

基于普通 Petri 网的梯形图中接通延时定时器的建模方法

温世刚 罗继亮 倪会娟 陈雪琨

(华侨大学信息科学与工程学院 厦门 361021)

摘 要 针对可编程逻辑控制器(PLC)的接通延时定时器(TON),提出了其普通 Petri 网建模方法。将状态变量描述为库所对,将 TON 中的指令运算描述为变迁,从而将 TON 模块描述为一个普通 Petri 网结构。结果表明该 Petri 网能够准确地描述 TON 的动态行为,从而为 PLC 梯形图程序的形式化设计或验证提供了建模方法。

关键词 可编程逻辑控制器, Petri 网, 梯形图, 接通延时定时器

中图分类号 TP273 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.07.031

Method for Modeling On-delay Timers in Ladder Diagrams Based on Ordinary Petri Nets

WEN Shi-gang LUO Ji-liang NI Hui-juan CHEN Xue-kun

(College of Information Science and Engineering, Huaqiao University, Xiamen 361021, China)

Abstract The method was proposed for modeling an on-delay timer (TON) of programmable logic controller (PLC) as an ordinary Petri net. A boolean variable is represented by a pair of places, while an instruction to be executed in PLC is represented by a transition. Consequently, a TON is modeled by an ordinary Petri net. The results show that the behavior of a PLC system can be represented by the states of its Petri net model. Evidently, this method can be used for the formal design and verification of PLC programs.

Keywords Programmable logic controller (PLC), Petri nets, Ladder diagram, On-delay timer (TON)

1 引言

梯形图^[1]是 可 编 程 逻 辑 控 制 器 (PLC) 最 为 常 用 的 设 计 语 言, 梯 形 图 的 执 行 过 程 类 似 继 电 器 电 路, 便 于 电 气 工 程 师 接 受。但是梯形图无法图形化地描述复杂控制系统的系统结构,并不是一种严谨的形式化语言。由于复杂 PLC 控制系统的状态是各控制单元状态的排列组合,存在状态空间爆炸问题,其传统测试方法难以确保梯形图程序的可靠性和正确性,因此试图发展形式化的梯形图程序设计方法,为此需要给出梯形图程序的形式化建模方法。

Petri 网是一种形式化数学模型,和自动机相比,具有较高的建模效率^[2]。韩赞东、刘继国和罗晟^[3]赋予了库所和变迁实际的控制属性,构造了可用于描述和设计控制系统的控制 Petri 网,并引入宏库所和宏操作,描述了网与网之间的相互作用,实现了对燃料元件装卸系统的分层设计。Suesut、In-ban 和 Nilas 等^[4]提出了通过布尔等式将 Petri 网转换为梯形图的方法,然后分析 Petri 网的死锁、冲突等性质,排除故障,转化为安全可靠的梯形图程序。通过这种方法,所建的 Petri 网模型比较简单,但是对于复杂指令(如定时器)可能没有办法转换。Wightkin、Buy 和 Darabi^[5]提出将顺序功能图(SFC)转换为时间 Petri 网的方法,即在 Petri 网的变迁上赋予时间

信息,以模拟出 SFC 的时间属性。但是作者为了更多的描述 PLC 中继电器的时间信息,所构建的 Petri 网模型比较复杂,很容易出现状态空间爆炸的问题,给读者的理解和未来的应用带来很大的不便。Tasi 和 Teng^[6]提出了一种布尔 Petri 网,将梯形图程序用布尔表达式表示,然后赋予变迁,并且运用 Petri 网去构造一种抽象的模型,通过这种模型可以诊断出梯形图中的错误。这种方法尽管网结构很小,但是网模型很抽象,再加上变迁是通过布尔表达式描述,使读者理解起来更加困难。Chen、Luo 和 Qi^[7]提出一种将梯形图程序转换为普通 Petri 网的方法,该方法是针对梯形图中位逻辑指令的建模方法,建模方法比较完整,能够形象地描述出指令在梯形图中的执行过程。但是,作者仅仅考虑的是一些梯形图中简单的逻辑指令,并没有涉及复杂的指令(例如定时器等),在实际应用中有一定的局限性。

