

Manuel Utilisateur

Proteus VSM
Circuits Animés Interactifs

(Proteus VSM - Août 2000)
© Multipower

SOMMAIRE

SOMMAIRE	2
TUTORIELS	4
TUTORIEL DE LA SIMULATION INTERACTIVE.....	4
<i>Introduction.....</i>	4
DESSIN DU CIRCUIT	4
<i>Placement des composants</i>	4
<i>Déplacement et orientation</i>	5
<i>Zoom et panoramique</i>	5
<i>Connexion des composants</i>	5
ÉCRIRE LE PROGRAMME	6
<i>Listing du source.....</i>	6
<i>Attacher le fichier source</i>	7
MISE AU POINT DU PROGRAMME	7
<i>Simulation du circuit.....</i>	7
<i>Mode de 'mise au point' (debug).....</i>	8
<i>Positionner un point d'arrêt.....</i>	8
<i>Trouver l'erreur.....</i>	9
SIMULATION INTERACTIVE	10
GÉNÉRALITÉS	10
<i>Le panneau de contrôle de l'animation.....</i>	10
<i>Les indicateurs et les actionneurs</i>	10
<i>Valider une simulation interactive</i>	10
LES EFFETS D'ANIMATION.....	11
<i>Vue d'ensemble.....</i>	11
<i>États logiques des pattes</i>	11
<i>Montrer les tensions des fils en couleur</i>	11
<i>Montrer le sens du courant avec des flèches</i>	12
LE CONTRÔLE DES INCRÉMENTS TEMPORELS DE LA SIMULATION	12
<i>Vue d'ensemble.....</i>	12
<i>Trames par seconde.....</i>	12
<i>Temps par trame.....</i>	12
<i>Temps d'un intervalle</i>	12
ASTUCES.....	12
<i>Échelle de temps.....</i>	12
<i>Échelle de tension</i>	13
<i>Mise à la masse (earthing)</i>	13
<i>Points à haute impédance</i>	13
INTRUMENTS VIRTUELS	14
VOLTMÈTRES ET AMPÈREMÈTRES.....	14
OSCILLOSCOPE.....	14
<i>Généralités</i>	14
<i>Utilisation de l'oscilloscope</i>	15
ANALYSEUR LOGIQUE	15
<i>Généralités</i>	15
<i>Utilisation de l'analyseur logique</i>	16
<i>Panoramique et zoom</i>	17
<i>Mesures</i>	17
ÉLÉMENTS D'INTERFACE UTILISATEUR.....	17
<i>Boutons rotatifs</i>	17

TRAVAILLER AVEC DES MICROPROCESSEURS	18
INTRODUCTION.....	18
CONTRÔLE DU CODE SOURCE.....	18
<i>Vue d'ensemble.....</i>	<i>18</i>
<i>Associer le code source à un projet.....</i>	<i>18</i>
<i>Travailler avec votre code source.....</i>	<i>19</i>
<i>Installation d'outil de génération tiers.....</i>	<i>19</i>
<i>Utilisation d'un programme MAKE.....</i>	<i>20</i>
<i>Utilisation de l'éditeur de source tiers.....</i>	<i>20</i>
FENÊTRES SUPPLÉMENTAIRES.....	20
MISE AU POINT AU NIVEAU DU SOURCE.....	21
<i>Vue d'ensemble.....</i>	<i>21</i>
<i>Fenêtres de code source.....</i>	<i>21</i>
<i>Pas à pas.....</i>	<i>21</i>
<i>Utilisation des points d'arrêts.....</i>	<i>22</i>
LA FENÊTRE WATCH.....	22
OBJETS DE DÉCLENCHEMENT DE POINT D'ARRÊT	23
<i>Vue d'ensemble.....</i>	<i>23</i>
<i>Déclenchement d'un point d'arrêt de tension – RTVBREAK.....</i>	<i>23</i>
<i>Déclenchement d'un point d'arrêt de courant – RTIBREAK.....</i>	<i>23</i>
<i>Déclenchement d'un point d'arrêt numérique – RTDBREAK.....</i>	<i>23</i>

TUTORIELS

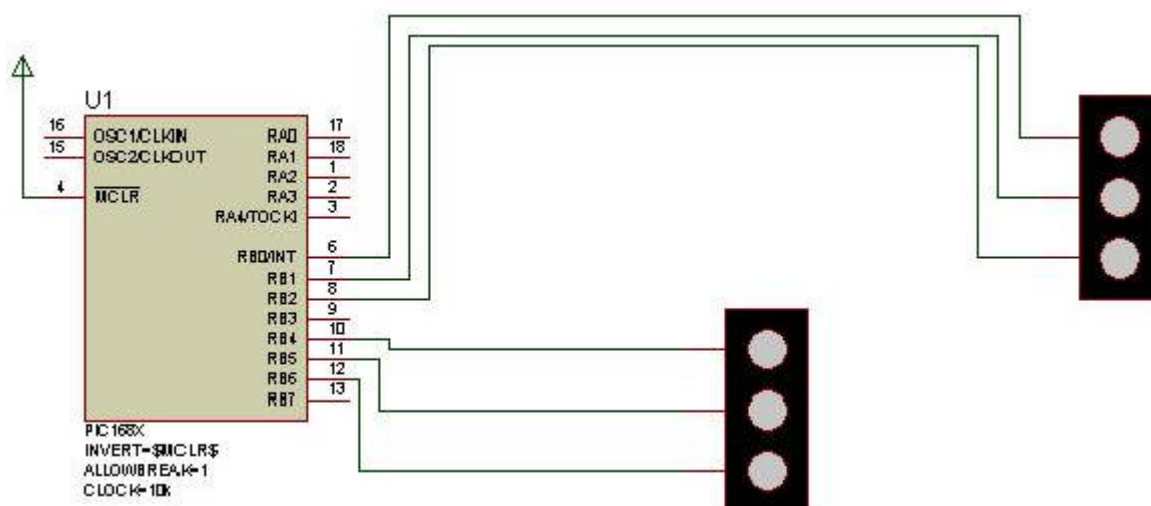
TUTORIEL DE LA SIMULATION INTERACTIVE

Introduction

L'objet de ce tutoriel est de vous montrer, à travers la création d'un schéma simple, comment conduire une simulation interactive avec Proteus VSM. Pendant que nous nous concentrerons sur l'utilisation des composants actifs et des possibilités de mise au point de l'éditeur ISIS, nous regarderons également les bases de la saisie de schéma – veuillez vous reporter au manuel de ISIS pour plus de détails.

Le circuit que nous utiliserons pour la simulation est composé de deux feux de contrôle du trafic connectés à un microcontrôleur PIC16F84 comme montré ci-dessous.

Nous expliquerons la construction du schéma à partir d'un projet vide, cependant une version terminée de ce schéma est présente dans le sous-répertoire 'Samples\Traffic Lights' du dossier d'installation du logiciel, sous le nom 'trafic.dsn', ainsi que le code source utilisé pour le PIC. Les utilisateurs, habitués avec la manipulation de ISIS, peuvent se rendre directement à la section qui présente le programme du microcontrôleur.



Si vous n'êtes pas un habitué de ISIS, nous allons vous présenter les bases de la manipulation de cet éditeur afin de vous familiariser avec l'environnement, avant de passer au programme.

DESSIN DU CIRCUIT

Placement des composants

Nous allons commencer par placer deux feux tricolores et un PIC16F84 sur une nouvelle feuille de notre projet. Commencez par créer un nouveau projet (design), sélectionnez l'icône de 'Mode Principal' (*Main Mode*) et l'icône 'Composant' (*Component*) (tous les icônes affichent une info bulle et une aide contextuelle dans la partie inférieure de l'écran), puis cliquez gauche sur le bouton 'P' placé au-dessus du sélecteur d'objets. La fenêtre de sélection des composants en bibliothèque se superpose à la fenêtre d'édition.

