

# Chapter 4 SFC (Sequential Function Chart)

---

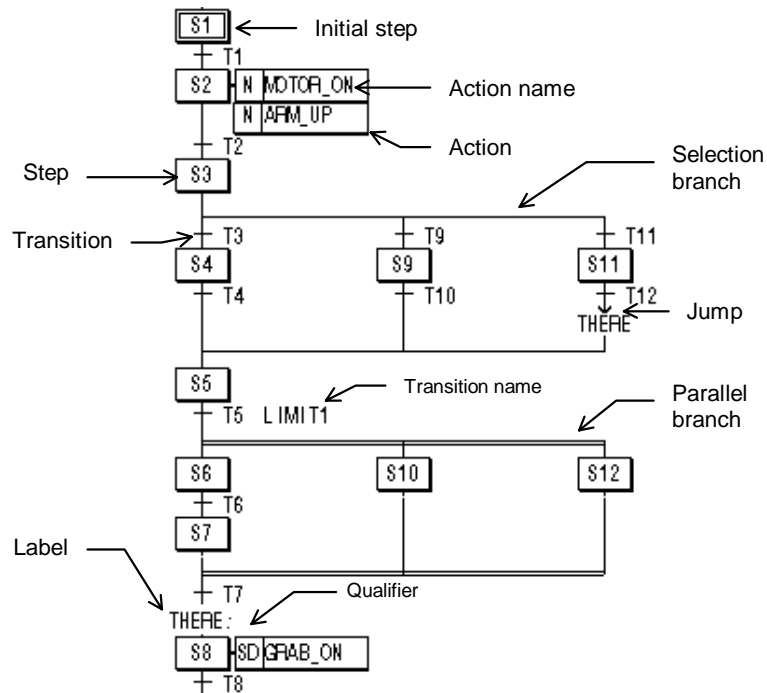
4.1. Overview .....	4-1
4.2. SFC structure .....	4-1
4.3. Rule of evolution .....	4-8



## 4. SFC(Sequential Function Chart)

### 4.1 Overview

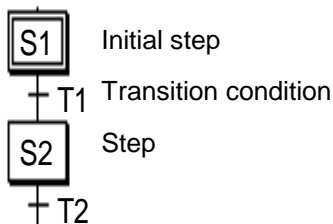
- SFC is the structured representation language which is executed by the flow chart format according to the application program's procedure using PLC language.
- SFC provides the connection method of the application program by dividing the step and transition and each step is related to the action and each transition is related to its condition.
- Only the program and function block of program type can use this SFC since SFC should have the status information.
- Type



### 4.2. SFC structure

#### 4.2.1. Steps

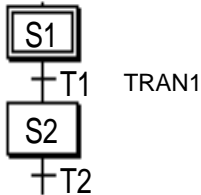
- Step indicates the unit of the sequence control by connecting the action.
- If the step is active, the attached action is executed.
- Initial step is initially active step.



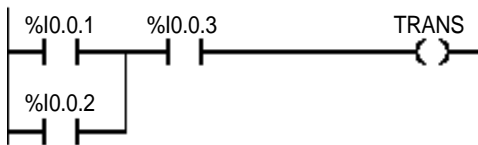
- If the transition condition is established after the initial step, current active step(S1) is changed to the inactive status and next step(S2) is active.

### 4.2.2 Transitions

- The transition indicates the execution process transition condition between steps.
- The transition condition shall be prepared by IL or LD of PLC language.
- If the result of transition is 1, current step will be inactive and next step will be active.
- The transition shall be arranged between steps.



Description of TRAN1



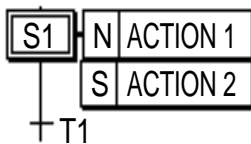
If TRANS is on, S1 will be inactive and S2 will be active.

TRANS variable is the declared one internally.

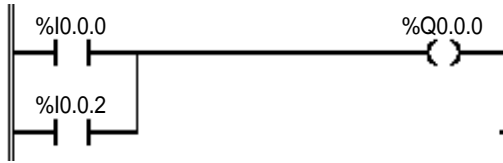
The transition condition shall be output by TRANS variable at all transition.

