Cybercity: a Practical Approach to Teach and Learn Cybersecurity in Smart Cities

Luca De Vito
Dept. of Engineering
University of Sannio
Benevento, Italy
devito@unisannio.it

Salvatore Bramante, Mauro D'Angelo
Perlatecnica
Maddaloni (CE), Italy
salvatore.bramante@perlatecnica.it
mauro.dangelo@perlatecnica.it

Galia Marinova
Department Technologies and
Management of Communication Systems
Technical University of Sofia
Sofia, Bulgaria
gim@tu-sofia.bg

Javier Orozco-Messana
Dept. of Mechanical and Materials Engineering
Technical University of Valencia
Valencia, Spain
jaormes@cst.upv.es

***Abstract*—This paper presents the activities of the Cybercity project. It aims to develop an innovative framework for teaching and learning cybersecurity in a practical way, using an educational platform consisting of a smart city scale model, where the students can practice security threats, attacks and countermeasures. In this paper, the main objectives of the project are presented, and the first activities are discussed. In particular, the design and the implementation of the educational platform are presented.**

***Index Terms*—Cybersecurity, Internet of Things, Smart City Education**

I. Introduction

As urban areas evolve into smart cities, integrating advanced technologies to enhance the quality of life, their main challenge becomes cybersecurity [1]. Smart cities leverage the Internet of Things (IoT), artificial intelligence (AI), and big data to optimize services such as transportation, energy management, healthcare, and public safety. However, this interconnectedness also introduces new vulnerabilities and threats that must be addressed to ensure the security and privacy of citizens.

The risks associated with cyberattacks are continually evolving, making cybersecurity education a dynamic field that can be challenging to teach at times [2]. Although universities are trying to cope with the demand for cybersecurity experts by including cybersecurity degrees in their programs, fully theoretical teaching is ineffective, particularly in the new networking scenarios, including Internet of Things (IoT) and edge computing [3].

Training environments and materials can support education activities in the field of cybersecurity by providing challenging situations and tools where the students can experiment under well-defined procedures and guidance. The training environment should allow different attack and defense mechanisms and must be able to adapt to a variety of different incidents [4].

Based on the above-mentioned considerations, the scientific literature has proposed several testbeds for practical teaching-learning and research. In [5], the authors proposed the Idaho CPS SCADA Cybersecurity testbed at the University of Idaho (ISAAC), an adaptive and reconfigurable cyber-physical systems testbed. This testbed has been created explicitly for conducting real-time experimental studies to analyze and monitor the effects of potential cyber-attacks, as well as assess the effectiveness of innovative cybersecurity measures on the smart grid. The testbed utilizes actual automation controllers, Intelligent Electronic Devices (IEDs), SCADA software, and cyber-physical networks. In [6], a testbed for IoT has been proposed, incorporating multiple networking layers and heterogeneous devices, to aid in networking research, anomaly detection, and the application of security principles targeted at IoT for educational and research purposes. This testbed offers a sophisticated yet practical hands-on setting that can be duplicated by those interested in developing a similar system. The testbed includes a Smart City model which utilizes a DirectLogic programmable logic controller (PLC), allowing students and researchers to interact with mechanical elements via the modbus protocol. In [7], the Authors developed a specialised cyber-physical testbed tailored for transportation critical infrastructure, featuring a simplified yet effective automated level-crossing system. The developed testbed enables the examination of the complex relationship between cyber-attacks and their physical impacts, allowing professionals to

This work was funded by the European Union, in the framework of the ERASMUS+ Programme, Action KA220-HED - Cooperation partnerships in higher education, Project Cybercity: a Practical Approach to Teach and Learn Cybersecurity in Smart Cities, Grant Agreement 2023-1-IT02-KA220-HED-000157723.

979-8-3315-0557-8$31.00 ©2024 IEEE

create and test cybersecurity solutions designed to protect vital infrastructure and industrial systems.

A smart city model, intended for education purposes but not specifically addressing cybersecurity, is presented in [8]. The proposal, called OpenCity, is an open architecture testbed for smart cities, featuring data collection and processing units, database management, distributed performance management algorithms, and real-time data visualization. The OpenCity architecture consists of decentralized software services and cyber-physical entities that are connected via the Message Queuing Telemetry Transport (MQTT) protocol.

