

Ladder Diagram based on State Diagram for Selection and Assembling Part on Dual Conveyor

Eka Iskandar, Mochammad Rameli, Nicco
Dept. of Electrical Engineering
Sepuluh Nopember Institute of Technology
Surabaya, Indonesia

iskandar@elect-eng.its.ac.id, rameli@ee.its.ac.id, lim.nicco@gmail.com

Abstract— In industry, conveyor is a means of transportation to move production materials. Conveyor not just move the goods but there are other processes such as counting the number of goods, filling the liquid and so forth because it is equipped with certain sensors and actuators. These processes can be automated by using PLCs as controllers by executing instruction programs in the form of ladder diagrams. The complexity of automation processes in conveyors requires the use of many relays in PLC programming and is difficult to apply to low-specification PLCs. Of course the procurement of PLC with high specifications require a lot of money. In order to overcome these problems it is necessary a method that can minimize the number of relays used so that the program becomes shorter. By using state diagram, the result of construction of ladder diagram is obtained by 32 rung and the program capacity is 3 KB.

Keywords— Conveyor, Flow-Table/State Diagram, Ladder Diagram, PLC, Rung.

I. INTRODUCTION

Generally the use of trucks as well as human resources in the transfer of materials have shortcomings such as delays in material transfer, the limitations of human stamina to work accidents which resulted in disruption of manufacturing industry activities[7]. The conveyor system allows the transport of materials more efficiently than using transport carriers such as trucks or manpower. Conveyor is an equipment used in industry to move goods from one location to another. With the conveyor, moving goods becomes easier, faster and safer and can save the operating time.

The process on the conveyor does not just move the goods but also there are other processes when the conveyor works like counting the quantity of goods, filling the liquid, separating or combining the product, and so forth. Of course, these processes can take place if the conveyor is equipped with certain sensors and actuators[3]. In operation, a complex set of processes can be automated using PLCs as controllers. PLC works based on user-provided instructions in the form of programs, where the commonly used programming language is the ladder diagram.

The complexity of automation processes on conveyors requires the use of many relays in PLC programming. Relay requirement is relatively limited to PLC low specification. Of course the procurement of PLC with high specifications require

a lot of money. In order to overcome these problems it is necessary a method that can minimize the number of relays used. In PLC programming, the number of fewer relays also makes the program shorter.

1.1 Dual Conveyor Workcell Systems

Figure 1 shows Dual conveyor workcell systems intended for material handling using two conveyors (top and bottom). In this study, dual conveyor is used as the main device that serves to process the selection and assembly of a workpiece based on the criteria of the appropriate height and type of material. Workpieces passed to the conveyor deliberately have three variations of height and two types of material variations[2].

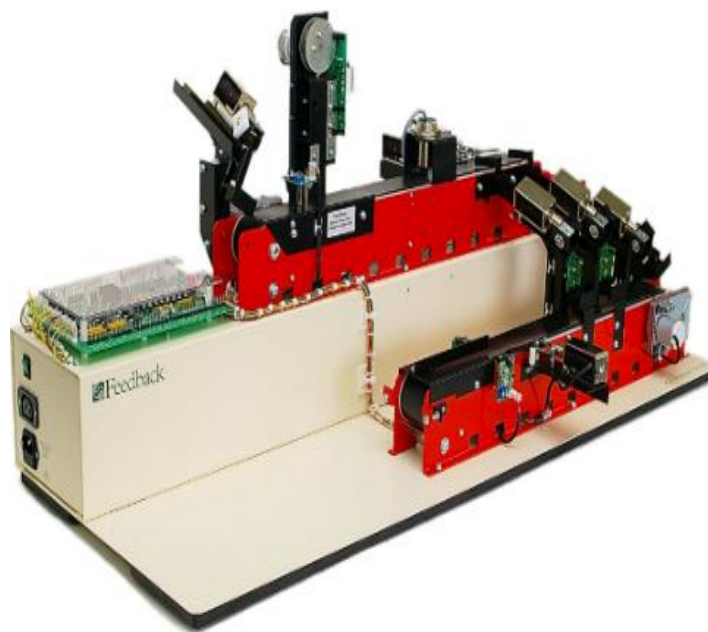


Fig. 1. Dual Conveyor Workcell

II. METHOD

2.1 State Diagram Design

Flow-table method or also called a state diagram is a graph that represents the event or state of a system in the form of a circle. The circle in question hereinafter referred to as the state term. The state contains the input and output information of a system written in binary digits[1].