Les feux tricolores sont présents dans la bibliothèque ACTIVE et le microcontrôleur PIC dans la bibliothèque MICRO. Pour transférer un composant vers le projet, positionnez la souris sur le nom

de l'objet désiré et double cliquez sur lui. Une sélection réussie fera apparaître le nom du composant dans le sélecteur d'objets à droite.

Lorsque vous avez sélectionné à la fois les objets TRAFFIC LIGHTS et PIC16F84, fermez le sélecteur de composants en bibliothèque et cliquez sur le PIC16F84 placé dans le sélecteur d'objet (une prévisualisation du composant apparaît dans la vue globale, dans le coin supérieur droit de l'écran). A présent, cliquez gauche dans la fenêtre d'édition pour placer le composant du schéma, et répétez ces opérations pour les deux feux tricolores.

Déplacement et orientation

A présent, nous avons placé les éléments de notre schéma mais il est fort possible que ceux-ci ne soient pas positionnés correctement. Pour déplacer un composant, positionnez la souris sur lui et cliquez droit (le composant doit être en surbrillance), puis enfoncez le bouton gauche et glissez le composant à la position désirée (vous devriez voir le contour du composant suivre le déplacement du pointeur de la souris). Lorsque la position désirée est atteinte, relâchez le bouton gauche pour terminer le déplacement. Notez qu'à cet instant le composant est toujours en surbrillance – cliquez droit dans une zone vierge de la fenêtre d'édition pour restituer l'état normal.

Pour orienter le composant, cliquez droit sur lui comme précédemment, puis cliquez sur l'icône d'orientation. La rotation s'effectue par pas de 90° - répétez autant que nécessaire. A nouveau, il est préférable de cliquer droit dans une zone vierge du schéma pour dévalider la sélection en cours.

Ce tutoriel ignorera les fonctions graphiques 2D impliquées dans la jonction des connexions pour nous concentrer sur la simulation du circuit.

Zoom et panoramique

Afin de connecter les éléments du schéma, il est utile de savoir zoomer sur une zone spécifique. La touche F6 zoom autour de la position de la souris ou, autre méthode, maintenez la touche SHIFT enfoncée et étirez un rectangle (bouton gauche) autour de la zone à zoomer. Un zoom arrière est obtenu avec la touche F7. La touche F8 effectue un zoom qui permet de visualiser la totalité du projet. Ces raccourcis claviers correspondent aux commandes du menu 'Affichage' (*View*).

ISIS possède une caractéristique puissante appelée 'Accrochage Temps Réel' (*Real Time Snap*). Lorsque le pointeur souris est positionné à proximité des terminaisons des pattes ou des fils, la position du curseur s'accroche sur ces objets. Ceci facilite l'édition et la manipulation du schéma. Cette fonctionnalité, active par défaut, est présente dans le menu 'Outils' (*Tools*).

Connexion des composants

La façon la plus simple de connecter un circuit est de valider l'option de 'Routeur de fils automatique' (*Wire Auto Router*) du menu 'Outils' (*Tools*).

Faites un zoom sur le PIC afin de bien visualiser les pattes du composant et placez le pointeur souris sur la patte 6 (RB0/INT). Vous devriez voir apparaître un petit curseur 'x' à l'extrémité de la souris. Ceci indique que la position de la souris permet de connecter un fil sur cette patte. Cliquez gauche pour débuter votre tracé et déplacez la souris vers le feu rouge d'un des feux tricolores. Lorsque vous visualisez un 'x' sur la patte d'arrivée, cliquez gauche pour terminer la connexion. Répétez le procédé afin d'obtenir le schéma souhaité.

Plusieurs points sont à noter:

- Vous pouvez connecter dans n'importe quel mode de fonctionnement – ISIS est suffisamment intelligent pour déterminer ce que vous faites.

- Lorsque l'autorouteur de fil est validé, il s'arrange pour trouver un chemin qui tient compte des obstacles. Ceci signifie que, de manière générale, il vous suffit de cliquer gauche sur les points de départ et d'arrivée de votre connexion, et ISIS se charge de trouver le bon chemin.
- Lorsque vous atteignez le bord de la zone d'édition, ISIS effectuera un panoramique automatiquement lors de la définition des extrémités de la connexion. Ceci signifie que vous pouvez zoomer au niveau qui vous convient, puis définir le point de départ et d'arrivée. Rien ne vous empêche d'utiliser les touches F6 et F7 pour agrandir ou réduire l'échelle de visualisation.

Pour terminer, nous devons connecter la patte 4 à un terminal 'd'alimentation' (*power*). Sélectionnez l'icône de 'mode gadgets' (*gadgets mode*) et l'icône 'terminal', puis validez le mot POWER dans le sélecteur d'objets. A présent, cliquez gauche dans la fenêtre d'édition. Rectifiez l'orientation si nécessaire et connectez la patte 4.

Votre circuit est terminé et, à ce stade, vous pouvez charger la version complète du circuit – ceci afin d'éviter toute confusion pour la suite des explications.

ÉCRIRE LE PROGRAMME

Listing du source

Pour les besoins du tutoriel, nous avons préparé le programme suivant qui utilisera le PIC pour contrôler les feux de gestion du trafic. Le programme est présent dans le fichier TL.ASM du sous-répertoire 'Samples\Traffic Lights' du répertoire d'installation.

```

LIST      p=16F84 ; PIC16F844 est le processeur cible

#include "P16F84.INC" ; inclure fichier entête

CBLOCK 0x10 ; zone tampon
    state
    11,12
ENDC

org      0 ; Vecteur de début
goto    setports ; Aller au code de lancement

org      4 ; Vecteur d'interruption
halt    goto    halt ; Stopper en cas de boucle sans fin et ne rien faire

setports clrw ; Zéro dans W.
movwf   PORTA ; S'assurer que PORTA vaut zéro avant validation
movwf   PORTB ; S'assurer que PORTB vaut zéro avant validation
bsf     STATUS,RP0 ; Sélectionner la Bank 1
clr     ; Masquer tous les bits comme sorties
movwf   TRISB ; Valider registre TRISB
bcf     STATUS,RP0 ; Resélectionner Bank 0.

initialise clrw ; Etat initial.
movwf   state ; Valider.

loop     call    getmask ; Convertir état.
movwf   PORTB ; Ecrire vers port.
incf   state,W ; Incrémenter état dans W.
andlw  0x03 ;
movwf   state ; Mettre en mémoire
call   wait ; Attendre :- )
goto   loop ; et boucler :- )

```

```

; Fonction qui renvoie un masque de bits pour l'état courant du port de
; sortie.
; Le nibble supérieur contient les bits pour un groupe de feux et celui
; inférieur
; pour l'autre groupe. Bit 1 pour rouge, 2 pour orange, 3 pour vert.
; Le bit 4 n'est pas utilisé.
getmask  movf    state,W    ; Obtenir état dans W.
         addwf   PCL,F      ; Ajoute offset dans W de PCL pour calc. goto.
         retlw   0x41      ; state==0 est Vert et Rouge
         retlw   0x23      ; state==1 est Orange et Rouge
         retlw   0x14      ; state==3 est Rouge et Vert
         retlw   0x32      ; state==4 est Rouge/Orange et Orange

; Fonction qui utilise deux boucles pour définir un retard.
wait     movlw   5
         movwf  l1

w1       call    wait2
         decfsz l1
         goto   w1

         return

wait2    clrf    l2
w2       decfsz l2
         goto   w2
         return
END

```

En fait, ce programme contient une erreur délibérée, afin que plus tard

Attacher le fichier source

L'étape suivante est d'associer le programme à notre projet afin de simuler son fonctionnement. Nous faisons cela grâce aux commandes du menu 'Source'. Sélectionnez la commande 'Ajouter/Supprimer fichiers sources' (*Add/Remove Source Files*). Cliquez sur le bouton 'Nouveau' (New), et parcourez les répertoires jusqu'à sélectionner '\Proteus\Samples\Traffic Lights' et le fichier TL.ASM. Cliquez sur 'Ouvrir' et là le fichier sera inséré dans la liste déroulante des fichiers sources.

