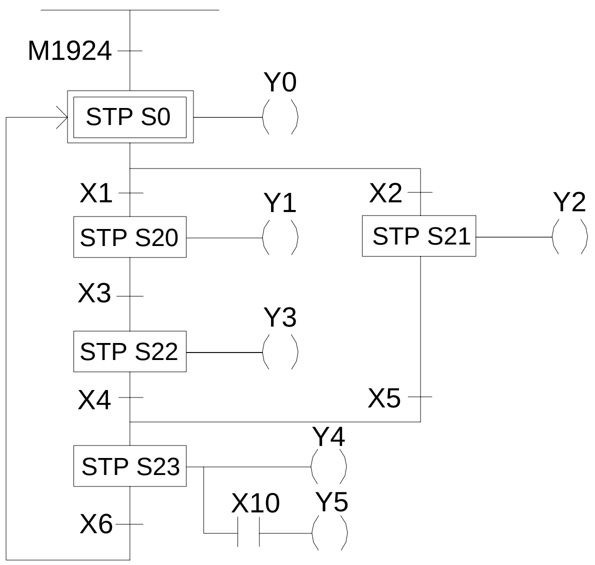
Chapter 8 Step Instruction Description

Structured programming design is a major trend in software design. The benefits are high readability, easy maintenance, convenient updating and high quality and reliability. For the control applications, consisted of many sequential tasks, designed by conventional ladder program design methodology usually makes others hard to maintain. Therefore, it is necessary to combine the current widely used ladder diagrams with the sequential controls made especially for machine working flow. With help from step instructions, the design work will become more efficient, time saving and controlled. This kind of design method that combines process control and ladder diagram together is called the step ladder language.

The basic unit of step ladder diagram is a step. A step is equivalent to a movement (step) in the machine operation where each movement has an output. The complete machine or the overall sequential control process is the combination of steps in serial or parallel. Its step-by-step sequential execution procedure allows others to be able to understand the machine operations thoroughly, so that design, operation, and maintenance will become more effective and simpler.

8.1 The Operation Principle of Step Ladder Diagram

【Example】



【Description】

1. STP Sxxx is the symbol representing a step Sxxx that can be one of S0～S999. When executing the step (status ON), the ladder diagram on the right will be executed and the previous step and output will become OFF.
2. M1924 is on for a scan time after program start. Hence, as soon as ON, the stop of the initial step S0 is entered (S0 ON) while the other steps are kept inactive, i.e. Y1～Y5 are all OFF. This means M1924 ON→S0 ON→Y0 ON and Y0 will remain ON until one of the contacts X1 or X2 is ON.
3. Assume that X2 is ON first; the path to S21 will then be executed.

   $$X2\ ON \Rightarrow \begin{cases} S21\ ON \\ S0\ OFF \end{cases} \Rightarrow \begin{cases} Y2\ ON \\ Y0\ OFF \end{cases}$$

   Y2 will remain ON until X5 is ON.
4. Assume that X5 is ON, the process will move forward to step S23.

   $$\text{i.e. } X5\ ON \Rightarrow \begin{cases} S23\ ON \\ S21\ OFF \end{cases} \Rightarrow \begin{cases} Y4\ ON \\ Y2\ OFF \end{cases}$$

   Y4 and Y5 will remain ON until X6 is ON.

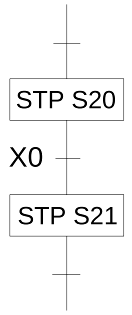
   ※If X10 is ON, then Y5 will be ON.
5. Assume that X6 is ON, the process will move forward to S0.

   $$\text{i.e. } X6\ ON \Rightarrow \begin{cases} S0\ \ ON \\ S23\ OFF \end{cases} \Rightarrow \begin{cases} Y0\ ON \\ Y4、Y5\ OFF \end{cases}$$

   Then, a control process cycle is completed and the next control process cycle is entered.

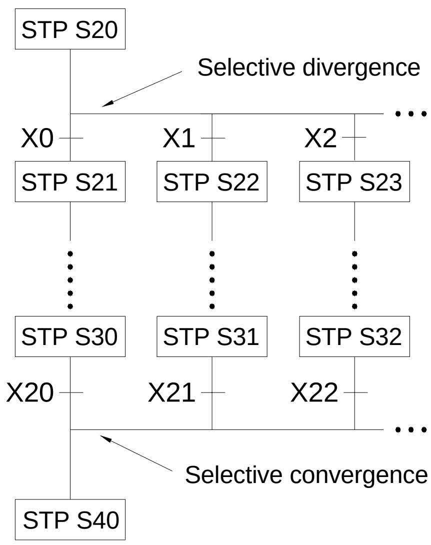
8.2 Basic Formation of Step Ladder Diagram

① Single path



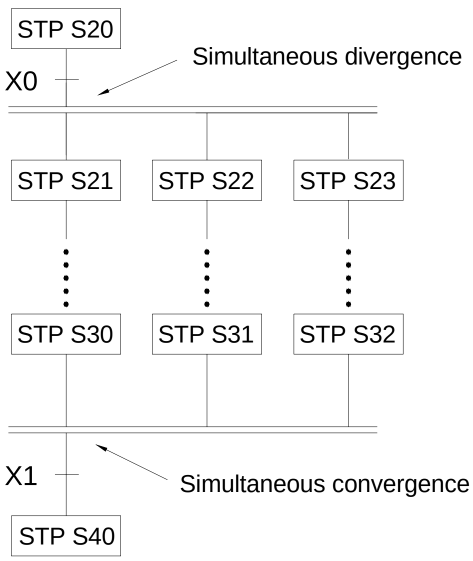
- Step S20 alone moves to step S21 through X0.
- X0 can be changed to other serial or parallel combination of contacts.

② Selective divergence/convergence



- Step S20 selects an only one path which divergent condition first met. E.g. X2 is ON first, then only the path of step S23 will be executed.
- A divergence may have up to 8 paths maximum.
- X1, X2, ….., X22 can all be replaced by the serial or parallel combination of other contacts.

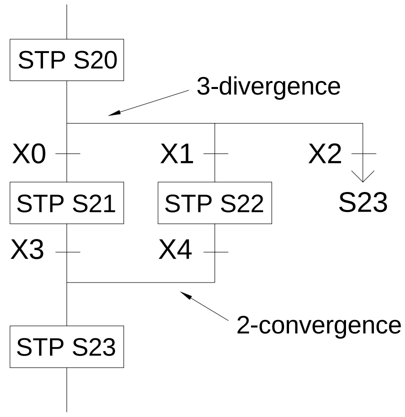
③ Simultaneous divergence/convergence



- After X0 is ON, step S20 will simultaneously execute all paths below it, i.e. all S21, S22, S23, and so on, are in action.
- All divergent paths at a convergent point will be executed to the last step (e.g. S30, S31 and S32). When X1 is ON, they can then transfer to S40 for execution.
- The number of divergent paths must be the same as the number of convergent paths. The maximum number of divergence/convergence path is 8.

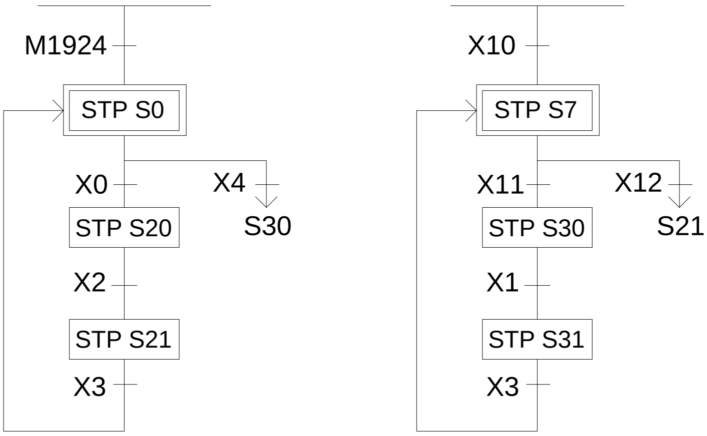
④ Jump

a. The same step loop



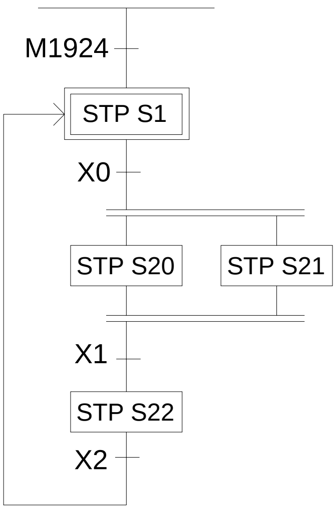
- There are 3 paths below step S20 as shown on the left. Assume that X2 is ON, then the process can jump directly to step S23 to execute without going through the process of selective convergence.
- The execution of simultaneous divergent paths can not be skipped.

b. Different step loop

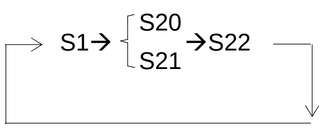


⑤Closed Loop and Single Cycle

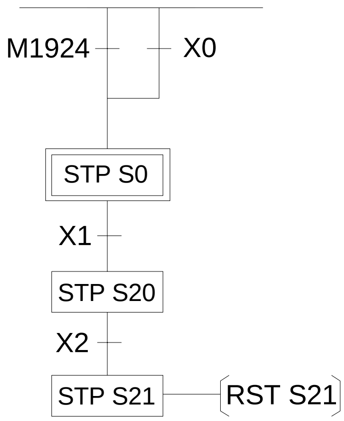
a. Closed Loop



- The initial step S1 is ON, endless cycle will be continued afterwards.