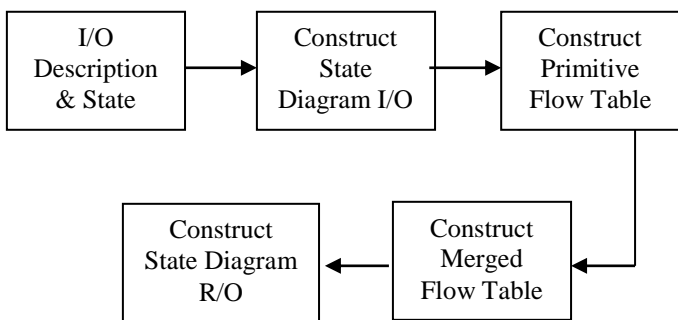


Fig. 2 Step by step Procedure

In this research, the design of state diagrams is described in Table 2. Begin by defining the input / output bits that will be used on dual conveyor system described in Table 1. In addition to this information, in Table 1 there is also an input / output address for PLC programming.

TABLE 1 I/O BIT, SISTEM AND ADDRESS

Bit	INPUT			OUTPUT		
	Information	Symbol	Address	Information	Symbol	Address
1	Relay Start	X ₁	10.00	HMDRV	Z ₁	100.02
2	HTDC	X ₂	0.04	DISP1	Z ₂	100.00
3	Opt1	X ₃	0.02	CONV1	Z ₃	100.01
4	T1 Contact	X ₄	T001	T1 Coil	Z ₄	T001
5	T2 Contact	X ₅	T002	T2 Coil	Z ₅	T002
6	Opt2	X ₆	0.06	T3 Coil	Z ₆	T003
7	Flag1	X ₇	11.06	Sol3	Z ₇	100.04
8	Flag2	X ₈	11.07	T4 Coil	Z ₈	T004
9	T3 Contact	X ₉	T003	Chute2	Z ₉	100.06
10	T4 Contact	X ₁₀	T004	T5 Coil	Z ₁₀	T005
11	Opt3	X ₁₁	0.09	Sol2	Z ₁₁	100.03
12	T5 Contact	X ₁₂	T005	T6 Coil	Z ₁₂	T006
13	T6 Contact	X ₁₃	T006	Chute1	Z ₁₃	100.05
14	Opt4	X ₁₄	0.08	DISP2	Z ₁₄	101.00
15	Opt5	X ₁₅	0.10	CONV2	Z ₁₅	100.07
16	Opt6	X ₁₆	0.11	Sol4	Z ₁₆	101.01

Events on dual conveyor systems are also translated into tables 2 before constructing state diagrams (I / O) to facilitate their preparation. Table 2 shows the input / output status occurring in each state, from the first to the last subprocess.

TABLE 2 STATE DIAGRAM (I/O) IN SEQUENCE

Sub Proses	State	Input Status	Output Status
1	1	Relay Start on	HMDRV on
	2	HTDC on	HMDRV off, DISP1 on, CONV1 on
	3	Opt1 on	DISP1 off, CONV1 off
2	1	Opt1 on	T1 Coil on
	2	T1 Contact on	T2 Coil on, CONV1 on
	3	T2 Contact on	HMDRV on, CONV1 off
	4	HTDC on	HMDRV off, CONV1 on
	5	Opt2 on	CONV1 off
3	1	Opt2, Flag1 & Flag2 on	T3 Coil on
	2	T3 Contact on	CONV1 on, Sol3 on, T4 Coil on
	3	T4 Contact on	CONV1 off, Sol3 off, Chute2 on
	4	Opt3 on	Chute2 off
Metal	1	Opt2 & Flag1 on	T5 Coil on
	2	T5 Contact on	CONV1 on, Sol2 on, T6 Coil on
	3	T6 Contact on	CONV1 off, Sol2 off, Chute1 on
	4	Opt4 on	Chute1 off
Me	1	Opt3 on	DISP2 on, CONV2 on, Sol4 on
	2	Opt5 on	DISP2 off, CONV2 off, Sol4 off
Plastik	1	Opt4 on	DISP2 on, CONV2 on
	2	Opt6 on	DISP2 off, CONV2 off
5	1	Opt2 on or;	DISP1 on, CONV1 on
	2	Opt5 on or;	
	3	Opt6 on	
	4	Opt1 on	DISP1 off, CONV1 off

