Fault tolerance HW2 吳辰鋐

1. Fault －A physical defect,imperfection or flaw within the hardware or software component

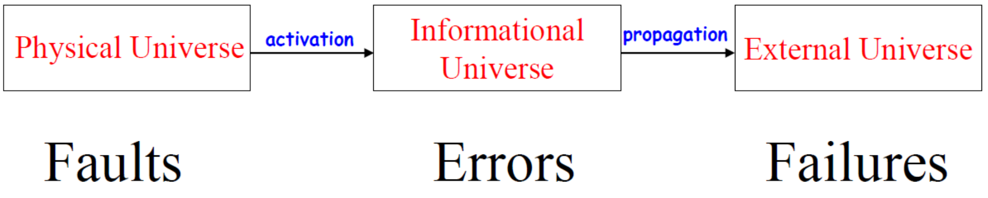
   Error － The manifestation of a fault. Specifically, it is a deviation from accuracy or correctness

   Failure － The deviation from expected actions or services or a non-performance of some action that is due or expected

   A programmer’s mistake is a fault; the consequence is a latent error in the written software; upon activation, the error becomes effectives; when this effective error produces erroneous output data a failure occurs.

   A short-circuit in an integrated circuit is a fault; the consequence is a line stuck at Boolean value is an error; when this produces wrong result of an arithmetic operation, it is a failure

   Relation between them:



2. (1) 20181021，台鐵普悠瑪出軌，火車 ATP(列車自動保護系統)故障而無法正常控制火車車速，導致超速而出軌

   (2) 20090601，法國航空 447 號班機空難，飛機的 pilot tube 結冰導致一連串系統錯誤未能回報正確訊息，載有 216 名乘客以及 12 名機組人員，在巴西聖佩德羅和聖保羅島嶼附近墜毀，機上人員全數罹難。

   (3) 19910225，在海灣戰爭期間，沙特阿拉伯達蘭的一枚美國愛國者導彈電池未能跟踪和攔截一枚伊拉克飛毛腿導彈。飛毛腿襲擊了美軍軍營，造成 28 名士兵死亡，另有約 100 人受傷，事後證明原因是由於計算機算術錯誤導致自引導以來的時間計算不准確。

3. Reliablity R(t) is the conditional probability that the system will perform correctly throught[0,t],given that it worked at time 0.

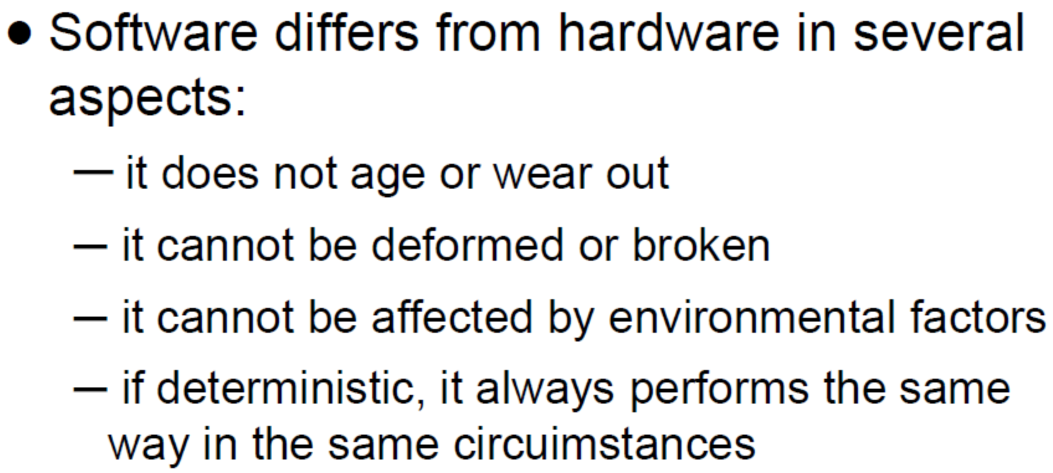
若reliablity愈高 則代表系統在0~t時間內可正常運作的機率愈大。
在不可修復錯誤的系統中reliablity相對重要。

4.
(1) reliablity定義為一段區間的系統正確機率，然而availabilty為特定瞬時的系統正確機率
(2)如Q3所述，若系統無法修復則reliablity必須很高，但因無法修復則availablity取在很長的時間之後則相對低
(3)若是不可修復的話則遲早壞掉，故steady-state availability為0

5.
(1)365*(1-0.9)=36.5 days
(2)365*(1-0.75)=91.25 days
(3)365*(1-0.5)=182.5 days

6.
365*24*60=525600 minutes/year
(525600-3)/525600 = 0.999994%
因為downtime 比3分鐘少，故steady-state availability會大於0.999994%

7.



