Annular Codes, © xeraina, Switzerland

Manuals

Author: C. Stamm

Date: September 19, 2023

Version: 1.23.38

1 Encoder

Encoder is an (interactive) console application for Windows to create, visualize and test Annular Codes.

In interactive mode the properties of an Annular Code can be set during code parametrization and stored in a JSON file. The code can be created, visualized, and tested (if a valid decoder license is available). The current settings of the encoder are stored in “settings.json”.

In command line mode a JSON file containing code properties can be read and the corresponding code can be created, visualized, and tested (if a valid decoder license is available).

1.1 Main Menu

Key	Menu	Description
c	change annular code name	Change the default name of the output file.
f	set output format	One of the following file formats: PNG, EMF, DXF (DXF is not available for all designs). Extended Metafile (EMF) has the best output quality.
g	logging on console	Enable/disable logging on console.
l	set logging level	Set the logging level to one of the following values: All, Debug, Info, Warning, Error, Fatal, Off
o	create and save annular code	Use the code properties, creates an annular code, and writes the code in the selected design and output format to an output file.
p	parametrize code	see on next page for further details
r	read code properties	Read code properties from a json file.
s	set code properties file name	Set the properties file name (default: config.json).
t	test created annular code	Try to decode the created annular code. Since this process needs access to the decoder, a valid license is necessary.
u	show command line parameters	Show the usage of this application in non-interactive command line mode.
v	view annular code	Visualize an annular code after it has been created. Press ESC with focus on the output window to terminate the visualization.
x	exit	Exit interactive mode and save settings.

1.2 Annular Code Parametrization Menu

Key	Menu	Description
a	set data type	One of the following payload data types: - Auto: automatically determined - digits: 0..9 - letters: 0..9, '.', ' ', A..Z, a..z - ASCII: 7 bit ASCII codes - Latin1: 8 bit Latin-1 codes
b	set background color	The background color (outside the annular code and in the center).
c	set code type	Black-and-White or Color
d	set data	The payload data.
e	set ECC strength	Set the error correction code strength in percent of the total code word. Supported values are: 7% (low), 15% (medium), 25% (quartile), and 30% (high).
f	set number of inner zebra rings	minimum: 1
g	set number of outer zebra rings	minimum: 1
i	set number of inner black rings	minimum: 1
j	set number of inner and outer quiet area rings	minimum: 1
k	set dpi	Resolution in dot per inch (dpi) 0: use screen resolution
l	set module size in mm	Module height in millimeters. At least two values of the set {l, m, n} or {o, p, r} must be set.
m	set inner diameter in mm	Diameter of the inner background area.
n	set outer diameter in mm	Diameter of the annular code area (including quiet zone).
o	set module size in px	Module height in pixels. At least two values of the set {l, m, n} or {o, p, r} must be set.
p	set inner diameter in px	Diameter of the inner background area.
q	quit	Quit parametrization mode without saving the code properties.
r	set outer diameter in px	Diameter of the annular code area (including quiet zone).
s	set code properties file name	Set the properties file name (default: config.json).
v	set line width in mm	Set virtual pen line width for drawing the code. Line widths are not supported in EMF output format.
w	set line width in px	Set virtual pen line width for drawing the code. Line widths are not supported in EMF output format.
x	exit	Write code properties to json file and return.
y	set design	Choose between one of the predefined code designs. The available designs (except outline) are depicted on Annular Codes – xeraina in the following order: normal, rounded corners, round modules, white crossing black crossing.
z	optimize capacity	Enable/disable capacity optimization.

