



Plan

1- Présentation de l'ENSERB.

- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- PCBUG11.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- μ C/OS II.
- 9- Conclusion.

Présentation de l'ENSERB



➤ **ENSERB** : l'Ecole Nationale Supérieure de Radioélectricité de Bordeaux est créée en 1920 au sein de la Faculté des Sciences de Bordeaux.

➤ **L'ENSERB possède 2 pôles de formation :**

- ➔ *le pôle Electronique.*
- ➔ *le pôle Informatique.*

➤ **Elle s'appuie sur des laboratoires de recherche de niveau international tels que :**

- ➔ *IXL, LaBRI, LAP...*

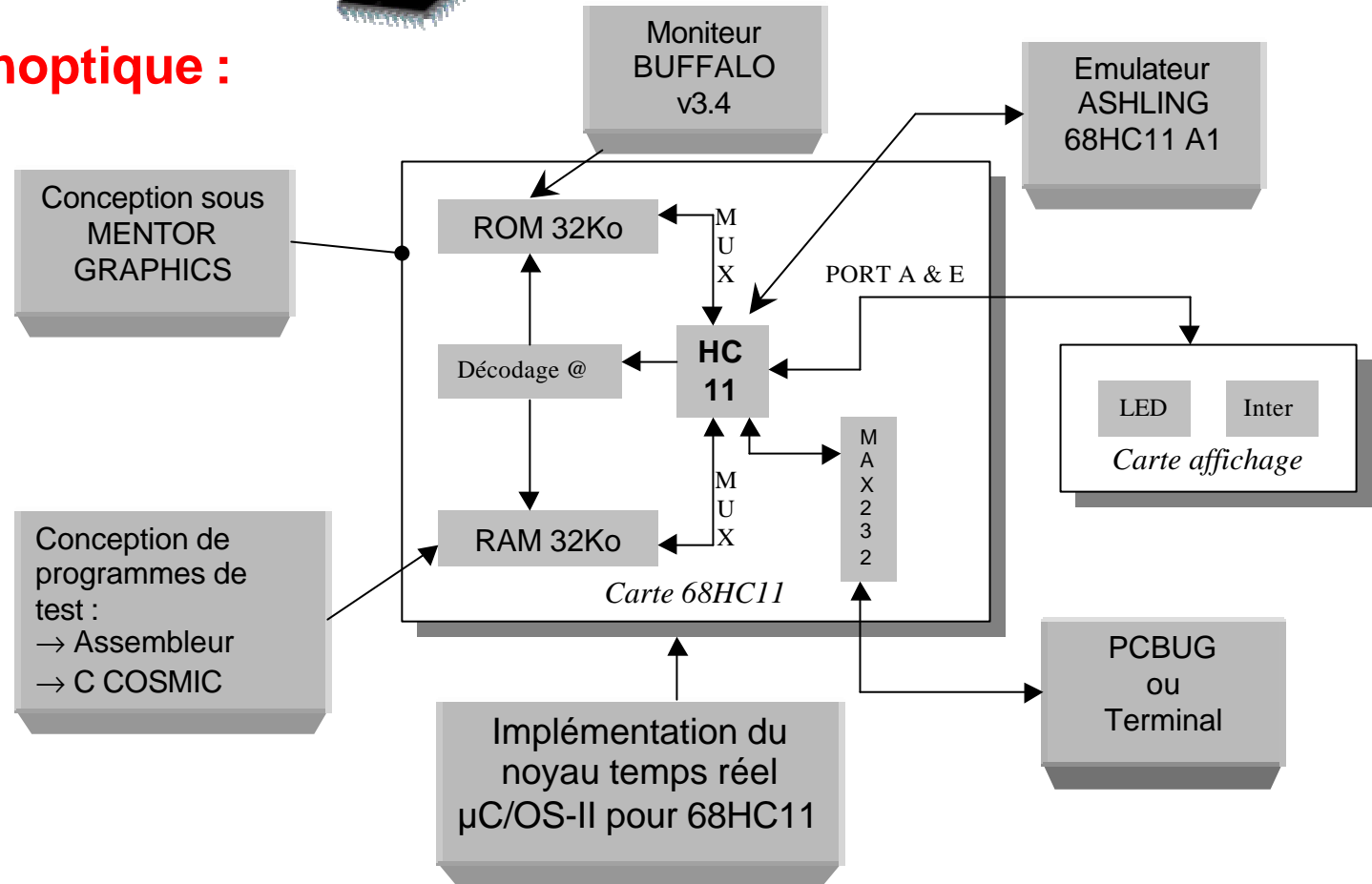




Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm. et C
- 8- μ C/OS II.
- 9- Conclusion.