Although several education testbeds and related materials have been developed to support cybersecurity education, most of the solutions are only based on virtualization and networking scenarios, and few of them concentrate on the replication of attacks to Cyberphysical Systems, Industrial Control Systems and operational technology assets. Moreover, among the analyzed literature proposals, only [6] includes a Smart City model, to allow the practical training of cybersecurity attack and protection of an upcoming smart city. However, it contains a single controller on the infrastructure side, without the possibility of emulating attacks on vehicles. To overcome such limitations, the Cybercity project has been proposed. It consists of a Smart City model where both infrastructure nodes and vehicles are equipped with programmable microcontrollers and can be victims of cybersecurity attacks. This paper describes the objectives of the Cybercity project and the main planned outcomes.

After the introduction, the paper is organized as follows: In Section II, the main objectives of the Cybercity project are drawn. Section III reports the design of the platform of the project. In Section IV, the planned didactical material that will be prepared to accompany the platform is discussed while Section V-A describes in detail the implementation of each component of the platform. Finally, in Section VI, conclusion and further developments are drawn.

II. Objectives of the project

As mentioned above, the dependence of Smart Cities on ICT makes them prone to cyber-attacks. The objective of the Cybercity project is to provide a field of action where students can practically experiment and become proficient in smart city management and understanding, focusing on cybersecurity, which is one of the most important aspects of their functioning. As it has been observed that most of the didactical efforts in the field of cybersecurity are theoretical, the project also aims to train and support university professors to prepare them for being able to use a practical didactical approach. Therefore, the Cybercity project chases two main objectives:

1) The development of an *Integrated system for cybersecurity*, consisting of a smart city model with self-driving vehicles, and
2) The preparation of a *Training course*, targeted to University teachers, to support them in the practical teaching of cybersecurity for Smart Cities.

In the next Sections, the two main objectives will be discussed in detail.

III. Integrated system for cybersecurity

The educational platform to be developed in the project will be a miniature city equipped with networked objects and systems that emulate the typical services of a smart city. The smart objects of the educational platform will be programmable with the possibility of introducing *ad hoc* vulnerabilities.

The following nodes in the network are foreseen:

- **Data Management and Control System (DMCS)**: The system managing the data collected from the sensors. All the sensor nodes of the city and the vehicles will send the information collected to a centralized data management and control system of the smart city systems.
- **Traffic Light Management System (TLMS)**: The traffic lights that manage vehicle flows at the city's intersections will be governed by a microcontroller-based control system that implements the traffic light cycle, transmits its status and receives commands from the DMCS.
- **Autonomous Vehicles (AV)**: Two types of self-driving vehicles will be developed, the former governed by an STM32H7 microcontroller and the latter by an Nvidia Jetson Nano type microprocessor board. Both will implement the same mission: ”the vehicle moves autonomously within its own lane, respecting road signs and traffic lights, stopping in the presence of an obstacle and then resuming navigation when the obstacle is removed”, and transmit their status and receive commands from the DMCS.
- **Hacker**: The intruder, emulating attacks on the city and the vehicles.

The vehicles, the traffic light control systems, and any additional system will communicate their status and will receive commands from the DMCS data management and control system (Fig. 1).

All the above-mentioned nodes will be in a local network, called SCEL-net (Smart City Edu Lab) (see Fig. 2). The network is created by a router, which will provide both Internet access and wireless access over WiFi to the local network for all the city's nodes.

The educational platform will support cybersecurity attacks, including:

- *lateral movement*, which is one of the phases of a cyber attack in which an attacker attempts to move laterally or horizontally across a network or computer system after gaining initial access. The attacker's goal during this phase is to expand his presence within the target environment, attempting to reach and compromise additional resources, user accounts, or devices;
- *man in the middle*, consisting of a device intercepting and selectively modifying communicated data to masquerade as one or more of the entities involved in a communication association;