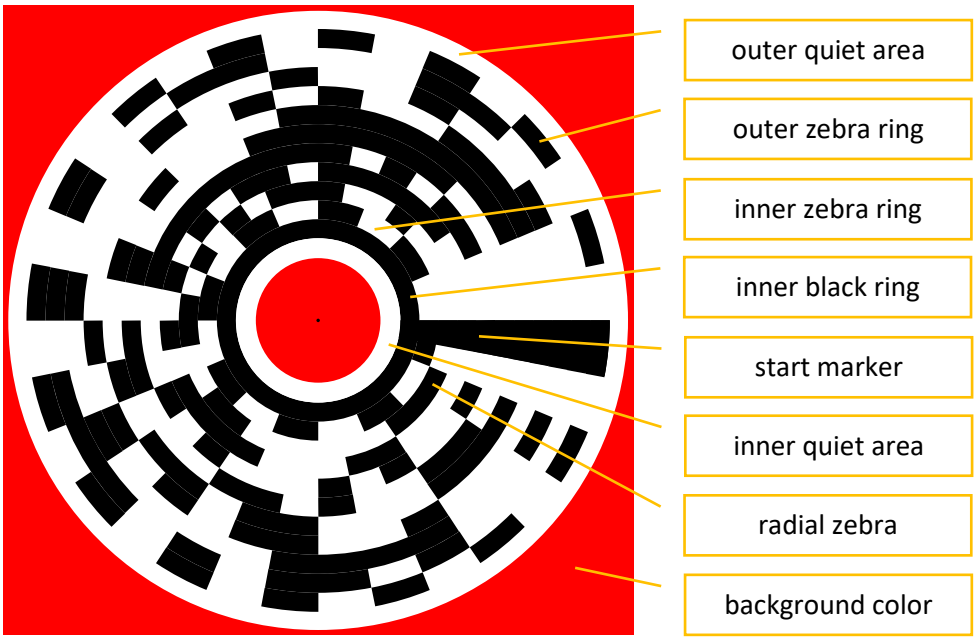

Figure 1: elements of annular codes

2 Decoder (Video Demo)

DecoderVideoDemo is an (interactive) console application for decoding annular codes using a camera or an image file. A valid decoder license is necessary.

In interactive mode a camera is used to create a snapshot of an annular code. Beford the camera is started, the camera backend, mode, and resolution can be set. The image of an annular code is decoded, and the output is listed. While the camera is taking snapshots the images can be inverted or distorted by noise, ellipses, and perspective transforms in different grades. The current settings of the decoder are stored in “decoder_settings.json”.

In command line mode an image file containing an annular code is read and decoded. The output is written to the console.

2.1 Main Menu

Key	Menu	Description
b	set camera backend	Set the camera backend to one of the following values: 0: default, 700: Directshow, 1400: Microsoft Media Foundation, 200: Video for Linux.
c	set code type	Set annular code type: Black-and-White or Color. Can be toggled during camera view.
d	set device number	To open default camera set 0. The device number is used only if the launch string is empty.
h	set frame height	Set the camera frame height. Only listed heights are allowed.
i	this help	Description of menu options.
l	set logging level	Set the logging level to one of the following values: All, Debug, Info, Warning, Error, Fatal, Off
m	set camera mode	Camera specific mode or -1 for camera default mode.
p	set camera launch string	Supported launch strings depend on the available camera backends. Typical examples are: "http://website/stream.mjpg" "rtsp://user:pwd@website:port/stream.sdp" "libcamerasrc camera-name=\"/base/soc/i2c0mux/i2c@1/imx219@10\" ! video/x-raw,width=640,height=480 ! appsink name=appsink"
s	start camera	Start camera and use the selected camera device, backend, and resolution. Stop the camera by pressing ESC inside of the camera view.
u	show command line parameters	Show the usage of this application in non-interactive command line mode.
v	show full camera launch string	Supported launch strings depend on the available camera backends.
w	set frame width	Set the camera frame width. Only listed widths are allowed.
x	exit	Exit interactive mode and save settings.

2.2 Camera View Menu

Key	Menu	Description
space	reset	Disable all distortions and set black and white code type.
0..9	set level	Set distortion level between 0 and 9 of the preset distortion state.
a	set area state	Set area destruction state. Use keys 0..9 to set destruction level.
b	toggle binarization	Toggle between color or binarized video frame.
c	toggle black-and-white/color	Toggle between black-and-white and color code type.
e	toggle image enhancement	Toggle between normal or enhanced video frame.
i	toggle image inversion	Toggle image inversion.
n	set noise state	Set noise state. Use keys 0..9 to set noise level.
p	set perspective state	Set perspective transform state. Use keys 0..9 to set perspective transform strength.
t	toggle decoding time	Toggle measuring decoding time.
v	toggle camera output	Toggle camera output.
x	stop camera	Stop camera and return to console menu.