Synoptique :



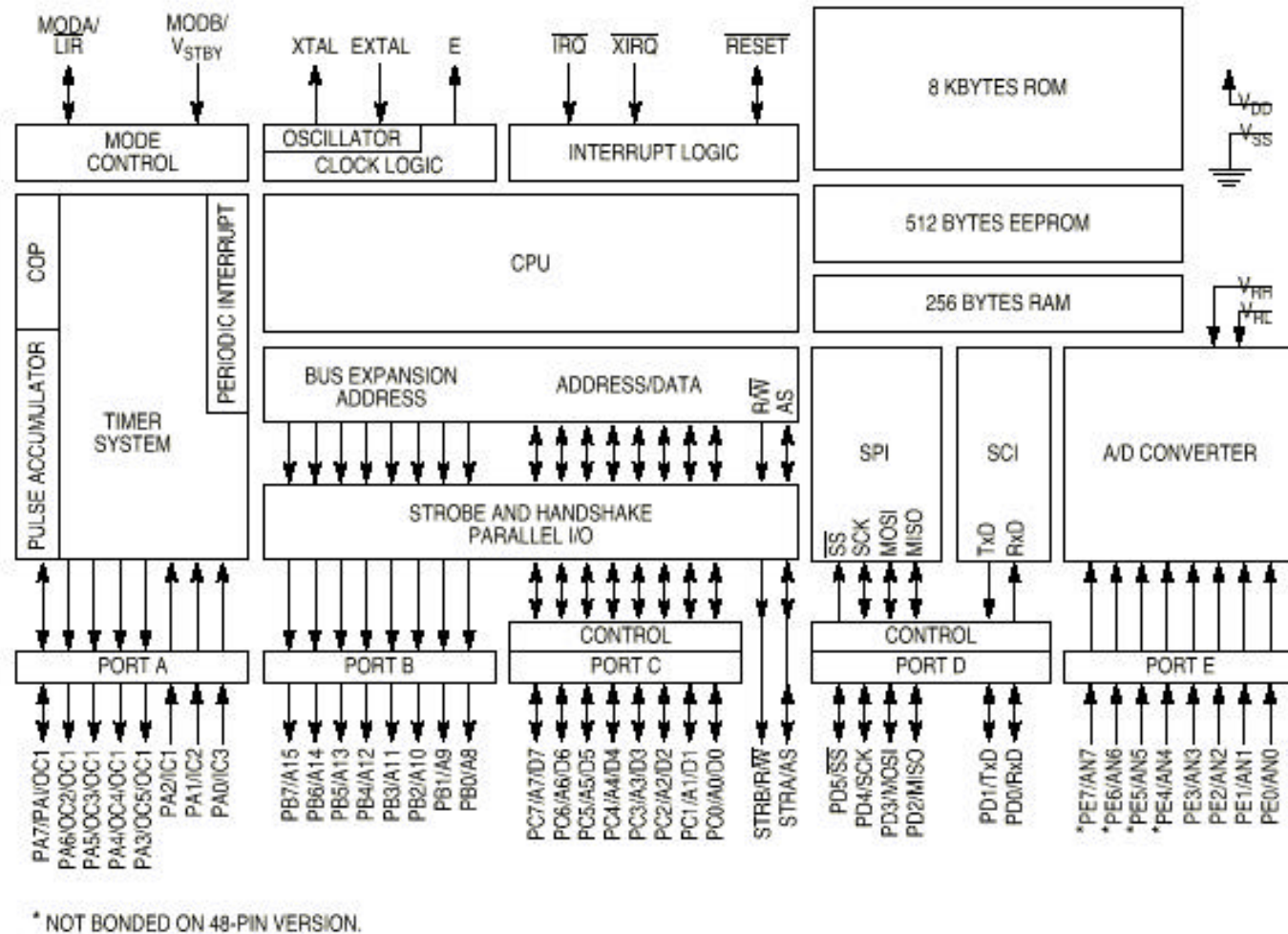
Mots clefs : Emulateur / débuggers / moniteur 68HC11, compilateur C Cosmic, noyau multitâche temps réel μ C/OS.



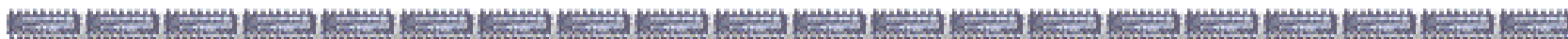


Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- **Le 68HC11A8.**
 - a- **Structure interne.**
 - b- Structure de la carte.
 - c- Décodage d'adresse.
 - d- Cartographie mémoire.
- 4- MENTOR GRAPHICS.
- 5- PCBUG11.
- 6- Emulateur Ashling.
- 7- Programmes asm. et C
- 8- μ C/OS II.
- 9- Conclusion.



