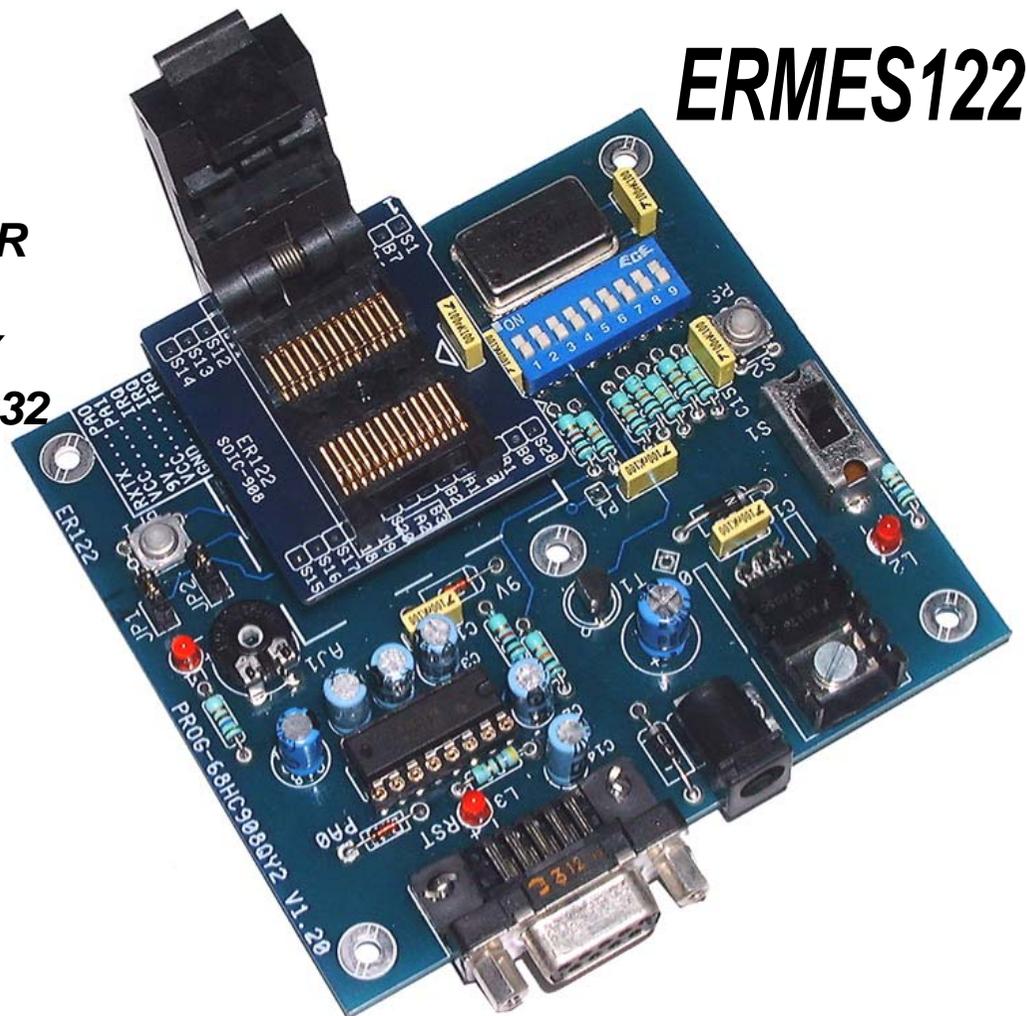


PROGRAMMATEUR ÉMULATEUR MC 68HC908QYXX ET MC68HC908GP32



ERMES122

Caractéristiques :

Alimentation : 12 VDC

Dimensions : 89x 95 mm

1 PRÉSENTATION

L'interface décrite ici permet la programmation des micro contrôleurs (μ C) MOTOROLA de la famille 68HC908, développer précédemment et spécifiquement pour le 68HC908QY (appelé NITRON). Il s'est avéré être compatible moyennant un adaptateur avec d'autres produits de la gamme comme le 68HC908GP32. Ces μ C ont des atouts majeurs par rapport à leurs concurrents qui les rendent bien plus attrayant dans tous les domaines, développement, intégration des périphériques, vitesse et coût.

Et surtout, ils intègrent tous un Debugger en ROM qui permet la programmation, l'effacement in situ (sur le circuit imprimé) puis le mode DEBUG avec point d'arrêt, mode temps réels. D'autres part tous les outils logiciels sont fournis gracieusement par MOTOROLA et garde la même ergonomie quel que soit le μ C de la gamme, ils sont disponibles sur INTERNET, mais vous trouverez sur le CD les logiciels ainsi que la documentation nécessaires. Vous trouverez quelques conseils d'utilisation dans cette notice, mais l'ensemble des documents MOTOROLA sont rédigés en anglais.

2

FONCTIONNEMENT

A) Le 68HC908QY :

C'est une μ C avec mémoire flash qui se décline en plusieurs versions QY1, QY2, QY4, ST1, ST2, QT4 dont chacun possède des caractéristiques plus ou moins importantes. Nous verrons plus spécifiquement le 68HC908QY2 μ C de milieu de gamme, qui permet pour un coût abordable des applications sérieuses.

Voici une présentation succincte du 68HC908QY2.