### 4.2.3. Actions

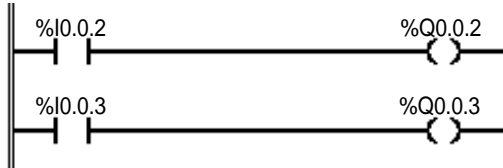
- Two actions can be connected to each step.
- The step without action is regarded as the standby until next transition condition is 1.
- The action consists of IL or LD of PLC language and the action is executed while the step is active.
- The qualifier is used to control the action.
- When the action is changed from active to inactive status, the contact output executed at the action will be 0. However, S, R, function, function block output remains the status before inactive status.



Content of ACTION1



Content of ACTION2



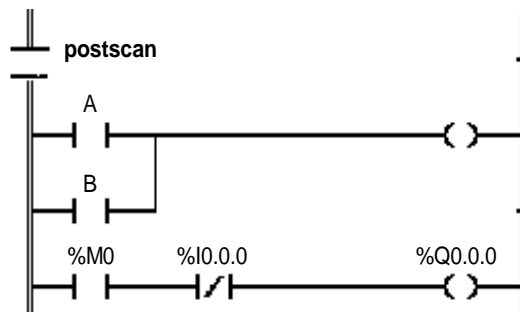
- ACTION1 is executed only when S1 is active.
- ACTION2 is executed till meets R qualifier after S1 is active. It will be continuously executed though S1 is inactive.
- At the moment of disabling of action, this action is post scanned and switched over next step.

**Reference**

**Post scan**

The action is scanned again when the action becomes inactive. The program output of contact program will be 0 because it is scanned regarding certain contact point(value = 0) is existed at the initial of action program.

The function, function block, S, R output and etc. are not included.



In above figure, C and %Q0.0.0 will be 0 since postscan's contact point is 0.

**4.2.4. Action qualifiers**

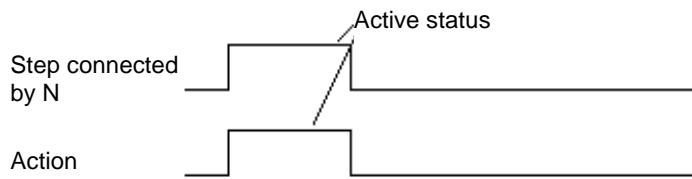
- Action qualifier is used whenever the action is used.
- The execution point/time of action relating to the step is defined according to the selected qualifier.
- The action qualifier is classified as follows;

## 4. SFC

---

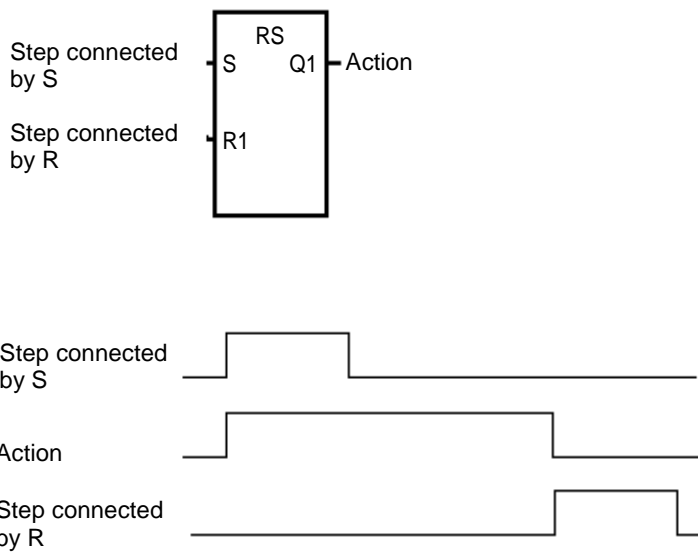
### 1) N(Non-Stored)

The action is executed as long as the step is active.



### 2) (S)Set

The action is executed, as soon as the step is active, till R qualifier is executed.

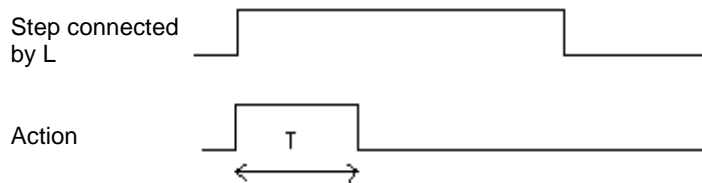
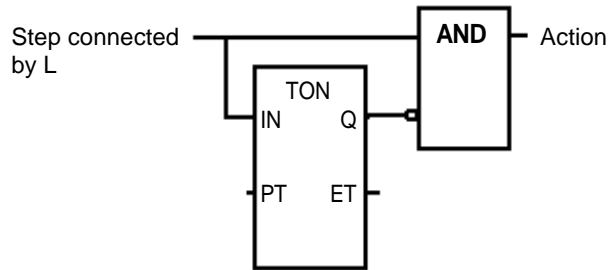


### 3) R(Overriding Reset)

The action executed by S, SD, DS and SL qualifier is aborted.