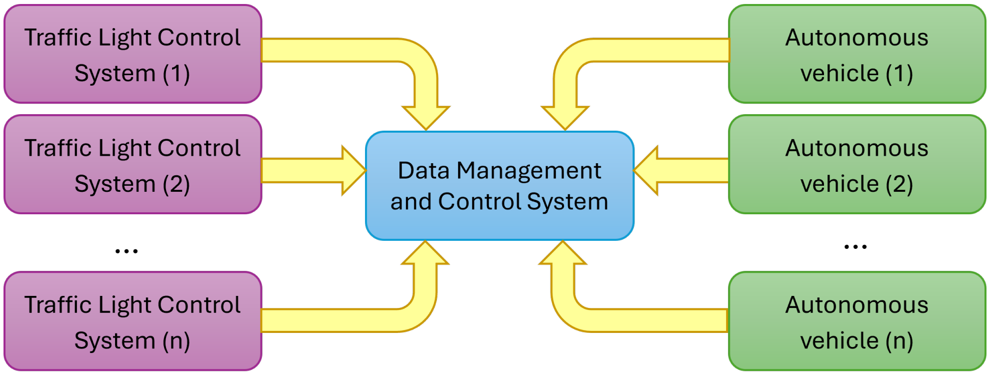

Traffic Light Control System (1)
Traffic Light Control System (2)
...
Traffic Light Control System (n)
Data Management and Control System
Autonomous vehicle (1)
Autonomous vehicle (2)
...
Autonomous vehicle (n)

Fig. 1. Overall system architecture of the Cybercity platform.

- *password cracking*, an attack in which an attacker tries to guess or crack an account password to gain unauthorized access;
- *Denial of Service (DoS)*, consisting of the prevention of authorized access to resources or the delaying of time-critical operations.

IV. Cybersecurity course

The second main outcome of the project will be the preparation of a course, targeted to University teachers and containing methodological and pedagogical guidelines as well as educational paths for structuring lessons, using the integrated Cybercity platform as a testbed.

The course will be implemented in asynchronous mode and then uploaded to the project learning management system, in order to be always available and accessible to all. The course will be complemented by an in-presence session of 5 days, which will put into practice what has been addressed theoretically in the course. This will give professors the necessary knowledge and skills to use the project outputs in their lessons as teaching support. Considering that the target group of the course contains experts in the cybersecurity domain and professional educators, the course will be centred on the use and functioning of the project tools from an educational standpoint and on the creation of tailored activities to make students reach the level of competence required. This will contribute to reach two of the project objectives, more specifically:

- Providing a transdisciplinary approach to enhance STEM disciplines and more specifically cybersecurity;
- Strengthening higher education organisations' digital skills;
- Providing new learning and teaching methods and approaches.

The course will be organised in 3 modules, including 10 lessons of 3 hours each. The topics that will be addressed during the lessons are:

Module 1 - Basics of the cybersecurity system and functioning of the city: this module provides a referential framework for the course introducing Intelligent transport in Connected Cities and Smart Cities from a cybersecurity perspective;

Module 2 - Addressing cybersecurity through the integrated system: this module will provide in the first part the basics of IoT sensor data information and IoT communication interfaces and protocols used in Smart Cities. Then, it will present the methodologies and tools for assessing the security risks and vulnerabilities and for evaluating and testing the security.

Module 3 - Attacks simulation: This module will present how to simulate attacks on the Cybercity platform and how to implement proper countermeasures.

V. Implementation of the integrated system for cybersecurity

A. Data Management and Control System

As mentioned above, the DMCS has the role of receiving messages containing status information or data from the nodes in the SCEL-net and sending commands to the managed nodes.

The DMCS has been implemented by splitting its functionalities into two subsystems:

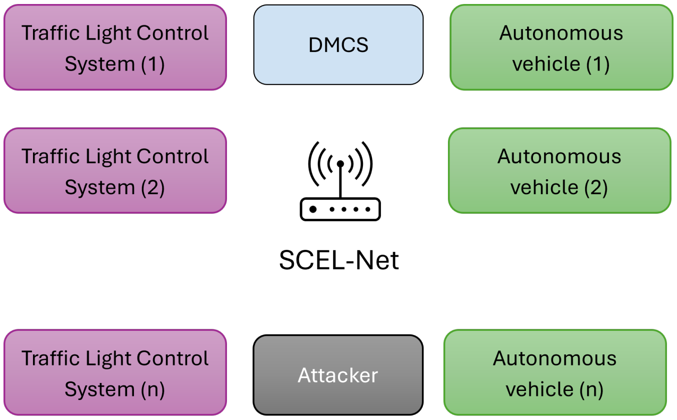

Traffic Light Control System (1)
DMCS
Autonomous vehicle (1)
Traffic Light Control System (2)
Autonomous vehicle (2)
SCEL-Net
Traffic Light Control System (n)
Attacker
Autonomous vehicle (n)

Fig. 2. Network connection of the Cybercity nodes.