因此,面对上述提到的问题,本文提出了一种将梯形图中接通延时定时器(TON)模块转化为普通 Petri 网的算法。文中将 TON 中的状态量用库所对表示,将指令运算描述为变迁,忽略时间信息,通过变迁使能的先后顺序,相应地表示了 TON 输入输出的切换动作。与文献[4-6]相比,本文采用了普通 Petri 网的建模方法,其语义比较简单,容易理解和分析,而且更清楚地描述了 TON 的工作原理。与文献[7]相比,本

到稿日期:2013-04-05 返修日期:2013-05-18 本文受国家自然科学基金(61203040),工业控制技术国家重点实验室开放课题(ICT1317),福建省高等学校新世纪优秀人才支持计划(11FJRC01),福建省高校杰出青年科研人才培育计划(JA10004),中央高校基本科研业务费专项基金(JB-SJ1006)资助。

温世刚(1987—),男,硕士,主要研究方向为可编程逻辑控制器的验证,E-mail:7079409@163.com;罗继亮(1977—),男,副教授,主要研究方向为离散事件系统和程序验证(通信作者)。

文考虑了定时器的建模,应用范围更广一些。并且文中还设计了梯形图模型的封装模块,这样应用起来更加方便。结果,可以通过 Petri 网理论去分析和模拟含有 TON 的梯形图程序,也为 PLC 程序仿真和验证建立了理论基础。

2 基础知识

2.1 接通延时定时器

如图 1 所示,可编程逻辑控制器(PLC)的接通延时定时器(TON)^[8]具有 4 个端子,IN 为 TON 输入端,Q 为 TON 的输出端,PT 为 TON 的设定值,ET 为 TON 的当前值。

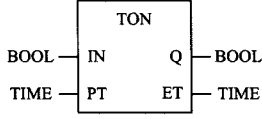


图 1 TON 模块示意图

当 IN 端布尔值变为 1 时,表示 TON 输入端与左母线接通,TON 启动,开始定时,ET 由 0 开始增加,当“ET=PT”时,定时时间到,Q 端布尔值变为 1,输出线圈动作。当 IN 端布尔值变为 0 时,则当前值 ET 变为 0,Q 端布尔值变为 0,输出线圈动作。

2.2 Petri 网基础知识

Petri 网可以被定义为一个四元组,记为 $\mathcal{N} = (\mathcal{P}, \mathcal{T}, \mathcal{F}, \mathcal{W})$ 。 $\mathcal{P}$ 是有限库所集; $\mathcal{T}$ 是有限变迁集; $\mathcal{P} \cap \mathcal{T} = \emptyset, \mathcal{T} \cup \mathcal{P} \neq \emptyset; \mathcal{F} \subseteq (\mathcal{P} \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P})$ 是库所和变迁或变迁和库所组成的二元组的集合,表示从库所到变迁或从变迁到库所的有向弧的集合; $\mathcal{W}: \mathcal{F} \rightarrow \mathbb{N}^+$ 表示在 $\mathcal{F}$ 中每一条弧上的权值。当 $\mathcal{F}$ 中弧上的权值均为 1 时,我们称此时的 Petri 网为普通 Petri 网。Petri 网系统是 $(\mathcal{N}, m_0)$,其中 m_0 是初始标识。标识是一个向量 $m: \mathcal{P} \rightarrow \{0, 1, 2, \dots\}$,其中第 i 维上的分量记为 $m(p_i)$,表示库所 p_i 的标识。关于 Petri 网的更多介绍,参考文献[9-11]。

3 TON 的建模方法

TON 尽管含有时间信息,并且分辨率有 1ms、10ms 和 100ms 3 种,但是在分析逻辑错误时并不需要考虑时间信息。因此,在将 TON 模块转换为普通 Petri 网的过程中可以将时间信息忽略,仅仅关心开关动作发生的顺序和 TON 输入与输出的状态切换顺序。

由于在 TON 设定完毕之后,设定值 PT 是一个固定的值,不会发生变化;当前值 ET 的变化是由 0 开始增加,直到达到设定值。因此,文中将当前值 ET 抽象为 3 种状态:TON 未工作状态(“ET=0”),开始定时状态(“0<ET<PT”)和达到设定值状态(“ET=PT”)。

为了方便和简洁,在梯形图中 TON 省略了内部的连接和构造,以模块的形式呈现给使用者。因此,文中也将 TON 的 Petri 网模型模拟为模块的形式,并且也会将模块内部用 Petri 网表示给读者。这样也为以后带有 TON 的梯形图程序向 Petri 网转换带来了方便。