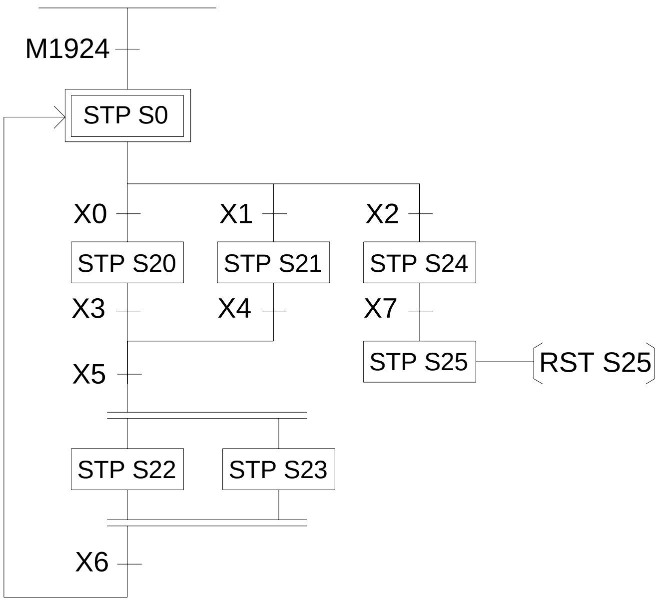


b. Single Cycle

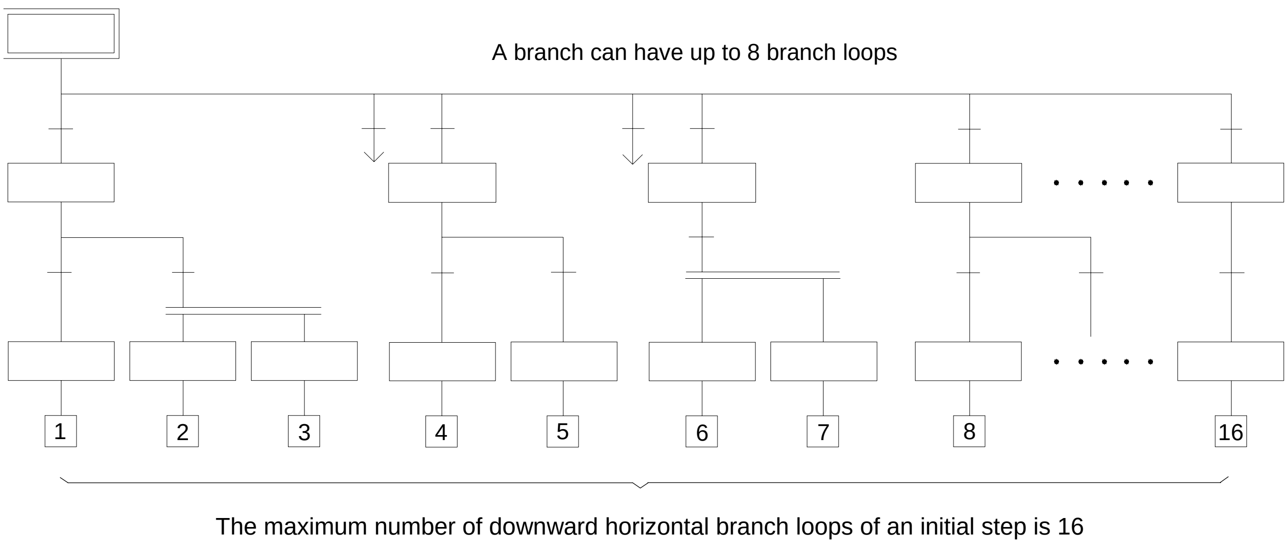


- When step S20 is ON, if X2 is also ON, then “RST S21” instruction will let S21 OFF which will stop the whole step process.

c. Mixed Process



⑥ Combined Application



8.3 Introduction of Step Instructions: STP, FROM, TO and STPEND

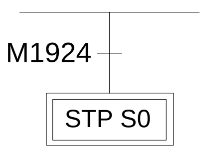
● STP Sx : S0≦Sx≦S7 (Displayed in WinProladder)

or

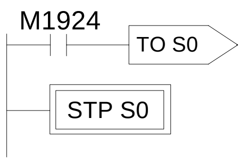
STP Sx ：S0≦Sx≦S7 (Displayed in FP-08)

This instruction is the initial step instruction from where the step control of each machine process can be derived. Up to 8 initial steps can be used in the FBs series, i.e. a PLC can make up to 8 process controls simultaneously. Each step process can operate independently or generate results for the reference of other processes.

【Example 1】 Go to the initial step S0 after each start (ON)



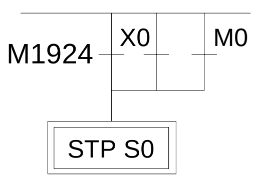
WinProladder



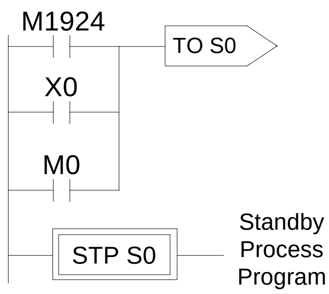
FP-08

```
ORG    M1924
TO     S0
STP    S0
```

【Example 2】 Each time the device is start to run or the manual button is pressed or the device is malfunction, then the device automatically enters the initial step S0 to standby.



WinProladder



FP-08

```
ORG    M1924
OR     X0
OR     M0
TO     S0
STP    S0
```

Standby process program

【Description】X0: Manual Button, M0: Abnormal Contact.

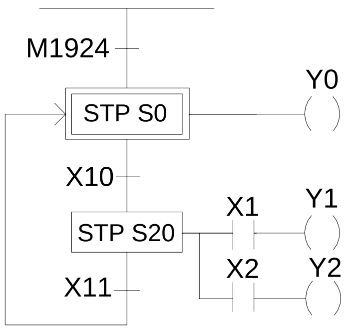
● STP Sxxx : S20≦Sxxx≦S999（Displayed in WinProladder）

or

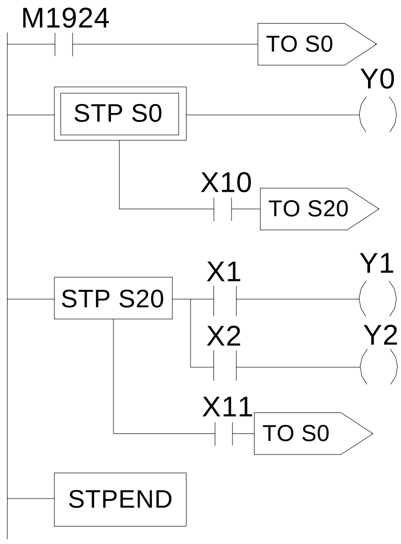
STP Sxxx : S20≦Sxxx≦S999（Displayed in FP-08）

This instruction is a step instruction, each step in a process represents a step of sequence. If the status of step is ON then the step is active and will execute the ladder program associate to the step.

【Example】



WinProladder



FP-08

ORG	M1924
TO	S0
STP	S0
OUT	Y0
FROM	S0
AND	X10
TO	S20
STP	S20
OUT	TR0
AND	X1
OUT	Y1
LD	TR0
AND	X2
OUT	Y2
FROM	S20
AND	X11
TO	S0
STPEND	

【Description】1. When ON, the initial step S0 is ON and Y0 is ON.

