IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Advanced Pipeline Inspection Robot Development Utilizing Non-Destructive Evaluation and Simulated with Verilog

XIANGBO CAI

Department of Electrical and Computer Engineering, Michigan State University, MI 48825, USA. Email: caixian3@msu.edu

ABSTRACT This project focuses on the creation of a pipeline inspection robot to deal with the pipeline inspection problem. The project utilizes the concept of non-destructive evaluation, which can achieve the function that without the need to damage the pipelines, the robot can be sent into the pipeline to execute the specific assignment given by human. The circuits are designed to control the robot's movements within the pipes. The robot is equipped with functions such as forward and backward movement, inspection of pipeline issues, and providing feedback on detected problems. Additionally, a Finite State Machine (FSM) is designed to diagram for the robot and the corresponding Verilog code is implemented. Besides, simulation tests using two sets of predefined data are conducted and the waveforms are generated to compare the detection results.

INDEX TERMS NDE, Verilog Simulation, FSM, Pipeline Inspection Robot

I. INTRODUCTION

Pipeline inspection in urban environments presents persistent challenges [1]. Pipes, intricate in their construction, are often buried beneath the ground, making detection a formidable task. Resorting to surface excavation for inspection not only risks damaging existing infrastructure but also contributes to soil erosion. To address these issues, non-destructive evaluation (NDE) technology has emerged as a solution. NDE utilizes principles such as radiation, ultrasonics [2], infrared [3], electromagnetic fields, and eddy current [4] to assess materials, parts, and equipment without compromising their future functionality or current operational status. In terms of the pipeline inspection, NDE is also widely used, like eddy current for high-speed pipeline inspection [5][6], current deflection for pipeline monitoring [7].

In this report, the proposed solution is introduced as follows: a pipeline inspection robot designed to aid in human efforts and obviate the need for disruptive soil excavation during underground inspections [8]-[10]. Compact and agile, the robot seamlessly navigates pipelines, collecting crucial parameters and information. It evaluates the condition of the pipeline, promptly notifying of any repair needs. Moreover, it autonomously executes maintenance tasks underground, effectively substituting for human intervention. This innovative pipeline robot epitomizes the essence of non-destructive testing.

The primary objective of this article is to elucidate the technical underpinnings of pipeline non-destructive inspection robots using Verilog. Input parameters that the robot may require are meticulously constructed, alongside their corresponding output parameters. Leveraging Vivado Verilog code, the functionalities of the pipeline robot are implemented. Furthermore, two comprehensive testbenches are also furnished, employing generated waveforms to simulate real-world challenges. Through this research on pipeline NDE robots, this robot aims to furnish indispensable technical support and assurance for the secure operation and sustainable development of urban pipelines.

II. DESIGN

A. *Pipeline Robot Parameters*

The pipe maintenance robot is equipped with essential features to navigate and perform maintenance tasks within pipelines. Those parameters can precisely control the robot and help achieve the result expected.

Pipe Maintenance Robot's Finite State Machine (FSM) has featured the following inputs. A feature with [On, Off] button for control and a sensor with an 8-bit register to store its starting location, represented as [4b_x, 4b_y]. Additionally, it possesses a sensor to detect maintenance requirements in each physical space, indicating tasks such as "Fire", "Cool", "Plunge", or "Free". A compass aid in orientating the robot, with directional indications of North, East, South, or West [N, E, S, W]. Furthermore, it employs sensors to detect wall locations relative to its forward-facing direction, distinguishing between Left, Right, and Forward [L, R, F]. Crucially, the robot maintains a register to store its current location, facing direction, and states, facilitating efficient navigation and task execution within the pipeline environment. All sensor inputs are active-high and represented as one-hot vectors, except for the location vector, ensuring precise control and effective operation of the robot.

The outputs of the pipe maintenance robot's FSM are essential commands and information necessary for its operation and navigation within the pipeline. It includes instructions for turning, denoted as Turn Left or Turn Right, to adjust its orientation as needed. Additionally, there is an output to drive forward, enabling the robot to progress through the pipeline efficiently. The current location output, represented as [4b_x, 4b_y], provides crucial positional information for tracking the robot's movement along the pipeline. Moreover, an output indicating the action to be performed, such as "Fire", "Cool", or "Plunge", guides the robot in executing maintenance tasks as dictated by the pipeline's condition. All outputs are active-high and conveyed as one-hot vectors, ensuring precise control and effective communication between the robot and its operational environment.