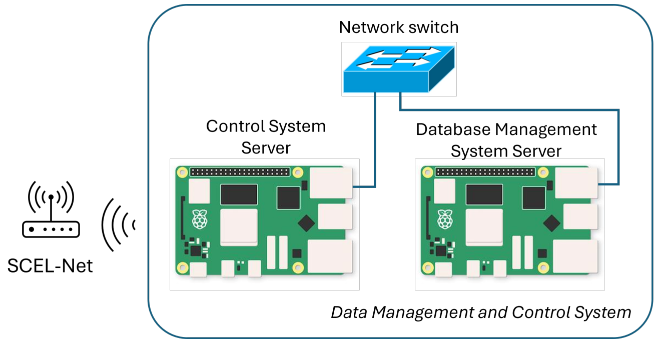

Network switch
Control System Server
Database Management System Server
SCEL-Net
Data Management and Control System

Fig. 3. Block scheme of the Data Management and Control System implementation.

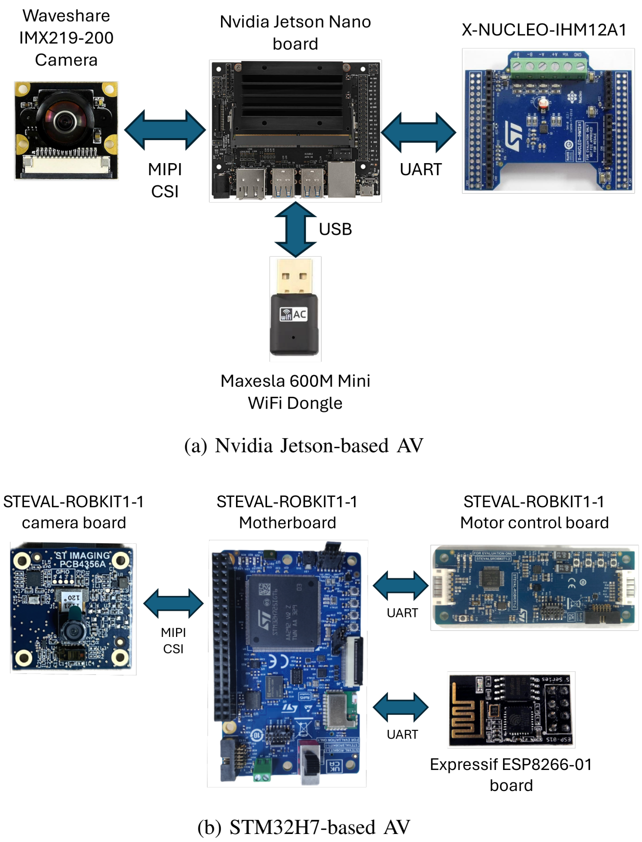
- A Control system Server, in charge of exporting the functionalities of the DMCS in the SCEL-Net;
- the Database Management System (DBMS) Server, consisting of the DBMS in charge of storing the DMCS data.

The two subsystems are connected between them in a private network, as shown in Fig. 3, while the sole Control System Server subsystem is directly connected to the SCEL-Net, by Wifi. Therefore, when a request from an element of the SCEL-Net comes for either reading the status of the network or updating the status with some data, the Control system Server will manage that request and will interface with the DBMS Server for reading/writing the data. Both the Control system Server and the DBMS Server are implemented with a Raspberry Pi 4, equipped with the Ubuntu operating system. The control system server runs a Python daemon, implementing a TCP (Transmission Control Protocol server that accepts JSON-coded requests. The DBMS Server consists of a MariaDB relational DBMS.

Access to the DMCS, and in particular to the information stored in the database is managed using 4 different user roles:

0. *Superuser* - Has full control over the database and can perform any action. Only one super user is allowed.
1. *Admin* - Admin users have extensive privileges within the system. They can perform all actions except restoring the entire database. They have the authority to delete the table content and perform actions related to accounts.
2. *Moderator/Referee* - Moderators, such as referees, have the ability to view the content of all nodes but are restricted from making modifications.
3. *Teacher/Student* - Teacher and student users have limited permissions. They can only perform actions on nodes related to them and are restricted from viewing content that does not belong to them.

