

VHDL

- RTL : Register Transfer Level

```

1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.std_logic_unsigned.all;
4  entity MON-ET is
5      generic (tp: time := 2ns);
6      port( A : in std_logic;
7            B : in std_logic;
8            S : out std_logic);
9  end entity MON-ET;
10 architecture FLOT of MON-ET is
11     begin
12         s <= a and b after tp;
13     end architecture FLOT;

```

Architecture structurelle

- and2 : entity work.MON-ET port map(A ⇒ A, B ⇒ B);

Comportemental

- Process : réalisation de parties séquentielles
 - Variables : abstraction algorithmiques
 - !!! Signaux ≠ Variables !!!

Conditionnelle IF :

```

1  library ieee;
2  use ieee.std_logic_1164.all;
3  entity decodeur is
4      port ( choix : in std_logic_vector(1
5            downto 0));
6      decode : out std_logic_vector(3
7            downto 0));
8  end entity decodeur;
9  architecture comport of decodeur
10     decodeage : process(choix) is
11     begin
12         IF (choix= "00") THEN decode <="0001
13         ";
14         ELSIF (choix="01") THEN decode <="
15         0010";
16         ELSIF (choix="10") THEN decode <="
17         0100";
18         ELSE decode <="1000";
19         END IF;
20     end process decodeage;
21 end architecture comport;

```

Conditionnelle CASE :

```

1  library ieee;
2  use ieee.std_logic_1164.all;
3  entity decodeur is
4      port ( choix : in std_logic_vector(1
5            downto 0));
6      decode : out std_logic_vector(3
7            downto 0));
8  end entity decodeur;
9  architecture comport of decodeur
10     decodeage : process(choix) is
11     begin
12         IF (choix= "00") THEN decode <="0001";
13         ELSIF (choix="01") THEN decode <="0010
14         ";
15         ELSIF (choix="10") THEN decode <="0100
16         ";
17         ELSE decode <="1000";
18         END IF;
19     end process decodeage;
20 end architecture comport;

```

Conditionnelle FOR :

```

1  process (A)
2  begin
3      Z <= "0000";
4      for I in 0 to 3 loop
5          if (A = I) then
6              Z(I) <= '1';
7          end if;
8      end loop;
9  end process;

```

Conditionnelle WITH/SELECT :

```

1  with a select b <=
2      "1000" when "00",
3      "0100" when "01",
4      "0010" when "10",
5      "0001" when "11";

```

Génération generate :

```

1  architecture GEN of REG.BANK is
2      component REG
3          port(D,CLK,RESET : in std_ulogic;
4                Q           : out std_ulogic);
5      end component;
6  begin
7      GEN_REG:
8          for I in 0 to 3 generate
9              REGX : REG port map
10                 (DIN(I), CLK, RESET, DOUT(I));
11          end generate GEN_REG;
12  end GEN;

```

FSM :

```

clocked : PROCESS(hor,raz)
BEGIN
  IF (raz = '0') THEN
    EtatPresent <= Etat0;
  ELSIF (hor'EVENT AND hor = '1') THEN
    EtatPresent <= EtatFutur;
  END IF;
END PROCESS clocked;
nextstate : PROCESS (EtatPresent,a)
BEGIN
  CASE EtatPresent IS
    WHEN Etat0 =>
      b <= '1';
      EtatFutur <= Etat1;
    WHEN Etat1 =>
      b <= '0';
      IF (a = '1') THEN
        EtatFutur <= Etat2;
      ELSIF (a = '0') THEN
        EtatFutur <= Etat1;
      ELSE
        EtatFutur <= Etat1;
      END IF;
    WHEN Etat2 =>
      b <= '0';
      EtatFutur <= Etat0;
    WHEN OTHERS =>
      EtatFutur <= Etat0;
  END CASE;
END PROCESS nextstate;

```

Testbenches :

```

file add1_tt : text open read_mode is "
add1_tt.dat";
variable ll : line;
variable tt : std_logic_vector(1 to 5);
while not endfile(add1_tt) loop
  readline(add1_tt,ll);
  read(ll,tt);

```

Machines à Etats

- Moore : Sortie dépendante de l'état courant
- Mealy : Sortie dépendante de l'état courant et des entrées