定义 1^[7] 在梯形图中,路径指由几个触点组成的序列,其中相邻的结点由导线相连。

定义 2 TON 的输入路径是指连接左母线和 TON 输入端的路径,通常表示为 π_p ,其集合表示为 Π_p 。

定义 3 TON 的复位路径是指连接左母线和 TON 复位

线圈的路径,通常表示为 π_r ,其集合表示为 Π_r 。

定义 2 中定义的 TON 的输入路径是使得 TON 输入端与左母线接通,而定义 3 中定义的 TON 的复位路径是能够使得 TON 复位线圈接通。

定义 4 TON 的输入割集是由 TON 的输入路径中的触点组成的集合,并且如果将该集合内的触点从梯形图中删除,那么就断开了所有 TON 的输入路径。如果一个 TON 的输入割集的任何真子集都不是输入割集,那么它称为该 TON 的最小输入割集,通常用符号 ψ_{ks} 表示,其集合表示为 Ψ_{ks} 。

定义 5 TON 的复位割集是由 TON 的复位路径中的触点组成的集合,并且如果将该集合内的触点从梯形图中删除,那么就断开了所有 TON 的复位路径。如果一个 TON 的复位割集的任何真子集都不是复位割集,那么它称为该 TON 的最小输入割集,通常用符号 ψ_{rs} 表示,其集合表示为 Ψ_{rs} 。

定义 4 中定义的 TON 最小输入割集是使得 TON 输入端与左母线断开,定义 5 中定义的 TON 最小复位割集是使得 TON 复位线圈断电。

定义 6^[12] 对于路径上的触点 v_s (s 为开关量的名称),它可以描述为一个 Petri 网模型 $\mathcal{N}_s = (\mathcal{P}_s, \mathcal{T}_s, \mathcal{F}_s)$ 。其中,库所集合 $\mathcal{P}_s = \{p_{s,off}, p_{s,on}\}$, $p_{s,off}$ 表示常开触点的断开状态(或常闭触点的闭合状态), $p_{s,on}$ 表示常开触点的闭合状态(或常闭触点的断开状态);变迁集合 $\mathcal{T}_s = \{t_{s,on}, t_{s,off}\}$, $t_{s,on}$ 表示由状态 $p_{s,off}$ 变为状态 $p_{s,on}$ 的过程,而 $t_{s,off}$ 表示由状态 $p_{s,on}$ 变为状态 $p_{s,off}$ 的过程;有向弧集合为 $\mathcal{F}_s = \{(p_{s,off}, t_{s,on}), (t_{s,on}, p_{s,on}), (p_{s,on}, t_{s,off}), (t_{s,off}, p_{s,off})\}$ 。初始标识 $m_0^s = [1 \ 0]^T$,即 $m_0^s(p_{s,off}) = 1, m_0^s(p_{s,on}) = 0$ 。

以 TON 的输入路径上的一个触点 v_x 为例,它可以描述为一个 Petri 网结构 $\mathcal{N}_x = (\mathcal{P}_x, \mathcal{T}_x, \mathcal{F}_x)$,如图 2 所示。

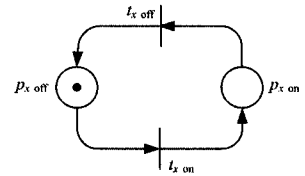


图 2 一个触点 v_x 的 Petri 网模型

定义 7 对于 TON 的一条输入路径 π_{ip} 和一条复位路径 π_{rp} ,它们分别被模拟为变迁 $t_{ip,on}$ 和变迁 $t_{rp,on}$,其集合分别用 $\mathcal{T}_{ip}$ 和 $\mathcal{T}_{rp}$ 表示;对于 TON 的一个最小输入割集 ψ_{ks} 和一个最小复位割集 ψ_{rs} ,它们分别被模拟为变迁 $t_{ks,off}$ 和变迁 $t_{rs,off}$,其集合分别用 $\mathcal{T}_{ks}$ 和 $\mathcal{T}_{rs}$ 表示。

定义 8 TON 内部结构部分和输出部分的指令运算描述为变迁 $t_{j,on}, t_{j,off}$ 。其中,变迁 $t_{j,on}$ 表示由状态 $p_{j,off}$ 变为状态 $p_{j,on}$ 的过程; $t_{j,off}$ 表示由状态 $p_{j,on}$ 变为 $p_{j,off}$ 的过程。

