An Introduction to Programmable Logic Controllers and How They Relate to Currently Taught Material

Programmable Logic Controllers (PLC's) are industrial controllers used to connect actuators, electronics, sensors and software systems to create "complex large applications" [1] in a variety of industry sectors: Energy, manufacturing, oil and gas, power, transportation, water and waste [2]. From the perspective of a mechatronics and robotics student starting in academic year 2016/17, they are not included in the curriculum despite many elements of PLC's being taught in various modules and their use in industry. What follows are a series of links connecting the understanding of PLC's to what is currently taught in the curriculum.

The first link to the curriculum is the structure of a PLC. They are built as microcontrollers with volatile memory segments, program memory segments and input and output connections. Microcontrollers are taught in several university modules: ELEC1620, ELEC2645, ELEC3662 and ELEC5620M. Embedded systems require a lot of understanding in order to set-up the clocks, inputs, outputs and accessories [3] [4] while PLCs use dedicated modules plugged into a rack for the set-up [5]. This reduces the set-up time for the outputs and allows the programmer to develop the control systems sooner.

The second connection to the curriculum is the programming framework. Microcontrollers are taught with a text-based programming system. While PLC's can be programmed using a text-based system, they are more commonly programmed using function blocks or ladder logic [6]. Ladder logic is a circuit-like method where logic components are placed on the rungs of a ladder while the sides consist of a positive rail on the left and a negative rail on the right [Figure 1]. Wires connect switches, outputs and other components together and to the rails. The convention is to have outputs on the right and inputs on the left of each rung. The current powering the components flows from positive to negative in the same way that current flows through electronic components however ladder logic uses only digital signals (ON or OFF). Similar to text-based languages in how lines of code are repeated sequentially in a bare-metal application [7], the rungs are executed from top to bottom in a continuous loop.

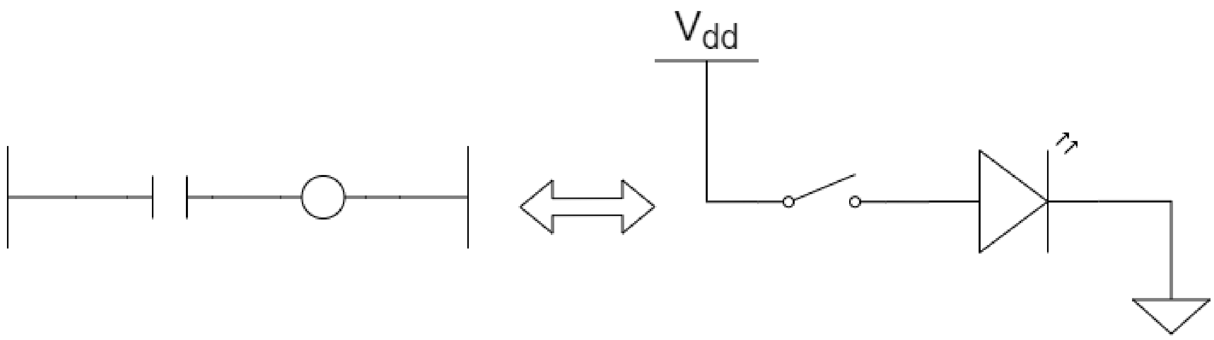

Figure 1 – A ladder logic circuit (left) and it's equivalent electrical circuit (right)

The inputs, while they look like capacitors, are in fact switches or relay contacts. Combine this with the circuit-like structure of ladder logic and inputs can be connected in similar manners to switches or transistors to create more complex forms of logic [Figure 2 and Figure 3].To make an AND logic gate, they can be connected in series. An OR logic gate is made by connecting them in parallel. The logical equivalents of buffers and inverters are included to demonstrate basic switch functionality. Transistors in logical networks are already taught at the University of Leeds in modules ELEC1620 [8] and as digital switches in ELEC1130 [9].