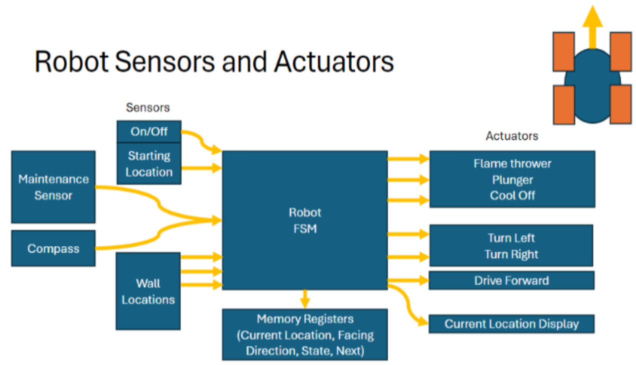

FIGURE 1. The Input and Output parameters of the robot.

B. *Pipeline Robot FSM Design*

Based on the configuration of relevant parameters and the preliminary design of the functions for the Pipeline Robot, the corresponding FSM diagram can be designed. The FSM consists of 20 states, including the "Off" state, "Idle" state, two turning states, three "Compass Shift" states, one movement recognition state, three "Hot" states, four "Plunge" states, and five "Frozen" states.

The detailed FSM Verilog code design for the pipeline inspection robot is documented in the repository [11]. The Verilog code can convert from the FSM diagram and can be executed to simulate and test the various properties of the pipeline inspection robot.

Different states connected by the input and output parameters defined in part A. According to the input and output information above, the related FSM diagram can be drawn as follows (with related inputs and outputs aside):

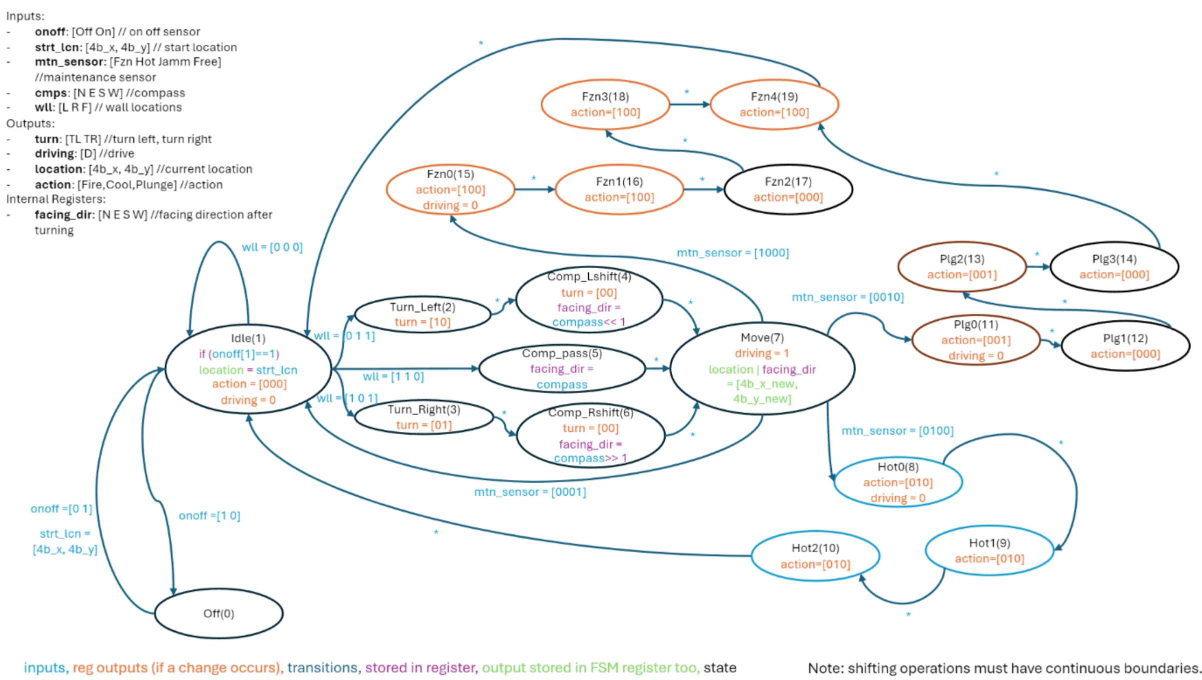

FIGURE 2. The FSM for the pipeline inspection robot

III. SIMULATION IN VIVADO VERILOG AND RELATED TESTBENCH

Verilog code based on the corresponding input and output can be built, and the corresponding functions can also be implemented. Corresponding code references are provided in the appendix. This code implements the state transition logic and action execution of the pipeline maintenance robot and ensures that it can accurately control the robot's behavior based on external input.