算法 1 基于普通 Petri 网的梯形图中 TON 建模方法设计

输入:TON 模块(含有输入、输出和复位部分)

输出:普通 Petri 网结构 $\mathcal{N} := (\mathcal{P}, \mathcal{T}, \mathcal{F}), (\mathcal{N}, m_0)$

第一部分 对 TON 输入、复位部分进行转换

S1 根据定义 6,将 TON 的输入路径和复位路径上的所有触点描述为 Petri 网模型。

S2 设计 5 个库所,3 个为 $p_{in,off}, p_{in,on}$ 和 $p_{in,rise}$,它们分别表示 TON 输入端与左母线断开、TON 输入端与左母线接通和 TON 输入端与左母线从接通到断开的切换状态,且初

始状态为 $m_0(p_{in,off})=1$ 、 $m_0(p_{in,on})=0$ 和 $m_0(p_{in,rise})=0$ ；2 个为 $p_{r,off}$ 和 $p_{r,on}$ ，它们分别表示 TON 复位线圈断电和 TON 复位线圈接通，且初始状态为 $m_0(p_{r,off})=1$ 和 $m_0(p_{r,on})=0$ 。

S3 根据定义 7 得到 $\Pi_p = \{\pi_{ip,1}, \pi_{ip,2}, \dots, \pi_{ip,m_{ip}}\}$ 、 $\Psi_{ks} = \{\psi_{ks,1}, \psi_{ks,2}, \dots, \psi_{ks,m_{ks}}\}$ 、 $\Pi_{rp} = \{\pi_{rp,1}, \pi_{rp,2}, \dots, \pi_{rp,m_{rp}}\}$ 和 $\Psi_{rs} = \{\psi_{rs,1}, \psi_{rs,2}, \dots, \psi_{rs,m_{rs}}\}$ 对应的变迁集合分别为 $T_{ip} = \{t_{ip,1,on}, t_{ip,2,on}, \dots, t_{ip,m_{ip},on}\}$ 、 $T_{ks} = \{t_{ks,1,off}, t_{ks,2,off}, \dots, t_{ks,m_{ks},off}\}$ 、 $T_{rp} = \{t_{rp,1,on}, t_{rp,2,on}, \dots, t_{rp,m_{rp},on}\}$ 和 $T_{rs} = \{t_{rs,1,off}, t_{rs,2,off}, \dots, t_{rs,m_{rs},off}\}$ ，其中 m_{ip} 、 m_{ks} 、 m_{rp} 和 m_{rs} 分别为 TON 的输入路径条数、最小输入割集个数、复位路径条数和最小复位割集个数。

S4 T_{ip} 中变迁表示由状态 $p_{in,off}$ 变为 $p_{in,on}$ 的过程，有向弧连接为 $1 \leq a \leq m_{ip}$ ， $(p_{in,off}, t_{ip,a,on})$ 、 $(t_{ip,a,on}, p_{in,on})$ 和 $(t_{ip,a,on}, p_{in,rise})$ ； T_{ks} 中变迁表示由状态 $p_{in,on}$ 变为 $p_{in,off}$ 的过程，有向弧连接为 $1 \leq b \leq m_{ks}$ ， $(p_{in,on}, t_{ks,b,off})$ 和 $(t_{ks,b,off}, p_{in,off})$ ； T_{rp} 中变迁表示由状态 $p_{r,off}$ 变为 $p_{r,on}$ 的过程，有向弧连接为 $1 \leq c \leq m_{rp}$ ， $(p_{r,off}, t_{rp,c,on})$ 和 $(t_{rp,c,on}, p_{r,on})$ ； T_{rs} 中变迁表示由状态 $p_{r,on}$ 变为 $p_{r,off}$ 的过程，有向弧连接为 $1 \leq d \leq m_{rs}$ ， $(p_{r,on}, t_{rs,d,off})$ 和 $(t_{rs,d,off}, p_{r,off})$ 。

