e9u_LSMD_get_pixel_pointer(0, 0)

# printing the sensor data
for pixel in range(0, SENSOR_PIXELS):
    print(str(pixel) + ": " + str(pointer[pixel]))

exit()
```

## 4 Fehlerbehandlung

Aus Gründen der Übersichtlichkeit enthält der obige *EasyAccess*-Code keine zusätzlichen Zeilen zur Fehlerbehandlung. Es wird jedoch empfohlen, nach jedem Aufruf einer DLL-Routine eine Fehlerbehandlung durchzuführen, um den korrekten Betrieb der Kamera und/oder des Computersystems zu überprüfen.

Zur Fehlerbehandlung stellen alle DLL-Routinen einen Rückgabewert zur Verfügung, der auf mögliche Fehler überprüft werden kann. Wurde die jeweilige Funktion ohne Fehler beendet, wird NULL zurückgegeben. Ist ein Fehler aufgetreten, wird ein Fehlercode zurückgegeben. Die genaue Bedeutung der Fehlercodes entnehmen Sie bitte der ausführlichen Kameradokumentation.

Nachfolgend ein Beispiel für eine geeignete Fehlerbehandlung anhand der `libe9u.e9u_LSMD_search_for_camera` DLL-Routine.

```
print("Searching for camera: ", end='')
i_status = libe9u.e9u_LSMD_search_for_camera(0)
if i_status == 0:
    print("OK!")
else:
    print("Error! Code: " + str(i_status))
    exit(1)
```