À présent nous devons choisir l'outil de génération de code pour ce fichier. L'outil MPASM convient parfaitement et doit être choisi dans la liste déroulante (si vous envisagez d'utiliser un autre assembleur ou compilateur, vous devrez l'enregistrer par la commande de définition d'outil de génération de code).

À ce stade, nous avons attaché le fichier source et spécifié l'outil de génération de code à utiliser.

MISE AU POINT DU PROGRAMME

Simulation du circuit

Afin de simuler le circuit, positionnez votre souris sur le bouton *Play* du panneau d'animation, placé dans la partie inférieure droite de l'écran, et cliquez gauche. La barre de message doit indiquer le temps écoulé depuis le lancement de l'animation. Vous noterez également qu'un des feux est vert alors que l'autre est rouge et que l'état logique des pattes est visualisé sur le schéma. Vous vous apercevrez cependant que les feux ne changent pas d'état. Ceci provient d'une erreur que nous

avons volontairement introduite dans le programme. A ce stade, il convient de mettre au point le programme pour isoler le problème.

Mode de 'mise au point' (debug)

Afin de passer en mode de mise au point, nous allons arrêter la simulation. Lorsque cette opération est effectuée, nous allons lancer le mode 'debug' en utilisant la combinaison CTRL+F12. Deux fenêtres apparaissent – une contient les valeurs courantes des registres et l'autre visualise le source du programme. L'une ou l'autre de ces fenêtres peut être activée par l'intermédiaire du menu 'Debug', de même qu'un bon nombre d'autres fenêtres d'informations. Nous désirons également activer une fenêtre 'd'analyse de variable' (*Watch Window*) afin de suivre les changements de l'état des variables.

Concentrons-nous sur la fenêtre 'Source' et notons la flèche rouge à gauche. Ceci, plus la ligne mise en surbrillance, indique la position courante du compteur programme. Pour placer un point d'arrêt ici appuyez sur la touche ENTREE (le point d'arrêt sera toujours positionné sur la ligne en surbrillance). Si vous désirez supprimer le point d'arrêt, vous devrez appuyer sur la touche ENTREE à nouveau, mais nous conserverons le point d'arrêt pour la suite des explications.

Positionner un point d'arrêt

Un coup d'œil au programme montre un bouclage continu. Il serait intéressant de placer un point d'arrêt au début de la boucle. Pour ce faire, nous mettons la ligne correspondante en surbrillance (adresse 000E) avec la souris, puis appuyons sur F9. Puis lancez l'exécution grâce à la touche F12. Vous devriez voir un message dans la barre inférieure de l'écran qui indique qu'un point d'arrêt numérique a été atteint, avec la valeur correspondante du compteur programme (PC). Celle-ci doit être la même que le point que vous avez placé.

Une liste des touches de mise au point est donnée dans le menu 'Debug', mais nous utiliserons principalement F11 pour avancer l'exécution en pas à pas. Utilisez F11 et vous constaterez que la flèche rouge s'est déplacée sur l'instruction suivante. Nous venons d'exécuter l'instruction '*clrw*', avant d'arrêter. Vous pouvez vérifier cela en regardant le registre W dans la fenêtre 'Registres' et vérifier qu'il vient d'être remis à zéro.

Ce que nous devons faire à présent est de déterminer ce que doit faire l'instruction suivante et de tester si l'exécution correspond. Par exemple, l'instruction suivante doit déplacer le contenu du registre 'W' vers le PORT A, c'est à dire que le port A doit être effacé. L'exécution de cette instruction et le contrôle de la fenêtre 'Registres' vérifie que c'est bien le cas. Poursuivez afin de constater que les deux ports ont été effacés (comme indiqué par le registre TRISB) et que la variable d'état a été mise à 0.

Comme nous arrivons sur un appel de fonction nous pouvons, soit exécuter son contenu sans entrer dans le détail (touche F10), soit exécuter chaque instruction de la fonction appelée (touche F11). Appuyez sur la touche F11 et vous serez positionné sur la fonction '*getmask*'. Un pas à pas montre que l'opération '*move*' s'effectue correctement et que nous arrivons au bon emplacement pour ajouter un offset de zéro à notre table de transcodage. Le masque est celui attendu lorsque nous revenons dans le programme principal. Au pas suivant; lorsque ce masque est écrit sur le port, la répercussion visuelle est correcte. Encore un pas pour incrémenter W de 1, confirmé dans la fenêtre 'Registre'.

Le pas à pas nous amène à l'instruction qui remet à zéro l'état lorsque sa valeur est supérieure à 3. Ceci, comme nous l'observons dans la fenêtre *Watch*, n'est pas réalisé comme désiré. L'état doit être clairement incrémenté de 1 de façon à valider correctement le masque pour l'instruction suivante de la boucle.

Trouver l'erreur

Une investigation plus poussée révèle que le problème est causé par un ET logique avec 4 au lieu de 3. Les états que nous désirons sont 0, 1, 2, 3 et un ET logique de chacune de ces valeurs avec le nombre 4 donne 0. Ceci explique pourquoi l'état ne change pas pendant la simulation. La solution est de modifier l'instruction AND avec 3 au lieu de 4.

Ce court exemple des techniques de mise au point, disponibles dans Proteus VSM, illustre les bases des multiples fonctionnalités disponibles.

SIMULATION INTERACTIVE

GENERALITES

Le panneau de contrôle de l'animation



De manière à contrôler le comportement des circuits animés qui ont été implémentés, nous avons ajouté un panneau de contrôle de type 'magnétoscope'. Cet ensemble de commandes est situé dans la partie inférieure droite de l'écran. S'il n'est pas visible, vous devez sélectionner l'option 'Circuit Animation' du menu 'Graph'. Il contient quatre boutons de contrôle de l'activité des circuits:

- Le bouton PLAY (1) est utilisé pour lancer le simulateur PROSPICE qui, à son tour active la simulation temps réel.
- Le bouton STEP (2) contrôle l'évolution pas à pas de l'animation à une vitesse définie. Si le bouton est appuyé puis relâché alors l'animation progresse d'une unité de temps; si le bouton est maintenu appuyé alors l'animation progresse de manière continue. L'incrément unitaire du temps de progression est ajusté dans la boîte de dialogue 'Animated Circuit Configuration'. Ce temps est utilisé pour contrôler finement le circuit et observer l'influence des vitesses faibles. Ceci peut être particulièrement utile dans un contexte éducatif où les étudiants peuvent visualiser les effets en pas à pas.
- Le bouton PAUSE (3) suspend l'animation qui pourra être relancée par un clic sur ce bouton, ou pourra avancer en pas à pas par un clic sur le bouton STEP. Le simulateur est également placé dans le mode PAUSE lorsqu'il rencontre un point d'arrêt.
La touche PAUSE du clavier agit comme un clic sur le bouton PAUSE.
- Le bouton STOP (4) demande à PROSPICE d'arrêter une simulation temps réel. Toutes les animations sont arrêtées et le simulateur n'est plus présent en mémoire. Tous les indicateurs sont remis dans un état inactif, mais les actionneurs conservent leur état actuel.
La combinaison SHIFT-PAUSE du clavier est équivalente à un clic sur le bouton STOP.