S5 当触点 v_y (y 为触点 v_y 对应的开关量名称) 为常开触点时， $p_{y1} = p_{y,on}$ ， $p_{y2} = p_{y,off}$ ；当触点 v_y 为常闭触点时， $p_{y1} = p_{y,off}$ ， $p_{y2} = p_{y,on}$ 。 $1 \leq a \leq m_{ip}$ ，当触点 $v_y \in \pi_{ip,a}$ 时，有向弧连接为 $(p_{y1}, t_{ip,a,on})$ 和 $(t_{ip,a,on}, p_{y1})$ ； $1 \leq b \leq m_{ks}$ ，当触点 $v_y \in \psi_{ks,b}$ 时，有向弧连接为 $(p_{y2}, t_{ks,b,off})$ 和 $(t_{ks,b,off}, p_{y2})$ ； $1 \leq c \leq m_{rp}$ ，当触点 $v_y \in \pi_{rp,c}$ 时，有向弧连接为 $(p_{y1}, t_{rp,c,on})$ 和 $(t_{rp,c,on}, p_{y1})$ ； $1 \leq d \leq m_{rs}$ ，当触点 $v_y \in \psi_{rs,d}$ 时，有向弧连接为 $(p_{y2}, t_{rs,d,off})$ 和 $(t_{rs,d,off}, p_{y2})$ 。 $a, b, c, d \in \mathbb{Z}^+$ 。

第二部分 对 TON 内部结构部分进行转换

S6 ET 被模拟为 3 种状态，需要用两对库所 $p_{e1,off}$ 和 $p_{e1,on}$ 、 $p_{e2,off}$ 和 $p_{e2,on}$ 表示，初始状态 $m_0(p_{e1,off})=1$ 和 $m_0(p_{e1,on})=0$ ， $m_0(p_{e2,off})=1$ 和 $m_0(p_{e2,on})=0$ 。当 $m(p_{e1,off})=1 \wedge m(p_{e2,off})=1$ 时，表示 TON 处于未工作状态；当 $m(p_{e1,on})=1 \wedge m(p_{e2,off})=1$ 时，表示 TON 处于定时状态，但还没有达到设定值状态；当 $m(p_{e1,on})=1 \wedge m(p_{e2,on})=1$ 时，表示 TON 处于达到设定值状态。

S7 根据定义 8，可以得到变迁 $t_{e1,on}$ 、 $t_{e2,on}$ 、 $t_{e1,r,off}$ 、 $t_{e1,i,off}$ 、 $t_{e2,r,off}$ 和 $t_{e2,i,off}$ 。这是因为由状态 $p_{e1,on}$ 变为状态 $p_{e1,off}$ 和由状态 $p_{e2,on}$ 变为状态 $p_{e2,off}$ 的过程均有两个。因此，添加有向弧的连接为 $(p_{e1,off}, t_{e1,on})$ 、 $(t_{e1,on}, p_{e1,on})$ 、 $(p_{e1,on}, t_{e1,r,off})$ 、 $(t_{e1,r,off}, p_{e1,off})$ 、 $(p_{e1,on}, t_{e1,i,off})$ 、 $(t_{e1,i,off}, p_{e1,off})$ 、 $(p_{e1,on}, t_{e2,on})$ 、 $(t_{e2,on}, p_{e1,on})$ 、 $(p_{e2,off}, t_{e2,on})$ 、 $(t_{e2,on}, p_{e2,off})$ 、 $(p_{e2,on}, t_{e2,r,off})$ 、 $(t_{e2,r,off}, p_{e2,off})$ 、 $(p_{e2,on}, t_{e2,i,off})$ 和 $(t_{e2,i,off}, p_{e2,off})$ 。

S8 根据 TON 的工作原理，TON 启动的条件是 $m(p_{in,rise})=1$ ，即 TON 输入端与左母线由断开到接通的瞬间。TON 不工作的条件是 $m(p_{in,off})=1 \vee m(p_{r,on})=1$ ，即只要 TON 输入端与左母线断开或者 TON 的复位线圈接通，TON 就会立刻停止计时并且复位。因此根据此原理，添加有向弧连接为 $(p_{in,rise}, t_{e1,on})$ 、 $(p_{in,off}, t_{e1,i,off})$ 、 $(t_{e1,i,off}, p_{in,off})$ 、 $(p_{in,off}, t_{e2,i,off})$ 、 $(t_{e2,i,off}, p_{in,off})$ 、 $(p_{r,on}, t_{e1,r,off})$ 、 $(t_{e1,r,off}, p_{r,on})$ 、 $(p_{r,on}, t_{e2,r,off})$ 和 $(t_{e2,r,off}, p_{r,on})$ 。