2. When transfer condition X10 is ON (in actual application, the transferring condition may be formed by the serial or parallel combination of the contacts X, Y, M, T and C), the step S20 is activated. The system will automatically turn S0 OFF in the current scan cycle and Y0 will be reset automatically to OFF.

i.e. X10 ON⇒ { S20 ON / S0 OFF } ⇒ { X1 ON →Y1 ON / X2 ON →Y2 ON / Y0 OFF }

3. When the transfer condition X11 is ON, the step S0 is ON, Y0 is ON and S20, Y1 and Y2 will turn OFF at the same time.

i.e. X11 ON⇒ { S0 ON / S20 OFF } ⇒ { Y0 ON / Y1 OFF / Y2 OFF }

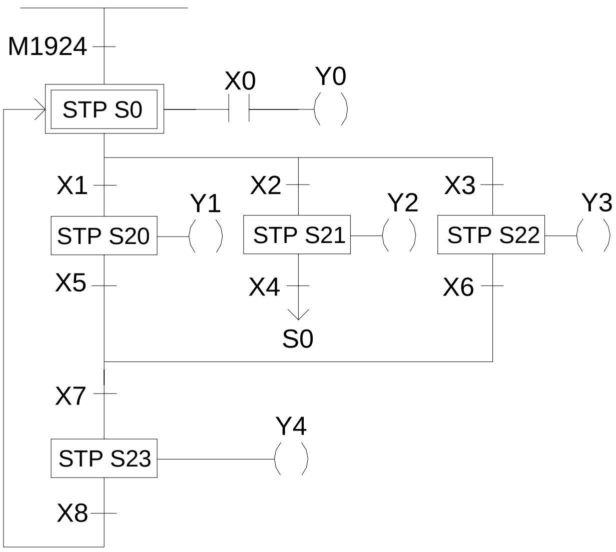
● FROM Sxxx : S0≦Sxxx≦S999（Displayed in WinProladder）

or

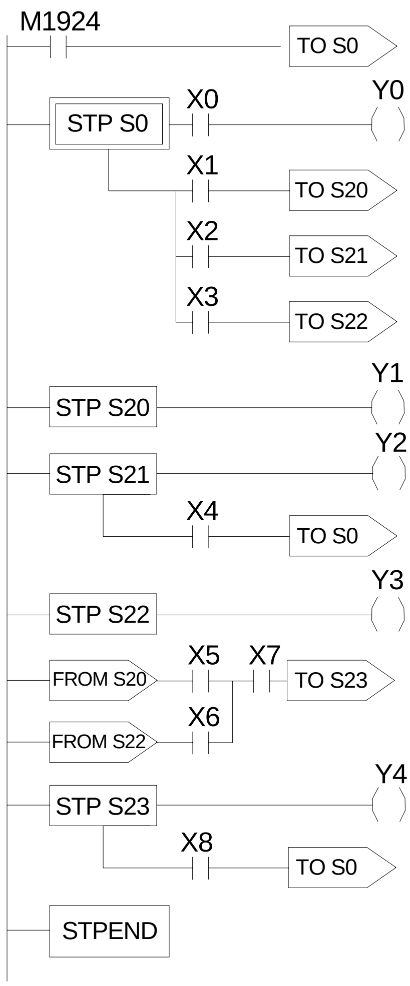
FROM Sxxx : S0≦Sxxx≦S999（Displayed in FP-08）

The instruction describes the source step of the transfer, i.e. moving from step Sxxx to the next step in coordination with transfer condition.

【Example】



WinProladder



FP-08

ORG	M1924
TO	S0
STP	S0
AND	X0
OUT	Y0
FROM	S0
OUT TR	0
AND	X1
TO	S20
LD TR	0
AND	X2
TO	S21
LD TR	0
AND	X3
TO	S22
STP	S20
OUT	Y1
STP	S21
OUT	Y2
FROM	S21
AND	X4
TO	S0
STP	S22
OUT	Y3
FROM	S20
AND	X5
FROM	S22
AND	X6
ORLD	
AND	X7
TO	S23
STP	S23
OUT	Y4
FROM	S23
AND	X8
TO	S0
STPEND	

【Description】: 1. When ON, the initial step S0 is ON. If X0 is ON, then Y0 will be ON.

2. When S0 is ON: a. if X1 is ON, then step S20 will be ON and Y1 will be ON.

   b. if X2 is ON, then step S21 will be ON and Y2 will be ON.

   c. if X3 is ON, then step S22 will be ON and Y3 will be ON.

   d. if X1, X2 and X3 are all ON simultaneous, then step S20 will have the priority to be ON first and either S21 or S22 will not be ON.

   e. if X2 and X3 are ON at the same time, then step S21 will have the priority to be ON first and S22 will not be ON.

3. When S20 is ON, if X5 and X7 are ON at the same time, then step S23 will be ON, Y4 will be ON and S20 and Y1 will be OFF.

4. When S21 is ON, if X4 is ON, then step S0 will be ON and S21 and Y2 will be OFF.

5. When S22 is ON, if X6 and X7 are ON at the same time, then step S23 will be ON, Y4 will be ON and S22 and Y3 will be OFF.

6. When S23 is ON, if X8 is ON, then step S0 will be ON and S23 and Y4 will be OFF.

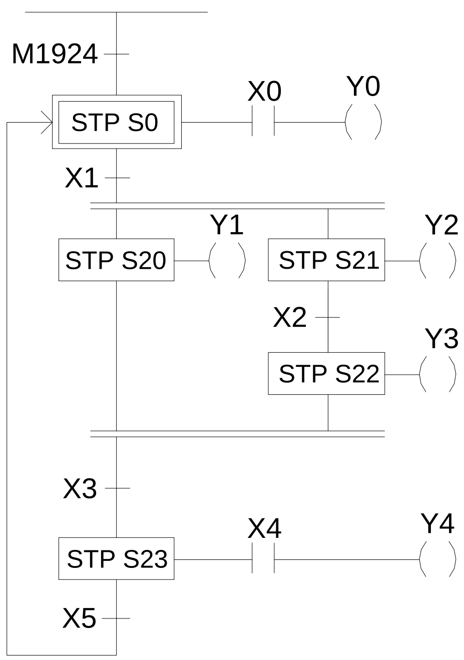
● TO Sxxx : S0≦Sxxx≦S999（Displayed in WinProladder）

or

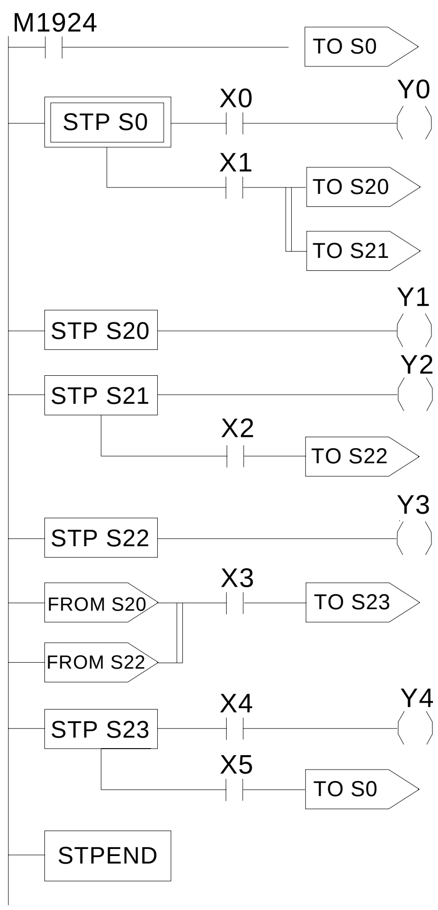
TO Sxxx : S0≦Sxxx≦S999（Displayed in FP-08）

This instruction describes the step to be transferred to.

【Example】



WinProladder



FP-08

ORG	M1924
TO	S0
STP	S0
AND	X0
OUT	Y0
FROM	S0
AND	X1
TO	S20
TO	S21
STP	S20
OUT	Y1
STP	S21
OUT	Y2
FROM	S21
AND	X2
TO	S22
STP	S22
OUT	Y3
FROM	S20
FROM	S22
AND	X3
TO	S23
STP	S23
AND	X4
OUT	Y4
FROM	S23
AND	X5
TO	S0
STPEND	

【Description】: 1. When ON, the initial step S0 is ON. If X0 is ON, then Y0 will be ON.

2. When S0 is ON: if X1 is ON, then steps S20 and S21 will be ON simultaneously and Y1 and Y2 will also be ON.
3. When S21 is ON: if X2 is ON, then step S22 will be ON, Y3 will be ON and S21 and Y2 will be OFF.
4. When S20 and S22 are ON at the same time and the transferring condition X3 is ON, then step S23 will be ON (if X4 is ON, then Y4 will be ON) and S20 and S22 will automatically turn OFF and Y1 and Y3 will also turn OFF.
5. When S23 is ON: if X5 is ON, then the process will transfer back to the initial step, i.e. So will be ON and S23 and Y4 will be OFF.