Pendant l'animation, le temps de simulation courant et la charge du processeur sont affichés dans la barre de message. Si la puissance processeur est insuffisante pour obtenir un temps réel, vous lirez 100% et le temps de simulation ne progressera plus en temps réel. Le système régule automatiquement le temps de simulation nécessaire par trame d'animation, même pour des montages complexes.

Les indicateurs et les actionneurs

La simulation interactive fait appel à des composants actifs. Ces composants possèdent un certain nombre d'états graphiques et sont subdivisés en deux groupes : les indicateurs et les actionneurs. Les indicateurs affichent un état graphique qui correspond au changement de la valeur de certains paramètres mesurés sur le circuit, alors que les actionneurs autorisent une modification de leur état par l'utilisateur, qui change certaines caractéristiques du circuit.

Les actionneurs sont repérés par la présence d'un marqueur rouge sur lequel l'utilisateur peut cliquer avec la souris. Si vous utilisez une souris avec molette, vous pouvez également agir sur les actionneurs en pointant sur eux et en tournant la molette dans la bonne direction.

Valider une simulation interactive

La plupart des techniques nécessaires à la mise en œuvre et à l'exécution d'une simulation animée interactive sont relatives au dessin d'un schéma avec ISIS. Nous vous conseillons de lire le tutoriel de ISIS. Cependant les grandes étapes du processus sont résumées ci-dessous:

- Choisissez les composants que vous voulez utiliser en utilisant le bouton 'P' du sélecteur d'objets. Tous les composants actifs (actionneurs et indicateurs) sont dans la librairie 'ACTIVE.LIB', mais vous pouvez utiliser n'importe quel composant qui possède un modèle de simulation.
- Placez le composant sur le circuit.
- Éditez-les - clic droit puis clic gauche, ou appuyer sur CTRL-E – afin d'affecter les propriétés et les valeurs adaptées. Certains modèles fournissent une aide intuitive contextuelle afin que les informations des propriétés individuelles soient visualisées par le placement du curseur dans le champ et l'appui sur la touche F1.
- Le code source du micro-processeur peut être placé sous le contrôle du Proteus VSM en utilisant les commandes dans le menu 'Source'. N'oubliez pas non plus d'assigner le code objet (HEXfile) au composant du micro-processeur dans le schéma.
- Branchez le circuit en cliquant sur les pattes.
- Effacez les composants avec un double clic droit.
- Déplacez les composant en cliquant droit puis par glissement avec le bouton de gauche.
- Si vous n'utilisez pas Proteus Lite, valider le circuit animation dans le menu 'Graph'.
- Cliquez sur le bouton PLAY du panneau de contrôle d'animation pour lancer la simulation.

Lorsque vous avez utilisé des instruments virtuels, ou les modèles de microprocesseur, la fenêtre surgissante liée à ces composants peut être affichée en utilisant les commandes du menu 'Debug'.

LES EFFETS D'ANIMATION

Vue d'ensemble

De même que pour les composants actifs dans le circuit, des effets d'animation peuvent vous aider à étudier l'opération du circuit. Ces options peuvent être validées en utilisant la commande 'Définir les options d'animation' (*Set Animation Options*) du System. Les valeurs sont sauveées avec le dessin.

États logiques des pattes

Cette option affiche un carré coloré pour chaque patte connectée à un lien numérique ou mixte. Par défaut, le carré est bleu pour la logique 0, rouge pour la logique 1 et gris pour un état flottant. Les couleurs peuvent être changées par la commande 'Valeur du projet par défaut' (*Set Design Defaults*) du menu 'Gabarit' (*Template*).

Bien que cette option alourdisse quelque peu le simulateur, elle peut être très, très pratique lorsqu'elle est utilisée en relation avec des points d'arrêt et le pas à pas car elle vous permet d'étudier l'état des ports de sortie d'un microcontrôleur.

Montrer les tensions des fils en couleur

Cette option affiche en couleur tout fil qui fait partie d'un réseau analogique : la couleur représente sa tension. Par défaut la dynamique de tension va de -6V (bleu), en passant par 0V (vert), jusqu'à 6V (rouge). Les tensions peuvent être modifiées par la commande 'Définir options d'animation' (*Set Animation Options*), et les couleurs par la commande 'Valeur du projet par défaut' (*Set Design Defaults*) du menu 'Gabarit' (*Template*).

Montrer le sens du courant avec des flèches

Cette option affiche une flèche pour tout fil sur lequel circule un courant. La direction des flèches correspond au flux conventionnel du courant, et est affichée lorsque l'amplitude du courant est supérieure à un seuil. Le seuil par défaut est de 1uA et peut être modifié par la commande 'Définir options d'animation' (Set Animation Options).

Le calcul des courants implique l'insertion d'une résistance de 1mΩ dans chaque segment de fil du schéma et induit un grand nombre de nœuds supplémentaires et, par conséquent, augmente la charge de calcul du simulateur. Cependant, les flèches sont une aide pédagogique importante pour l'apprentissage de l'électricité ou l'électronique de circuits simples.

LE CONTROLE DES INCREMENTS TEMPORELS DE LA SIMULATION

Vue d'ensemble

Deux paramètres contrôlent l'évolution temps réel de la simulation. Le champ 'Trames par secondes' (*Animation Frame Rate*) détermine le nombre de rafraîchissement d'écran par seconde, alors que le champ 'Temps de simulation par trame' (*Animation Timestep*) détermine l'incrément temporel par trame. Pour une simulation temps réel, le temps par trame est la fonction réciproque du nombre de trames par seconde.

Trames par seconde

Normalement il n'est pas nécessaire de modifier la valeur par défaut qui est de 20 trames par seconde, qui donne une animation lissée sans surcharger les performances des PCs actuels. Cependant, il est quelquefois utile de diminuer cette valeur lors de la mise au point de modèles de circuits animés.

Temps par trame

Le temps par trame permet de ralentir le fonctionnement des circuits, ou d'accélérer le fonctionnement des circuits lents. Pour une simulation temps réel, le temps par trame est la fonction réciproque du nombre de trames par seconde.

Bien évidemment, le temps par trame entré par l'utilisateur sera toujours une valeur désirée. La valeur effective dépendra de la puissance CPU disponible pour calculer les ressources de simulation nécessaires pendant le temps d'une trame. La charge CPU affichée pendant l'animation représente le ratio de ces deux temps. Lorsque la puissance du CPU est insuffisante, la charge CPU sera de 100% et le temps par trame atteint baissera.

Temps d'un intervalle

Le troisième champ contrôle le pas à pas temporel. La simulation progressera de cet incrément lorsque vous cliquez sur le bouton 'STEP' (2) du panneau d'animation.

ASTUCES

Échelle de temps

Les simulations interactives seront vues, le plus souvent, en temps réel, donc vous ne pourrez pas utiliser des horloges de 1MHz, ou des entrées sinusoïdales à 10KHz, à moins d'adapter le temps par trame.

Si vous tentez de simuler quelque chose qui fonctionne très rapidement, vous devez garder certains points en mémoire.

- Pour une puissance CPU donnée, seule une certaine valeur de temps de simulation peut être calculée dans un temps réel fixé. Proteus VSM est conçu pour s'ajuster au nombre de trames par seconde, et raccourcira toute trame qui n'est pas terminée dans le temps disponible. Par conséquent, tout circuit très rapide fonctionnera plus lentement (relativement au temps réel), mais de manière plus lisse.
- Les modèles de simulation des composants analogiques sont simulés plus lentement que les modèles numériques. Sur un PC Pentium II vous pouvez simuler des circuits numériques qui opèrent à plusieurs MHz en temps réel, mais les circuits analogiques seront gérés à environ 10 kHz. Ainsi, il est idiot de tenter de simuler les oscillateurs d'horloges pour circuits numériques dans le domaine analogique. Utilisez plutôt une 'horloge numérique' (*digital clock*) et cochez la case 'Isoler avant' (*Isolate before*) afin de ne pas simuler l'horloge analogique.