Alimentation 3 et 5V, Code compatible 68HC05

Mémoire EPROM FLASH 1536 octets

Mémoire RAM 128 octets

Oscillateur Interne (ajustable) ou externe 3.2Mhz et 9.8Mhz

Mode AutoWakeup

Mémoire FLASH auto programmable

2 Timers 16bits (commutable sur pins externes)

4 convertisseurs analogiques, numériques 8Bits

13 entrées/sorties (I/O) bi-directionnel

1 entrée IRQ

1 entrée RST

6 interruptions pour clavier

Pull up interne sur port A et B, pin RST et pin IRQ

Boîtier PDIP16 Pins et SOIC

1 entrée INTERRUPTION

1 watchdog interne

1 contrôleur d'alimentation intégré

3

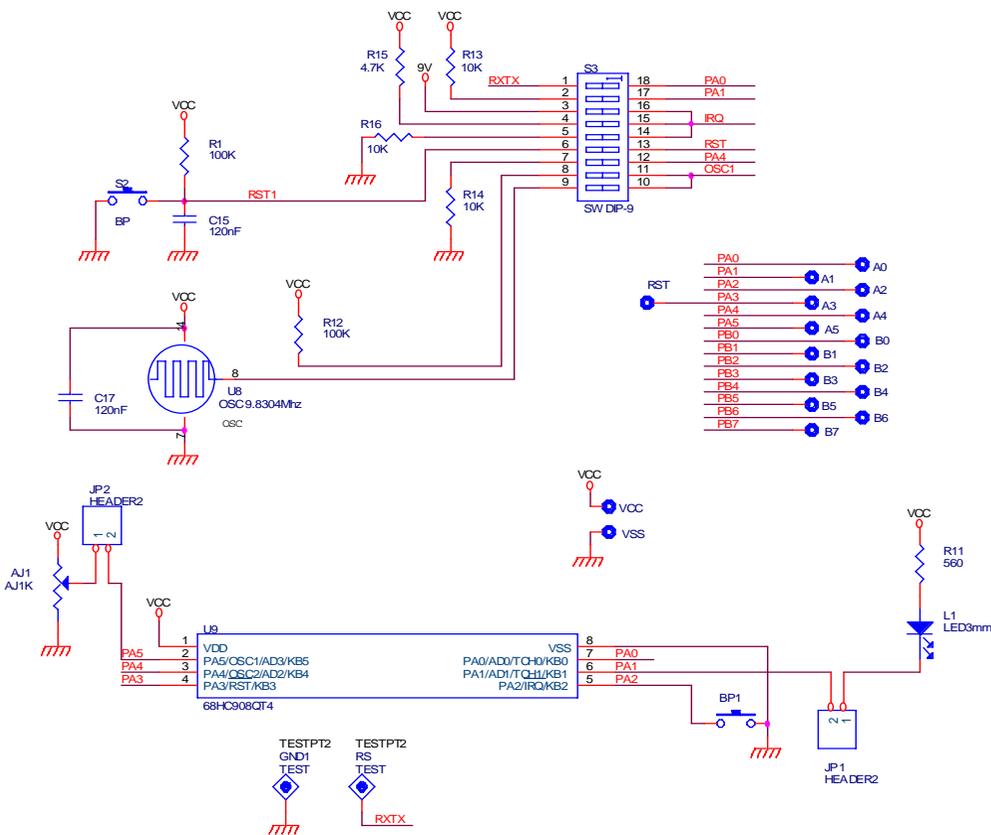
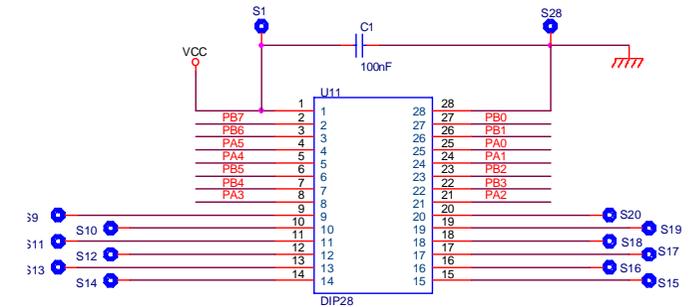
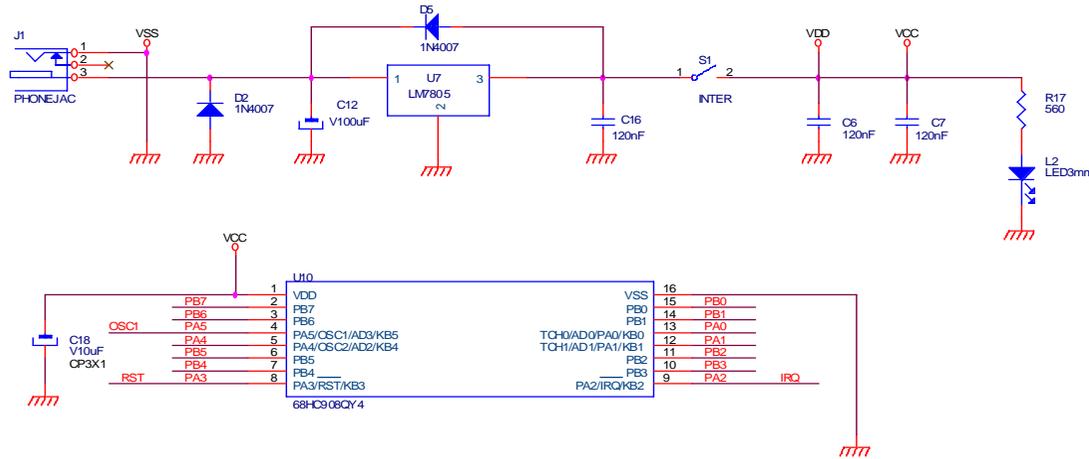
PRINCIPE

L'alimentation est issue d'un bloc secteur 12V qui va délivrer le 5V à travers le régulateur U7 et le 9V à travers la diode zener D3. Ce 9V va forcer le passage du μ C en mode DEBUG. La diode D2 protège le montage des inversions de polarité, à éviter quand même. La led L2 signale la mise sous

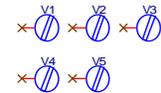
ERMES122

tension du μC . L'interface vers le PC, adaptation des signaux du port série, est réalisée classiquement par un CI MAX232 qui convertit le 5V en $\pm 12V$. Une pin de ce circuit, visualisé par la LED L3, est utilisée pour réaliser le RESET du μC .

Les mini interrupteurs type Dip de S3 permettent la configuration du μC suivant le mode désiré. L'oscillateur intégré à quartz U8 sert principalement lors de la programmation.