● STPEND :（Displayed in WinProladder）

or

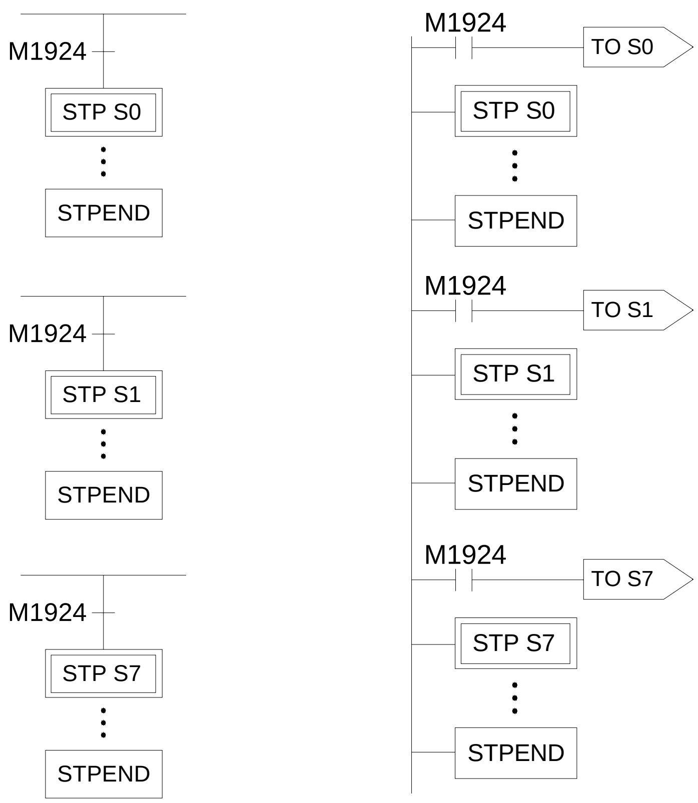
STPEND :（Displayed in FP-08）

This instruction represents the end of a process. It is necessary to include this instruction so all processes can be operated correctly.

A PLC can have up to 8 step processes (S0～S7) and is able to control them simultaneously. Therefore, up to 8 STPEND instructions can be obtained.

【Example】

WinProladder FP-08



ORG	M1924
TO	S0
STP	S0
•	
•	
•	
STPEND	
ORG	M1924
TO	S1
STP	S1
•	
•	
•	
STPEND	
ORG	M1924
TO	S7
STP	S7
•	
•	
•	
STPEND	

【Description】 When ON, the 8 step processes will be active simultaneously.

8.4 Notes for Writing a Step Ladder Diagram

【Notes】

- In actual applications, the ladder diagram can be used together with the step ladder.

- There are 8 steps, S0～S7, that can be used as the starting point and are called the “initial steps”.

- When PLC starts operating, it is necessary to activate the initial step. The M1924 (the first scan ON signal) provided by the system may be used to activate the initial step.

- Except the initial step, the start of any other steps must be driven by other step.

- It is necessary to have an initial step and the final STPEND instruction in a step ladder diagram to complete a step process program.

- There are 980 steps, S20～S999, available that can be used freely. However, used numbers cannot be repeated. S500～S999 are retentive(The range can be modified by users), can be used if it is required to continue the machine process after power is off.

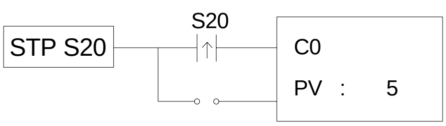
- Basically a step must consists of three parts which are control output, transition conditions and transition targets.

- MC and SKP instructions cannot be used in a step program and the sub-programs. It’s recommended that JMP instruction should be avoided as much as possible.

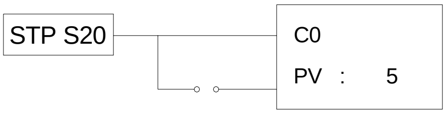
- If the output point is required to stay ON after the step is divergent to other step, it is necessary to use the SET instruction to control the output point and use RST instruction to clear the output point to OFF.

- Looking down from an initial step, the maximum number of horizontal paths is 16. However, a step is only allowed to have up to 8 branch paths.

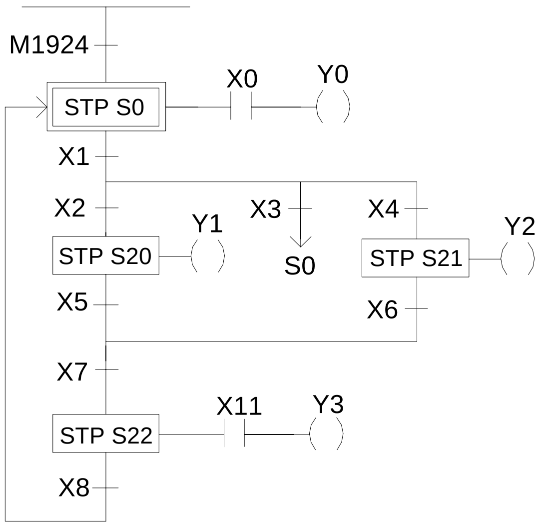
- When M1918=0（default）, if a PULSE type function instruction is used in master control loop (FUN 0) or a step program, it is necessary to connect a TU instruction before the function instruction. For example,



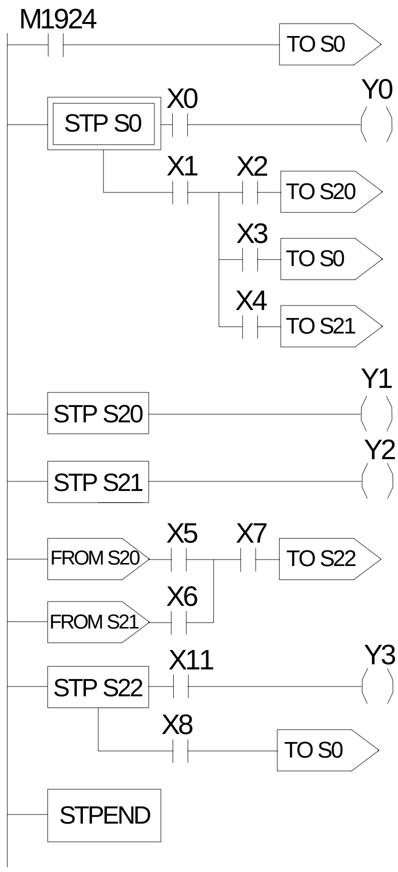
When M1918=1, the TU instruction is not required, e.g.:



Example 1



WinProladder



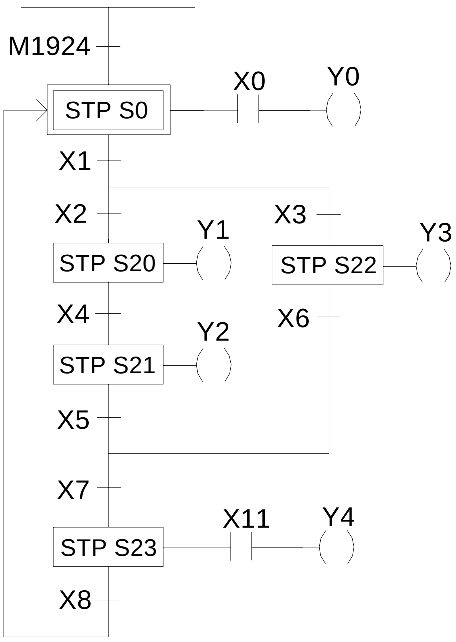
FP-08

Net0	ORG	M1924
	TO	S0
Net1	STP	S0
	AND	X0
	OUT	Y0
	FROM	S0
	AND	X1
	OUT TR	0
	AND	X2
	TO	S20
	LD TR	0
	AND	X3
	TO	S0
	LD TR	0
	AND	X4
	TO	S21
Net2	STP	S20
	OUT	Y1
Net3	STP	S21
	OUT	Y2
Net4	FROM	S20
	AND	X5
	FROM	S21
	AND	X6
	ORLD	
	AND	X7
	TO	S22
Net5	STP	S22
	AND	X11
	OUT	Y3
	FROM	S22
	AND	X8
	TO	S0
Net6	STPEND	