Échelle de tension

Si vous utilisez des fils colorés pour indiquer une tension, vous devez indiquer la gamme des tensions présentes dans votre circuit. La dynamique par défaut est de +/-6V donc, si la tension de fonctionnement de votre circuit est différente, vous devez changer la valeur 'Tension maximale' (*Maximum Voltage*).

Mise à la masse (earthing)

PROSPICE tentera de définir un point de masse pour tout circuit actif qui n'en possède pas un explicitement (*ground terminal*). En pratique, il sera choisi au point milieu d'une batterie ou au point central d'une alimentation (*split supply*). Donc le terminal positif d'une batterie sera au-dessus de zéro et le terminal négatif sera inférieur (couleurs des fils rouge et bleu respectivement). Si le comportement n'est pas celui attendu, vous pouvez définir explicitement un point de référence de masse en utilisant un terminal 'GROUND' de ISIS.

Points à haute impédance

La logique de mise à la masse automatique contrôle également tous les terminaux des composants qui ne sont pas connectés à la masse et insérera automatiquement une résistance de grande valeur afin d'assurer la convergence du simulateur SPICE. Ceci signifie que tous les circuits constitués de composants non connectés (non connectés ou partiellement connectés) seront simulés – bien que, quelquefois, les résultats soient étranges.

INTRUMENTS VIRTUELS

VOLTMETRES ET AMPEREMETRES

Plusieurs modèles de voltmètres et d'ampèremètres sont présents dans la bibliothèque ACTIVE. Ils opèrent en temps réel et peuvent être connectés au circuit comme tout autre composant. Lorsque la simulation est lancée, ils indiquent la tension à leurs bornes ou le courant qui circule à l'aide d'un afficheur numérique.

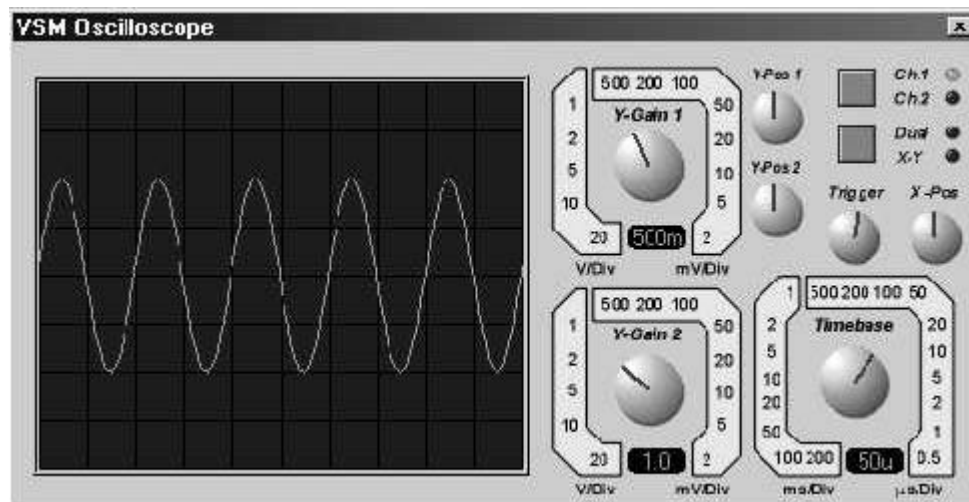
Les modèles fournis couvrent la pleine échelle de 100, 100m et 100u avec une résolution de 3 chiffres visualisés et un maximum de 2 chiffres pour la partie décimale. Ainsi l'objet VOLTMETER peut visualiser des valeurs comprises entre 0.01V et 99.9V, alors que la gamme de l'AMMETER-MILLI va de 0.01mA à 100mA, et ainsi de suite.

Les modèles de voltmètres intègrent une résistance de charge interne de 100M par défaut, qui peut être modifiée par simple édition du composant. Le fait de fournir une valeur vierge dévalide la résistance de charge du modèle.

OSCILLOSCOPE

Généralités

L'oscilloscope VSM est présent, en standard, dans le package professionnel de PROTEUS VSM, mais est une option de la gamme PROTEUS VSM Lite.



L'oscilloscope VSM modélise une unité analogique à deux voies comme spécifié ci-dessous:

- Double voies ou opération X-Y.
- Gain de 20V/div à 2mV/div.
- Temps de base de 200ms/div à 0.5us/div.
- Déclenchement automatique sur niveau de tension verrouillé sur une des voies.

Utilisation de l'oscilloscope

Pour visualiser les formes d'ondes analogiques:

1. Prenez un objet OSCILLOSCOPE dans la bibliothèque 'ACTIVE' et placez-le sur le schéma. Connectez ses entrées sur le signal que vous voulez mémoriser.
2. Lancez une simulation interactive en appuyant sur le bouton 'Play' du contrôle d'animation. La fenêtre oscilloscope doit apparaître.
3. Si vous désirez voir deux signaux, sélectionnez le mode 'Dual'.
4. Validez le temps de base à une valeur convenable, compte tenu du circuit. Vous devez penser aux fréquences présentes dans votre circuit et les convertir en temps de cycle en prenant leurs inverses.
5. Ajustez le gain Y et la position Y afin d'adapter la taille et la position. Si la forme d'onde est constituée d'un petit signal AC superposé en haut d'une forte tension DC, vous pouvez connecter un condensateur entre le point de test et l'oscilloscope, car la compensation de la position Y est limitée en amplitude.
6. Choisissez le canal de déclenchement et assurez-vous que la led correspondante de Ch1 ou Ch2 est active.
7. Tournez le bouton de déclenchement jusqu'à ce que l'afficheur soit verrouillé sur la portion souhaitée du signal. Le verrouillage se fait sur des fronts montants si le bouton pointe vers le haut, et sur des fronts descendants s'il pointe vers le bas.

Modes d'opération

L'oscilloscope peut opérer en 3 modes, indiqués ci-dessous:

- Simple voie – ni la led Dual, ni la led X-Y ne brille. Dans ce mode, la led de Ch1 ou Ch2 indique quel canal est affiché.
- Double voie – la led Dual brille. Dans ce mode, les leds Ch1 ou Ch2 indique quelle est la voie de déclenchement.
- Mode X-Y – la led X-Y brille.

La commutation du mode courant s'obtient en cliquant sur le bouton à côté des leds Dual et X-Y.

Déclenchement

L'oscilloscope VSM fournit un mécanisme de déclenchement automatique qui synchronise le temps de base avec la forme d'onde incidente.

Les leds Ch1 et Ch2 indiquent quelle voie est utilisée pour déclencher.

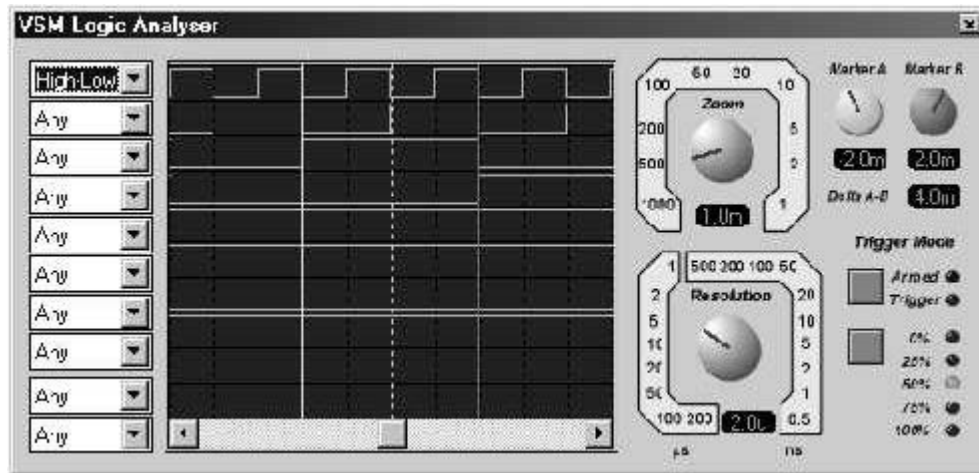
Le bouton de déclenchement tourne sur 360° et valide le niveau de tension et la pente de déclenchement. Lorsque le marqueur pointe vers le haut, le scope déclenche sur des fronts montants; lorsque le marqueur pointe vers le bas, le déclenchement se fait sur des fronts descendants.