These access rights definitions ensure that different entities within the system have appropriate levels of access and permissions, thereby maintaining data integrity and security. Please note that these access rights should be assigned carefully to ensure that users have the necessary permissions to perform their tasks without compromising the security of the database.


Waveshare IMX219-200 Camera
Nvidia Jetson Nano board
X-NUCLEO-IHM12A1
MIPI CSI
UART
USB
Maxesla 600M Mini WiFi Dongle
STEVAL-ROBKIT1-1 camera board
STEVAL-ROBKIT1-1 Motherboard
STEVAL-ROBKIT1-1 Motor control board
MIPI CSI
UART
UART
Expressif ESP8266-01 board

(a) Nvidia Jetson-based AV

(b) STM32H7-based AV

Fig. 4. Block scheme of the autonomous vehicles: (a) Jetson-based AV, (b) STM32H7-based AV.

B. Autonomous vehicles

As mentioned in Section III, two different models of AVs have been developed within the project, sharing the same structure. The former has the block scheme shown in Fig. 4a. Its core is based on a Nvidia Jetson Nano board. This board has been selected due to its capabilities of processing images as the objective of the AV is to navigate autonomously based on the camera information, recognizing obstacles and traffic lights. The Jetson board is interfaced with:

- a Waveshare IMX219 camera, equipped with a Sony IMX219 8M pixel camera sensor, to acquire images about the surrounding environment, through a MIPI (Mobile Industry Processor Interface) Camera Serial Interface (CSI);
- an ST Microelectronics X-NUCLEO-IHM12A1 expansion board [9], equipped with an STSPIN240 low voltage dual brush dc motor driver, for the motor control, through UART (Universal Asynchronous Receiver-Transmitter) interface, and
- a Maxesla IEEE 802.11ac 600M Mini WiFi Dongle, for the wireless connection to the SCEL-Net.

The latter AV model is based on the STEVAL-ROBKIT1-1 by ST Microelectronics. It consists of the following 3 boards:

- a motherboard, equipped with an STM32H725 microcontroller, which is in charge of executing the most compu-

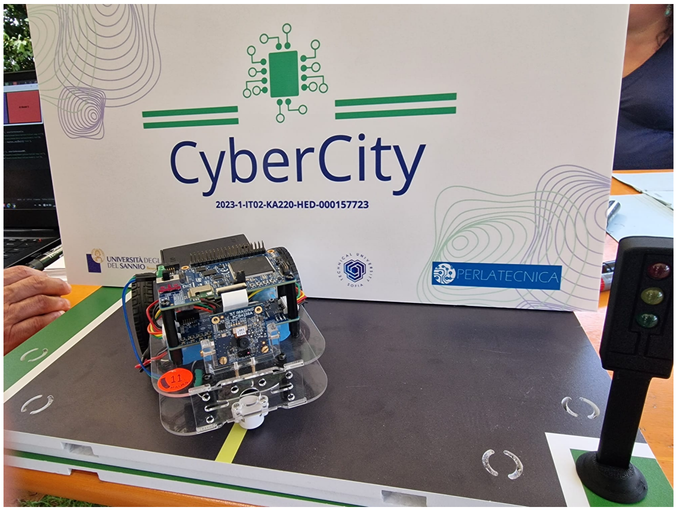


Fig. 5. Picture of the STM32H7-based AV.

tationally intensive tasks, such as image processing;

- a camera board, equipped with an ST Microelectronics VD56G3 1.53 M Pixel sensor module [10], for acquiring images of the surrounding environment, and with a VL53L8 8x8 multizone Time-of-Flight sensor [11];
- a motor control board, equipped with an STM32G071CBT6 microcontroller, and a STSPIN240 dual brush dc motor driver.

The three boards are connected as shown Fig. 4b. The camera board is connected to the motherboard by the MIPI CSI, while the motor control board is connected to the motherboard by UART interface. In addition, an Expressif ESP8266-01 board has been connected to the STEVAL-ROBKIT1-1 motherboard through UART interface for the wireless connection with the SCEL-Net. A picture of the STM32H7-based AV is shown in Fig. 5.

C. *Traffic Light Control System*

The Traffic Light Control System has been implemented according to the block scheme of Fig. 6. It consists of a Nucleo-F401RE board [12] from ST Microelectronics, equipped with an STM32F401RE microcontroller. This board interfaces with a set of KY-016 RGB LEDs, which realize the traffic light functionality through the General Purpose Input/Outputs, and with a further Expressif ESP8266-01 board for the wireless connection to the SCEL-Net.