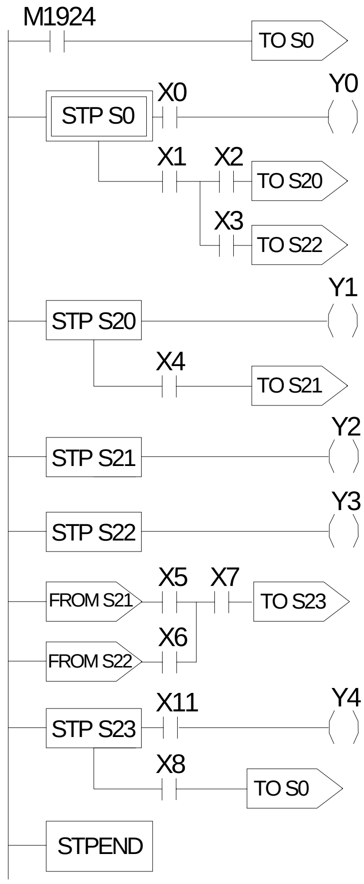
Description

1. Input the condition to initial step S0
2. Input the S0 and the divergent conditions of S20, S0 and S21
3. Input the S20
4. Input the S21
5. Input the convergence of S20 and S21
6. Input the S22

Example 2



WinProladder



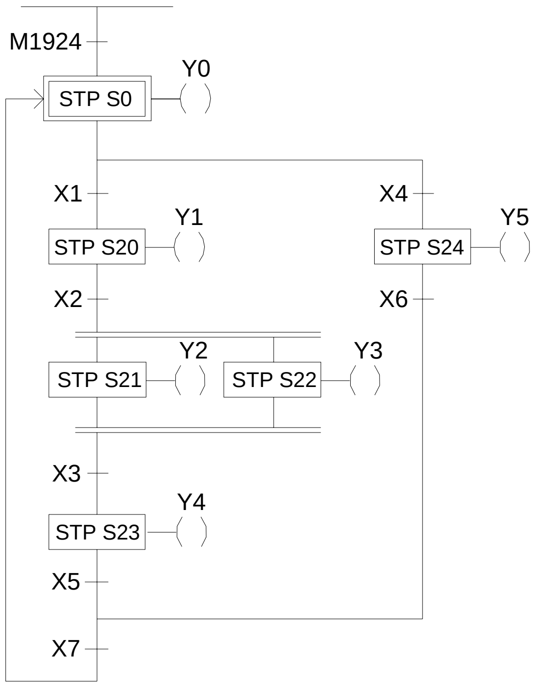
FP-08

Net0	ORG	M1924
	TO	S0
Net1	STP	S0
	AND	X0
	OUT	Y0
	FROM	S0
	AND	X1
	OUT TR	0
	AND	X2
	TO	S20
	LD TR	0
	AND	X3
	TO	S22
Net2	STP	S20
	OUT	Y1
	FROM	S20
	AND	X4
	TO	S21
Net3	STP	S21
	OUT	Y2
Net4	STP	S22
	OUT	Y3
Net5	FROM	S21
	AND	X5
	FROM	S22
	AND	X6
	ORLD	
	AND	X7
	TO	S23
Net6	STP	S23
	AND	X11
	OUT	Y4
	FROM	S23
	AND	X8
	TO	S0
Net7	STPEND	

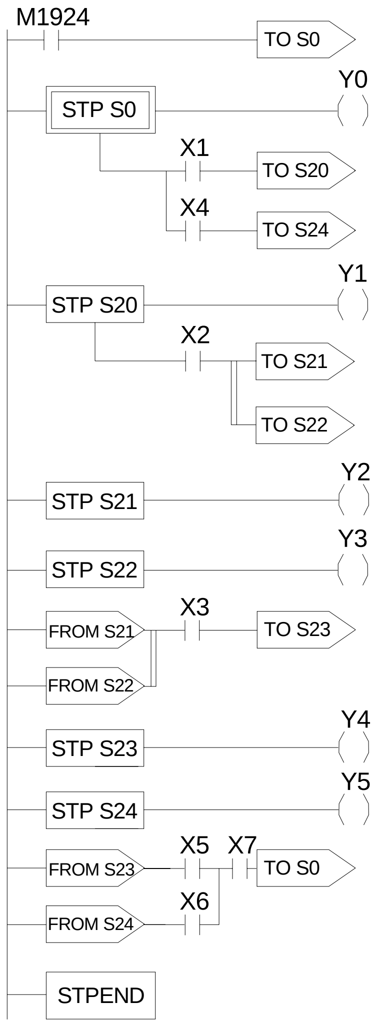
Description

1. Input the condition to initial step S0
2. Input the S0 and the divergent condition of S20 and S22
3. Input the S20
4. Input the S21
5. Input the S22
6. Input the convergence of S21 and S22
7. Input the S23

Example 3



WinProladder



FP-08

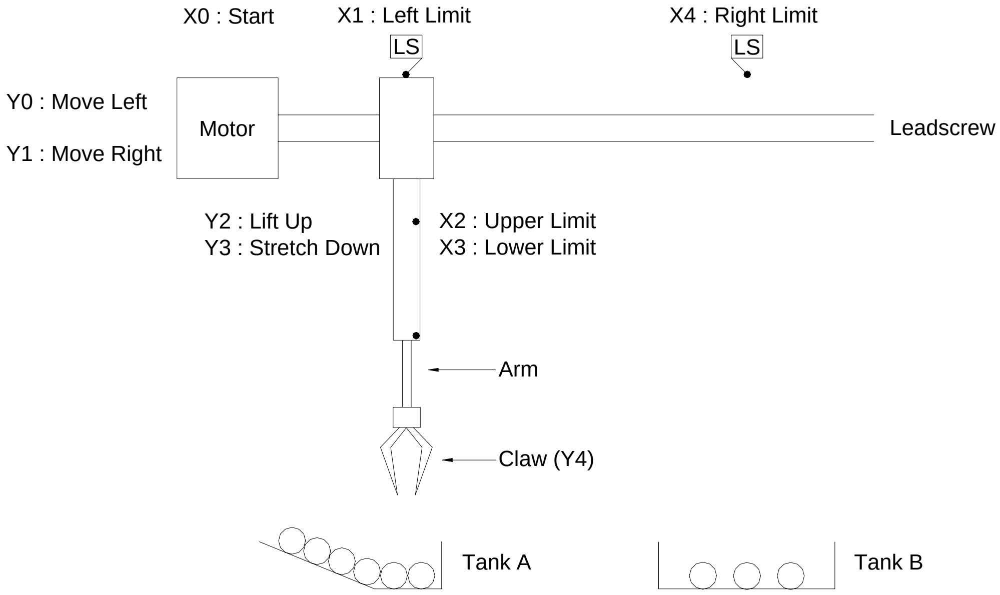
Net0	ORG	M1924
	TO	S0
Net1	STP	S0
	OUT	Y0
	FROM	S0
	OUT TR	0
	AND	X1
	TO	S20
	LD TR	0
	AND	X4
	TO	S24
Net2	STP	S20
	OUT	Y1
	FROM	S20
	AND	X2
	TO	S21
	TO	S22
Net3	STP	S21
	OUT	Y2
Net4	STP	S22
	OUT	Y3
Net5	FROM	S21
	FROM	S22
	AND	X3
	TO	S23
Net6	STP	S23
	OUT	Y4
Net7	STP	S24
	OUT	Y5
Net8	FROM	S23
	AND	X5
	FROM	S24
	AND	X6
	ORLD	
	AND	X7
	TO	S0
Net9	STPEND	