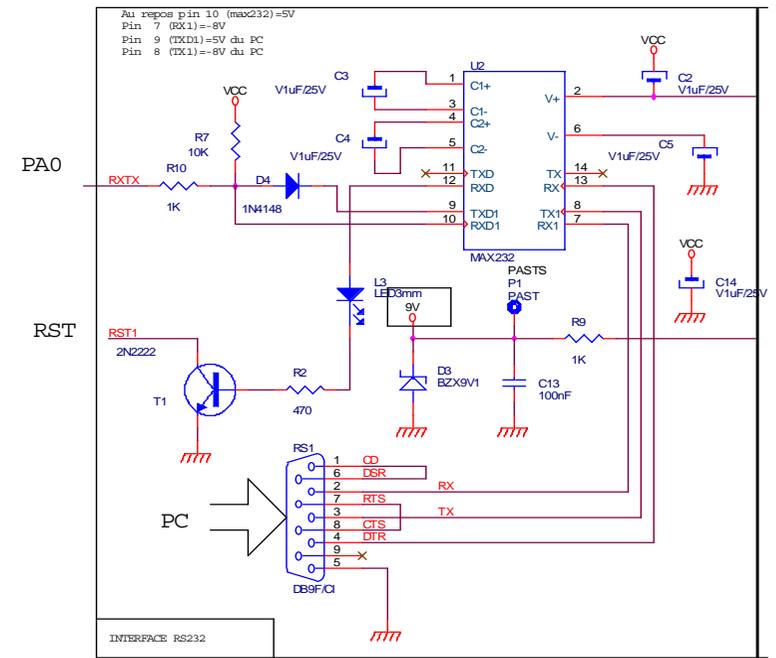


V1.20



SHEET PRQY2A.SCH
 Vast Monitor Mode
 IRQ=Vtst
 RST=VDD
 PA1=1
 PA4=0
 CLK=9.8304Mhz
 Bus=2.4576Mhz
 COP=Disabl ed

Schéma de principe :



ERMES122

Trois composants ont été intégrés sur la carte pour tester des logiciels, il s'agit d'un bouton poussoir BP1, d'une led L1 commutable par JP1 et d'un ajustable AJ1 commutable par JP2. Avec ces trois composants on peut déjà réaliser de nombreux montage à base de Timer, PWM, Convertisseur analogique numériques et Etc. **JP1 et JP2 seront enlevés en fonctionnement normal.**

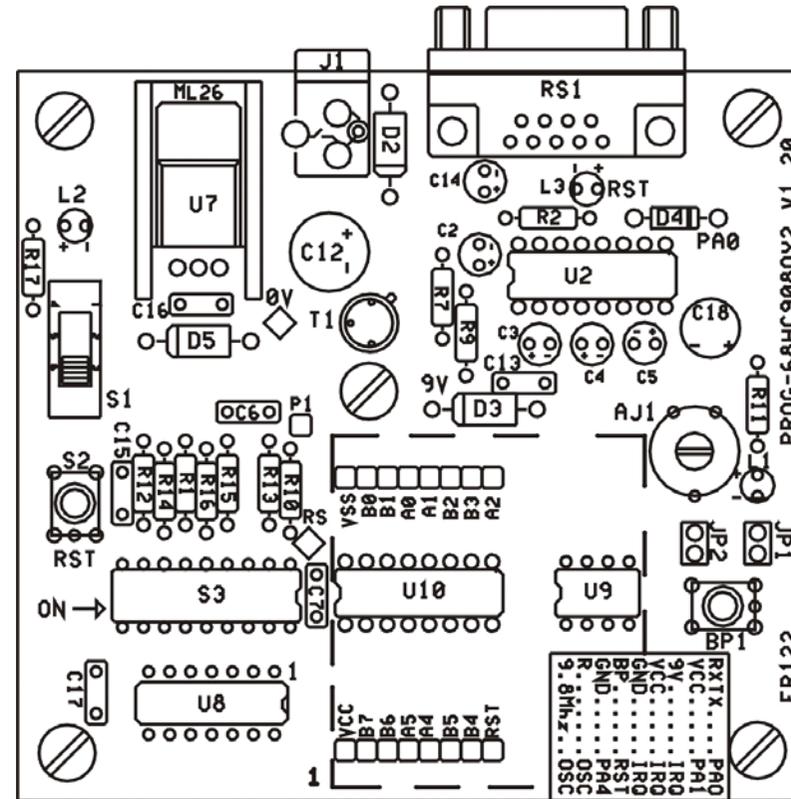
Pour programmer et utiliser les circuits 908 Cms, il à été prévu une interface vers un support à insertion nulle qui s'insère dans les barrettes sécables de A0, A1 à B7 ..

Ces barrettes pourront servir aussi à récupérer les signaux d'un 68HC908QY qui est sur la carte ER122 lors de la programmation ou pendant le mode DEBUG. Il suffit de câbler quelques fils pour travailler sur une carte externe. Ceci évite la réalisation d'une autre interface PC sur la carte (ou l'embryon de carte) en cours de développement. L'auteur utilise souvent ces connecteurs à travers des barrettes sécables ou par un adaptateur sur U9, pour activer une carte externe. Il ne faut que quelques signaux pour programmer, effacer un 68HC908, ou émuler un logiciel. Une fois le logiciel développé on peut isoler les signaux de la carte grâce aux mini-inter de S3. D'autre part, on peut utiliser la liaison série de la carte ER122 via PA0, pour établir un dialogue avec le PC. On peut programmer le 68HC908QT4 sur le support 8 Pins.

3 RÉALISATION

ATTENTION :

Implantation composants :



Suite aux retours SAV, nous avons constaté certaines erreurs dues à l'inattention ou au manque d'application lors de la réalisation des KITS. Vous trouverez ci-après les erreurs classiques généralement constatées.

1/ La soudure froide : Elle se produit lorsque la panne du fer ne chauffe pas assez les deux éléments à souder, la soudure ne peut pas accrocher, car la température n'est pas atteinte. Une panne peut se produire de suite ou après quelques temps d'utilisation, lorsque l'oxydation fait son œuvre. Vérifiez que la soudure est brillante et qu'elle forme un cône autour de la patte du composant, de plus

rappelons qu'il ne faut jamais souffler sur une soudure (même pour aller plus vite) .