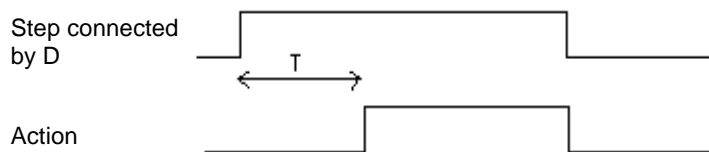
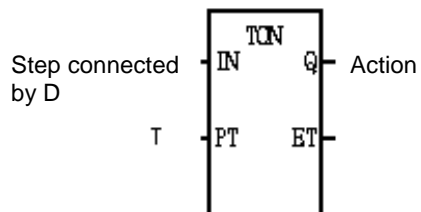
## 4) L(Time Limited)

The action is executed for a preset length of time T, as long as the step is active.



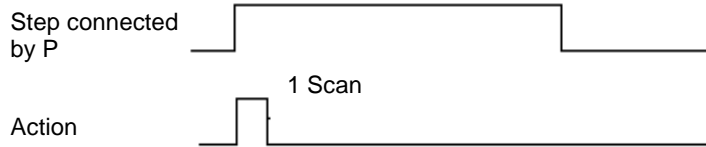
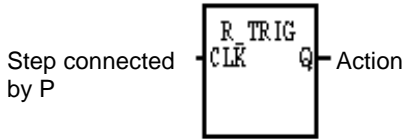
## 5) D(Time Delayed)

The action is executed after a preset time T has elapsed and remains executed for as long as the step is active.



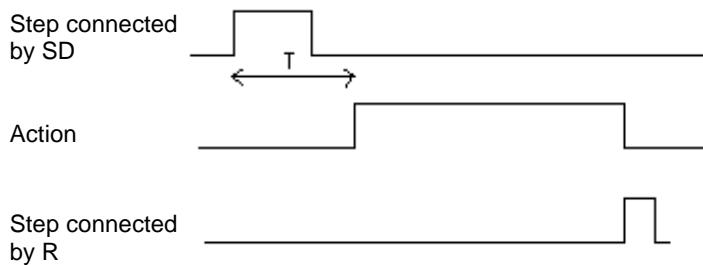
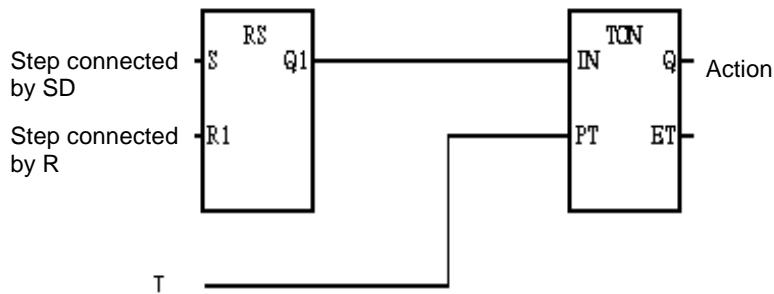
6) P(Pulse)

As soon as the step is active, the action is executed for one cycle.



7) SD(Stored & Time Delayed)