Si aucun déclenchement n'intervient sur plus d'une période du temps de base, le temps de base devient libre.

ANALYSEUR LOGIQUE

Généralités

L'analyseur logique VSM est présent, en standard, dans le package professionnel de PROTEUS VSM, mais est une option de la gamme PROTEUS VSM Lite.



Un analyseur logique opère en enregistrant de manière continue les données numériques dans un grand tampon de stockage. C'est un processus d'échantillonnage, donc il existe une résolution ajustable qui définit la plus petite impulsion pouvant être mémorisée. Une section de déclenchement contrôle les données entrantes et force l'arrêt du processus de capture après que la condition de déclenchement ait été atteinte; la capture commence en armant l'instrument. Le résultat est que le contenu du tampon de capture peut être affiché avant et après le temps de déclenchement. Comme le tampon de capture est très large (10000 échantillons, dans ce cas), ceci rend possible un zoom et un panoramique de l'affichage. Enfin, des marqueurs de mesures permettent des mesures précises telles que des largeurs d'impulsion, et ainsi de suite.

Les caractéristiques de l'analyseur logique VSM 24 canaux sont:

- Courbes 8 x 1 bit et bus 2 x 8 bits.
- Tampon de capture 10000 x 24 bits.
- Résolution de la capture résolution de 200us par échantillon à 0.5ns par échantillon qui correspond à des temps de capture de 2s à 5ms.
- Zoom d'affichage réglable de 1000 échantillons par division à 1 échantillon par division.
- Déclenchement sur combinaison ET de l'état des entrées et/ou fronts, et valeurs des bus.
- Position du déclenchement à 0, 25, 50, 75 et 100% du tampon de capture.
- Deux curseurs sont disponibles pour des mesures temporelles précises.

Utilisation de l'analyseur logique

Pour capturer et visualiser des données numériques:

1. Prenez l'objet LOGIC ANALYSER dans la bibliothèque ACTIVE, placez-le sur le schéma et connectez ses entrées sur le signal que vous voulez enregistrer.
2. Lancez la simulation interactive en appuyant sur le bouton 'Play' du panneau de contrôle de l'animation. La fenêtre de l'analyseur logique doit apparaître.
3. Utilisez le bouton de résolution pour définir une valeur convenable. Celle-ci représente la plus petite largeur d'impulsion pouvant être enregistrée. Plus la résolution est petite, plus petit sera le temps d'enregistrement.
4. Validez les listes déroulantes à gauche de l'instrument pour définir les conditions de déclenchement. Par exemple, si vous désirez déclencher lorsque le signal sur la voie 1 est à l'état haut, et que celui sur la voie 3 est une transition montante, vous devez valider la première liste avec '*High*' et la troisième avec '*Low-High*'.
5. Décidez de la proportion des données à voir avant et après que la condition de déclenchement soit intervenue, et cliquez sur le bouton placé à côté des leds de pourcentage pour choisir la position souhaitée.
6. Lorsque vous êtes prêt, cliquez sur le bouton placé à gauche de la led '*armed*' pour 'armer' l'instrument.

La led 'armed' brillera et la led de déclenchement sera éteinte. L'analyseur logique capture les données incidentes de manière continue tout en testant les conditions de déclenchement des entrées. Lorsque les conditions sont vérifiées, la led de déclenchement brille. La capture des données se poursuit jusqu'à remplissage de tampon. Dès cet instant, la led 'armed' s'éteint et le tampon est visualisé.

Panoramique et zoom

Comme le tampon de capture contient 10 000 échantillons, et que la largeur de l'afficheur est de 250 pixels, il est nécessaire de permettre un panoramique et un zoom du tampon de capture. Le bouton Zoom détermine le nombre d'échantillons par division, tandis que l'ascenseur associé à l'afficheur permet un panoramique vers la gauche ou la droite.

Noter que le texte placé au-dessous du bouton de Zoom affiche le temps courant par division en secondes, et non pas la valeur courant du bouton lui-même. Le temps de division est calculé en multipliant la valeur du zoom par la résolution.

Mesures

Deux marqueurs ajustables sont disponibles pour faire des mesures précises. Chaque marqueur peut être positionné en utilisant le bouton coloré associé. Le texte sous le bouton affiche le point temporel occupé par le marqueur relativement au temps de déclenchement, tandis que le texte 'Delta A-B' affiche la différence de temps entre les marqueurs.

ELEMENTS D'INTERFACE UTILISATEUR

Boutons rotatifs

Les instruments virtuels VSM utilisent la souris pour agir sur des boutons qui ajustent certains paramètres. La procédure d'ajustement est la suivante:

1. Pointez quelque part à l'intérieur du bouton.
2. Appuyez sur le bouton gauche et maintenez-le enfoncé.
3. Éloignez le pointeur souris du bouton et tournez autour du centre du bouton, en traçant un arc circulaire pour tourner le bouton jusqu'à la valeur désirée.
4. Le bouton suit l'angle formé par le pointeur souris. Plus vous éloignez la souris du centre, plus vous obtenez un contrôle fin.
5. Relâchez le bouton de la souris pour valider la nouvelle position du bouton.

TRAVAILLER AVEC DES MICROPROCESSEURS

INTRODUCTION

La combinaison du mode de simulation mixte et de l'animation des circuits devient particulièrement efficace lorsqu'elle est utilisée avec des systèmes à microcontrôleurs. Beaucoup de ces systèmes nécessitent une interface utilisateur, ou au minimum des événements externes qui ne peuvent pas être simulés facilement dans un environnement non interactif, et PROTEUS VSM a été développé principalement pour répondre à ces problèmes. Par conséquent, un nombre important de ses fonctionnalités est centré autour du développement des systèmes à microcontrôleurs.

En particulier, l'édition et la compilation du code source sont intégrées dans l'environnement du projet afin que vous puissiez éditer le code source et observer facilement les effets d'une modification. Un fichier source peut être édité grâce à deux touches clavier et l'exécution d'une nouvelle simulation avec deux touches également.

CONTROLE DU CODE SOURCE

Vue d'ensemble

Le système de contrôle du code source dispose de deux fonctions majeures:

- L'enregistrement du nom de fichier source dans ISIS afin de pouvoir l'éditer sans commuter dans un autre environnement.
- La déclaration des règles de compilation du source afin de créer le code objet. Lorsqu'elles sont validées, elles sont automatiquement exécutées à chaque demande de simulation, afin que le code objet utilisé soit le plus à jour.

Associer le code source à un projet

Pour ajouter un fichier source à un projet:

1. Sélectionnez la commande 'Ajouter/Enlever fichiers source' (*Add/Remove source files*) du menu 'Source'.
2. Sélectionnez l'outil de génération de code pour le fichier source. Si vous désirez utiliser un nouvel assembleur ou un compilateur, la première fois vous devrez l'enregistrer par la commande 'Définir les outils de génération de code' (*Define Code Generation Tools*).
3. Cliquez sur le bouton 'Nouveau' (New) et sélectionnez ou entrez le nom d'un fichier source via le sélecteur de fichiers. Vous pouvez entrer explicitement un nom de fichier qui n'existe pas encore.
4. Validez les 'indicateurs' (Flags) nécessaires à la compilation du source. Les indicateurs à inclure à chaque fois qu'un outil est utilisé peuvent être entrés lors de l'enregistrement de l'outil.
5. Cliquez sur OK pour ajouter le fichier source au projet.