To better demonstrate the research results, the created corresponding simulation data is provided. As shown in Figures 4 and 5, this article provides simulation routes of two different robots and their corresponding data. These data not only verify the effectiveness of the model, but also demonstrate its application potential in different scenarios. Relevant test code and more detailed data can be found in the appendix to facilitate readers to deeply understand and reproduce the research results.

Inputs:
- **onoff**: [Off On] // on off sensor
- **strt_lcn**: [4b_x, 4b_y] // start location
- **mtn_sensor**: [Fzn Hot Jamm Free] //maintenance sensor
- **cmps**: [N E S W] //compass
- **wll**: [L R F] // wall locations

Outputs:
- **turn**: [TL TR] //turn left, turn right
- **driving**: [D] //drive
- **location**: [4b_x, 4b_y] //current location
- **action**: [Fire,Cool,Plunge] //action

Internal Registers:
- **facing_dir**: [N E S W] //facing direction after turning

FIGURE 3. FSM input and Output Vectors

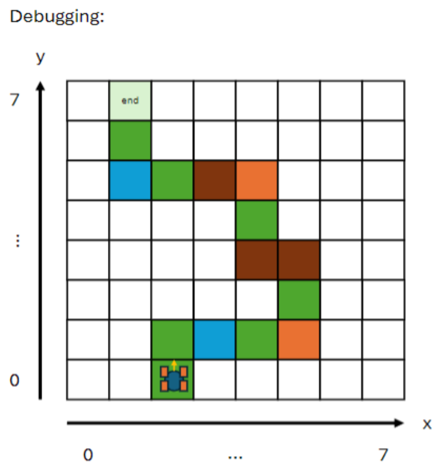

FIGURE 4. The first simulation test process

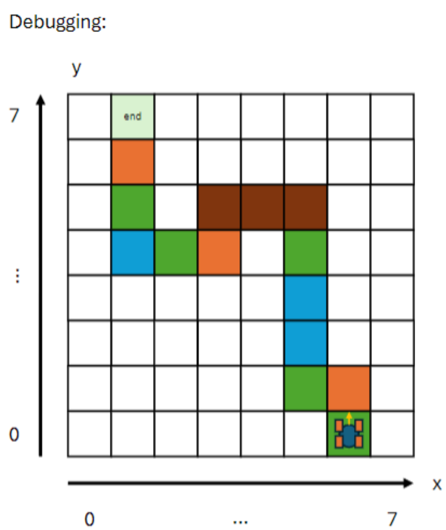

FIGURE 5. The second simulation test process

IV. SIMULATION RESULT – WAVEFORM ANALYSIS

The waveform plots of two different simulation results create representative models for this report. These waveforms are obtained by executing Verilog code in the Vivado environment, specifically after running the corresponding testbench. These waveform diagrams not only visually display the simulation results, but also present the dynamic change process of the data, reflecting the system's response and performance under different test conditions. Through these detailed and informative waveform diagrams, readers can more clearly understand and analyze the effectiveness of the design and its potential application value.

For the simulation results, two group tests were conducted. The first group simulated the path shown in Fig. 4, with the testbench code set according to the predefined trajectory in the figure and simulated using Verilog. Similarly, the second group simulated the path shown in Fig. 5, with the testbench code set accordingly. For further testbench code related to the pipeline FSM, the design and implementation details of the testbench can be found in the repository [11].

A . *First Simulation Test Result*

Based on the Figure 4 pathway, appropriate testbench code can be generated. Figure 6 showcases the overall simulation results, with Figures 7, 8, and 9 showing the "Frozen", "Hot" and "Plunge" individual waveforms.