2.2 Arrangement of State Diagram (I/O)

At this stage, the information in Table 2 is arranged one by one into the state diagram (I / O) based on each subprocess and the I / O bit beam information of each state. Figure 3 shows one of the state diagrams (I / O) that have been compiled.

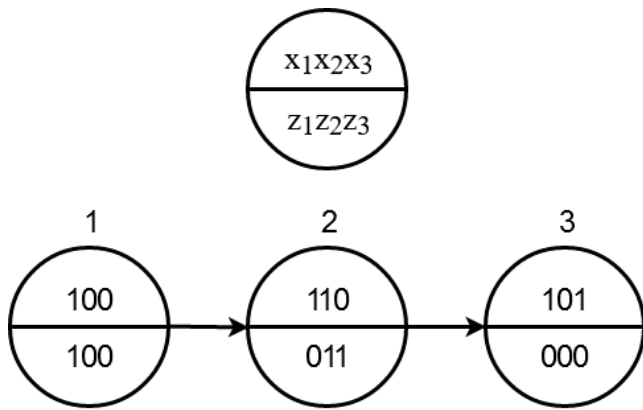


Fig. 3. State Diagram (I/O) for Sub Proses 1

In writing, all inputs / outputs in binary digits are written into a state sequenced from the first to the last such as writing a fractional number with an input-to-output (I / O) array. An input / output is said to be active if it is declared in binary digit '1', otherwise it is said to be inactive if stated in binary digit '0'. The number of inputs is written with the symbol "xi" and the output with the symbol "zj" and the numbering state is written with the symbol "s" as shown in Figure 3. The arrows show the transition from the previous state to the next state. A state will switch to the next state if there is a change in the input bit that can result in changes in output bits.

2.3 Arrangement of Primitive Flow Table

Information from the state diagram (I / O) as a whole on each sub process is used to construct a primitive flow table of dual conveyor systems. Here is shown one of the primitive flow table that has been prepared.

TABLE 3 PRIMITIVE FLOW TABLE FOR SUB PROSES 1

Row	Inputs $x_1x_2x_3$			Outputs $z_1z_2z_3$		
	100	110	101	z_1	z_2	z_3
1	1	2	-	1	0	0
2	-	2	3	0	1	1
3	-	-	3	0	0	0

Preparation of state diagrams (I / O) that have been completed followed up by creating a primitive flow table. This table contains information about stable state positions, unstable states, and do not care on input bits, and output bits in the s state. Begin by filling in the Row field, Inputs $x_{i+1} \dots x_m$, and Outputs $z_{j+1} \dots z_n$ corresponding to the number of input / output bits in the state diagram (I / O). The 'Row' column is the column for the state number from the first state "s" to the last state "t".

The "Inputs" column contains a combination of different input bits from each state. The combination of input bits is written from the first input bit to the last bit in the 1 cell of the table (eg 000, 100, etc.). If there is a combination of the same input bits from different states then it is sufficient to write 1 (one) time only. Suppose there is a combination of input bits

'100' at state 1 and state to-5, it does not need to be written 2 (two) times in 2 cells but enough once written in 1 cell.

Stable state is the state position where in that column the s-state has the same input bit combination with the input bit combination in the state diagram (I / O). Stable states are marked by bolded numbering.