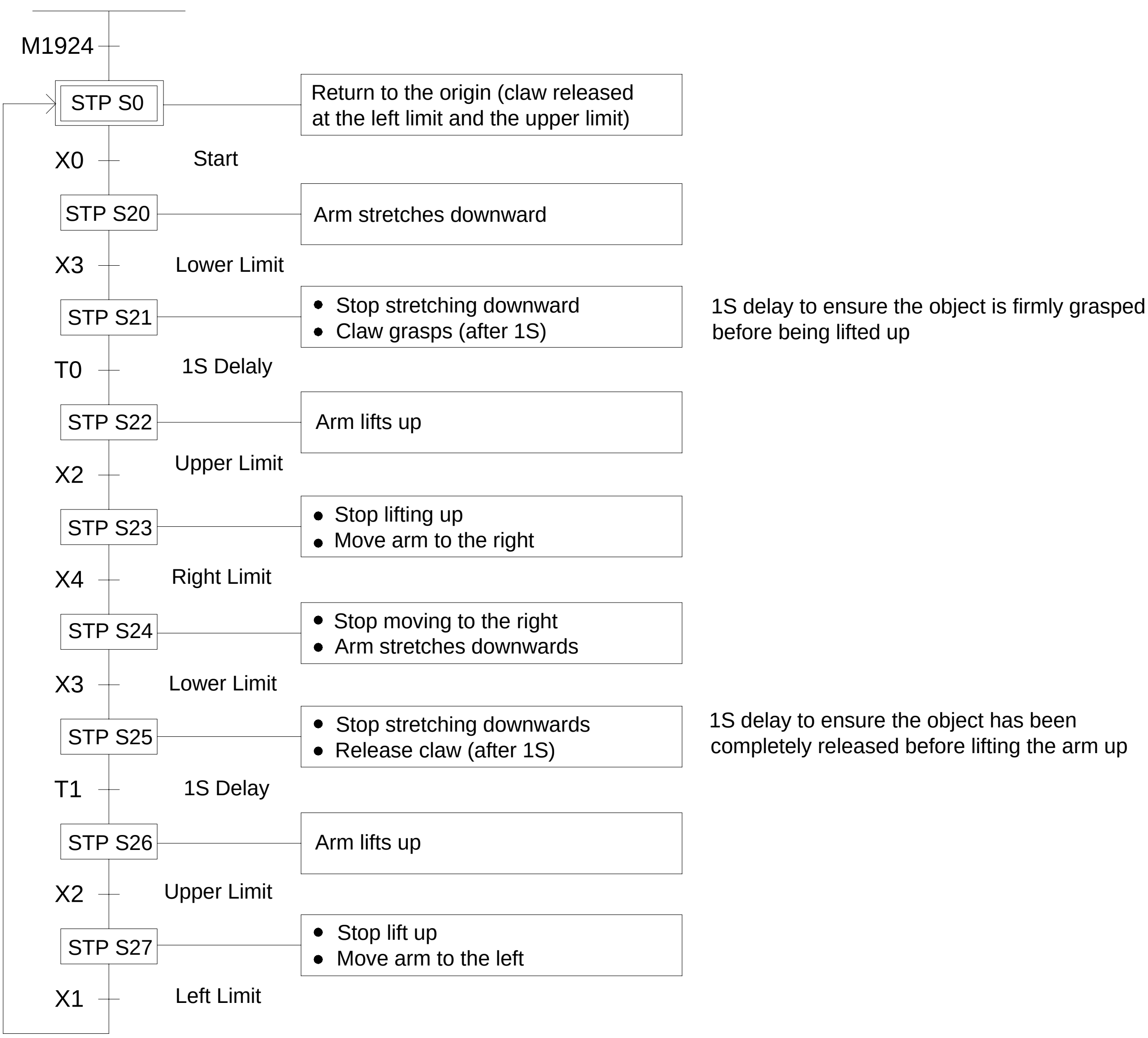
Description

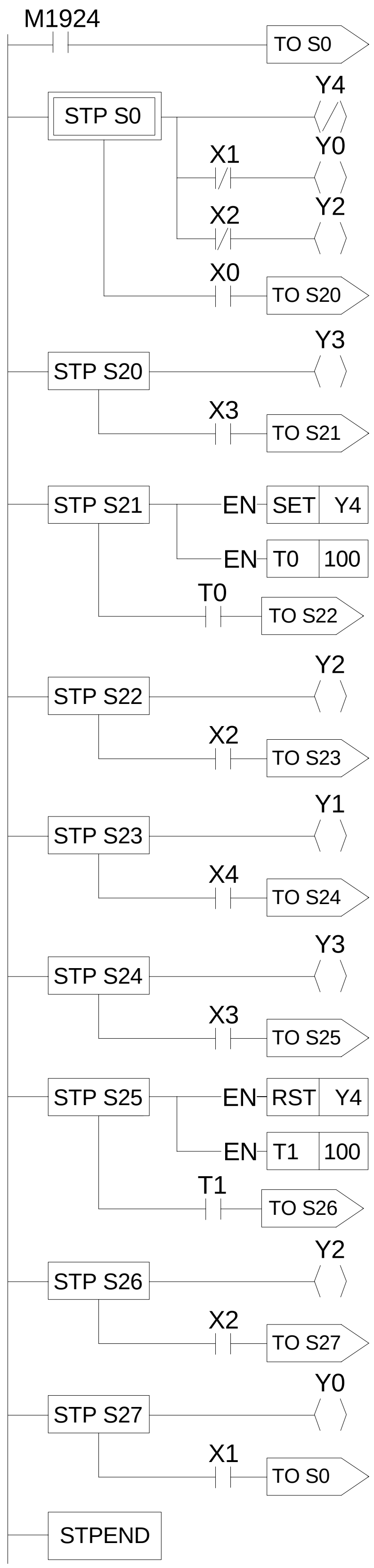
1. Input the condition to initial step S0
2. Input the S0 and the divergences of S20 and S24
3. Input the S20
4. Input the S20 and the divergences of S21 and S22
5. Input the S21
6. Input the S22
7. Input the convergences of S21 and S22
8. Input the S23
9. Input the S24
10. Input the convergences of S23 and S24

8.5 Application Examples

Example 1 Grasp an object from tank A and put it in Tank B







Release claw

Return to the left limit

Return to the upper limit

Turn the switch ON before moving to S20

Stretch arm downward

Move to S21 after stretching to the lower limit

Claw grasps (since the SET instruction is used, Y4 should remain ON after departing from STP S21)

Divergent into S22 after 1S

Lift the arm up

Divergent into S23 after reaching the upper limit

Move arm to the right

Divergent into S24 after moving to the right limit

Stretch the arm downward

Divergent into S25 after stretching to the lower limit

Release claw

Delay for 1S

Transfer into S26 after 1S

Lift the arm up

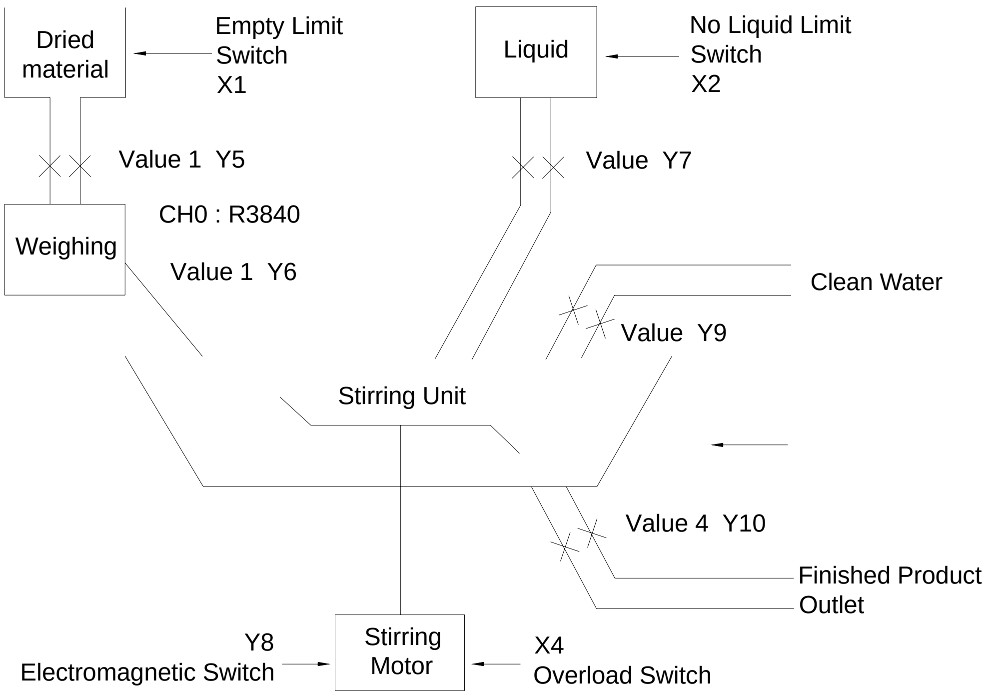
Divergent into S27 after reaching the upper limit

Move the arm to the left

Divergent into S0 after moving to the left limit (a complete cycle)

ORG	M1924
TO	S0
STP	S0
OUT TR	0
OUT NOT	Y4
AND NOT	X1
OUT	Y0
LD TR	0
AND NOT	X2
OUT	Y2
FROM	S0
AND	X0
TO	S20
STP	S20
OUT	Y3
FROM	S20
AND	X3
TO	S21
STP	S21
SET	Y4
T0 PV:	100
FROM	S21
AND	T0
TO	S22
STP	S22
OUT	Y2
FROM	S22
AND	X2
TO	S23
STP	S23
OUT	Y1
FROM	S23
AND	X4
TO	S24
STP	S24
OUT	Y3
FROM	S24
AND	X3
TO	S25
STP	S25
RST	Y4
T1 PV:	100
FROM	S25
AND	T1
TO	S26
STP	S26
OUT	Y2
FROM	S26
AND	X2
TO	S27
STP	S27
OUT	Y0
FROM	S27
AND	X1
TO	S0
STPEND	