2/ La " gougoutte " de soudure entre deux pattes très proches : La solution est simple, vérifiez avant la soudure les connexions aboutissant à la pastille que vous allez souder, et contrôlez après. N'oubliez pas, que plus un composant est petit (condensateur, transistor), plus il a du mal à évacuer la chaleur, ne restez pas trop longtemps (<5s) sur une patte et espacez le soudage sur un composant actif.

ERMES122

3/ N'hésitez pas à plaquer correctement les éléments sur le circuit imprimé (support CI, poussoir etc..), dans le cas contraire lors de l'utilisation (insertion, extraction, serrage) les efforts ne seront pas transmis sur l'élément, mais sur les pistes du circuit imprimé d'où rupture de celles-ci. La méthode consiste par exemple pour un support C.I., à faire 2 soudures en diagonale puis appuyez sur le support et chauffez les 2 soudures, l'une après l'autre, vous serez surpris de voir que le support s'enfonce encore. Une exception à cette règle, concerne les éléments qui sont amenés à chauffer (risque de brûlure sur le circuit imprimé).

Un dernier conseil : Pour le positionnement des composants, nous vous conseillons de les implanter dans le même sens de lecture (la vérification des valeurs sera grandement facilitée), et de bien les plaquer sur le circuit-imprimé, la résistance mécanique sera bien meilleure. Le circuit imprimé étant percé et sérigraphié, la difficulté réside juste dans le placement des composants.

ATTENTION :

Certains composants sont polarisés, ils ont donc un sens d'insertion particulier. Il s'agit des diodes, des leds, des circuits intégrés et de leurs supports. Il est conseillé d'implanter les composants par ordre de taille croissante, veuillez donc, de préférence, procéder comme suit.

Le régulateur 7805 doit être fixé sur le refroidisseur et serré sur le circuit imprimé avant d'être soudé. Vérifiez bien l'insertion de toutes les pins sur la prise DB9 avant de commencer à souder. Attention en coupant les barrettes

sécables de ne pas les mettre en petits morceaux, utilisez une pince coupante fine, éventuellement un couteau sur une planche. Pour les barrettes sécables JP1 et JP2, mettre le cavalier avant de les souder pour éviter de se brûler les doigts en appuyant dessus lors de la soudure. Pour les barrettes sécables de chaque cotés de U8, vérifier quelles soient bien verticales et enfoncées. Soudez U8 directement en faisant attention aux polarités.

Mise en route ER122-1

Montez les résistances :

R9, R10 : 1KR (marron, noir, rouge)
R7, R13, R14, R15, R16 : 10KR (marron, noir, orange)
R2 : 470R (Jaune, violet, marron)
R1, R12 : 100KR (marron, noir, jaune)
R11, R17 : 560R (vert, bleu, marron)

Montez les diodes :

D2, D5 : 1N4007 (attention au sens)
D4 : 1N4148 (attention au sens)
D3 : BZX9V zener (attention au sens)

Montez les supports de circuit intégrés :

16 broches : U2, U10 (attention au sens)
8 broches : U9 (attention au sens)

Montez les boutons poussoir :

BP1, S2 : (à souder 4 broches)

Montez l'oscillateur à Quartz :

U8 : 9,8304Mhz (4 broches)

Montez les LEDS :

L1, L2, L3 : Rouge (attention à la polarité)

Montez la résistance ajustable :

AJ1 : 1KR (3 broches)

Montez le dip switch :

S3 : 9 contacts (18 broches)

Montez les condensateurs milfeuil :

C6, C13, C15, C16, C17 : 100nf

Montez le transistor :

T1 : 2N2222 (attention au sens)

Montez les supports pour cavalier :

JP1, JP2 : 2 x 2 broches

Montez les condensateurs chimiques :

C2, C3, C4, C5, C14 : 1uF/25V radial (respectez la polarité)
C18 : 10uF/25V radial (respectez la polarité)
C12 : 100uF/25V radial (respectez la polarité)

Montez le l'interrupteur :

S1 : glissière 2 positions (6 broches + 2)

Montez le régulateur 5V :

U7 : LM7805 (ne pas oublier le radiateur)

Montez le Jack d'alimentation :

J1 : à souder (3 broches)

Montez la prise Sub D :

RS1 : à souder (9 broches)

Montez les circuits intégrés sur leurs supports :

U10 : 68HC908QY2 (voir sens support)
U2 : MAX232 (voir sens support)

Platine ER122-2

La carte d'extension ER122-2 a été prévue pour la programmation des circuits CMS, elle s'insère sur la carte ER122. Le support choisi est un 28 broches, car il permettra pour un coût qui est guère plus important, la programmation de μ C de plus grande taille. Il suffira de prévoir sur les futurs programmeurs une interface similaire.

*Attention : Les barrettes sécables doivent être soudées du côté piste et ne pas dépasser sur la partie supérieure du circuit, Si possible câbler cette carte en premier elle servira de guide pour câbler les barrettes sécables sur ER122 et éviter un décalage lors de l'insertion.

ERMES122

Montez le condensateur milfeuïl

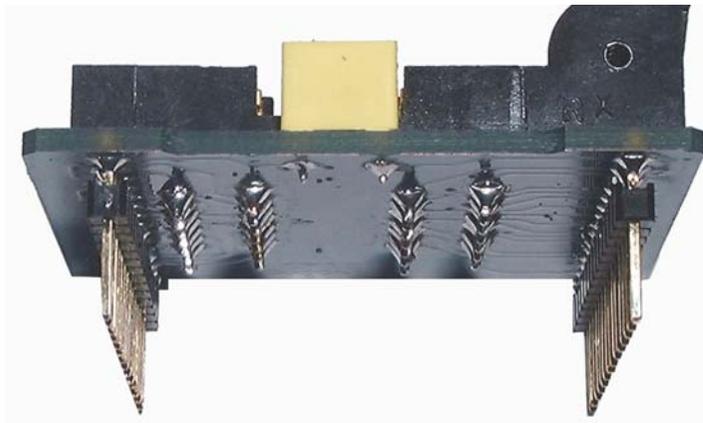
C1, voir photo ci-après : 100nf

Montez le support à insertion nulle:

U11, voir photo ci-dessous : (28 broches)

Montez les barettes sécables mâle coté soudures.