Unstable states are representations of arrows or transitions in state diagrams (I / O). The arrow in question is the arrow coming out of a state and going to the next state. Suppose there is an arrow coming out of state 1 to state 2, then the arrow is an "unstable state 2", which means state 1 can go to state 2 by "unstable state 2". The table is written by giving the same number as the destination state but not bolded

Conditions do not care is a condition that does not affect the preparation of state diagrams so that it can be ignored. The "-" to indicate the condition of do not care.

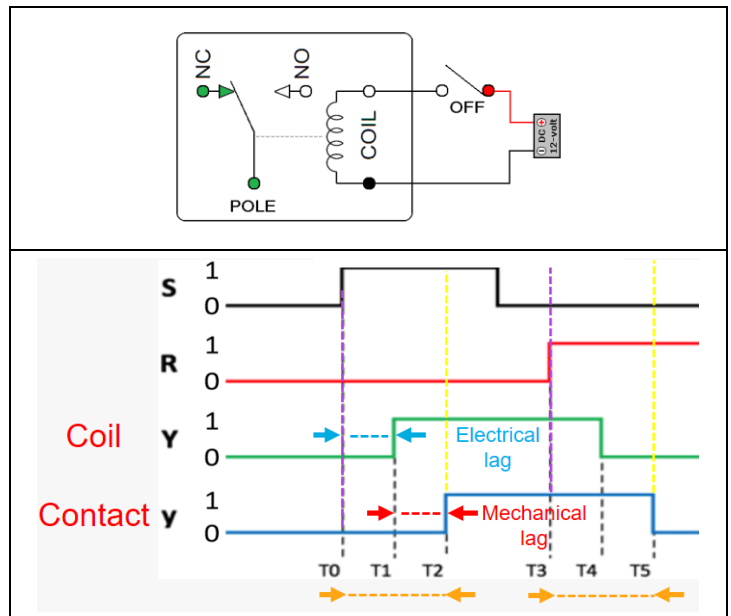


Fig. 4. Illustration Stable state, and Unstable state

In the "Outputs" column, the table is populated by inserting a combination of output bits from each state into each output bit column. Suppose there is a combination of the '001' output bit in the s state, then the first output bit (bit '0') is loaded in column "zj", the second output bit (bit '0') is filled in column 'zj + 1', and the last bit of output (bit '1') is loaded in the "zn" column in sequence. In general Table 8 shows the preparation of primitive flow table based on the state diagram (I / O) in Figure 6.

2.4 Arrangement Merged Flow Table

In the compilation of the merged flow table, the authors combine lines 1, 2 and 3 in sub-process 5. Sub processes 1 through 4 do not merge rows because they do not have the same output bits.

TABLE 4 MERGED FLOW TABLE FOR SUB PROCESS 1

Row	Inputs $x_1x_2x_3$			Outputs $z_1z_2z_3$			System's	
	100	110	101	z_1	z_2	z_3	y_1	y_2
1	1	2	-	1	0	0	1	0
2	-	2	3	0	1	1	1	1
3	-	-	3	0	0	0	0	1

Merged flow table is a table to simplify the output and determine the number of relays. The information contained therein will be used to construct state diagrams (R / O) and conversion to ladder diagrams. The first step in constructing a merged flow table is to combine the lines. The rows to be merged are suggested to select rows that have the same bits of output combinations in the primitive flow table. The rows to be merged must meet the following rules, including:

1. If there is a stable state and an unstable state in the same column, the coupled row gets a stable state.
2. If there are numbers and "-" marks in the same column, the rows that are combined get numbers (with or without bold, as the case may be).
3. If there is only a "-" sign in the column, the merged rows get a "-" sign.

In the table there is a row whose output is the same even though the state is different, ie row "s" and "s + 2". The rows can be combined if they meet the previously mentioned rules.

After combining the line, the next step is to determine the number of relays to use. The number of relays to be used should be able to cover the number of rows or states that have been merged. This is related to the flip-flop RS circuit theory that can produce two states, ie when the output is '0' or '1'. Therefore, n flip flops or relays can differentiate between 2n different states.

The completed merged flow table illustration is shown by Table 4. The arrows in the table show the direction of state change.