A8 BLOCK

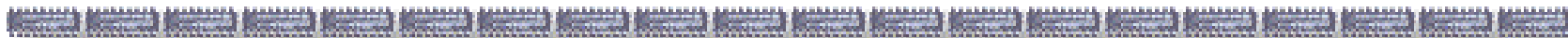
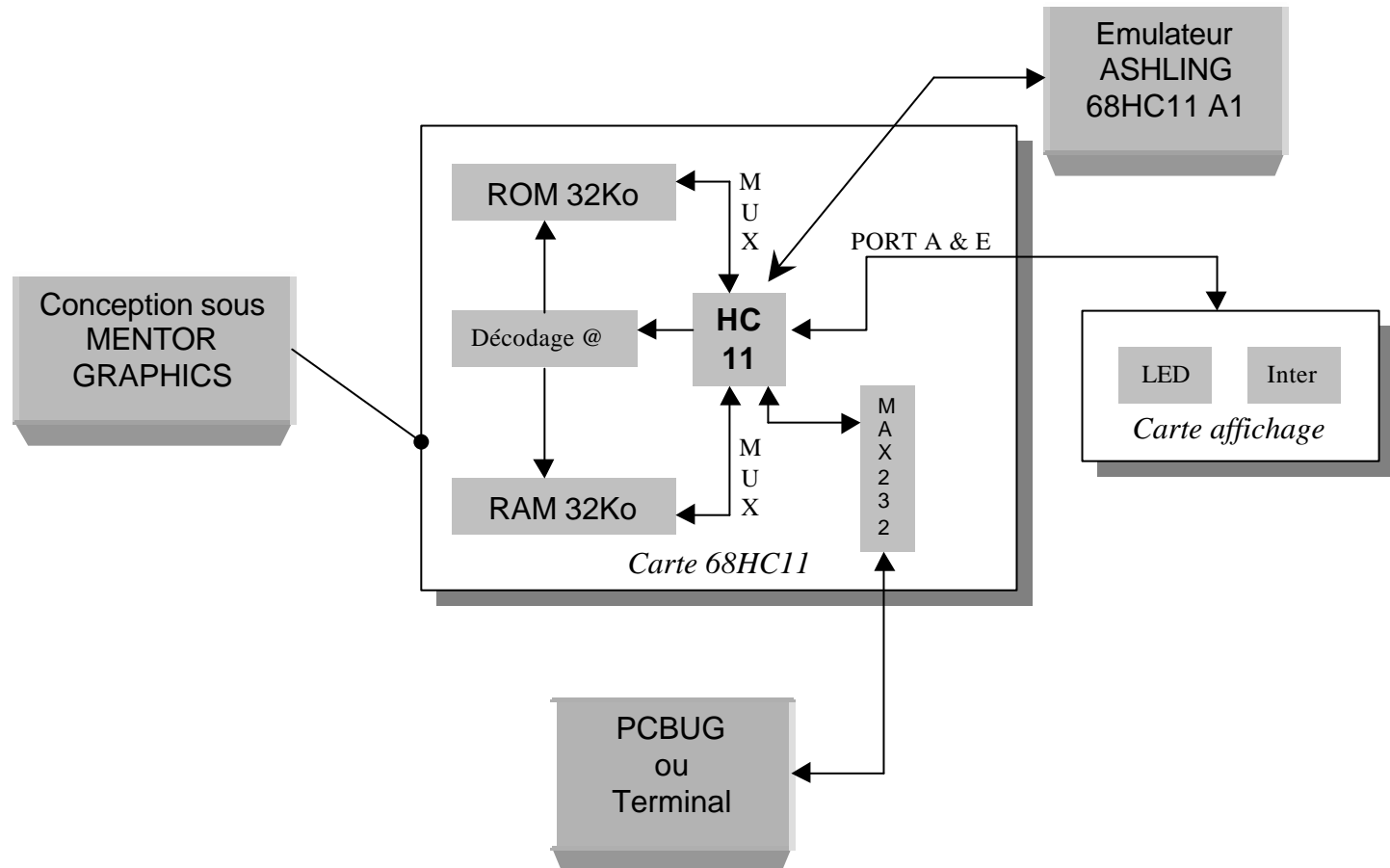




Structure de la carte.

Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- **Le 68HC11A8.**
 - a- Structure interne.
 - b- Structure de la carte.**
 - c- Le décodage d'adresse.
 - d- Cartographie mémoire.
- 4- MENTOR GRAPHICS.
- 5- PCBUG11.
- 6- Emulateur Ashling.
- 7- Programmes asm. et C
- 8- μ C/OS II.
- 9- Conclusion.