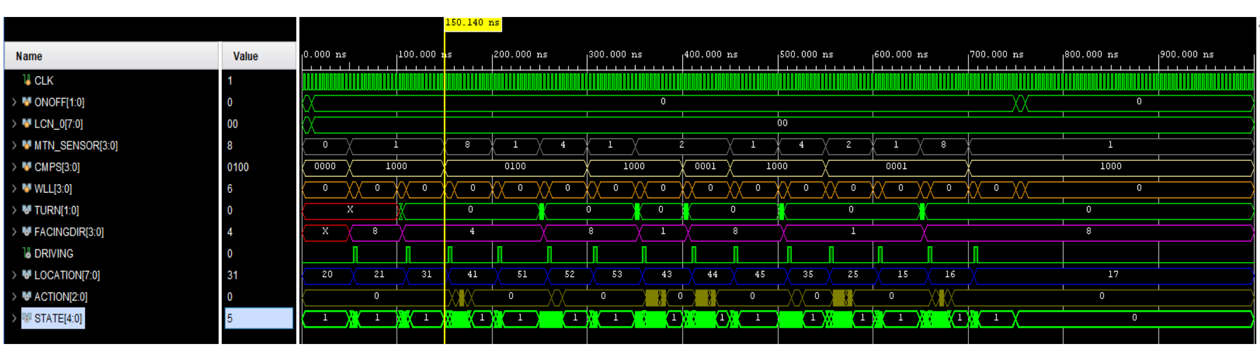

FIGURE 6. An entire overview of the "First Test" simulation waveform.

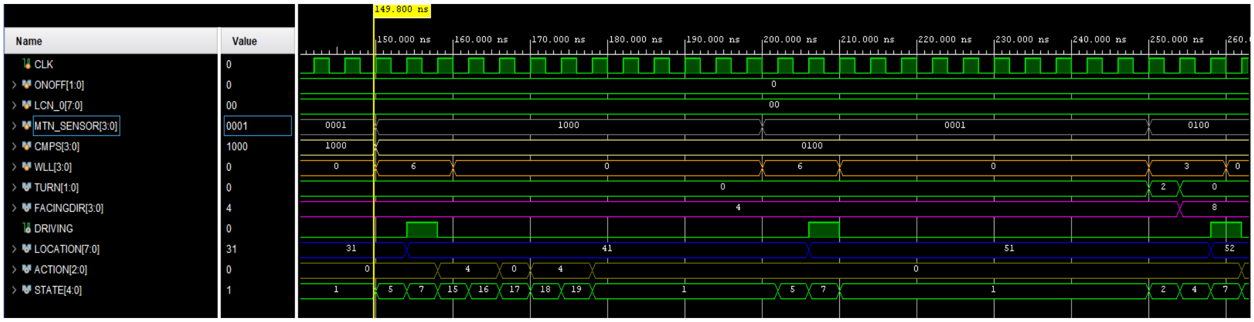

FIGURE 7. When the mtn_sensor=1000, "Frozen" mode executed, iterate the 15, 16, 17, 18, and 19 states.

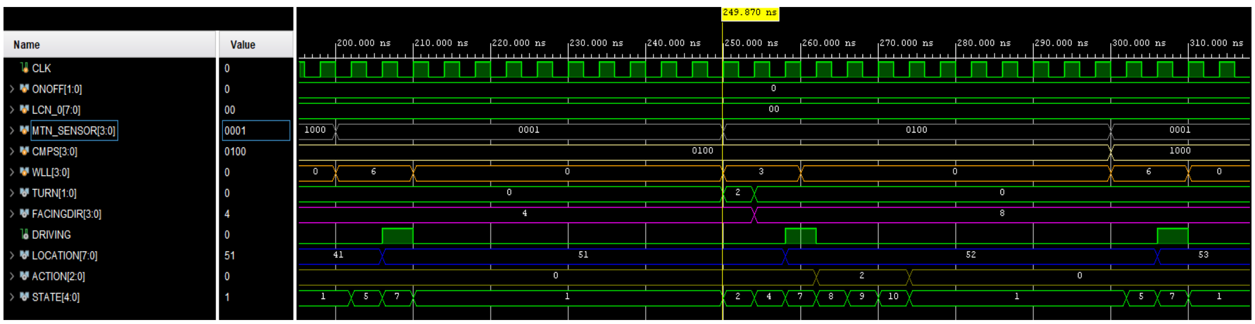

FIGURE 8. When the mtn_sensor = 0100, "Hot" mode executed, iterate 8, 9, and 10 states.