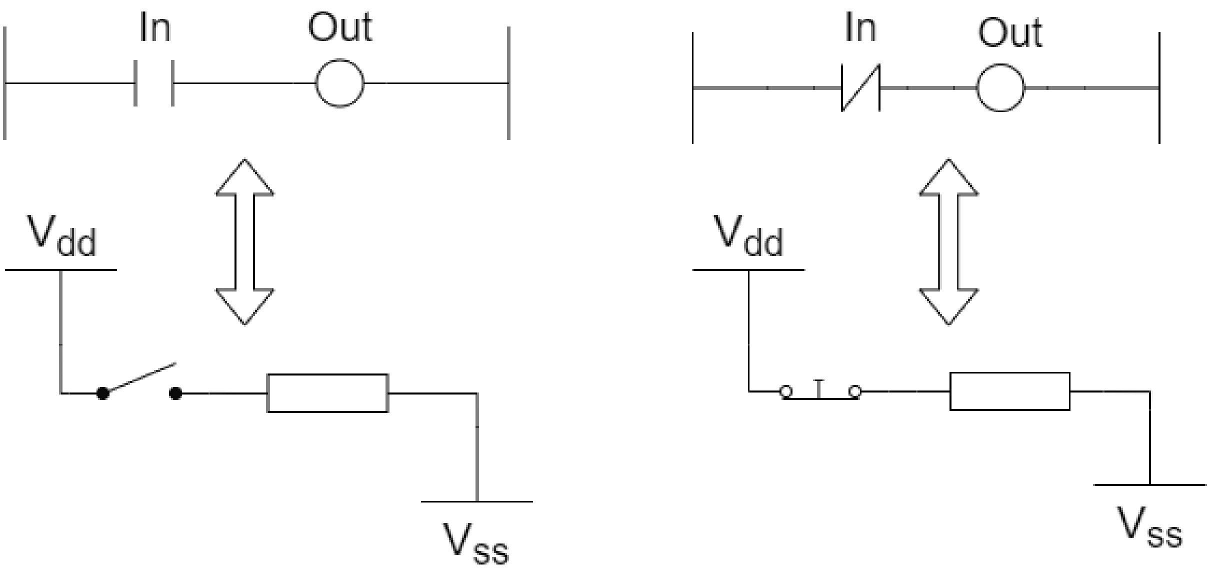

Figure 2 – Ladder logic circuits for a buffer (top left) and inverter (top right) and the electrical equivalent circuits

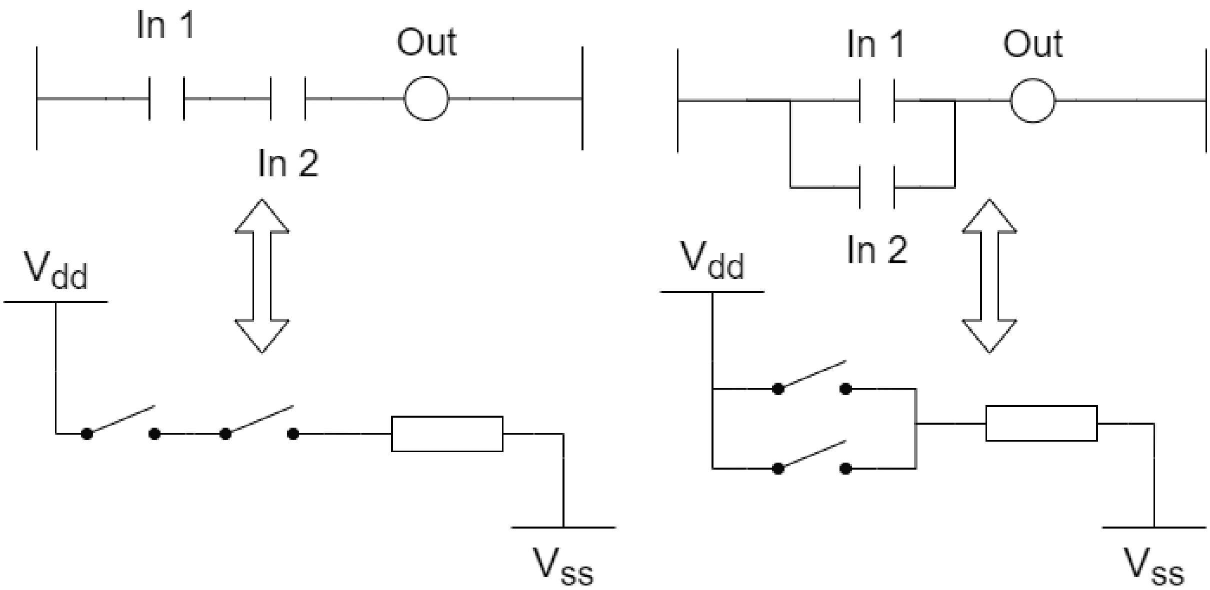

Figure 3 – Ladder logic circuits for an AND gate (top left) and an OR gate (top right) with the electrical equivalent circuits

The inputs can be connected to control outputs but also change the values of internal memory. The memory changes are implemented using output blocks to move and manipulate internal data. This is a different method of data control to an embedded system as taught in Year 3 Embedded Systems where values are set in a line of code. Both methods result in similar effects of changing specific bitfields in memory (register or otherwise). An embedded system's programmer accesses the registers to set internal hardware (clocks, ADCs and interrupts) and a PLC programmer uses the fields to control hardware outputs and alter the internal registers of the controller.