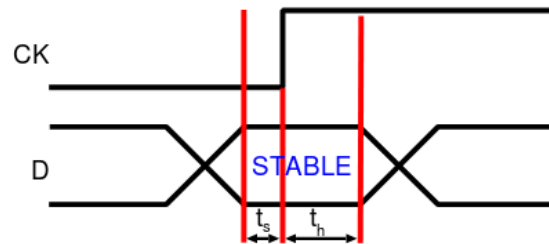
Méthode de réalisation

- Spécification du cahier des charges
- Détermination des états
- Identification des E/S de la MAE
- Graphe de transition
- Table de transition
- Nombre de bascules?
- Équation de l'état futur et des sorties
- Réalisation

Encodages

- NBC : Le nombre b est une somme de puissances de 2
- One-Hot : Le bit n représente le nombre n

Métastabilité

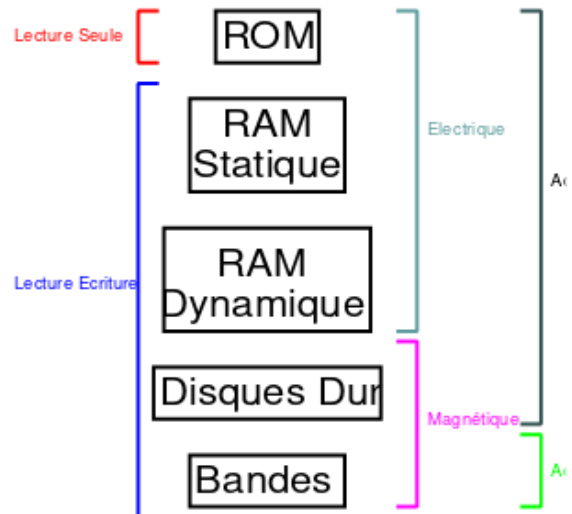


- t_s le temps de résolution pour que la donnée passe à un état stable.
- t_h le temps de front montant d'horloge où la donnée est stable.
- si $t_h \leq 0$, risque de métastabilité!
- MTBF : Mean Time Between Failure

Mémoires

- ROM : Read-Only Memory
- RAM : Random Access Memory
- ROM = RAM, exemple de NON-RAM : bobine magnétique, carte perforée, etc

Classification



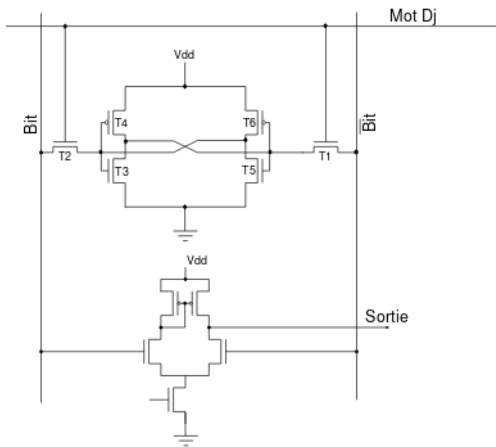
- bit : Binary digIT, plus petite quantité binaire
- octet : information codée sur 8 bits, unité binaire référence
- mot : taille du bus de données
- Data Bus (M) : rend un mot de M bits depuis la mem.
- Addr Bus (N) : sélectionne un mot parmi 2^N .
- CS (Chip Select) : Activation du circuit
- R : Accès = lecture.
- W : Accès = écriture.

Mémoires RAM

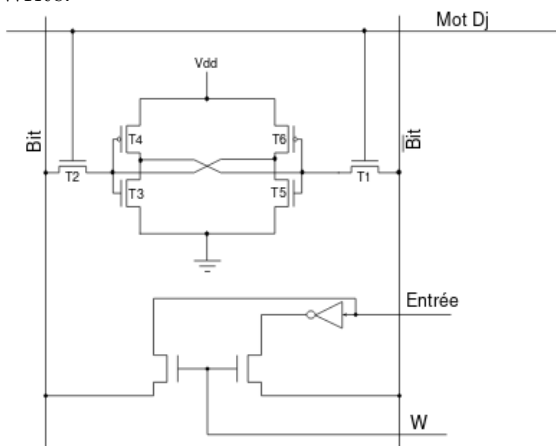
Technologie SRAM

- Variante CAM : adressée par contenu (caches associatifs).
- Variante biport : Deux accès simultanés en mémoire.