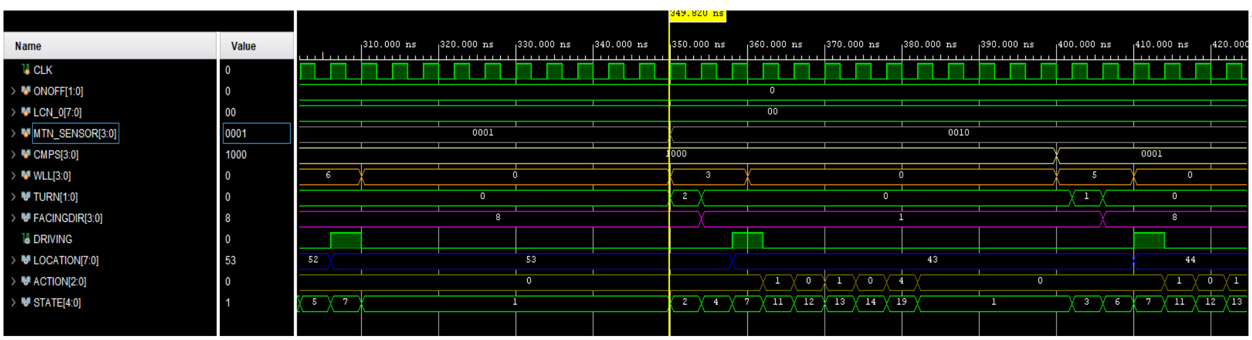

FIGURE 9. When the mtn_sensor=0010, "Plunge" mode executed, iterate 11, 12, 13, 14, and 19 states.

B. *Second Simulation Test Result*

Based on the pathway showcase in Figure 5, the related simulation testbench, similar to the First Simulation, can be shown. Figure 10 is an overall simulation result, while Figures 11, 12, and 13 show the "Frozen", "Hot", and "Plunge" testbench-based circle resu

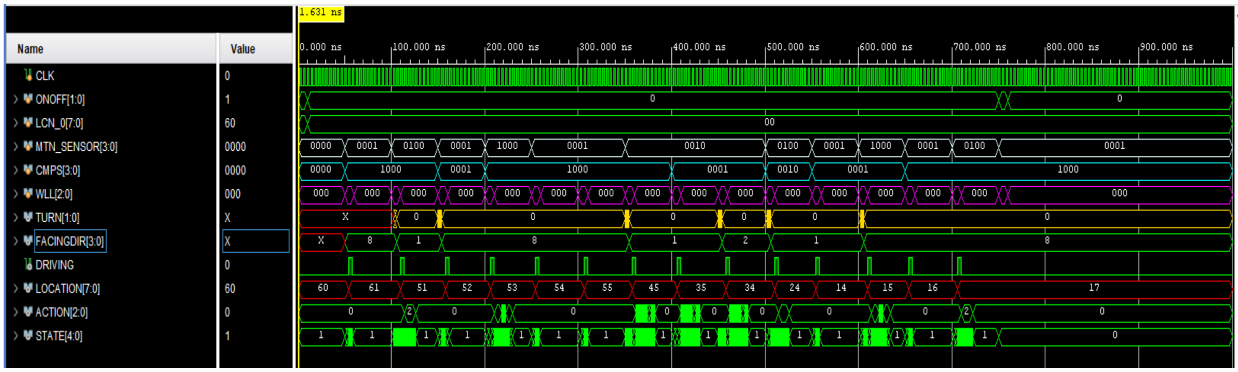

FIGURE 10. An entire overview of the "Second Test" simulation waveform.