(2x14 broches), voir détail ci-dessous.



Liste des composants

| Désignation : | Qté | Repère | Observation | |
|---------------------------|------------------------------|--------|------------------------|----------------------------|
| Résistance métal 5%..... | 1KR (marron, noir, rouge) | 2 | R9, R10 | |
| Résistance métal 5%..... | 10KR (marron, noir, orange) | 5 | R7, R13, R14, R15, R16 | |
| Résistance métal 5%..... | 470R (jaune, violet, marron) | 1 | R2 | |
| Résistance métal 5%..... | 100KR (marron, noir, jaune) | 2 | R1, R12 | |
| Résistance métal 5%..... | 560R (vert, bleu, marron) | 2 | R11, R17 | |
| Résistance ajustable..... | 1KR | 1 | AJ1 | |
| Cond. Milfeuïl..... | 100nF/63V | 5 | C6, C13, C15, C16, C17 | |
| Cond. Chim. Radial..... | 1uF/25V | 1 | C2, C3, C4, C5, C14 | 25V ou plus suivant appro. |
| Cond. Chim. Radial..... | 10uF/25V | 1 | C18 | 25V ou plus suivant appro. |
| Cond. Chim. Radial..... | 100uF/25V | 1 | C12 | 25V ou plus suivant appro. |
| Diode..... | 1N4007 | 2 | D2, D5 | |
| Diode..... | 1N4148 | 1 | D4 | |
| Diode zener..... | BZX9V | 1 | D3 | |
| Oscillateur à quartz..... | 9,8304Mhz | 1 | U8 | |
| Diode Led 3mm..... | Rouge | 3 | L1, L2, L3 | |
| Transistors..... | 2N2222 | 1 | T1 | |
| Régulateur 5V pos..... | LM7805 | 1 | U7 | |
| Micro contrôleur..... | 68HC908QY2 | 1 | U10 | |
| Circuit intégré..... | MAX232 | 1 | U2 | |

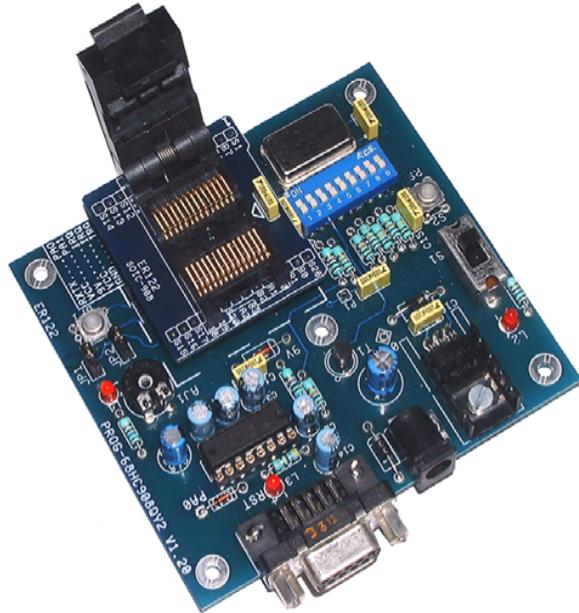
Accessoires de montage :

| | | | |
|-------------------------------------|---------------|---|---------------|
| Supports C.I..... | 16 broches | 2 | Pour U2, U10 |
| Support C.I..... | 8 broches | 1 | Pour U9 |
| Dip switch 9 contacts..... | 18 broches | 1 | S3 |
| Supports pour cavalier..... | 2x2 broches | 2 | JP1, JP2 |
| Supports ER122-2..... | 1x8 broches | 2 | Pour VCC, VSS |
| Interrupteur à glissière 2 pos..... | 6 broches + 2 | 1 | S1 |
| Jack alimentation à souder..... | 2,5 mm | 1 | J1 |
| Prise SUB D à souder..... | 9 broches | 1 | RS1 |
| Radiateur pour régulateur..... | en U | 1 | Pour U7 |
| Vis pour radiateur..... | ø 3 x 10mm | 6 | |
| Ecrou pour radiateur..... | ø 3 mm | 6 | |
| Rondelle éventail..... | ø 3 mm | 6 | |
| Entretoise mâle/femelle..... | 10mm | 5 | |
| Pied caoutchouc..... | | 5 | |

Garantie :

Les Kits ERMES ont été élaborés et testés de façon rigoureuse. Un soin tout particulier est apporté dans le choix des composants et le circuit imprimé est d'une qualité irréprochable. Si toutefois vous deviez rencontrer un problème lors de la réalisation, veuillez avant toute chose vérifier l'implantation des composants (sens et valeur), les soudures, le câblage. Vérifier de plus l'alimentation des circuits intégrés. Si le phénomène persiste, notre service technique est à votre disposition pour vous aider. Envoyez-nous un courrier, accompagné d'une enveloppe timbrée pour la réponse (délai réponse env. une semaine), en nous donnant le maximum d'informations. Nous garantissons le bon fonctionnement des kits ERMES. En cas de problème, ramenez le kit chez votre distributeur. La réparation sera effectuée gratuitement, sauf en cas de mauvais assemblage évident. Nous déclinons toute responsabilité pour tout dommage causé par l'utilisation ou la défectuosité d'un kit ERMES.

ERMES122



| Désignation | Pin µC | Pin S3 | Observation |
|-------------|--------|--------|---------------------------------------|
| RXTX | PA0 | 1 | Liaison vers le PC |
| VCC | PA1 | 2 | |
| 9V | IRQ | 3 | Choix du signal sur IRQ |
| VCC | IRQ | 4 | |
| GND | IRQ | 5 | |
| BP | RST | 6 | Liaison vers le BP S2 pour RST manuel |
| GND | PA4 | 7 | |
| R | OSC | 8 | Oscillateur par résistance |
| 9.8Mhz | OSC | 9 | Oscillateur par quartz |

Programmation d'un 68HC908QY2 :

Utilisation du programmeur :

Vous trouverez le manuel qu'il est souhaitable d'imprimer dans les fichiers : "MC68HC908QY4.pdf", "m68ics08somb_v_1_08 in circuit simulator manuel.pdf", et dans qtqy_addendum_v_1_00.pdf".

