

B-ASIC 3.2

Logon System

Instruction	Description
INKF Encre, Couleur	Modification couleur d'une encre écran Encre=[0..15] Couleur=[0..4095]
INKFRVB Encre, R, V, B	Modification couleur d'une encre écran définie par ses composantes rouge,vert,bleu Encre=[0..15] R/V/B = [0..15]
BORDERP Couleur	Modification couleur du bord Couleur=[0..4095]
BORDERRVB R,V,B	Modification couleur du bord définie par ses composantes rouge, vert, bleu R/V/B = [0..15]
INKS Encre, Couleur	Modification couleur d'une encre sprite Encre=[1..15] Couleur=[0..4095]
INKSRVB Encre, R,V, B	Modification couleur d'une encre sprite définie par ses composantes rouge, vert, bleu Encre=[1..15] R/V/B=[0..15]
SPROFF NumSprite	Désactiver l'affichage d'un sprite NumSprite=[0..15]
SPRON NumSprite	Activer l'affichage d'un sprite NumSprite=[0..15]
SPRZOOM NumSprite, ZoomX, ZoomY	Zoomer un sprite Le sprite doit être activé pour être affiché NumSprite=[0..15] ZoomX/ZoomY=[0..3]
SPRXY NumSprite, PosX, PosY	Positionner un sprite Le sprite doit être activé pour être affiché NumSprite=[0..15] PosX=[] PosY=[]
SPRTDEF NumSprite, Adresse, Duree, DelaiDepart	Permet de définir un chemin pour un sprite Le chemin est défini par des coordonnées x, y définies en ram. Chaque coordonnée est définie sur 2 octets [donc 4 octets par position], avec l'octet de poids faible en 1 ^{er} et l'octet de poids fort en 2 nd . La dernière coordonnée x doit être F0F0 pour indiquer la fin du chemin avant bouclage. La durée indique le temps pendant lequel le sprite doit parcourir le chemin (0 pour toujours), et le délai de départ indique le temps à partir duquel le sprite commencera son chemin lorsque le chemin sera activé (voir instruction SPRTON) NumSprite=[0..15] Adresse=[0..FFFF] Duree=[0..FFFF], DelaiDepart=[0..FFFF] Les temps sont définis par pas de 1/50 ^{ème} de sec.
SPRTON N1,N2,, Nx	Démarrage de 1 ou plusieurs chemins définis par SPRTDEF (simultanément). Le sprite doit être activé pour être affiché (SPRON) Nx=[0..15]
SPRTOFF N1,N2,.....,Nx	Arrêt de 1 ou plusieurs chemins définis par SPRTDEF (simultanément) Nx=[0..15]
SPRSWAP NumSpr1, NumSpr2	Echange le contenu de 2 sprites NumSprX=[0..15]
SPRPLOT NumSprite, PosX, PosY, Encre	Positionne un pixel dans un sprite NumSprite=[0..15] PosX=[0..15] PosY=[0..15] Encre=[0..15] (0 encre transparente)

SPRCOPY NumSpr1, NumSpr2	Copie du contenu du sprite2 dans le sprite 1 NumSprX=[0..15]
SPRTURNX NumSprite	Retourne un sprite horizontalement NumSprite=[0..15]
SPRTURNY NumSprite	Retourne un sprite verticalement NumSprite=[0..15]
SPRFILL NumSprite, Encre	Remplir un sprite avec une encre NumSprite=[0..15] Valeur=[0..15]
SPLIT Ligne, Offset	Rupture Ligne =[0..199] , Offset=[0..FFFF] Cette instruction doit évoluer. Ne plus utiliser
RETARDX Decalage	Gestion de l'avance retard horizontale Décalage=[0..15] Cette instruction doit évoluer. Ne plus utiliser
RETARDY Decalage	Gestion de l'avance retard verticale Decalage=[0..7] Cette instruction doit évoluer. Ne plus utiliser
MASQUE Type	Activation du masquage d'avance retard horiz. Type=0 ou 1 (inactif, actif) Cette instruction doit évoluer. Ne plus utiliser
SPRSAVE [NoSprite],NoBlock,Filename	Sauvegarde d'un sprite ou d'un bloc de sprites Le B-Asic dispose d'une banque de 64 sprites en plus des 16 déjà en ram asic. Ils sont gérés comme 4 blocs de 16 sprites supplémentaires. Si le Numéro de sprite est précisé, alors un seul sprite est sauvegardé, sinon le bloc de 16 sprites est sauvegardé. Le bloc 0 représente les sprites réels. NoSprite=[0..15] champ facultatif NoBlock=[0..4], Filename=Nom du fichier sauvé
SPRLOAD [NoSprite],NoBlock, Filename	Lecture d'un sprite ou d'un bloc de sprites Le B-Asic dispose d'une banque de 64 sprites en plus des 16 déjà en ram asic. Ils sont gérés comme 4 blocs de 16 sprites supplémentaires. Si le Numéro de sprite est précisé, alors un seul sprite est chargé, sinon le bloc de 16 sprites est chargé. Le bloc 0 représente les sprites réels. NoSprite=[0..15] champ facultatif NoBlock=[0..4], Filename=Nom du fichier à lire
SPRTORAM Sprite, SpriteBank	Transfère un sprite réel vers la banque Sprite=[0..15] SpriteBank=[0..63]
RAMTOSPR SpriteBank, Sprite	Transfère un sprite banque vers un sprite réel SpriteBank=[0..63] Sprite=[0..15]
CATCH NumSprite, PosX, PosY	Capture un sprite à partir d'un écran en mode 0 NumSprite=[0..15] PosX=[0..144] PosY=[0..184]
INKCOPY	Copie la palette de l'écran vers la palette sprites
SPRLINK, NumSpr1,NumSpr2, Xrel,Yrel	Attache un sprite à un autre sprite Le sprite 2 est attaché au sprite 1 avec des coordonnées relative. Si sprite 1 est en position X1,Y1, alors la position du sprite 2 est calculé comme suit X2=X1+Xrel et Y2=Y1+Yrel
SPRUNLINK NumSpr1, NumSpr2	Détache un sprite d'un autre sprite.
ERALINK	Efface tous les attachements.

POKEASIC Adresse, val1,, valn	Poke de 1 ou plusieurs valeurs dans les registres de l'asic. Adresse=[4000..7FFF] ValX=[0..FF] Réservé aux utilisateurs expérimentés
POKEVR Adresse, val1,, valn	Poke de 1 ou plusieurs valeurs dans la Ram Video 2 Adresse=[4000..7FFF] ValX=[0..FF]
DEFORGXY, Sprite, X, Y	Définir la position absolue à partir de laquelle la position d'un sprite est définie. Si un sprite est positionné en 10, 10 et que l'instruction défini une position d'origine 30, 40, alors le sprite sera affiché réellement en 40, 50 Sprite = [0..15] X= [] Y=[]
DEFORGSPR, Sprite, SpriteAttache	Définir la position absolue d'un sprite par rapport à celle d'un autre sprite. Si un sprite est positionné en 10,10 et que le sprite d'attache est en 30, 40, alors le sprite attaché sera en 40, 50 Sprite et SpriteAttache = [0..15]