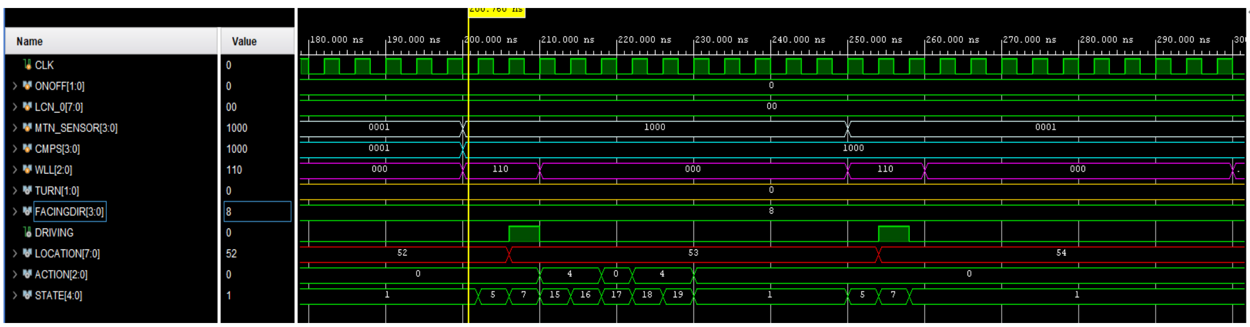

FIGURE 11. When the mtn_sensor = 1000, "Frozen" mode executed, iterate 15, 16, 17, 18, and 19 States.

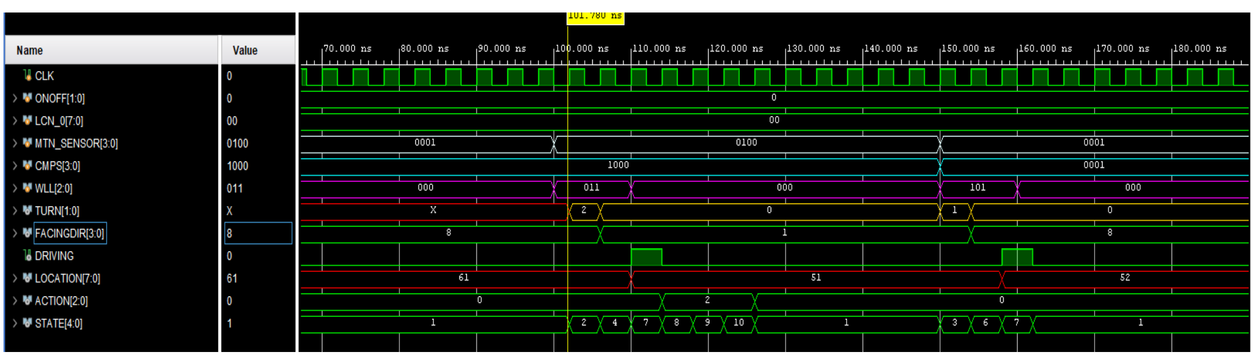

FIGURE 12. When the mtn_sensor = 0100, "Hot" mode executed, iterate 8, 9, and 10 states.

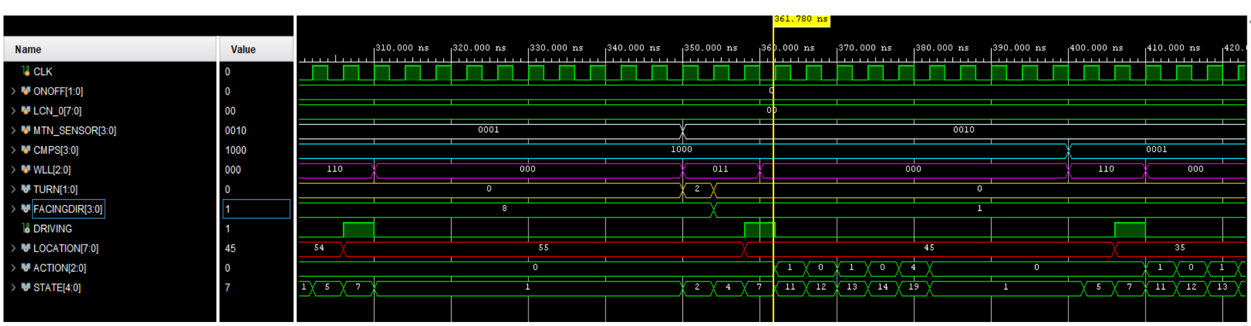

FIGURE 13. When the mtn_sensor = 0010, "Plunge" mode executed, iterate 11, 12, 13, 14, and 19 states.

V. CONCLUSION

With an increasing demand for more life-easing ideas, a steady stream of engineering challenges encountered will follow. The importance of NDE technology becomes increasingly prominent—and the need for developing more technology like this is becoming ever-necessary. The pipeline maintenance robot introduced in this article is a typical example of NDE technology in real-life applications. The robot utilizes advanced non-destructive testing methods, such as the robot instructional concept, and, through Verilog programming, implements related functions. This design significantly promotes the inspection and maintenance of urban pipelines with the utilization of NDE methods.