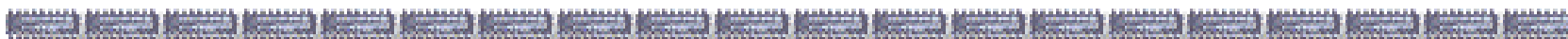
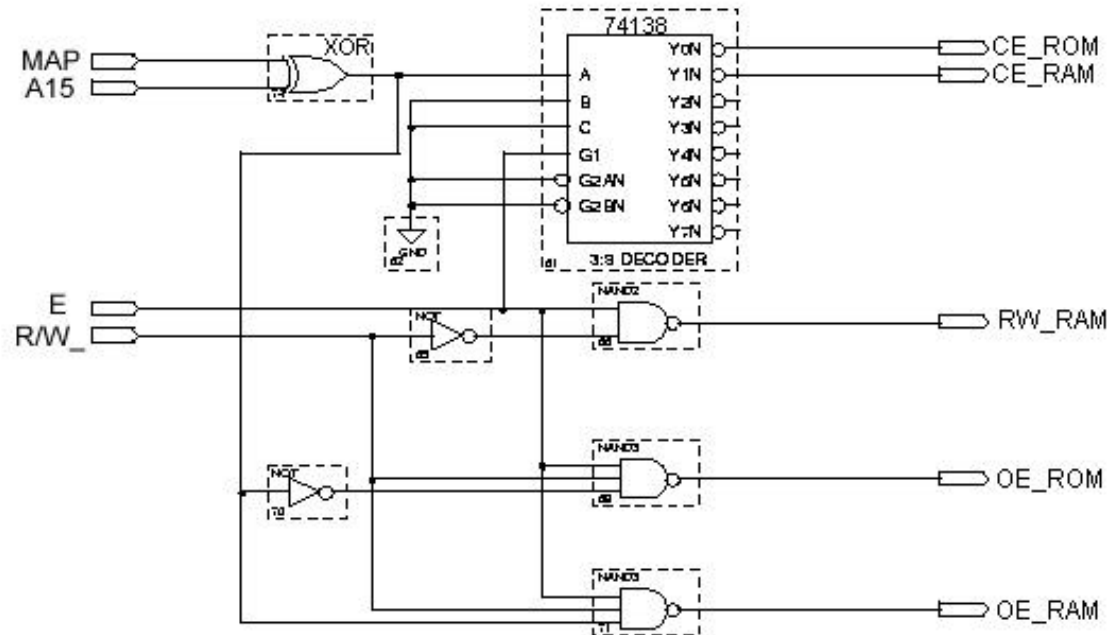




Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.**
 - a- Structure interne.
 - b- Structure de la carte.
 - c- Le décodage d'adresse.**
 - d- Cartographie mémoire.
- 4- MENTOR GRAPHICS.
- 5- PCBUG11.
- 6- Emulateur Ashling.
- 7- Programmes asm. et C
- 8- μ C/OS II.
- 10- Conclusion.

Le décodage d'adresse avec le composant programmable ALTERA.

