**ELI-NP target identification software**

**using TAG barcode**

*Rev\_1: 07.03.2017*

*Author: Dragos C. Popescu*

*Coordinator: Mihail O. Cernaianu*

**Description:**

The purpose of this software application is to identify and decode a TAG barcode which is used for targets identification.

The TAG, as defined in the previous deliverable D6.1, is a CODE39 barcode type that can be milled/printed/laser engraved/attached onto the target frame or target wafer itself, depending on experiment configuration and constraints. The data from the barcode consists of 16 alphanumeric characters divided in three parts:

* 5 alphanumerical characters (A-Z & 0-9) encoding the registering facility

*ex:* EXFEL, ELINP, ELIBL, DESY1, etc.

* 3 alphanumerical characters (A-Z & 0-9) to identify the design of the target frame in the database

*ex: 4IN, 9X9, FOI, etc.*

* 6 digits encoding multiple frames unique serial numbers

*ex: 000000 – 999999*

Below is an example of a barcode having EXFEL4IN000001 encoding:



**Software application interfaces**:

The software is executed as a Command Line application having as arguments the name and extension of the input image containing a barcode and another optional argument for displaying an output image with the identified barcode. The encoding of the barcode is returned in the Command Line. It is also returned in a text file that is created having the same name as the input image but with .TAG extension.

**Implementation**

The application is built around two open-source libraries:

* **OpenCV** - <http://opencv.org/>
* **Zbar** - <http://zbar.sourceforge.net/>

OpenCV library is used for handling input image reading and output image display. The Zbar library is used for searching and decoding the barcode from the input image.

The application is developed in C++ and all data processing is CPU based

**Characteristics:**

Some benchmark tests have been made using the application to highlight decoding precision and execution time. We came up to the following conclusions:

* The barcode should be at least 400px long in the input image for a succesfull decoding.Auth
* The ambient light doesn’t affect the decoding process as long as the barcode is visible.
* The image resolution incereases the execution time.