Example 2 Liquid Stirring Process



- Input Points: Empty limit switch X1
  No liquid limit switch X2
  Empty limit switch X3
  Over-load switch X4
  Warning clear button X5
  Start button X6
  Water washing button X7

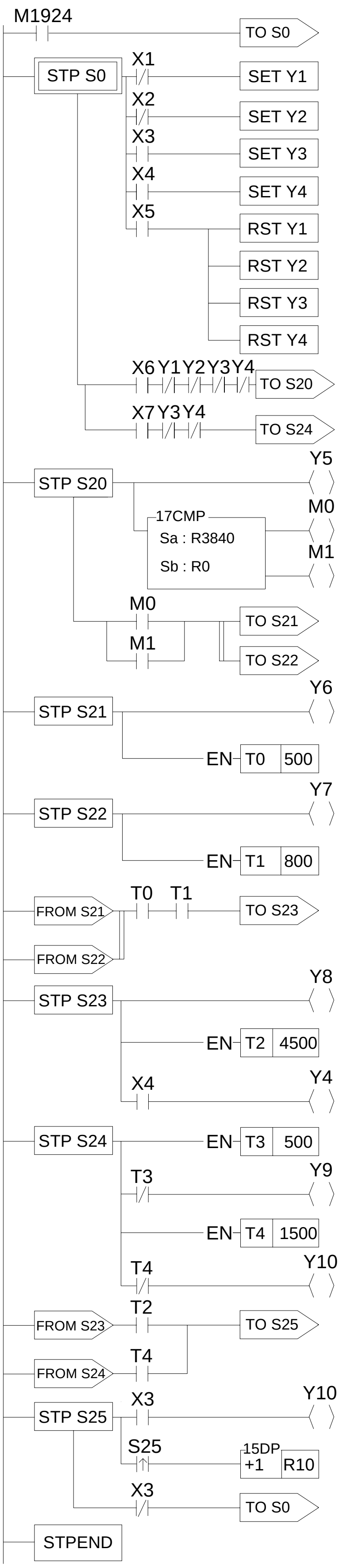
- Warning Indicators: Empty dried material Y1
  Insufficient liquid Y2
  Empty stirring unit Y3
  Motor over-load Y4

- Output Points: Dried material inlet valve Y5
  Dried material inlet valve Y6
  Liquid inlet valve Y7
  Motor start electromagnetic valve Y8
  Clean water inlet valve Y9
  Finished product outlet valve Y10

- Weighing Output: CH0（R3840）

- M1918=0

WinProladder



Warning indicators

Reset warning

Production start

Water washing start

Input weighing

Status after weighing

Divergent into S21 and S22

Input material to stirring unit

Add liquid to stirring unit

Complete dried material and liquid input, transfer the status to S23

Stirring timer

Wash stirring unit

Input clean water

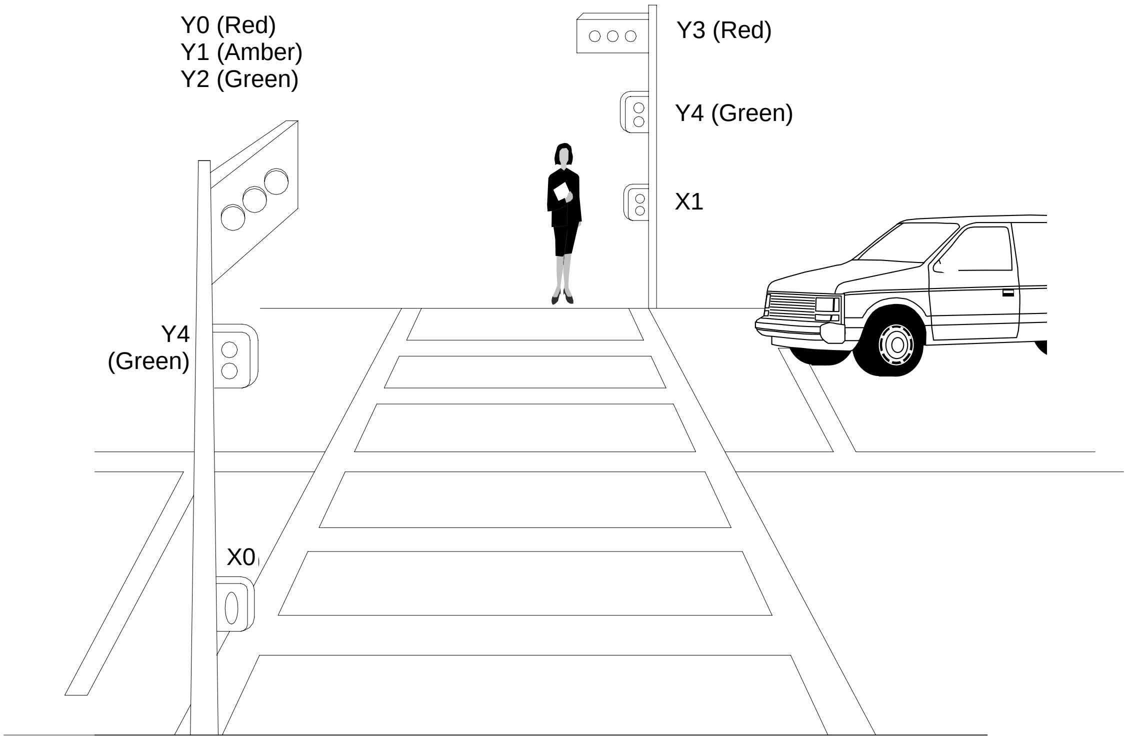
Drain water out

Output finished product and accumulate the cycle

FP-08

```
ORG       M1924
TO        S0
-----------------
STP       S0
OUT TR    0
AND NOT   X1
SET       Y1
LD  TR    0
AND NOT   X2
SET       Y2
LD  TR    0
AND       X3
SET       Y3
LD  TR    0
AND       X4
SET       Y4
LD  TR    0
AND       X5
RST       Y1
RST       Y2
RST       Y3
RST       Y4
FROM      S0
OUT TR    1
AND       X6
AND NOT   Y1
AND NOT   Y2
AND NOT   Y3
AND NOT   Y4
TO        S20
LD  TR    1
AND       X7
AND NOT   Y3
AND NOT   Y4
TO        S24
-----------------
STP       S20
OUT       Y5
FUN       17
  Sa:R3840
  Sb:R0
FO        0
OUT       M0
FO        1
OUT       M1
FROM      S20
LD        M0
OR        M1
ANDLD
TO        S21
TO        S22
-----------------
STP       S21
OUT       Y6
T0   PV:  500
-----------------
STP       S22
OUT       Y7
T1   PV: 800
FROM      S21
FROM      S22
AND       T0
AND       T1
TO        S23
-----------------
STP       S23
OUT  TR   0
OUT       Y8
LD   TR   0
T2   PV:  4500
LD   TR   0
AND       X4
OUT       Y4
-----------------
STP       S24
OUT  TR   0
T3   PV:  500
LD   TR   0
AND NOT   T3
OUT       Y9
LD   TR   0
T4   PV:  1500
LD   TR   0
AND NOT   T4
OUT       Y10
FROM      S23
AND       T2
FROM      S24
AND       T4
ORLD
TO        S25
-----------------
STP       S25
OUT  TR   0
AND       X3
OUT       Y10
LD   TR   0
AND TU    S25
FUN       15DP
  D:R10
FROM      S25
AND NOT   X3
TO        S0
-----------------
STPEND
```