S9 设计 TON 的封装模块图，如图 3 所示。长方形框上边的 5 个端口是 5 个变迁，右起变迁使能时分别表示的意

思是 TON 输入端与左母线接通、TON 处于到达设定值时输入端与左母线断开、TON 处于没到达设定值时输入端与左母线断开、TON 处于到达设定值时 TON 复位线圈接通和 TON 处于没到达设定值时 TON 复位线圈接通；下边的两个端口是 2 个库所，右起分别表示 TON 未达到设定值状态时输出端口为 off 状态和 TON 达到设定值状态时输出端口为 on 状态。模块内部的连接部分就是步骤 7、8 所叙述的，如图 4 所示。

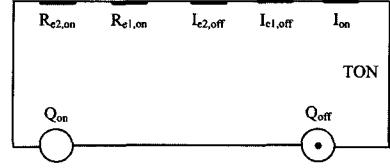


图 3 TON 的 Petri 网模块的封装图

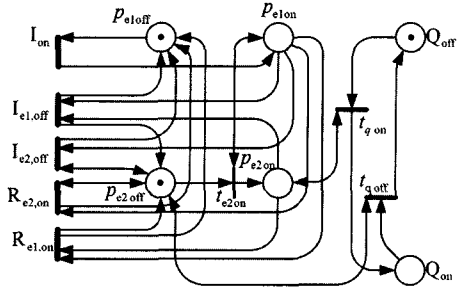


图 4 TON 的 Petri 网模块

第三部分 对 TON 输出部分进行转换

S10 设计两个库所 $p_{q,on}$ 和 $p_{q,off}$ 分别表示与 TON 输出端连接的一个输出线圈（不考虑置位线圈、复位线圈和取反线圈）的闭合和断开状态，当与 TON 输出端连接的线圈个数为 n 时，则设计的库所组成的库所集合为 $P_q = \bigcup_{0 < k \leq n} \{p_{q,k,on}, p_{q,k,off}\}$ ，初始状态为 $1 \leq k \leq n$ ， $k \in \mathbb{Z}^+$ ， $m_0(p_{q,k,off})=1$ ， $m_0(p_{q,k,on})=0$ 。

S11 根据定义 8，可以得到变迁集合 $T_q = \bigcup_{0 < k \leq n} \{t_{q,k,on}, t_{q,k,off}\}$ ， $k \in \mathbb{Z}^+$ ，其中， $t_{q,k,on}$ 使能表示对应的输出线圈闭合过程，即 TON 达到设定值状态； $t_{q,k,off}$ 使能表示对应的输出线圈断开过程。库所与变迁之间的有向弧连接为 $1 \leq k \leq n$ ， $k \in \mathbb{Z}^+$ ， $(p_{q,k,off}, t_{q,k,on})$ 、 $(t_{q,k,on}, p_{q,k,on})$ 、 $(p_{q,k,on}, t_{q,k,off})$ 、 $(t_{q,k,off}, p_{q,k,off})$ 、 $(p_{e2,on}, t_{q,k,on})$ 、 $(t_{q,k,on}, p_{e2,on})$ 、 $(p_{e2,off}, t_{q,k,off})$ 和 $(t_{q,k,off}, p_{e2,off})$ 。

4 示例

以西门子 S7-200PLC 中 TON 为例，电梯自动开门延时控制。当轿厢触碰到指定楼层限位开关，电梯延时 2s 开门。如图 5 所示是根据控制要求编写的 PLC 梯形图程序。I0.0 表示楼层限位开关，Q0.1 表示电梯运行状态，Q1.0 表示电梯轿厢门开关状态，I1.0 表示开门限位，I1.1 表示停止运行的开关。

当 I1.0 接通表示到达指定楼层，Q0.1 为假的时候，表示电梯停止上升或下降，停在指定楼层，此时 TON 启动，2 秒后，TON 接通，输出 Q1.0 为真，电梯轿厢门自动打开。I1.0 接通表示电梯门开启完毕，I1.1 接通表示电梯已停在指定楼层，但是电梯门还没有开启，按下复位指令。这时两者均可以使 TON 复位，之后有有效的输入信号时 TON 可以重新定时。