Read:

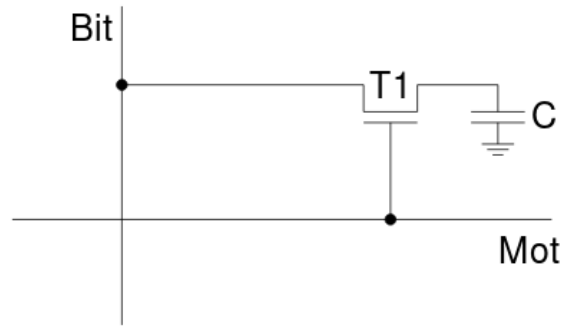


Write:



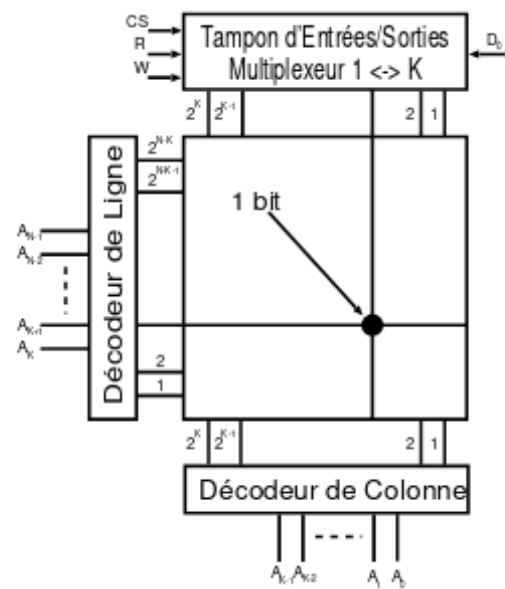
- [+] Rapide
- [+/-] Statique : rétention infinie tant qu'alimentée.
- [+] Techno CMOS : faible consommation.
- [-] Volumineuse.
- [-] Chère à produire.
- [--] Volatile.

Technologie DRAM

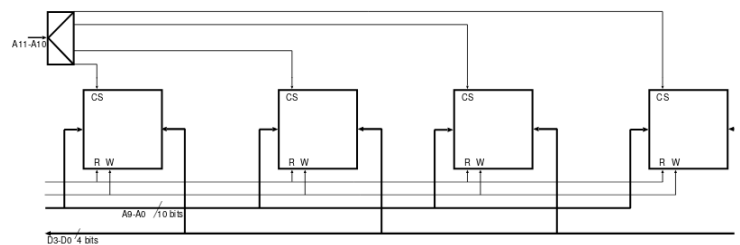


- [++] Compact.
- [+] Techno CMOS : faible consommation.
- [-] Lecture destructrice : réécriture après lecture.
- [-] Déchargement du condensateur : rechargement régulier.
- [--] Volatile.

Organisation interne



Organisation externe



Technologie SDRAM

- Synchronous DRAM.
- Mémoire câblée pour ne plus dépendre des délais matériels.
- Introduction du mode "rafale" (BURST)

Technologie DDRAM

- Double-Rate DRAM.
- Lectures sur le front montant et descendant de l'horloge.

Technologie QDRAM

- Quad-rate DRAM.
- Dissocier Entrée de Sortie.
→ Lecture et écriture simultanée sur le cycle.

Mémoires ROM

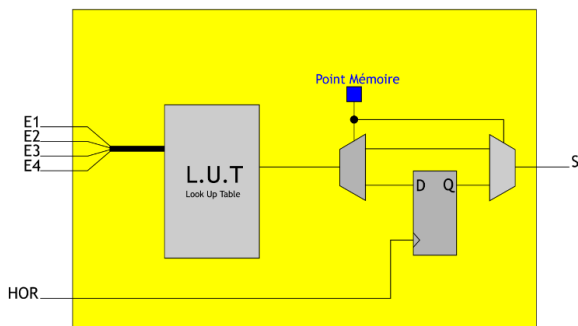
- Nécessité de mémoires non-volatile
- Ex: BIOS de μ -ordinateurs.
- Ex2: stockage de programmes en systèmes embarqués.