Example 3 Pedestrian Crossing Lights

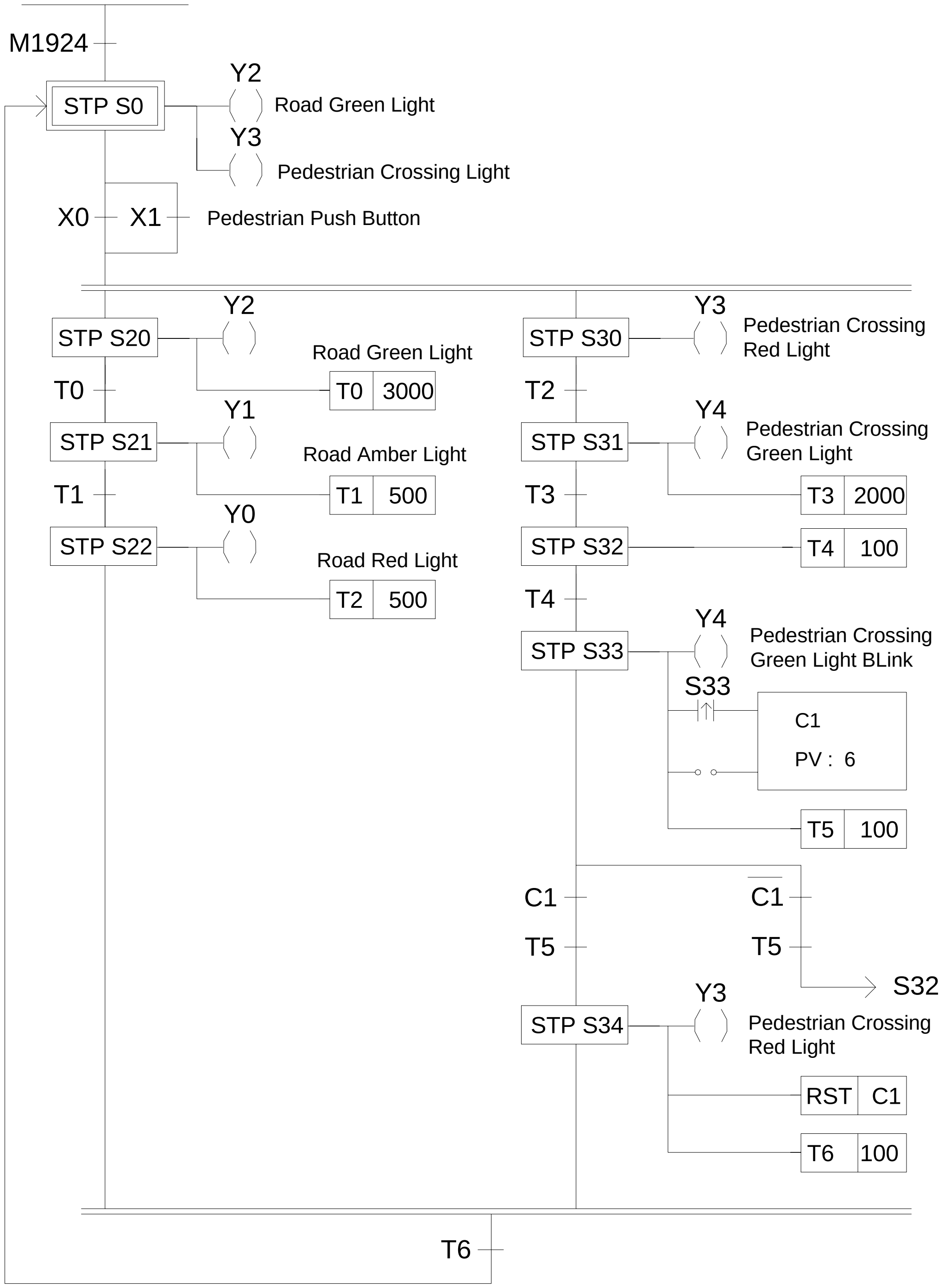


- Input Points: Pedestrian Push Button X0
  Pedestrian Push Button X1

- Output Points: Road Red Light Y0
  Road Amber light Y1
  Road Green Light Y2
  Pedestrian Crossing Red Light Y3
  Pedestrian Crossing Green Light Y4

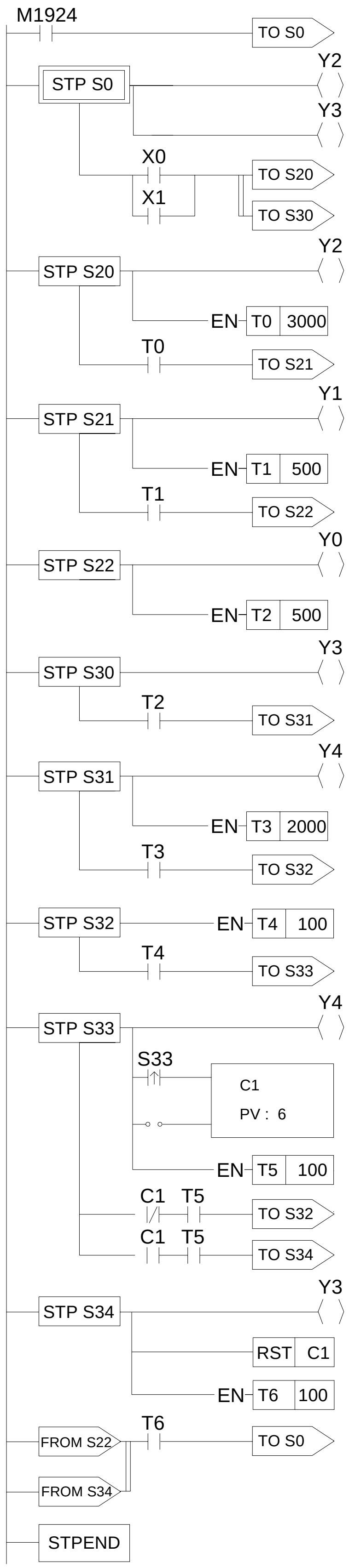
- M1918=0

- Pedestrian Crossing Lights Control Process Diagram



- Pedestrian Crossing Lights Control Program

WinProladder



FP-08

```
ORG      M1924
TO       S0
-----------------------
STP      S0
OUT      Y2
OUT      Y3
FROM     S0
LD       X0
OR       X1
ANDLD
TO       S20
TO       S30
-----------------------
STP      S20
OUT      Y2
T0  PV:  3000
FROM     S20
AND      T0
TO       S21
-----------------------
STP      S21
OUT      Y1
T1  PV:  500
FROM     S21
AND      T1
TO       S22
-----------------------
STP      S22
OUT      Y0
T2  PV:  500
-----------------------
STP      S30
OUT      Y3
FROM     S30
AND      T2
TO       S31
-----------------------
STP      S31
OUT      Y4
T3  PV:  2000
FROM     S31
AND      T3
TO       S32
```

```
STP      S32
T4  PV:  100
FROM     S32
AND      T4
TO       S33
-----------------------
STP      S33
OUT TR   0
OUT      Y4
LD  TR   0
AND TU   S33
LD       OPEN
C1  PV:  6
LD  TR   0
T5  PV:  100
FROM     S33
OUT TR   1
AND NOT  C1
AND      T5
TO       S32
LD  TR   1
AND      C1
AND      T5
TO       S34
-----------------------
STP      S34
OUT      Y3
RST      C1
T6  PV:  100
-----------------------
FROM     S22
FROM     S34
AND      T6
TO       S0
-----------------------
STPEND
```

8.6 Syntax Check Error Codes for Step Instruction

The error codes for the usage of step instruction are as follows:

E51 : TO(S0-S7) must begin with ORG instruction.

E52 : TO(S20-S999) can't begin with ORG instruction.

E53 : TO instruction without matched FROM instruction.

E54 : To instruction must comes after TO, AND, OR, ANDLD or ORLD instruction.

E56 : The instructions before FROM must be AND, OR, ANDLD or ORLD

E57 ：The instruction after FROM can’t be a coil or a function

E58 ：Coil or function must before FROM while in STEP network

E59 : More than 8 TO# at same network.

E60 : More than 8 FROM# at same network.

E61 : TO(S0-S19) must locate at first row of the network.

E62 : A contact occupies the location for TO instruction.

E72 : Duplicated TO Sxx instruction.

E73 : Duplicated STP sxx instruction.

E74 : Duplicated FROM sxx instruction.

E76 : STP(S0~S19) without a matched STPEND or STPEND without a matched STP(S0~S19).

E78 : TO(S20~S999), STP (S20~S999) or FROM instructions comes before or without STP(S0~S19).

E79 : STP Sxx or FROM Sxx instructions comes before or without TO Sxx.

E80 : FROM Sxx instruction comes before or without STP Sxx.

E81 : The max. level of branches must <=16.

E82 : The max. no. of branches with same level must <=16.

E83 : Not place the step instruction with TO->STP->FROM sequence.

E84 : The definition of STP# sequence not follow the TO# sequence.

E85 : Convergence do not match the corresponding divergence.

E86 : Illegal usage of STP or FROM before convergent with TO instruction.

E87 : STP# or FROM# comes before corresponding TO#.

E88 : During this branch, STP# or FROM# comes before the corresponding TO#.

E89 : FROM# comes before corresponding TO# or STP#.

E90 : Invalid To# usage in the simultaneous branch.

E91 : Flow control function cannot be used in the step ladder region