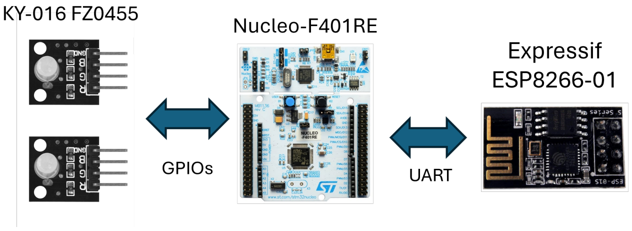


Fig. 6. Block scheme of the Traffic Light Control System.

VI. Conclusion and further work

In this paper, the first activities of the Cybercity project have been presented. The project has two main objectives: (i) the design and implementation of a small-scale smart city, which could serve as an educational platform for cybersecurity training; (ii) the definition and the deployment of a cybersecurity course with the support of the developed platform. In the first year of the project, the first objective has been faced and the design of the educational platform has been presented in this paper. The design was driven by the double requirement of practising IoT and Smart-City concepts while allowing the emulation of attacks with the possibility of moving from one system to another.

Further work will be directed to a characterization of the developed platform and to the detailed definition of the content of the cybersecurity course.

References

[1] M. Alamer and M. A. Almaiah, “Cybersecurity in smart city: A systematic mapping study,” in *2021 International Conference on Information Technology (ICIT)*, 2021, pp. 719–724.

[2] M. A. Khan, A. Merabet, S. Alkaabi, and H. El Sayed, “Game-based learning platform to enhance cybersecurity education,” *Educational and Information Technologies*, vol. 27, pp. 5153–5177, 2022.

[3] M. Ficco and F. Palmieri, “Leaf: An open-source cybersecurity training platform for realistic edge-iot scenarios,” *Journal of Systems Architecture*, vol. 97, pp. 107–129, 2019.

[4] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, and M. A. Ferrag, “Cyber ranges and testbeds for education, training, and research,” *Applied Sciences*, vol. 11, no. 4, 2021.

[5] I. A. Oyewumi, A. A. Jillepalli, P. Richardson, M. Ashrafuzzaman, B. K. Johnson, Y. Chakhchoukh, M. A. Haney, F. T. Sheldon, and D. C. de Leon, “Isaac: The idaho cps smart grid cybersecurity testbed,” in *2019 IEEE Texas Power and Energy Conference (TPEC)*, 2019, pp. 1–6.

[6] J. Thom, T. Das, B. Shrestha, S. Sengupta, and E. Arslan, “Casting a wide net: An internet of things testbed for cybersecurity education and research,” in *2021 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2021.

[7] S. Hosseinzadeh, D. Voutos, D. Barrie, N. Owoh, M. Ashawa, and A. Shahrabi, “Design and development considerations of a cyber physical testbed for operational technology research and education,” *Sensors*, vol. 24, no. 12, 2024.

[8] N. Zohrabi, P. J. Martin, M. Kuzlu, L. Linkous, R. Eini, A. Morrissett, M. Zaman, A. Tantawy, O. Gueler, M. A. Islam, N. Puryear, H. Kalkavan, J. Lundquist, E. Karincic, and S. Abdelwahed, “Opencity: An open architecture testbed for smart cities,” in *2021 IEEE International Smart Cities Conference (ISC2)*, 2021.

[9] ST Microelectronics, “X-NUCLEO-IHM12A1 - Low voltage dual brush DC motor driver expansion board based on STSPIN240 for STM32 Nucleo,” https://www.st.com/en/ecosystems/x-nucleo-ihm12a1.html.

[10] ——, “CAM-56G3 - VD56G3 promodules: camera module evaluation samples for VD56G3 image sensor,” https://www.st.com/en/evaluation-tools/cam-56g3.html.

[11] ——, “VL53L8CX - Low-power high-performance 8x8 multizone Time-of-Flight sensor (ToF),” https://www.st.com/en/imaging-and-photonics-solutions/vl53l8cx.html.

[12] ——, “NUCLEO-F401RE - STM32 Nucleo-64 development board with STM32F401RE MCU, supports Arduino and ST morpho connectivity,” https://www.st.com/en/evaluation-tools/nucleo-f401re.html.