The software for a PLC can also be designed in a similar fashion despite the difference in programming methods. Finite State Machines (FSM's) as taught in ELEC2645 [10] can be used in PLC's. Various states can be selected using a portion of ladder logic to control the state (aka. Bottle Logic) [Figure 4 (top rung)]. The conditions in the control rung can be expanded to allow connections from multiple states and for handling connections to multiple states. Another portion of the ladder logic can be used to control the outputs during each state [Figure 4 (bottom rung)]. The design of

FSM's has been covered [10] yet the implementation of them in a system was not given as many case studies as the LinkedIn PLC course [11]. Going through from FSM to bottle logic in a case study may be beneficial for students to see the relevance of state machines.

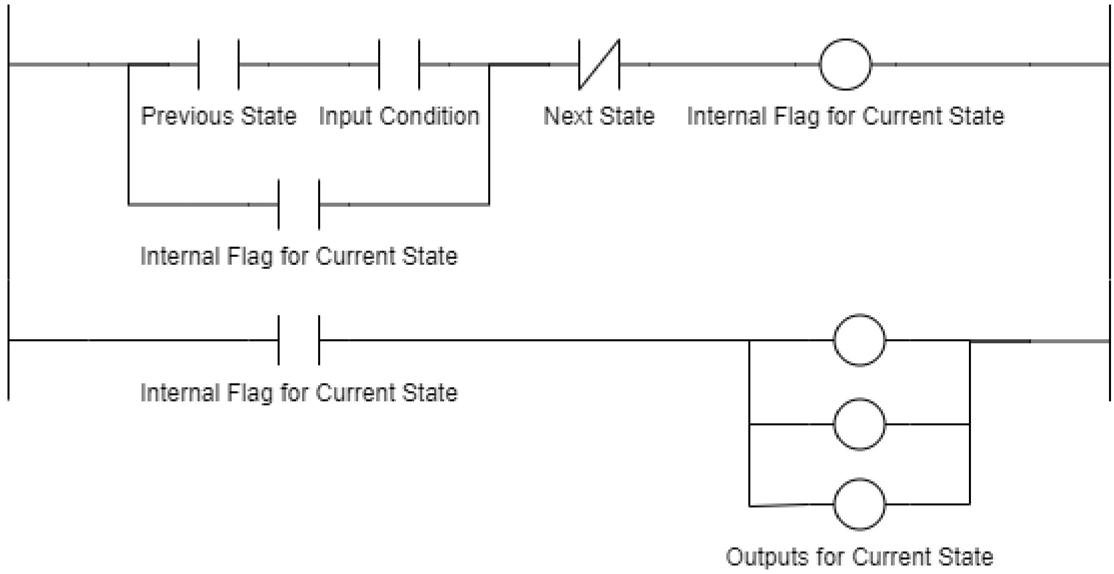

Figure 4 – Ladder logic for controlling a single state and its outputs in two rungs

An industrial norm for user input (that is not a pushbutton) is the Binary Coded Decimal (BCD) wheel. These are used to allow users to enter decimal numbers to the system. They are set up as one wheel per digit and each wheel is given 4 input lines so that the binary values from 0 to 9 are covered. This is not the most effective use of binary digits and functions are used to convert from BCD into the corresponding binary equivalent of the number the user entered on the wheels. An example of this and the conversion can be seen below [Figure 5]. This would allow students to practice binary mathematics as taught in ELEC1620 [12].

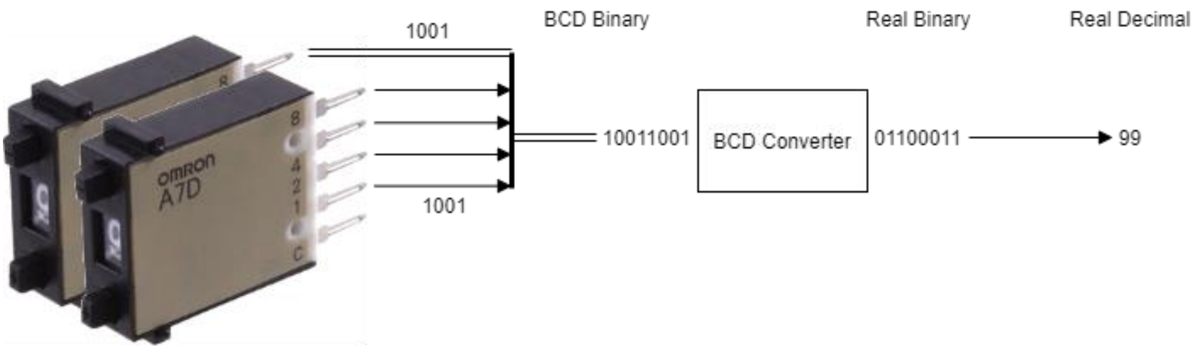

Figure 5 – Two BCD thumbwheels with connections shown [13] (altered for demonstration purposes) and an example of the output converted into binary

Lastly, working with the PLC's often requires the use of simulators or vendor-specific programming tools in the same light that Logisim and Keil IDE were used to teach students logic circuits [14] and programming devices respectively [15]. One simulator is LogixPro (aka. RSLogix) [16] but there are other software packages from Entertron [17], VIPA [18] and PLCLogix [19] and programmers with built-in simulators [20] from OEMs such as Allen-Bradley and Siemens. These packages can be used for laboratory work to practice the fundamentals and to devise coursework in the same way that Logisim was used in ELEC1620.