Types de ROM

- Mask-ROM : réalisé lors de la fabrication du circuit.
- PROM : Programmable Read-Only Memory, programmable qu'une fois.
- EPROM : Erasable PROM, effaçable par exposition à des rayons U.V.
- EEPROM : Electrically EPROM, effaçable électriquement.
- NOVRAM : RAM à alimentation indépendante (pile).

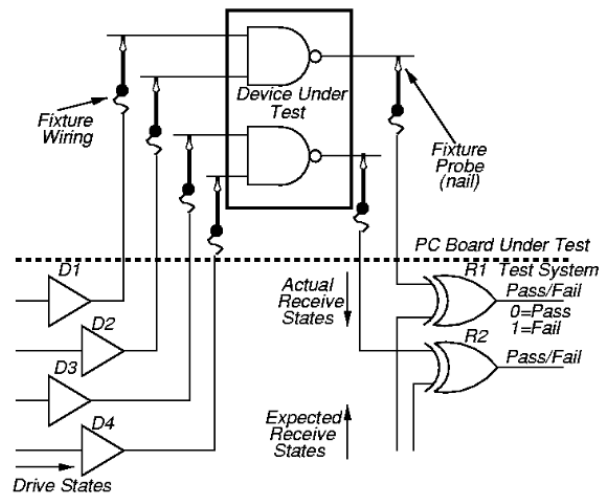
Structure d'un FPGA

- FPGA sous-divisé en cellules.
 - LUT : Look-Up Tables (combinatoire).
 - Bascules (séquentiel).
- Entrées/Sorties.
- Logique de routage.

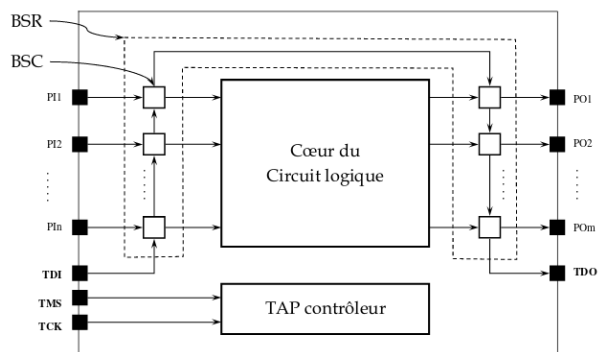


Circuits reprogrammables/Tests

Bed-Of-Nails



Boundary-Scan (JTAG)



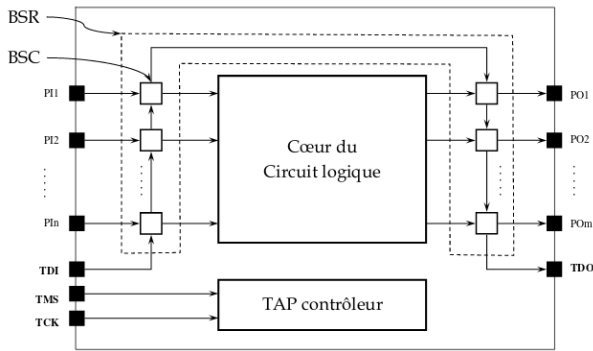
JTAG :

Joint Test Action Group

Norme IEEE1149.1 : "Standard Test Access Port and Boundary-Scan Architecture"

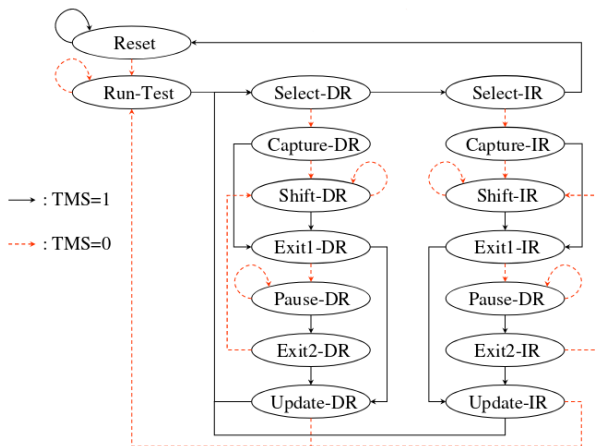
- [++] Permet de tester des SoC.
- [++] Permet le In-System Programming des FPGA.
- [++] Permet de faire du diagnostic de circuit sans instrument supplémentaire.
- [-] Quatre plots obligatoires à ajouter à la carte.
- [-] Routage
- [-] Dégradation des délais.
- [-] Rallongement du TTM (Time-To-Market)