2.5 Arrangement of State Diagram (R/O)

In the previous step has been obtained merged flow table. The information in the table will be used to construct a state diagram (R / O) that has a relay / output arrangement on each state. Figure 5 shows one of the state diagrams (R / O) that have been compiled.

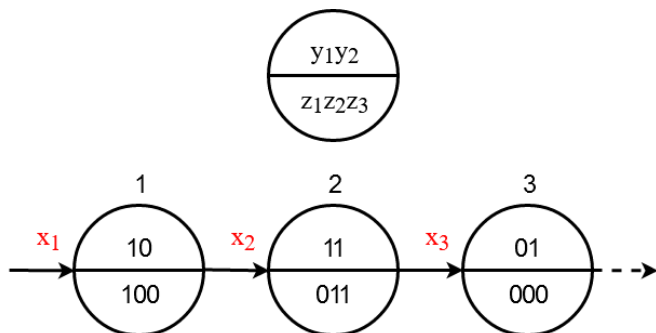


Fig. 5. State Diagram (R/O) for Sub Proses 1

The final step of the state diagram method is to arrange the state diagram (R / O) by using the information in the merged flow table. For writing the same as state diagram (I / O), it's just the contents of each state not in input / output format but in relay / output format. In transition (arrow) is given what input bit data converts previous state to next state like "Xi" and so on. The writing illustration of the state diagram (R / O) is shown in Figure 5

2.6 Construction Ladder Diagram from State diagram

From the results of designing the state diagram obtained the use of 12 relays and 16 pieces of output. Therefore, a construction of 12 rung relays (y_1 to y_{12}) and 16 rung outputs (z_1 to z_{16}) are used. However, there are additional rung as much as 4 pieces to support the work process of dual conveyor system. Individuals for the Start, Stop, Flag1 and Flag2 buttons. Ladder of rung relay y_1 is shown in Fig. 4.



Fig. 6. Rung Relay y_1

In Figure 6, there is one relay bits (" y_1 ") and no output bits. The construction is done on the ladder part of each relay and followed by the ladder part of each output[4]. For ladder compilation of the relay, it must first know the change of relay bit on each state. The relay bit is said to be active if the bit is '1' and is said to be inactive if the bit is '0'.

Note the state (R / O) of Figure 6, relay y_1 changes from off (state 1) to active (on state 2) by input x_1 . Then relay y_1 becomes disabled when there is input x_3 . If you want to create ladder for relay " y_1 " then y_1 becomes output (coil) on the rung and searched set and reset it. The set for relay " y_1 " is the input x_1 and the relay y_1 itself (self-holding). x_1 and y_1 are written in NO (Normally Open) form. If it is written, the set for relay " y_1 " is x_1 in-or (parallel) with y_1 .

In the ladder translation to reset, there is no special provision so it can be determined itself depending on the order of events on the system. Suppose the output " z_1 " will be disabled when there is an input x_3 or z_3 output on. While output z_2 will be active when relay y_1 is active. From here the programmer can choose whether to use x_3 or z_3 as reset on relay " y_1 ".

In the example in Figure 6, the authors chose x_3 as reset for relay " y_1 " so that y_2 is written in NC and placed before the output (coil) on the program rung. Just like the relay " y_1 ", for relay " y_2 " is also searched first which becomes set and reset it and after that translated to ladder.

The ladder part for output is enough to know the combination of relay bits where the output bit is active. Suppose that in state 2, the output bit " z_1 " is active (binary value '1') with the relay bit combination '10' then the '10' bit is converted to y_1 . (y_2) . For that the writing of ladder from

output "z1" is to write "y1. (y_2)" into input and z1 is written into a coil on the program rung.

III. RESULT AND DISCUSSION

Tests on the system performed to determine whether the implementation is running as expected. The author performs the test using a standard workpiece by running the system from the selection process until the assembly process is complete. Data retrieval is done using standard workpiece to know how long time from one cycle (cycle) of selection process and assembly occurs. The test data is shown by Table 5.