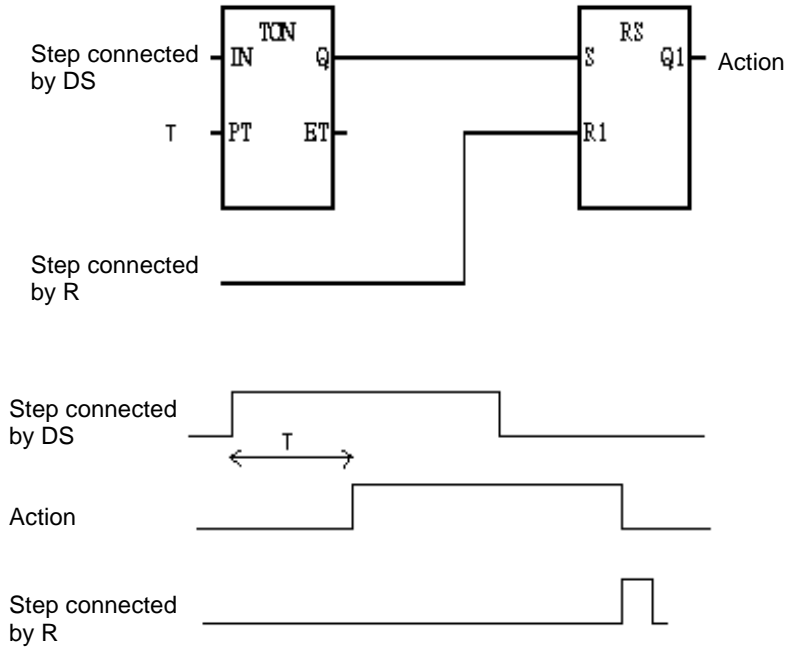
The action is executed when a preset period of time T has elapsed after step activation, even if the step becomes inactive. This condition persists until the R qualifier is executed.





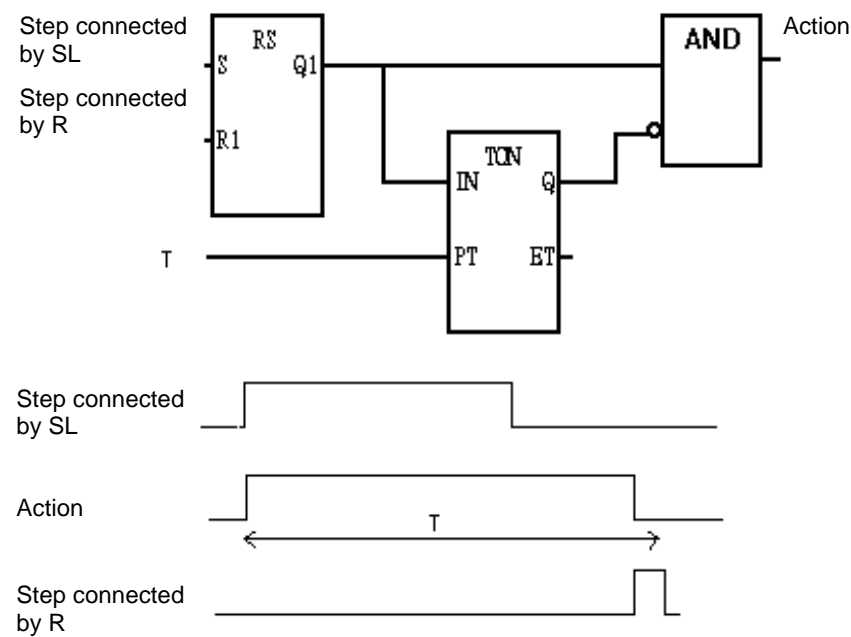
8) DS(Delayed & Stored)

As for the SD qualifier, the action is also executed with a time delay. It differs from the SP qualifier in that the step must remain active during the time delay.



9) SL(Stored & Time Limited)

The action is executed for a preset length of time T as soon as the step is active until the R qualifier is executed.



### 4.3. Rules of evolution

#### 4.3.1. Serial connection

- Two steps are not connected directly and always divided by the transition always.
- Two transitions are not connected directly and divided by the step always.



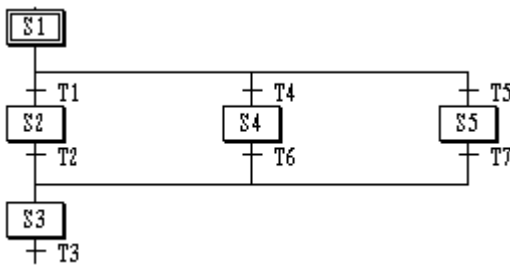
[Right example]    [Wrong example]

- Regarding the transition between steps of serial connection, lower step will be active when next transition condition becomes 1 under the activation of upper step.

#### 4.3.2. Selection branch

- The step next to the transition condition of 1 will be active under the activation of upper step in case of selection branch. The others are same to the serial connection.