Lancer l'exécutable : "ics08qtqyz113b.exe" qui va installer le logiciel sur votre ordinateur.

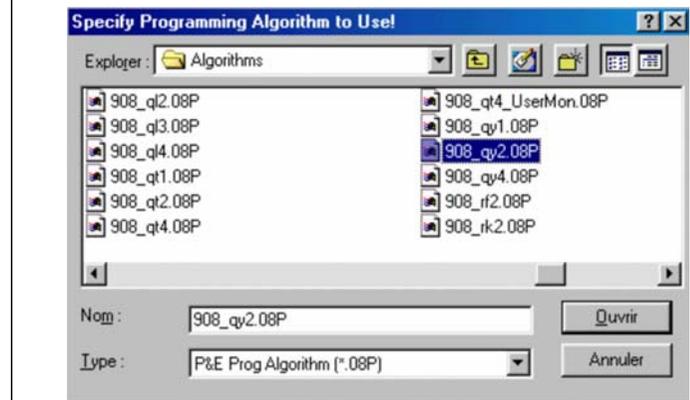
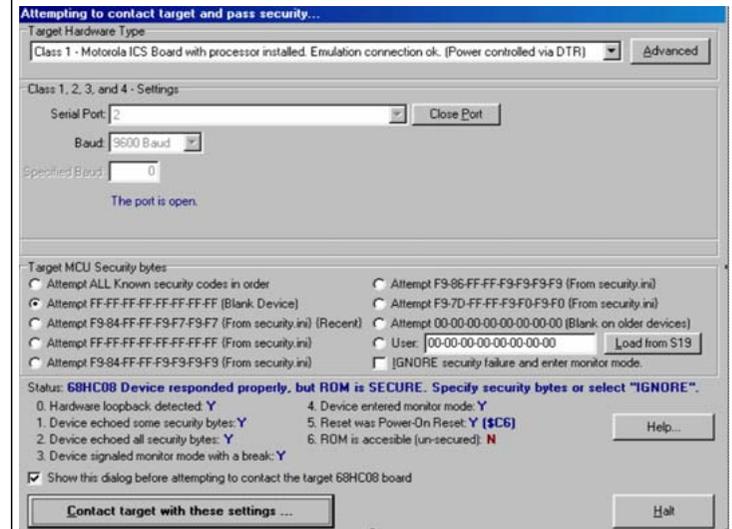
-Sélection des signaux sur S3 pour dialoguer avec le µC.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|----|
| X | X | X | X | X | X | X | X | X | ON |
| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |

- Mettre le µC (attention au sens)
- Alimenter le programmeur, par un bloc secteur 14V (+ sur milieu du Jack).
- Lancer WINIDE32, puis " Environement ", " Ouvrir Projet, DEMO BLINKER "
- Puis appuyer sur F4 pour assembler le code

Appuyer sur le 3° bouton jaune (programmer) puis Faire OK.

On aboutit dans un tableau qui permet la configuration du port série, du mode utilisé et la sélection du mot de passe Il affiche dans la partie status, les différentes étapes de connection. Mettre sous tension la carte. Appuyer sur "Contact Target with these settings"



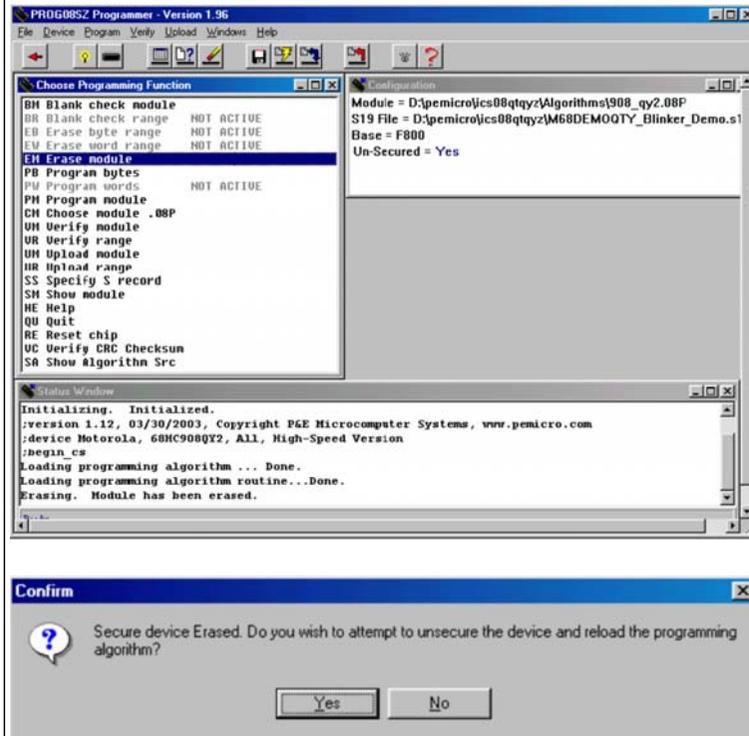
Choisissez l'algorithmme du 908QY2, on voit ici la multitude de µC que ce logiciel peut programmer.

Pour mettre en oeuvre le montage et essayer les logiciels nous allons utiliser un programme de test très complet, qui va nous permettre d'utiliser une partie importante du µC à savoir, les ports d'entrées sorties, le timer et les convertisseurs analogiques numériques. Nous utiliserons pour cela les quelques composants prévus à cet usage sur la carte (Led, bouton poussoir, ajustable). Bienvenu dans le monde du µC, et de la ... réflexion.

Le tableau sur la sérigraphie du circuit imprimé permet la sélection des mini-interrupteurs de S3 sans documentation particulière.

ERMES122

Effacer le µC, s'il n'est pas vierge.