More generally, the design of a pipeline robot requires precise design of the result graph of its finite state machine (FSM), with further writing and executing corresponding code based on these design results being vital. During operation, the robot should identify the direction, detect various conditions within the pipeline, such as whether the temperature is abnormal (overheating or undercooling), and generate corresponding feedback through internal programs based on the detection results. In addition, the robot should determine, based on the position information on the left, right and front, the coordinate position of its next move.

Notably, there is still room for improvement in the design of pipeline robots. When inputting temperature abnormality information and judging direction parameters, the robot currently only sets 0 or 1 to indicate its status. This simplified processing method cannot meet the needs of more complex practical applications. It is expected that in future development, with the advancement of technology and optimization of parameter settings that, as this article has stepped towards, these problems will be rendered solved.

REFERENCES

[1] M. Z. Ab Rashid *et al.*, "Modeling of the in-pipe inspection robot: A comprehensive review," *Ocean Engineering*, vol. 203, p. 107206, May 2020. doi:10.1016/j.oceaneng.2020.107206

[2] J. H. Kurz, A. Jüngert, S. Dugan, G. Dobmann, and C. Boller, "Reliability considerations of NDT by probability of detection (POD) determination using ultrasound phased array," *Engineering Failure Analysis*, vol. 35, pp. 609–617, Dec. 2013. doi:10.1016/j.engfailanal.2013.06.008

[3] C. Ibarra-Castanedo *et al.*, "Infrared vision for artwork and Cultural Heritage Nde Studies: Principles and case studies," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 59, no. 5, pp. 243–248, May 2017. doi:10.1784/insi.2017.59.5.243

[4] Guang Yang *et al.*, "Pulsed eddy-current based giant magnetoresistive system for the inspection of Aircraft Structures," *IEEE Transactions on Magnetics*, vol. 46, no. 3, pp. 910–917, Mar. 2010. doi:10.1109/tmag.2009.2032330

[5] M. Z. Ab Rashid *et al.*, "Modeling of the in-pipe inspection robot: A comprehensive review," *Ocean Engineering*, vol. 203, p. 107206, May 2020. doi:10.1016/j.oceaneng.2020.107206

[6] H. M. Kim and G. S. Park, "A new sensitive excitation technique in nondestructive inspection for underground pipelines by using differential coils," *IEEE Transactions on Magnetics*, vol. 53, no. 11, pp. 1–4, Nov. 2017. doi:10.1109/tmag.2017.2713042

[7] R. Jarvis, P. Cawley, and P. B. Nagy, "Current deflection NDE for the inspection and monitoring of pipes," *NDT & E International*, vol. 81, pp. 46–59, Jul. 2016. doi:10.1016/j.ndteint.2016.03.006

[8] Y.-S. Kwon and B.-J. Yi, "Design and motion planning of a two-module collaborative indoor pipeline inspection robot," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 681–696, Jun. 2012. doi:10.1109/tro.2012.2183049

[9] A. Nayak and S. K. Pradhan, "Design of a new in-pipe inspection robot," *Procedia Engineering*, vol. 97, pp. 2081–2091, 2014. doi:10.1016/j.proeng.2014.12.451

10] I. N. Ismail *et al.*, "Development of in-pipe inspection robot: A Review," *2012 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (STUDENT)*, Oct. 2012. doi:10.1109/student.2012.6408425

[11] X. Cai. "ECE230-DP04-pipeFSM." GitHub. https://github.com/caishamble/ECE230-DP04-pipeFSM. May 16, 2024.

Xiangbo Cai is a second-year undergraduate student at Michigan State University pursuing a B.S.E in Computer Engineering within the Department of Electrical and Computer Engineering. As an Honors College student, he is currently an undergraduate researcher, under Dr. Yiming Deng, at the Non-Destructive Evaluation Laboratory (NDEL) at MSU.

His research currently centers on the development and application of electromagnetic and robot-detection-oriented non-destructive evaluation (NDE) techniques, with a particular focus on pipeline detection. Xiangbo is dedicated to advancing NDE technologies, helping to innovate practical solutions in modern electronics within the field of engineering.