- Software differs from hardware in several aspects:
  - — it does not age or wear out
  - — it cannot be deformed or broken
  - — it cannot be affected by environmental factors
  - — if deterministic, it always performs the same way in the same circuimstances

相較之下硬體錯誤受年齡及環境影響，且同樣條件下不一定每次都會發生

8.
(1)redundancy可解決硬體的老化問題，然而軟體並不存在老化問題，同樣輸入會有同樣輸出，故重複的部分也僅會發生同樣的錯誤，所以不適用

redundancy
(2)最直接有效的方式就是 N-version programming，利用 design diversity 來降低錯誤，可同時解決軟體硬體 faults，壞處是成本較高

9.

$$(f_{sd})' = \left(x_{n+1} f(x_1, x_2 \dots x_n) + x'_{n+1} f'(x'_1, x'_2, \dots x'_n)\right)'$$

→左右同取 dual

$$\Rightarrow \left(x'_{n+1} f(x'_1, x'_2, \dots x'_n) + x_{n+1} f'(x_1, x_2 \dots x_n)\right)'$$

$$= \left[x_{n+1} + \underbrace{f'(x'_1, x'_2 \dots x'_n)}_{f_d}\right] \times \left[x'_{n+1} + \underbrace{f(x_1, x_2 \dots x_n)}_{f}\right]$$

$$= \underbrace{x_{n+1} x'_{n+1}}_{0} + x_{n+1} f + x'_{n+1} f_d + f \times f_d$$

$$= x_{n+1} f + x'_{n+1} f_d + f \times f_d \underbrace{(x_{n+1} + x'_{n+1})}_{=1}$$

$$= x_{n+1} f \underbrace{(1 + f_d)}_{=1} + x'_{n+1} f_d \underbrace{(1 + f)}_{=1}$$

$$= x_{n+1} f + x'_{n+1} f_d = f_{sd}$$

10.
Input 有 00 01 10 11 四種 各 25%機率發生

(1)兩個 input 都正確的機率(1-P1-P0)^2
(3)兩個都 stuck at 1 僅 00 會答案錯誤 -> (P1^2)*(3/4)
(4)兩個都 stuck at 0 僅 00 答案正確 ->(P0^2)*(1/4)
(5)一個 stuck at 0 一個 stuck at 1 僅 00 答案錯誤 ->P0*P1*2*(3/4)
(6)一個 stuck at 0 另一正常 僅 10 或 01 答案錯誤 ->P0(1-P1-P0)*2*(3/4)
(7)一個 stuck at 1 另一正常 僅 00 答案錯誤 ->P1(1-P1-P0)*2*(3/4)

->Reliability =(1-P1-P0)^2 + (P1^2)*(3/4) + (P0^2)*(1/4) + P0*P1*(3/2) + P0(1-P1-P0)*(3/2) + P1(1-P1-P0)*(3/2)

11.

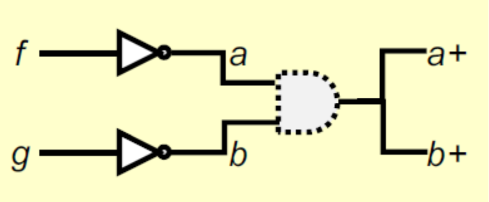
(1)permanent fault 是持續存在的 fault，transient fault 是沒有週期性、偶發的 fault，intermittent fault 是時好時壞，週期性發生的 fault

(2)True，ex:若電路 input 發生 stuck at fault，則整個 output 都會受到影響

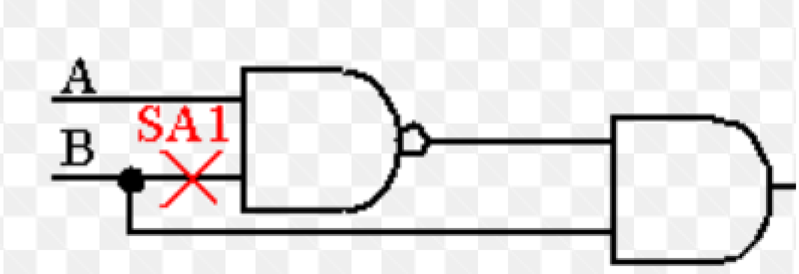
12.

(1)若 wire w1 和 w2 發生 bridge fault，則 w1 和 w2 值會受到彼此影響，可能同時 0 或同時 1，若僅用 stuck at 則固定為 0 或 1 而以

(2)常見 bridge fault model 包含了 wire AND、wire OR，視實際電路 CMOS 的驅動能力決定

Ex: 

13.



A redundant SA fault
(reconvergent fanout)

AB = 00 01 10 11

Output= 0 1 0 0

SA1 = 0 1 0 0 ->任何輸入情況下皆不會影響 output

14.

(1)365*24*5*0.121*(10^-8)=0.000052998

(2)1/(0.121*(10^-8))=826446280.992

15.