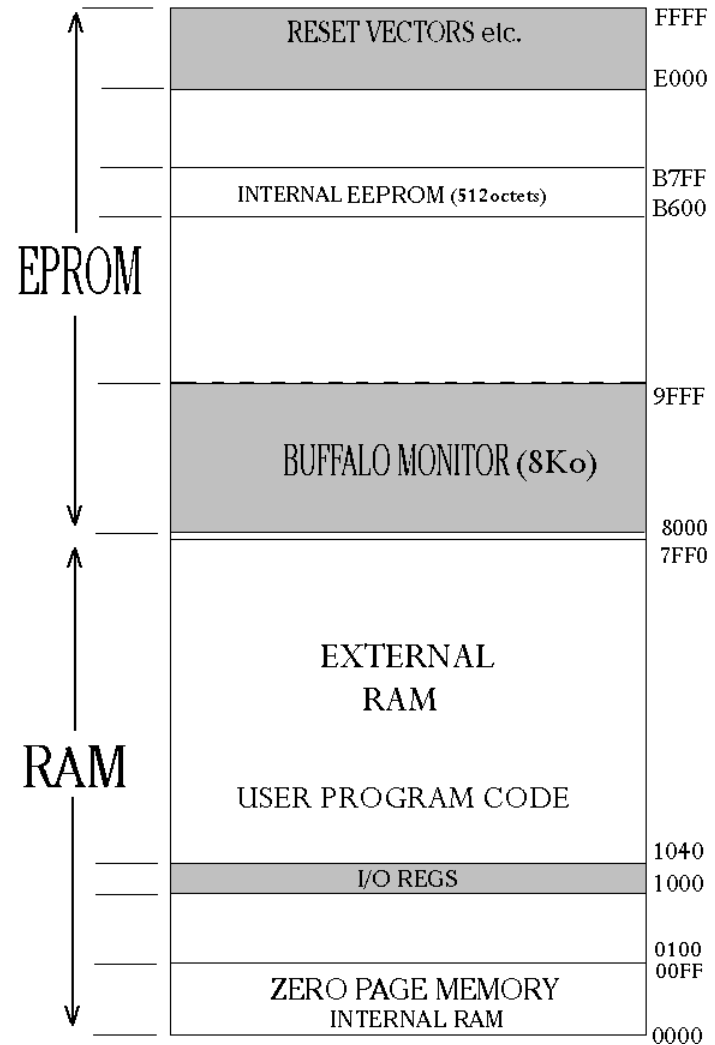




Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.**
 - a- Structure interne.
 - b- Structure de la carte.
 - c- Le décodage d'adresse.
 - d- Cartographie mémoire.**
- 4- MENTOR GRAPHICS.
- 5- Debuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm. et C
- 8- μ C/OS II.
- 9- Conclusion.

La cartographie mémoire de la carte.





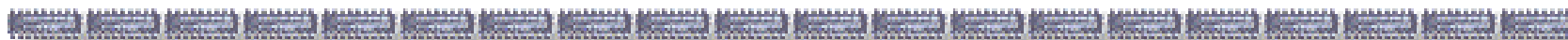
Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- **MENTOR GRAPHICS.**
 - a- Design Architect.
 - b- Librarian.
 - c- Package.
 - d- Layout.
 - e- Fablink.
- 5- PCBUG11.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- μ C/OS II.
- 9- Conclusion.

Les différentes étapes de conception sous MENTOR GRAPHICS



- 1^{ère} étape : la saisie du schéma sous **Design Architect.**
- 2^{ème} étape : l'édition des géométries sous **Librarian.**
- 3^{ème} étape : le lien entre le symbole électrique et la forme géométrique sous **Package.**
- 4^{ème} étape : le placement des composants et le routage de la carte sous **Layout.**
- 5^{ème} étape : la génération du fichier perçage sous **Fablink.**





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
 - a- PCBUG11.
 - b- BUFFALO v3.4.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- µC/OS II.
- 9- Conclusion.

Le logiciel PCBUG11.

```
PCBUG342 Ax
Total bytes loaded: $0246
Total bytes written: $0246
Total bytes loaded: $0270
Total bytes written: $002A
C000 4F      > CLRA
C001 CEF000  > LDX  #$F000
C004 2003    > BRA  $C009
C006 A700    > STAA $00,X
C008 08      > INX
C009 8CF002  > CPX  #$F002
C00C 26F8    > BNE  $C006
C00E 8E00FF  > LDS  #$00FF
C011 BDC0A1  > JSR  >$C0A1
C014 20FE    > BRA  $C014
C016 3C      > PSHX
C017 3C      > PSHX

PC  ACCA  ACCB  X      Y      CCR <SXHINZUC> SP  MCU: 68HC11A8
$0000 $40   $2C   $1000 $0100 $40 %1..... $00EB RTS Level :0N
State: STOPPED
Base : HEX
User RST $XXXX
User SWI $XXXX
User XIRQ $XXXX

restart
loads testio2
asm c000
>>_
```





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.**
 - a- PCBUG11.
 - b- BUFFALO v3.4.**
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- µC/OS II.
- 9- Conclusion.

BUFFALO 3.4 Ex (ENSERB) - Bit User Fast Friendly Aid to Logical Operation

```
68HC11E9 CPU
8K BUFFALO MONITOR PROGRAM EPROM: $8000 TO $9FFF
DEFAULT INTERNAL RAM & REGISTER ALLOCATION
EEPROM: $B600 TO $B7FF
EXTERNAL RAM 32K : $0000 TO $7FFF
>
```

```
ASM [<addr>] Line asm/disasm
[/,=] Same addr,      [^,-] Prev addr,      [+ ,CTLJ] Next addr
[CR] Next opcode,    [CTLA,.] Quit
BF <addr1> <addr2> [<data>] Block fill memory
BR [-][<addr>] Set up bkpt table
BULK Erase EEPROM,          BULKALL Erase EEPROM and CONFIG
CALL [<addr>] Call subroutine
GO [<addr>] Execute code at addr,          PROCEED Continue execution
REMOD [<addr> [<addr>]] Modify EEPROM range
LOAD, VERIFY [T] <host dwnld command> Load or verify S-records
MD [<addr1> [<addr2>]] Memory dump
MM [<addr>] or [<addr>]/ Memory Modify
[/,=] Same addr,      [^,-,CTLH] Prev addr,  [+ ,CTLJ,SPACE] Next addr
<addr>0 Compute offset, [CR] Quit
MOVE <s1> <s2> [<d>] Block move
OFFSET [-]<arg> Offset for download
RM [P,Y,X,A,B,C,S] Register modify
STOPAT <addr> Trace until addr
T [<n>] Trace n instructions
TM Transparent mode (CTLA = exit, CTLB = send brk)
[CTLW] Wait,          [CTLX,DEL] Abort          [CR] Repeat last cmd
>
```

