**數位系統導論期末範例4**

**機器人控制器離散事件建模和電路合成**

**陳慶瀚**

**2022-06-10**

**1.機器人運動控制規格**

一個自走機器人的運動控制共有7個狀態：

- Start(初始狀態)

- Move\_forward(前進)

- Move\_back(後退)

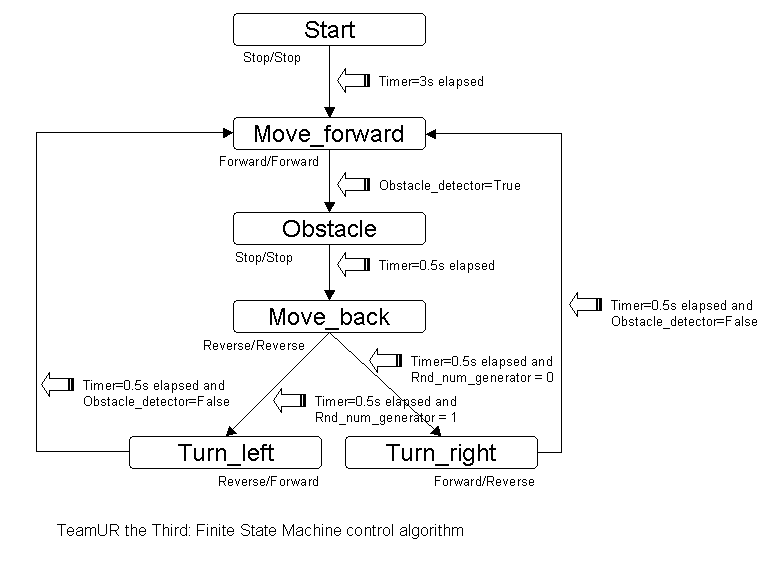
- Obstacle(偵側障礙物)

- Turn\_left(左轉)

- Turn\_right(右轉)

輸出是左右兩個馬達(Left\_motor/Right\_motor)的 控制訊號，每個馬達有3個可能的控制訊號值：Stop(00), Forward, Reverse

**2、機器人運動控制器狀態機**



**3、Grafcet離散事件建模**



**4. 機器人控制器電路合成**

**Grafcet控制器電路合成**

grafcet: process(CLK,RST)

begin

if RST='0' then

X0<='1'; X1<='0'; X2<='0'; X3<='0';

X4<='0'; X5<='0';

elsif CLK'event and CLK='1' then

if X0='1' and Tim="0110" then --3

X0 <= '0'; X1 <= '1';

elsif X1='1' and Obstacle ='1' then

X1 <= '0'; X2 <= '1';

elsif X2='1' and Tim="0001" then--0.5

X2 <= '0'; X3 <= '1';

elsif X3='1' and Tim="0001" and Rnd\_num\_generator = '1' then

X3 <= '0'; X4 <= '1';

elsif X3='1' and Tim="0001" and Rnd\_num\_generator = '0' then

X3 <= '0'; X5 <= '1';

elsif X4='1' and Tim="0001" and Obstacle\_detector = '1' then

X4 <= '0'; X0 <= '1';

elsif X5='1' and Tim="0001" and Obstacle\_detector = '1' then

X5 <= '0'; X0 <= '1';

end if;

end if;

end process grafcet;

**Datapath電路合成**

datapath: process(CLK,RST)

begin

if CLK'event and CLK='1' then

if X0='1' then

Left\_motor <= "10";

Right\_motor <= "10";

elsif X1='1' then

Left\_motor <= "11";

Right\_motor <= "11";

elsif X2='1' then

Left\_motor <= "10";

Right\_motor <= "10";

elsif X3='1' then

Left\_motor <= "00";

Right\_motor <= "00";

elsif X4='1' then

Left\_motor <= "00";

Right\_motor <= "11";

elsif X5='1' then

Left\_motor <= "11";

Right\_motor <= "00";

end if;

end if;

end process datapath;

**系統合成**

LIBRARY IEEE;

USE IEEE.STD\_LOGIC\_1164.all;

USE IEEE.STD\_LOGIC\_ARITH.ALL;

USE IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity grafcet\_vhdl4 is

port(

CLK : in std\_logic;

RST : in std\_logic;

Obstacle : in std\_logic;

Obstacle\_detector : in std\_logic;

Rnd\_num\_generator : in std\_logic;

Tim : in std\_logic\_vector (3 downto 0);-- 0.5

-- motor control : 00 Reverse; 11 Forward; 10 Stop

Left\_motor : out std\_logic\_vector (1 downto 0); Right\_motor : out std\_logic\_vector (1 downto 0)

);

end grafcet\_vhdl4;

architecture arch of grafcet\_vhdl4 is

signal X0,X1,X2,X3,X4,X5 : std\_logic;

signal T : std\_logic\_vector (3 downto 0);

begin

T <= Tim;

grafcet: process(CLK,RST)

begin

……

end process grafcet;

datapath: process(CLK,RST)

begin

……

end process datapath;

end arch;