N'oubliez pas d'éditer le microprocesseur afin d'affecter le fichier objet (fichier HEX) à la propriété PROGRAM. ISIS ne peut pas faire cela automatiquement car vous pourriez avoir plusieurs processeurs dans le schéma!

Travailler avec votre code source

Pour éditer un fichier source:

1. Utilisez la combinaison ALT-S.
2. Validez le numéro d'ordre du fichier source dans le menu 'Source'.

Pour revenir à ISIS, construire le code objet et lancer la simulation:

1. Depuis l'éditeur de texte, utilisez la combinaison ALT-TAB pour revenir à ISIS.
2. Appuyez sur F12 pour lancer la simulation, ou CTRL-F12 pour lancer le debug.

ISIS examine les dates des fichiers source et objet afin d'utiliser l'outil de génération approprié lorsque c'est nécessaire.

Pour produire à nouveau tous les codes objets:

1. Sélectionner la commande 'Tout compiler' (*Build All*) du menu 'Source'.

ISIS exécutera tous les outils de génération de code nécessaires pour obtenir le code objet, sans tenir compte des dates et des heures des fichiers objets actuels. Une fenêtre affiche les indications fournies par les outils afin de contrôler les messages d'erreurs ou d'avertissements retournés.

Installation d'outil de génération tiers

Plusieurs assembleurs ou compilateurs (*sharewares*) peuvent être installés dans le répertoire \PROTEUS\TOOLS depuis le CD, et seront validés automatiquement comme outil de génération de code par le programme d'installation de PROTEUS. Cependant, il vous est possible d'utiliser d'autres outils; pour ce faire, lancez la commande Définition des outils de génération de code' (*Define Code Generation Tools*) du menu 'Source'.

Pour enregistrer un nouvel outil:

1. Sélectionnez la commande 'Définition des outils de génération de code' (*Define Code Generation Tools*) du menu 'Source'.
2. Cliquez sur le bouton 'Nouveau' (*New*) et utilisez le sélecteur de fichiers pour localiser le fichier exécutable de l'outil. Vous pouvez aussi utiliser un fichier 'batch' pour outil de génération.
3. Entrez les extensions des fichiers source et objet. Ceci indiquera à ISIS quel est l'outil à utiliser en fonction de l'extension choisie. Si vous cochez l'option 'Construire toujours' (*Always build*), alors l'outil est toujours utilisé et l'extension du code objet n'est pas nécessaire.
4. Spécifiez une ligne de commande pour l'outil. Utilisez %1 pour faire référence au fichier source et %2 pour le fichier objet. Vous pouvez également utiliser %\$ pour le répertoire de PROTEUS et %~ pour le répertoire dans lequel est placé le fichier DSN (schéma ISIS).

C'est à cet endroit qu'il est préférable d'ajouter les arguments de la ligne de commande afin de faciliter l'automatisation du processus de génération de code (c'est à dire sans pause pour l'entrée de données utilisateur), et de préciser les répertoires des fichiers entêtes spécifiques au processeur à inclure.

Si vous voulez utiliser les fonctionnalités du debugger source de Proteus VSM, vous aurez besoin d'un extracteur de données de debug pour votre assembleur ou compilateur. C'est une ligne de commande supplémentaire qui extrait les informations des références croisées code source/objet du fichier liste généré par l'assembleur ou le compilateur.

Si vous possédez un programme DDX, entrez l'extension de votre fichier liste ou du fichier qui contient les informations symboliques de mise au point issues de l'outil de génération de code, et

cliquez sur le bouton 'Parcourir' (*Browse*) pour indiquer le chemin et le nom de votre programme DDX.

Utilisation d'un programme MAKE

Dans certains cas, les règles simples de construction de votre application présentes dans ISIS ne suffisent pas – spécialement lorsque votre projet comprend des sources et des codes objets multiples, ainsi qu'une phase d'édition de liens. Dans de tels cas, il faudra utiliser un programme externe (*MAKE*) de création d'applications. Dans ce cas, procédez de la façon suivante:

Valider la construction du projet avec un programme MAKE externe:

1. Installez votre programme 'make' (habituellement livré avec votre assembleur/compilateur) comme outil de génération de code. Validez l'extension source avec MAKE et cochez 'Construire toujours' (*Always build*). Pour un programme "make" typique, validez la ligne de commande suivante:
2. Utilisez la commande 'Ajouter/Enlever des fichiers sources' pour inclure le fichier de description (*makefile*) – par exemple : MONPROJET.MAK.
3. Ajoutez également les fichiers sources, mais sélectionnez 'Aucun' (*NONE*) comme outil de génération de code.

Utilisation de l'éditeur de source tiers

Proteus VSM est fourni avec un éditeur de texte simple – SRCEDIT qui vous permet d'éditer vos sources. SRCEDIT est une version modifiée du bloc notes (NOTEPAD) qui peut ouvrir plusieurs fichiers sources, et peut répondre à une demande DDE pour sauver tout tampon mémoire modifié.

Si vous possédez un éditeur de texte plus évolué, tel que UltraEdit, vous pouvez indiquer à ISIS de l'utiliser en lieu et place de SrcEdit. Il est nécessaire que votre éditeur puisse répondre à une demande DDE.

Valider un éditeur de source:

1. Sélectionnez la commande 'Valider un éditeur de texte externe' (*Setup external text editor*) du menu 'Source'.
2. Cliquez sur le bouton 'Parcourir' (*Browse*) et utilisez le sélecteur de fichier pour localiser le fichier exécutable de votre éditeur.
3. ISIS pilote l'ouverture de l'éditeur et la sauvegarde via le protocole DDE. Référez-vous à la documentation de votre éditeur ou à votre fournisseur pour la syntaxe de la commande.

FENETRES SUPPLEMENTAIRES

La plupart des modèles de microprocesseurs écrits pour Proteus VSM créeront des fenêtres d'informations qui peuvent être affichées ou cachées grâce au menu 'Debug'. Il existe trois groupes de fenêtres:

- Fenêtre de statuts : un modèle de processeur utilisera une telle fenêtre pour afficher l'état des registres.
- Fenêtre mémoire – il en existe typiquement une par espace mémoire du microprocesseur. Les composants mémoires (RAM ou ROM) créent également ces fenêtres.
- Fenêtre source – une par fichier source spécifié par la propriété PROGRAM du processeur.

Pour afficher une fenêtre supplémentaire:

1. Lancez le mode de mise au point par CTRL-F12 ou, si le mode est déjà lancé, cliquez sur le bouton Pause du contrôle de l'animation.
2. Pressez ALT-D, puis sélectionnez le numéro d'ordre de la fenêtre désirée dans le menu 'Debug'.

Ces fenêtres ne peuvent être affichées que lorsque la simulation est en pause, et sont masquées automatiquement lors d'une exécution pour vous donner accès aux composants actifs du schéma. Lorsque la simulation est en pause (soit manuellement, soit par l'intermédiaire d'un point d'arrêt), les fenêtres qui doivent l'être apparaissent.

Toutes les fenêtres de mise au point ouvrent un menu contextuel sur le clic droit de la souris. Ce menu vous permet de contrôler l'apparence et le formatage des données dans la fenêtre.

La position et la visibilité des fenêtres de mise au point sont sauveées automatiquement dans un fichier PWI, avec un nom de fichier identique à celui du projet courant. Le fichier PWI contient également les positions de tous les points d'arrêts et le contenu de la fenêtre 'Watch'.