00:02:37 connecté

ANSI

9600 8-N-1

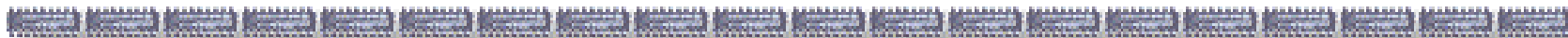
Défil

Maj

Num

Capturer

Imprimer l'écho

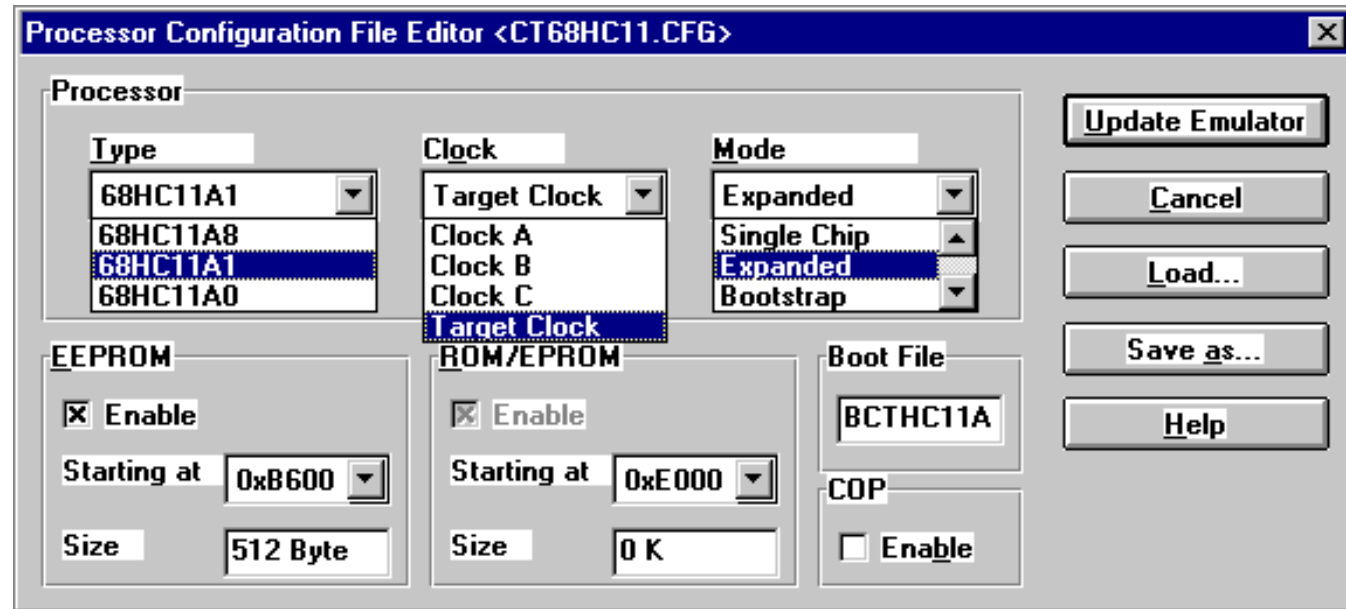




Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
- 6- Emulateur Ashling.**
- 7- Programmes asm et C.
- 8- μ C/OS II.
- 9- Conclusion.

L'Emulateur CT68HC11



➤ 64 Ko de RAM interne.

➤ Choix du mapping (Emulateur/Cible).

➤ Interface conviviale avec le logiciel Pathfinder.





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
- 6- Emulateur Ashling.

7- Programmes asm et C.

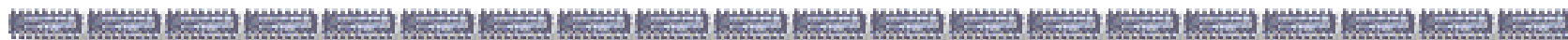
- 8- μ C/OS II.
- 9- Conclusion.

Programme en assembleur.

- Edition du programme avec Notepad :
➔ testio.asm ou .asc
- Compilation du fichier source avec As11.exe :
.asm ➔ .S19
- Récupération de la table des symboles avec srec2ash.exe :
.S19 ➔ .obj
- Débuggage sous Pathfinder.

Programme en C.

- Edition du programme avec Notepad :
➔ testio.c
- Edition de la bibliothèque de fonctions :
➔ hc11lib.c
- Compilation des fichiers sources avec cx6811.exe :
.C ➔ .O (objet)
- Edition des liens avec clnk.exe :
Configuration du fichier .lcf
- Débuggage sous Pathfinder.