**Example**

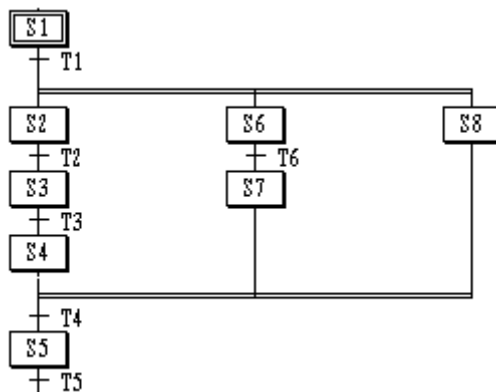


- \* When the transition condition of T1 is 1,  
The selection branch will be active in order of S1 → S2 → S3.
- \* When the transition condition of T4 is 1,  
The selection branch will be active in order of S1 → S4 → S3.
- \* When the transition condition of T5 is 1,  
The selection branch will be active in order of S1 → S5 → S3.  
When the transition condition is 1 simultaneously, left transition will be prior to the right transition.
- \* When the transition condition of T1 and T4 is simultaneously,  
The selection branch will be active in order of S1 → S2 → S3.
- \* When the transition of T1 and T5 is 1 simultaneously,  
The selection branch will be active in order of S1 → S4 → S3.

### 4.3.3. Parallel branch

- All steps which are connected below this transition will be active if the transition condition of next transition is 1 under the activation of upper step in case of parallel branch. At this time, the active step will be as many as the number of branch.
- If parallel branches are joined, next step will be active when last step of each branch is active and transition condition is 1.

#### **Example**



- \* When S1 is active and transition condition T1 is 1, S2, S6 and S8 will be active and S1 will be inactive.
- \* When S4, S7 and S8 are active and transition condition T4 is 1, S5 will be active and S4, S7 and S8 will be inactive.

\* Active order

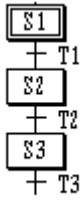
```

S1-+->S2--->S3--->S4+->S5
    +->S6--->S7-----+
    +->S8-----+
  
```

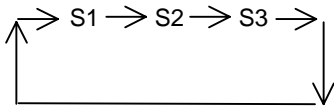
### 4.3.4. Jump

- SFC initial step will be active if next transition condition of the last step is 1 after SFC last step is active.

**Example**



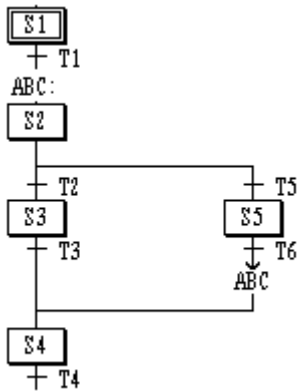
\* Active order



- The flow can be controlled to the aimed position using the jump.
- The jump can be extended only the end of SFC program or selection branch. The jump can not be done toward or outward parallel branch. But, the jump is possible within the parallel branch.

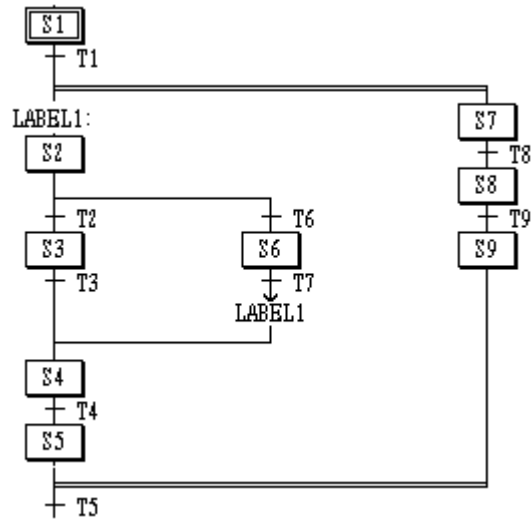
**Example**

1) Jump at the end of selection jump.

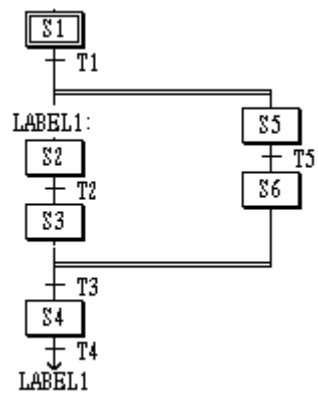


- S2 is active after S5.

- 2) Jump in parallel branch



- 3) The jump can not be done into parallel branch.



MEMO