Composition du Boundary Scan



- TCK : Horloge lors du Test.
- TMS : Test-Mode Select
- TDI : Test Data Input
- TDO : Test Data Output

TAP Controller



- Test-Logic-Reset : État de Reset.
- Run-Test-Idle : Mise en attente.
- Select-IR/DR : Choix [Instr/Data/Reset].
- Capture-IR/DR : Le registre à décalage IR/DR est chargé en parallèle sur le front montant de TCK. Les valeurs sont décalées de l'entrée TDI vers la sortie TDO dans le registre à décalage IR/DR sur le front montant de TCK.
- Exit1-IR/DR : État temporaire.
- Pause-IR/DR : Attente de valeurs sur TDI.
- Exit2-IR/DR : État temporaire.
- Update-IR/DR : Les valeurs contenues du registre à décalage IR/DR sont chargées dans le registre parallèle IR/DR sur le front descendant de TCK. Prise en compte des nouvelles données par le circuit.

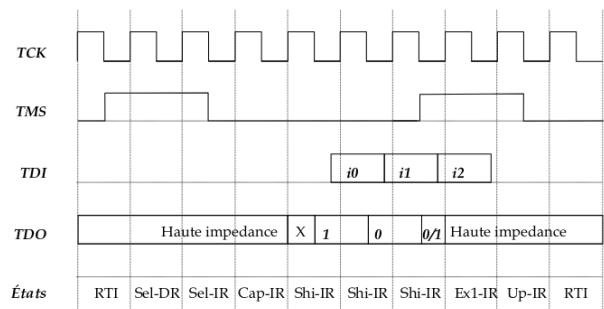
Remarques :

- Si Etat=Shift-IR/DR → driver TDO activé. sinon → Haute Impédance.
- Les données sur TDI sont décalées sur le front montant de TCK, les données sont décalées à la fin de Shift-IR/DR.

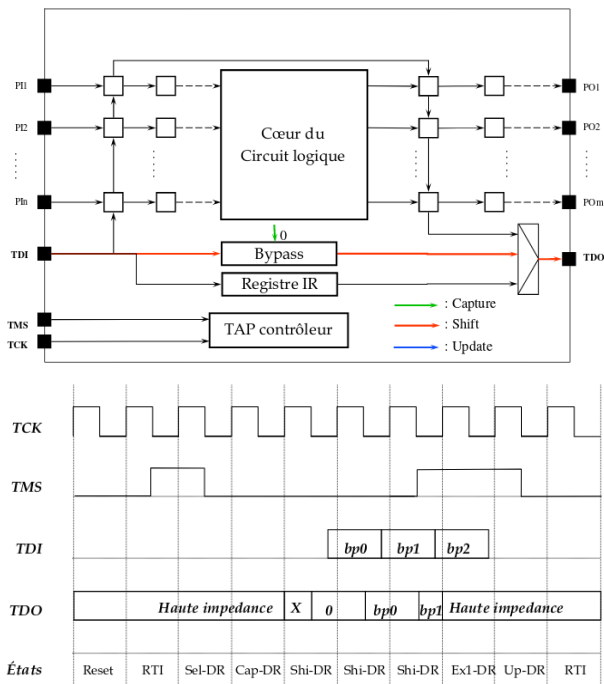
- Les données sur TDO sont décalées sur le front descendant de TCK, la donnée est présente sur TDO $\frac{1}{2}$ cycle avant que la donnée sur TDI soit lue.

Operations de IR

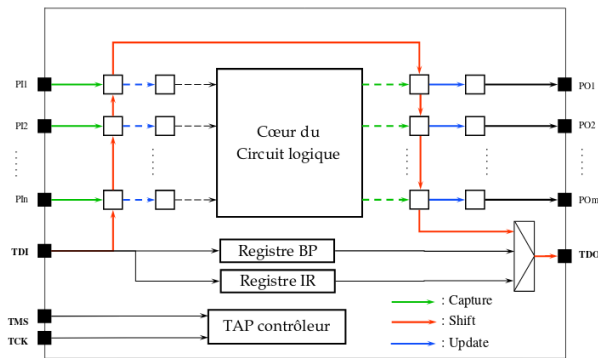
État du TAP Contrôleur	Registre a décalage	Registre Parallèle
Test-Logic-Reset	Indéfini	IDCODE s'il existe, BYPASS sinon
Capture-IR	Chargement de "01" dans LSBs	État précédent
Shift-IR	Décalage des bits d'instruction	État précédent
Exit1-IR	État précédent	État précédent
Pause-IR		
Exit2-IR	État précédent	Chargement des instructions du registre a décalage
Update-IR		