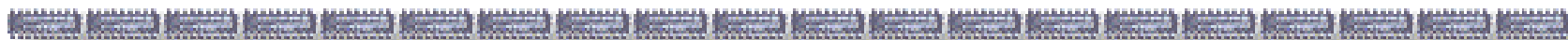
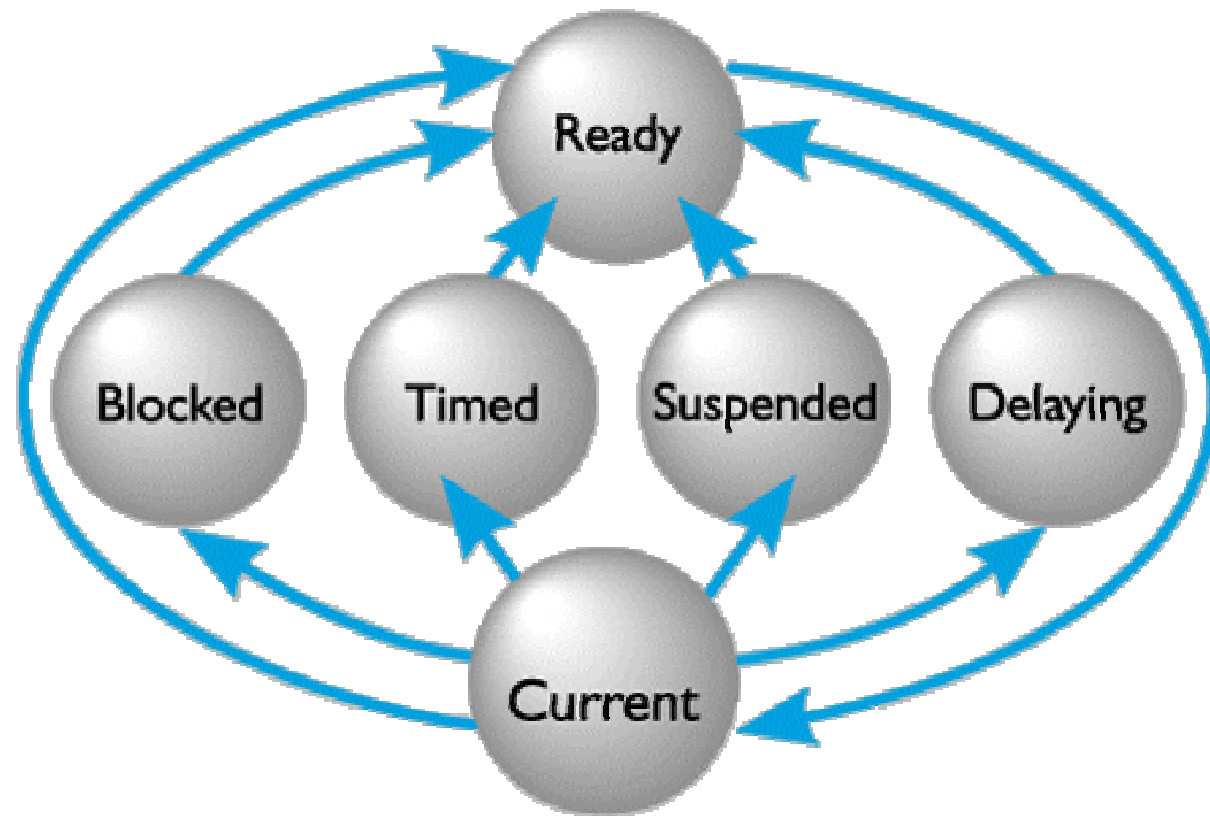




Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- **µC/OS II.**
 - a- **Qu'est-ce qu'un système multitâche temps réel?**
 - b- Présentation de µC/OS II.
 - c- Mise en place du noyau µC/OS II.
- 9- Conclusion.

L'organisation des tâches.





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Débuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.

- 8- μ C/OS II.**
 - a- Qu'est-ce qu'un système multitâche temps réel ?
 - b- Présentation de μ C/OS II.**
 - c- Mise en place du noyau μ C/OS II.

- 9- Conclusion.

Le noyau multitâche μ C/OS II.

Caractéristiques essentielles :

- Création et gestion de 63 tâches maximum.

- Création et gestion de sémaphores.
 - Fonction d'attente de tâche.

- Changement de priorité des tâches.
 - Effacement de tâche.

- Envoi de messages depuis une routine d'interruption (ISR) ou d'une tâche vers une autre tâche.





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Debuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.