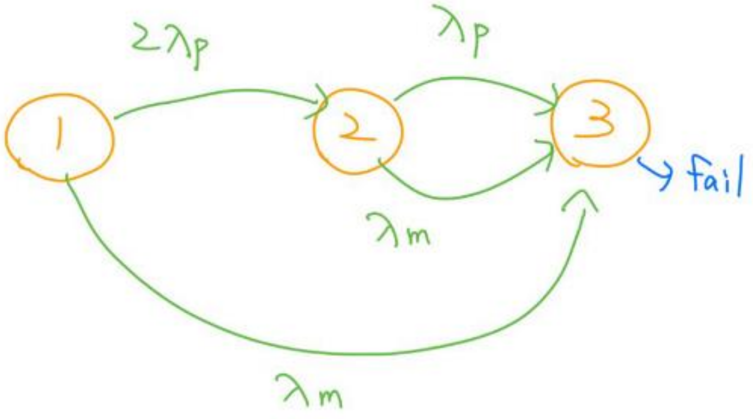
(1)eˆ-(a*365*24)=0.96 ->a = 0.00000466004

(2)1-(1-0.96)*(1-0.96)=0.999984

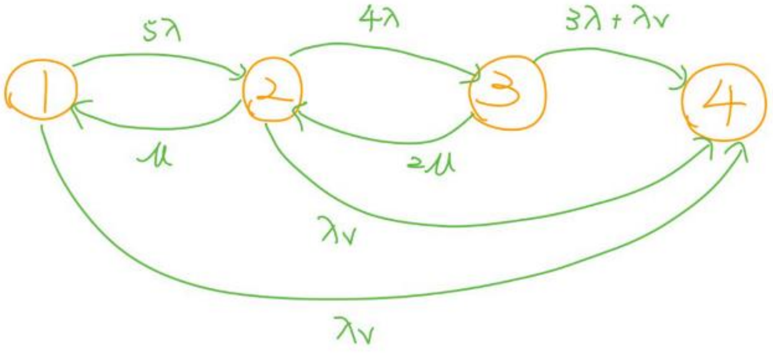
16.

$$
\begin{aligned}
R(i) &= R_1\left[1-(1-R_4)\left(1-R_2\left[\underbrace{1-(1-R_3)^2}_{1-(1+R_3^2-2R_3)}\right]\right)\right]\\
&= R_1\left[1-(1-R_4)\left(1-R_2\left[-R_3^2+2R_3\right]\right)\right]\\
&= R_1\left[1-(1-R_4)(1+R_2R_3^2-2R_2R_3)\right]\\
&= R_1\left[1-(1+R_2R_3^2-2R_2R_3-R_4-R_2R_3^2R_4+2R_2R_3R_4)\right]\\
&= -R_1R_2R_3^2+2R_1R_2R_3+R_1R_4+R_1R_2R_3^2R_4-2R_1R_2R_3R_4
\end{aligned}
$$

17.

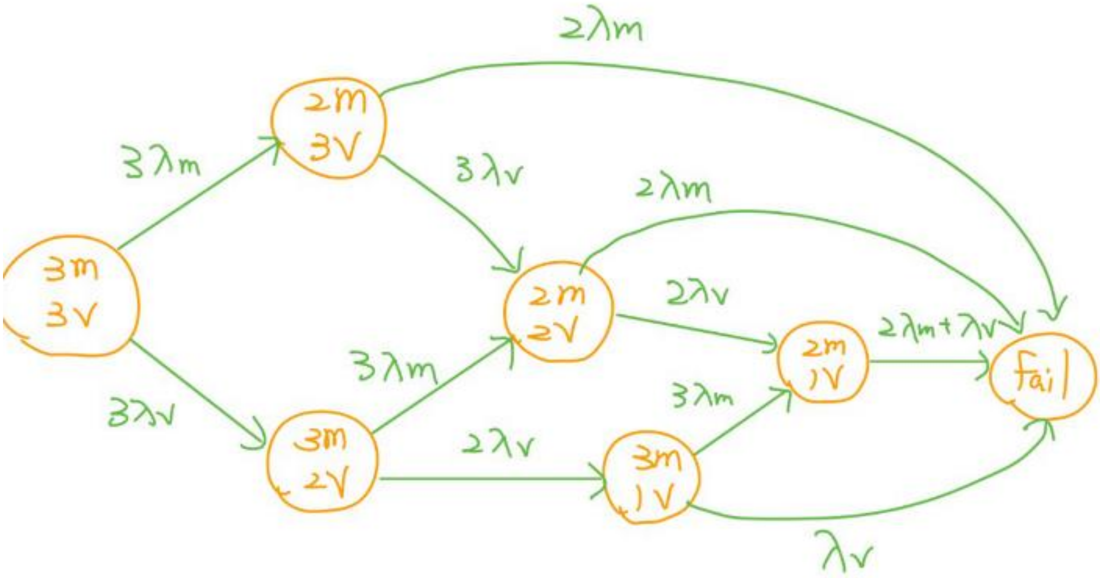


18.



可以，可將各個部分的reliability或availability以機率形式寫入矩陣，則整個系統的數字即可同樣由矩陣乘出

19.



20.
Advantage of watchdog
->cheap 、 improve availability
Disadvantage of watchdog
->coverage is limited,as neither the data nor the results are checked
、full restart causes loss of data

Advantage of heartbeats
->can handle more complex problem(ex:caused by traffic)
Disadvantage of heartbeats
->more complex