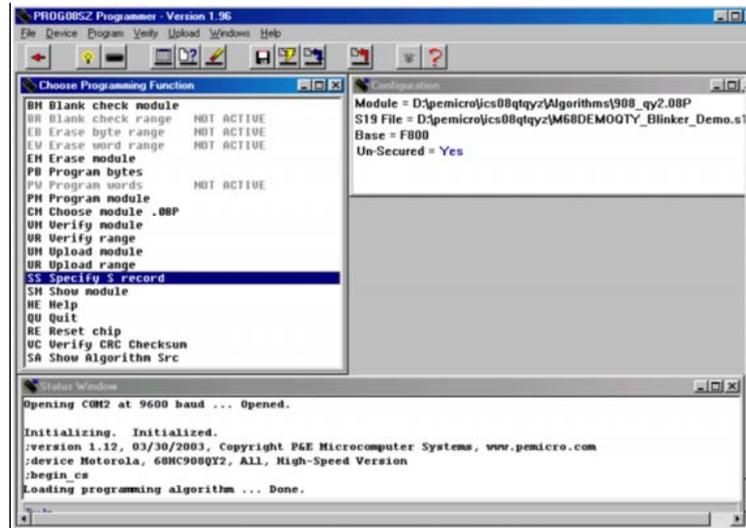


Tapez sur le bouton Yes.

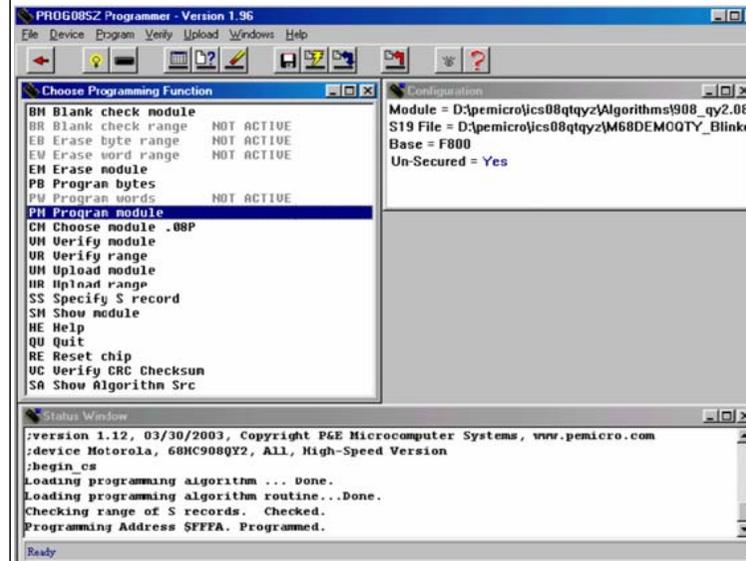
Il faut dans ce cas représenter un code d'accès. (µC vierge ' code = \$FFFF).

Faire un Marche/arrêt de la carte (U alim < 0.1V)
Cliquer sur le BP " Yes "

Choisir le fichier à programmer (SS Specify S Record) (*.S19), penser à changer le nom du fichier à chaque nouveau programme développé car le fichier "S19 File" n'est pas mis automatiquement à jour.



Lancer la programmation (PM).



-Quittez (file - exit).

-Couper l'alimentation du montage.

-Sélection des signaux sur S3 pour lancer le programme.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| X | X | X | X | X | X | X | X | X | X | ON |
| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | - | - |

-Mettre les cavaliers JP1 et JP2 en place

-Mettre sous tension la carte

-Le programme va démarrer

Fonctionnement du programme de test.

-La Led L1 va clignoter

-Toute action sur l'ajustable AJ1 va faire varier la vitesse de clignotement

-Si on appuie sur le bouton poussoir BP1, le fonctionnement change, une action sur l'ajustable AJ1 fera varier la luminosité de la led. Une autre action sur BP1 rétabli le mode initial.

Mode DEBUG :

-Couper l'alimentation du montage

-Sélection des signaux sur S3 pour lancer le programme.

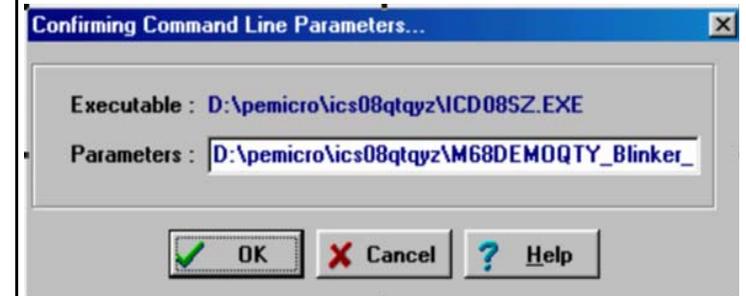
| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| X | X | X | X | X | X | X | X | X | X | ON |
| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | - | - |

-Mettre les cavaliers JP1 et JP2 en place

Lancer WINIDE

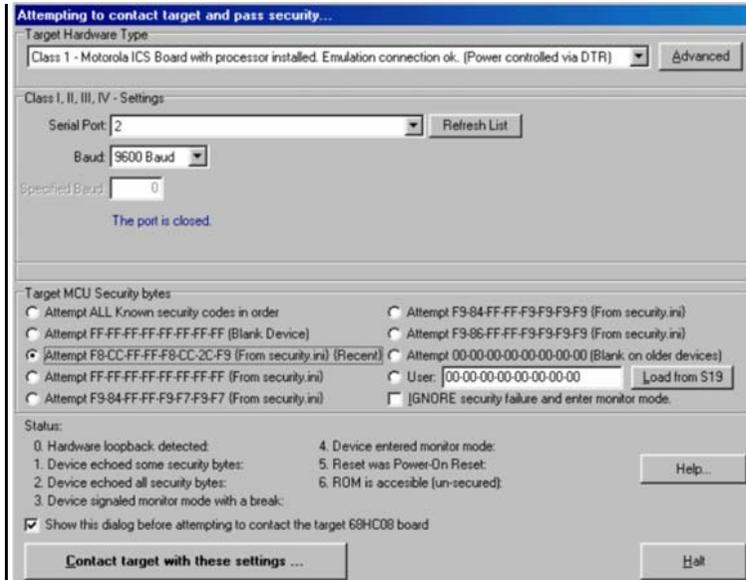
-Mettre sous tension la carte

-Appuyer sur le 4° bouton (In Circuit simulator).



On retrouve le panneau de dialogue avec la carte.