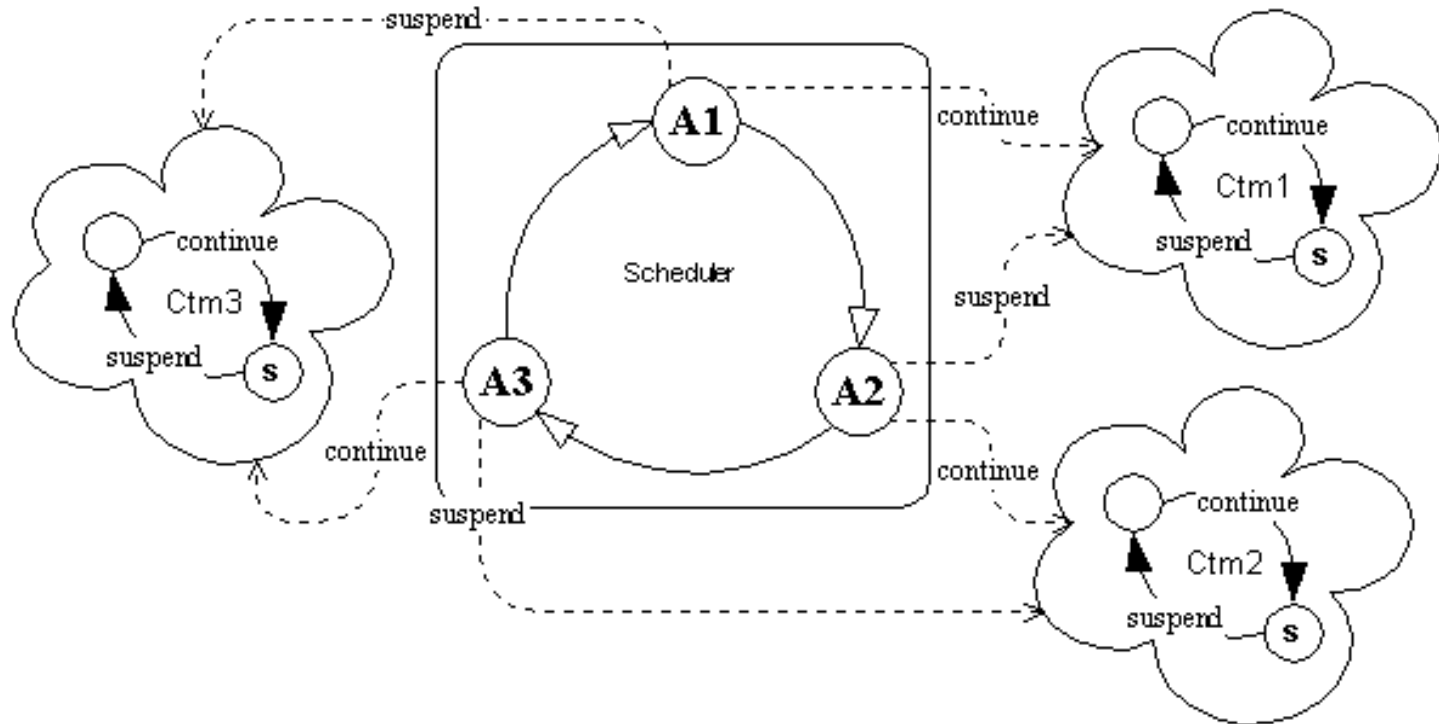
8- μ C/OS II.

- a- Qu'est-ce qu'un système multitâche temps réel ?
- b- Présentation de μ C/OS II.
- c- **Mise en place du noyau μ C/OS II.**

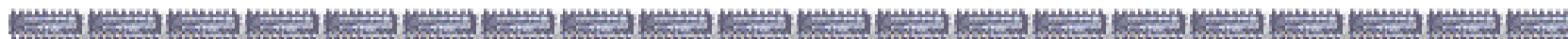
- 9- Conclusion.

Mise en place de μ C/OS II: Exemples de 3 tâches.

L'ordonnancement des 3 tâches :



tâche #1 : recopie du PORT E sur le PORT A,
tâche #2 : chenillard PORT A, tâche #3 : écho sur la liaison série.





Plan

- 1- Présentation de l'ENSERB.
- 2- Présentation du sujet.
- 3- Le 68HC11A8.
- 4- MENTOR GRAPHICS.
- 5- Debuggers.
- 6- Emulateur Ashling.
- 7- Programmes asm et C.
- 8- μ C/OS II.

9- Conclusion.

Conclusion



- Utilisation de nombreux logiciels : MENTOR GRAPHICS, Emulateur ASHLING, PCBUG11, ALTERA MAX + Plus, programmeur ART...
- Approfondissement de la connaissance des 68HC11A1 et F1.
- Programmation en langage assembleur HC11 et langage C : compilation, édition de liens.
- Notion de noyau multitâche temps réel :
Fonctionnement du noyau μ C/OS II.
Conception de tâches.
Notion de sémaphores.