TABLE 5 TEST RESULTS FROM 10 CYCLES

Standard Workpiece Testing Data (8 mm)		
Cycle number	Metal (second)	Plastic (second)
1	37.23	33.32
2	36.74	33.20
3	36.97	32.97
4	37.11	33.20
5	37.74	33.13
6	35.88	32.88
7	36.86	32.06
8	38.02	32.88
9	38.18	33.10
10	36.95	33.40
Avg. Time per Cycle (second)	37.168	33.014

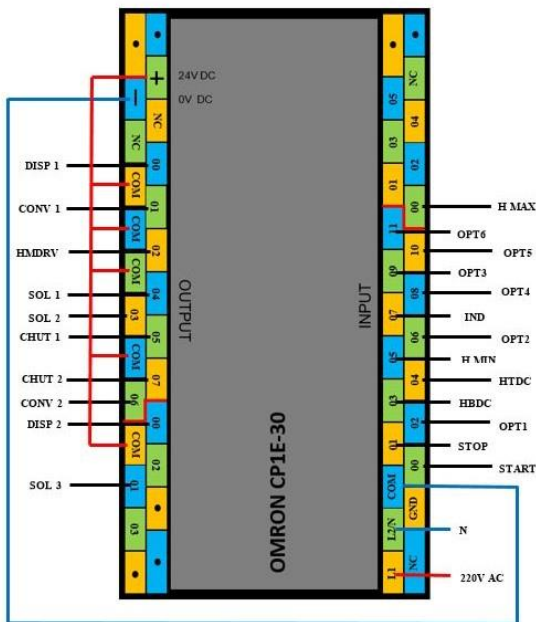


Fig. 7. Wiring I/O on PLC

IV. CONCLUSION

Based on the ladder diagram construction process that has been done in this final project research, it can be concluded several things, among others:

1. Construction of ladder diagram with State Diagram method for selection and assembly process on Dual Conveyor produce total of 32 rung ladder with detail of 4 rung support process, 12 rung relay and 16 rung output.
2. There are 6 Timers and 16 memory addresses.
3. Memory capacity for program files generated by 3 KB
4. Per cycle of the selection process and assembly of standard workpieces has an average time of 37.168 seconds for metal parts and 33,014 seconds for plastic parts.
5. For a very complex system, the State Diagram method is not suitable to be applied because it will produce a ladder diagram that is not optimal. However, these problems can be solved by dividing the entire system process into subprocesses.

ACKNOWLEDGMENT

The authors wish to thank Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember (ITS) that has supported this research.

REFERENCES

- [1] D, Persen., "Industrial Automation : Circuit Design and Components", Israel: John Wiley & Sons. 1989.
- [2] Anonim, "Single Conveyor & Workcell Systems 34-001 to 34-004". Crowborough England: FI Ltd. 2003
- [3] Hugh Jack, Automating Manufacturing Systems with Plcs, _____. 2010
- [4] PLC CQM1 Instruction manual Omron
- [5] Thapa, Devinder, Suraj Dangol, and Gi-Nam Wang. "Transformation from Petri nets model to programmable logic controller using one-to-one mapping technique." International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06). Vol. 2. IEEE, 2005.
- [6] Baydar, C. M., & Saitou, K. (2004). Off-line Error Prediction, Diagnosis and Recovery Using Virtual Assembly Systems. Journal of Intelligent Manufacturing, 15, 679 - 692.
- [7] Boothroyd, , G. (2005). Assembly Automation and Product Design . Boca Raton, Florida, USA, Taylor & Francis.
- [8] Buzan, D., Sclaroff, S., & Kollios, G. (2004). Extraction and Clustering of Motion Trajectories in Video. Proceedings from the 17th International Conference on Pattern Recogn
- [9] Davies, E. R. (2005). Machine Vision Theory, Algorithms, Practicalities. San Francisco, California, USA Elsevier.
- [10] Enabling Manufacturing Competitiveness and Economic Sustainability_ Proceedings of the 5th International Conference on Changeable
- [11] Isermann, R. (2006). Fault-Diagnosis Systems An Introduction to Fault Tolerance. Berlin, Germany Springer-Verlag.