MISE AU POINT AU NIVEAU DU SOURCE

Vue d'ensemble

Proteus VSM supporte un mode de mise au point au niveau du code source par l'intermédiaire de programmes d'extraction de données de debug (*DDX : Debug Data eXtraction*). Les programmes DDX sont des utilitaires qui parcourent les fichiers produits par les assembleurs ou les compilateurs, extraient les informations des références croisées entre les adresses du code objet et les lignes du code source, et les formatent dans un format standard.

Si vous avez installé un programme DDX pour votre assembleur ou compilateur, Proteus VSM créera une fenêtre source pour chaque fichier source du projet, et ces fenêtres seront listées dans le menu 'Debug'.

Fenêtres de code source

Les fenêtres de code source possèdent un certain nombre de caractéristiques:

- Une barre bleue indique le numéro de la ligne courante, et un point d'arrêt peut être placé en appuyant sur la touche F9 (commutateur de point d'arrêt), et vers lequel le programme s'exécutera si vous utilisez la combinaison CTRL-F10 (Aller vers).
- Un chevron rouge indique la position courante du compteur programme.
- Un circuit rouge repère la ligne sur laquelle est placé le point d'arrêt.

Le menu contextuel disponible sur clic droit fournit une des options supplémentaires telles que :

Aller à la ligne, Aller à l'adresse, Chercher ainsi que des commutateurs pour afficher les numéros de lignes, les adresses et les octets du code objet.

Pas à pas

Des options de pas à pas sont également disponibles dans le menu 'Debug'.

Step Over – 'avance d'une ligne', à moins que l'instruction soit un appel à une fonction. Dans ce cas l'ensemble de la fonction est exécuté.

Step Into – 'exécute une instruction du code source'. Si aucune fenêtre source n'est active, il exécute une instruction de code machine.

Step Out – 'exécute jusqu'à la fin de la fonction courante'.

Step To – 'exécute jusqu'à ce que le programme arrive à la ligne courante'. Cette option n'est disponible que lorsque la fenêtre de code source est active.

Notez que, mis à part pour *Step To*, les commandes d'exécution en pas à pas fonctionneront sans fenêtre de code source. Il est possible – bien que cela ne soit pas facile – de mettre au point un programme généré par un outil pour lequel il n'existe aucun support DDX.

Utilisation des points d'arrêts

Un point d'arrêt fournit un moyen d'investigation puissant, des problèmes de logiciel ou de couplage logiciel/matériel d'un projet. Typiquement vous placerez un point d'arrêt à l'entrée d'une fonction qui pose problème, lancerez la simulation, puis interagirez avec le projet afin que le programme passe par le point d'arrêt. A ce point, la simulation sera suspendue. Puis vous tracerez le programme en pas à pas, en observant les valeurs des registres, la mémoire et les autres conditions de fonctionnement du circuit. Validez la 'visualisation de l'état logique des pattes' (*Show Logic State of Pins: ON*) est également très instructif.

Lorsque la fenêtre de code source est active, le point d'arrêt peut être placé ou enlevé de la ligne courante à l'aide de la touche F9. Vous pouvez placer un point d'arrêt sur toute ligne dont le code objet existe.

Si le code source change, Proteus VSM tentera de retrouver la position du point d'arrêt sur la base des adresses des sous-programmes du fichier et, par association, dans les octets du code objet. Évidemment, lorsque vous modifiez le code radicalement, la tentative sera impossible, mais généralement ceci fonctionne très bien.

LA FENETRE WATCH

Alors que les fenêtres des registres et des mémoires associées au modèle du processeur ne sont affichées que lorsque la simulation est 'suspendue' (pause), la fenêtre 'Watch' fournit un moyen de visualiser les valeurs qui sont mises à jour en temps réel. Elle permet également d'affecter des noms à des positions mémoires individuelles, ce qui rend leur gestion plus simple que la recherche dans une fenêtre mémoire.

Pour ajouter une rubrique dans la fenêtre 'Watch' :

1. Pressez CTRL-F12 pour lancer le mode de mise au point, ou cliquez sur le bouton 'Pause' si la simulation est déjà lancée.
2. Affichez la fenêtre 'mémoire' qui contient la rubrique à scruter, et la fenêtre 'Watch' grâce au numéro d'ordre du menu 'Debug'.
3. Marquez la position mémoire ou plusieurs positions avec le bouton gauche de la souris. Les positions sélectionnées doivent apparaître en couleur inversée.
4. Glissez les rubriques sélectionnées dans la fenêtre 'Watch'.

Vous pouvez également ajouter les rubriques dans la fenêtre 'Watch' en utilisant la commande 'Ajouter rubrique' (*Add Item*) grâce au menu contextuel.

Modifier les rubriques dans la fenêtre 'Watch'

Lorsque vous avez placé une ou plusieurs rubriques dans la fenêtre 'Watch', vous pouvez sélectionner une rubrique avec le bouton gauche de la souris et :

- Renommer la rubrique avec CTRL-R ou F2.

- Modifier la taille de la donnée avec l'une des options disponibles dans le menu contextuel (clic droit). Pour les tailles de données de plusieurs octets (exemple: mots de 16 ou 32 bits ou chaînes de caractères), les octets sont supposés être consécutifs à l'adresse de la rubrique. Par conséquent, pour afficher un mot de plusieurs octets ou une chaîne, il vous suffit de glisser le premier octet depuis la fenêtre mémoire.
- Modifier le format de numération en binaire, octal, décimal ou hex.

OBJETS DE DECLENCHEMENT DE POINT D'ARRET

Vue d'ensemble

Certains objets composants permettent de suspendre la simulation lorsqu'une condition particulière survient dans le circuit. Ils sont particulièrement utiles lorsqu'ils sont combinés aux possibilités de pas à pas, bien qu'un circuit puisse être simulé normalement jusqu'à apparition d'une condition particulière, puis tracé en pas à pas afin d'observer finement ce qui se passe.

Les objets de déclenchement spécifiques sont présents dans la bibliothèque REALTIME.

Déclenchement d'un point d'arrêt de tension – RTVBREAK

Cet objet est disponible sous la forme 1 ou 2 pattes, et déclenche un point d'arrêt lorsque la tension sur la patte, ou la tension entre 2 pattes, est supérieure à la valeur spécifiée. Vous pouvez coupler cet objet à une source de tension contrôlée arbitraire (AVCS) pour déclencher un point d'arrêt sur une formule complexe qui implique des tensions multiples, des courants, etc.

Lorsque le déclenchement est intervenu, l'objet ne peut déclencher à nouveau que lorsque la tension est redescendue sous le seuil de déclenchement, puis croit à nouveau.

Déclenchement d'un point d'arrêt de courant – RTIBREAK

Cet objet possède deux pattes et déclenche un point d'arrêt lorsque le courant qui le parcourt est supérieur à une certaine valeur.

Lorsque le déclenchement est intervenu, l'objet ne peut déclencher à nouveau que lorsque le courant est redescendu sous le seuil de déclenchement, puis croit à nouveau.

Déclenchement d'un point d'arrêt numérique – RTDBREAK

Cet objet est disponible avec un nombre de pattes variable et vous pouvez définir d'autres tailles si c'est nécessaire. Il déclenche un point d'arrêt lorsque la valeur binaire sur ses entrées égale la valeur affectée au composant. Par exemple, le fait de spécifier la valeur 0x80 à RTDBREAK_8, provoquera un déclenchement lorsque D7 est à l'état haut et D0-D6 sont à l'état bas.

Lorsque le déclenchement est intervenu, l'objet ne peut déclencher à nouveau que lorsque la tension sur ses différentes entrées a changé.