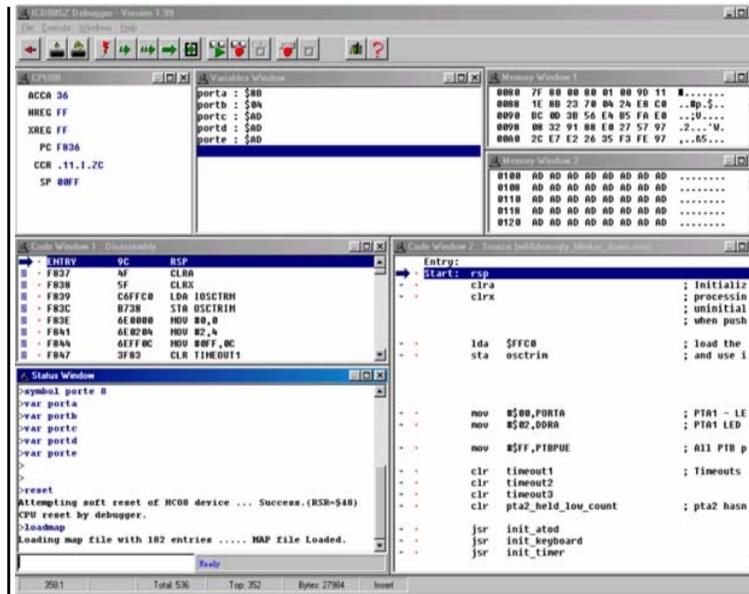
ERMES122



Comme vous avez pu le voir, l'accès au programme est protégé par un mot de passe que l'on doit présenter lors de la programmation, de la simulation ou tout autres actions qui permettent d'avoir un regard sur la mémoire. Ce mot de passe est constitué d'octets du programme, vecteurs etc., il est mémorisé dans un fichier de configuration et présenté lors de l'accès au µC. Le code présenté doit-être associé au logiciel du µC.

Mais, il se peut que vous ayez changé de programme ou de µC et le code ne correspond plus. Dans ce cas une seule solution, cocher la case (Ignore sécurité failure), puis effacer le µC et programmez à nouveau le µC, initialiser à nouveau la carte (Marche/Arrêt).

Le DEBUGGER est activé.



Taper : LOADMAP pour charger la table des symboles (nom des variables)

Puis RESET ; pour initialiser le programme

Puis T pour Trace (mode pas à pas)

Puis appuyer sur F9 pour réactiver la dernière commande saisie (=T).

Les commandes les plus importantes sont (voir aussi HELP)

GOTIL (une étiquette) ' Exécute le programme jusqu'à l'étiquette désignée

MM (Memory Modif)' Modification de la mémoire du µC

MM PORTA 0 '(PORTA à 0)

GO ' Mode RUN

CY 0 ' Remet le compteur de cycle machine à 0

BR (une adresse) Pose un point d'arrêt, le programme stoppe et permet la modification et le contrôle des adresses.

Un click du bouton droit de la souris permet la pose d'un point d'arrêt ou l'exécution d'autres ordres.

Bien sûr toutes les commandes, et elles sont nombreuses, sont décrites dans le logiciel et le manuel.

Quelques informations :

Plusieurs conditions doivent être remplies, pour que le moniteur intégré à la ROM du µC soit accessible. (Voir tableau dans le manuel des configurations des pins). En effet pour éviter que le µC ne rentre tout seul en mode DEBUG, MOTOROLA a imposé un certain nombre de conditions sur l'état des pins du µC. Mais, il est tellement pratique de pouvoir programmer, effacer et "Debugger" le µC sur la carte, qu'on accepte bien volontiers ces quelques contraintes. Pour éviter la paralysie de ces ports, principalement PA0, PA1, PA4, IRQ, il y a des solutions simples. La carte finale pourra avoir 2 fonctions, le fonctionnement normal, puis un fonctionnement ponctuel de programmeur voire d'émulateur.

Il faut donc pour cela, pouvoir changer l'état des Pins dédiées au mode Programmeur sans pour cela générer de court-circuit ou de catastrophe du côté de l'utilisation normale de la carte. Mais en y réfléchissant un peu, il existe quelques solutions simples.

-PA0 est utilisé pour le dialogue avec le PC, mais n'a t'on pas besoin d'une liaison série sur la carte finale ?

-PA1 doit être relié au 5V, mais justement on peut avoir une charge (Led) connectée au 5V

ERMES122

-IRQ=9V pour la programmation, mais si c'est un bouton poussoir qui est relié à la pin IRQ, rien ne l'empêche.

-OSC1 doit recevoir l'horloge extérieure pour être synchrone avec le PC, dans ce cas une résistance peut isoler la charge.

On voit ici, que l'on peut concevoir (au prix de quelques réflexions) un double usage des Pins. Il faut d'autre part prévoir quelques connections pour relier le μ C vers la carte ER122, connections qui pourront par ailleurs servir de point test.

Il reste une autre solution peu gourmande en ressource si l'on veut reprogrammer le μ C soudé sur la carte. Elle consiste à intégrer un peu de code dans le μ C pour auto-effacer le μ C si une action caractéristique se produit sur ses PINs lors de la mise sous tension, par exemple un Bouton Poussoir appuyé. En effet, dans le cas où le μ C est vierge (vecteur RESET à \$FFFF), il suffit de connecter PA0 au PC via la carte ER122 et maintenir IRQ à 0, l'horloge interne est utilisée, pour communiquer avec WINIDE. Il y a dans ce cas une seule Pin de monopolisée (PA0), et encore elle est partageable avec une autre fonction (led reliée au plus ou transistor PNP etc.)

Lire les limitations du DEBUGGER dans " MONO8 DEBUGGING LIMITATION AND TIPS "

Avec un peu d'expérience on comprend la technique d'activation de la ROM et on apprécie les fonctions intégrées qui peuvent par ailleurs être réutilisées dans votre programme.

Cette série de μ C 908 mérite vraiment que l'on fasse un effort de réflexion (juste au départ), car elle intègre une puissance rarement rencontrée chez d'autres constructeurs.