PLC's are widely used in industry but not taught at the University of Leeds despite the aim of the mechatronics and robotics course being to build "sophisticated intelligent systems" and equip and train students for a career in industry [21]. PLC's should be in the curriculum as they are a manifestation of all the aspects in the course's aim. They allow for smart control of motors and sensors using low-level computing that is simple and reliable in both software and hardware. They are a useful resource for learning many of the basics of software development and are a valuable preparation for industrial work.

References

[1] Z. Khalil, "1. Industrial Automation: Control PLCs and DCS, and management"," in *Learn Industrial Automation*, LinkedIn Learning, 2018.

[2] Z. Khalil, "3. Supervisory Control and Data Acquisition (SCADA): Usage," in *Learn Industrial Automation*, LinkedIn Learning, 2018.

[3] L. Mhamdi, *ELEC3662: Embedded Systems Lecture 2: TM4C123 Architecture,* Leeds: University of Leeds, 2018.

[4] L. Mhamdi, *ELEC3662: Embedded Systems Lecture 3: Programming & TM4C123 I/O Ports,* Leeds: University of Leeds, 2018.

[5] Z. Khalil, "2. Programmable Logic Controllers (PLC): Main Components," in *Learn Industrial Automation*, LinkedIn Learning, 2018.

[6] Z. Khalil, "2. Programmable Logic Controllers (PLC): PLC Programming Languages," in *Learn Industrial Automation*, LinkedIn Learning, 2018.

[7] D. D. Cowell, P. S. Freear and T. Carpenter, *ELEC5620: Embedded Micorprocessor System Design - Pre-Read 1,* Leeds: University of Leeds, 2019.

[8] C. A. Evans, *ELEC1620 Digital Electronic and Microcontrollers: 4 - Boolean Algebra,* Leeds: University of Leeds, 2016.

[9] H. Eisele, "Lecture 9 - The MOSFET as a Switch," in *ELEC1130 Circuit Analysis and Design*, Leeds , University of Leeds, 2017, pp. 23-28.

[10] C. A. Evans, "Finite State Machines," in *ELEC2645 Embedded Systems Project Module Handbook*, Leeds, University of Leeds, 2017, pp. 32-36.

[11] Z. Khalil, "5. Sequence Logic," in *PLC Ladder Logic Essential Training*, LinkedIn Learning, 2019.

[12] C. A. Evans, *ELEC1620 Digital Logic and Microcontrollers 5 - Binary Arithmetic,* Leeds: University of Leeds, 2016.

[13] Omron Electronics, "A7D-106-1 Omron Electronics Inc-EMC Div | Switches | DigiKey," [Online]. Available: https://www.digikey.com/product-detail/en/A7D-106-1/SW279-ND/36132. [Accessed 18 February 2020].

[14] C. A. Evans, “ELEC1620: Screencast 2.1 - Introduction to Logisim,” University of Leeds, Leeds, 2016.

[15] University of Leeds, “ELEC3662 - Embedded Systems Laboratory Lessons,” University of Leeds, Leeds, 2018.

[16] The Learning Pit, “LogixPro 500 PLC Simulator by TheLearningPit.com,” The Learning Pit, 2017. [Online]. Available: http://thelearningpit.com/lp/logixpro.html. [Accessed 31 December 2019].

[17] Entertron Industries Inc. , “Software - Entertron,” Entertron Industries Inc., 2015. [Online]. Available: http://www.entertron.com/software.htm. [Accessed 31 December 2019].

[18] VIPA USA, “Products - Software - Programming - Page 1 - VIPAControlsAmerica,” VIPA USA, 2020. [Online]. Available: https://vipausa.com/categories/products/software/programming.html. [Accessed 14 January 2020].

[19] Logic Design Inc., “PLCLogix - Welcome to PLCLogix,” Logic Design Inc., 2020. [Online]. Available: http://www.plclogix.com. [Accessed 31 December 2019].

[20] Siemens, “TIA Portal iPDF English,” 2019. [Online]. Available: https://assets.new.siemens.com/siemens/assets/public.1575896550.0c66eeeb-b67d-4c98-aa0d-290f82d5c0d2.7801-09-tia-p-ipdf-en-181029-1.pdf. [Accessed 21 February 2020].

[21] University of Leeds, “Mechatronics and Robotics MEng, BEng | University of Leeds,” 2019. [Online]. Available: https://courses.leeds.ac.uk/f543/mechatronics-and-robotics-meng-beng. [Accessed 8 December 2019].