Operations de DR



EXTEST Mode



Definitions :

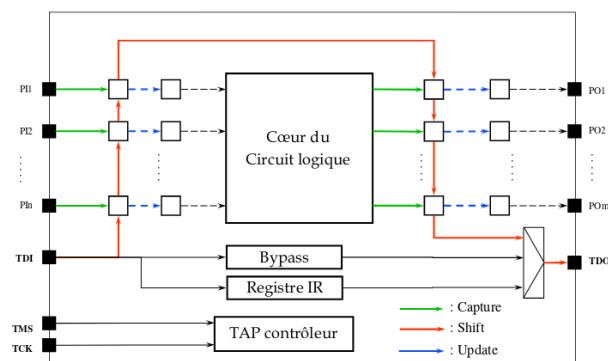
- PTV : Parallel Test Vector
- STV : Sequential Test Vector
- PRV : Parallel Response Vector
- SRV : Sequential Response Vector

Test en Boundary-Scan :

1. Initialiser le TAP en mode Test-Logic-Reset.
2. Charger l'instruction SAMPLE/PRELOAD.
3. "Shift" le 1er vecteur STV.
4. Charger l'instruction EXTEST.
5. "Capture" la réponse du STV précédent dans le BSR.
6. "Shift" la réponse SRV, en chargeant le vecteur suivant STV.
7. "Update" le nouveau vecteur STV.
8. Répéter les étapes 5,6 & 7 jusqu'au dernier vecteur STV.
9. "Capture" la réponse du dernier vecteur STV.
10. "Shift" la réponse SRV du dernier vecteur.
11. Retourner à l'état Test-Logic-Reset.

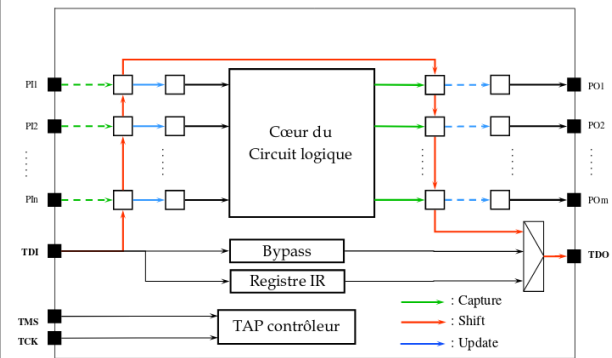
⇒ Tester les interconnexions entre les circuits d'une carte.

SAMPLE/PRELOAD Mode



⇒ Permet de pré-charger/échantillonner un circuit.

INTEST Mode



Test en Boundary-Scan:

- Sélectionner le mode INTEST au lieu de EXTEST.
- Contrairement au mode EXTEST, les vecteurs sont à appliquer aux plots d'entrée du circuit et non aux plots de sortie.
- Les lectures sont à faire sur les plots de sortie du circuit et non aux plots d'entrées.

⇒ Isoler et tester indépendamment chaque circuit logique d'une carte.

RUNBIST Mode

1. Initialiser le TAP à Test-Logic-Reset.
2. Charger l'instruction SAMPLE/PRELOAD.
3. "Shift" le vecteur de sécurité
4. Charger l'instruction RUNBIST, ce qui va sélectionner le registre de sauvegarde de signature.
5. Aller à l'étape Run-Test-Idle.
6. Effectuer le nombre de coups d'horloge nécessaires pour le BIST.
7. "Capture" le registre de signature.
8. "Shift" la signature pour vérification.
9. Revenir à l'état Test-Logic-Reset.

⇒ RUNBIST : RUN Built-In Self-Test

Kevin Mambu
M1 SESI 2017/2018
kev.mambu@protonmail.com

Sources :

Bertrand Granado - Support de cours de Sysprog 2014
Alain Vachoux - Le langage VHDL
Abdelhakim Khouas - Conception en vue du Test : Le Boundary Scan

