

## Sprite demo: Breakout

[break.zip](#)

*This module is part of the original MMBasic library. It is reproduced here with kind permission of Hugh Buckle and Geoff Graham. Be aware it may reference functionality which has changed or is deprecated in the latest versions of MMBasic.*

Colour Maximite V4.0 Demos from Fabrice Muller, France

**Arcade:** Arcade game where you have to shoot enemies until you die 3 times. It shows how to use multi-sprite enemies, audio modules as sound effects and ADC input for moving your ship Right/Left with a potentiometer + a Button to fire.

### BREAKTST.BAS:

```
'Breakout Game demo for Colour Maximite v4.0
'Fabrice Muller
'2012
'Connect a potentiometer to Pin 2
'
'
'      Pin 2
'      |
'      |
'      / \
' +3v3 -----| o |----- GND
'      \___/
'
'      10Kohm Potentiometer
'
'
'All Arrays start at 0
Option base 0
'Setup the Pin 2 as ADC input
SetPin 2,1
'array for a 15 x 8 bricks wall
Dim Brick(14,7)
'Load the Sprites
Drive "b:"
Sprite Load "Break.spr"
RackSize = 32
'Put the screen in 240 x 216 pixels
Mode 4
Cls
Print @(100 , 80) "Press a key to start"
Do While Inkey$ = "" : Loop
Cls
Lives = 3
Level = 1
'Maximum number of level you created
Max_Level = 4
```

```
'Load level 1 at start
Load_Level(1)
Racket = Pot_Read(2)
Randomize Timer
ballx = Rnd * (MM.HRes - 40) + 20
bally = 90
dx = 0
Do While dx = 0
  dx = Rnd(5) - 2
Loop
dy = -2
'Draw the Racket and the Ball
Sprite On 1,racket,200
Sprite On 2,racket + 16,200
Sprite On 3,ballx,bally
Sprite_Flip = 0
'Main loop
Do While 1
  'Get racket position
  Racket = Pot_Read(2)
  RackRight = Racket + RackSize
  'Move the Racket and the Ball
  If Sprite_Flip = 1 Then
    Sprite off 1
    Sprite off 2
    Sprite off 3
    Sprite on 1,racket,200
    Sprite on 2,racket + 16,200
    Sprite on 3,ballx,bally
    Sprite_Flip = 0
  Else
    Sprite off 3
    Sprite off 2
    Sprite off 1
    Sprite on 3,ballx,bally
    Sprite on 2,racket + 16,200
    Sprite on 1,racket,200
    Sprite_Flip = 1
  EndIf
  ballx = ballx + dx
  bally = bally + dy
  If bally >= 190 Then Racket_Test
  If bally <= 3 Then dy = Abs(dy)
  If ballx > (MM.HRes - 7) Then dx = -dx
  If ballx < 1 Then dx = Abs(dx)
  ShootY = Int(bally / 8)
  If (ShootY >= 0) And (ShootY <= 7) Then
    ShootX = Int(Ballx / 16)
    If ShootX <= 14 Then Test_the_Bricks
  EndIf
```

```
Pause 20
Loop

Sub Racket_Test
  If (ballx < (Racket - 7)) Or (ballx > RackRight) Then
    Lives = Lives - 1
    If Lives = 0 Then
      Print @(100 , 90) "GAME OVER !"
      Sound 500,500
      Print @(60 , 110) "Press a key to restart"
      Do While Inkey$ = "" : Loop
      Run
    Else
      Sprite Off 3
      Bally = 90
      Ballx = Int(Rnd * (MM.HRes - 30) + 10)
      Sprite On 3,Ballx,Bally
      Exit Sub
    EndIf
  EndIf
  Sound 700,10
  dy = -dy
  If (ballx >= Racket) And (ballx >= Racket + Int(RackSize / 4)) Then
    If dx > 0 Then dx = -3
  EndIf
  If (Ballx >= Racket + Int(RackSize/4)) And (Ballx <= Racket +
Int(RackSize/2)) Then
    If dx > 0 Then dx = -2
  EndIf
  If (Ballx >= (Racket + Int(RackSize / 2)+Int(RackSize / 4))) And (Ballx <=
RackRight) Then
    If dx < 0 Then dx = 2
  EndIf
End Sub

Sub Test_the_Bricks
  If ShootX < 0 Then ShootX = 0
  If brick(ShootX,ShootY) = 1 Then
    brick(ShootX,ShootY) = 0
    Sound 1000,20
    Sprite Off 3
    Line (ShootX*16,ShootY*8)-((ShootX*16)+15,(ShootY*8)+7),0,bf
    Sprite On 3,ballx,bally
    score = score + 10
    Print @(0 , 200) score;" "
    If (score Mod 1200) = 0 Then
      'Load the next level
      Level = Level + 1
      If Level > Max_Level Then
        Print @(100 , 80) "You win !"
        Print @(70 , 100) "Press a key to restart"
```

```
    Do While Inkey$ = "" : Loop
    Run
  EndIf
  Sprite Off 1
  Sprite Off 2
  Sprite Off 3
  Bally = 90
  Ballx = Int(Rnd * (MM.HRes - 30) + 10)
  Cls
  Load_Level(Level)
  Sprite On 1,racket,200
  Sprite On 2,racket + 16,200
  Sprite On 3,Ballx,Bally
  Sprite_Flip = 0
  dy = Abs(dy)
  Exit Sub
EndIf
If dy > 0 Then
  dy = -dy
Else
  dy = Abs(dy)
EndIf
EndIf
End Sub

Sub Load_Level(Number)
  'Here we load a bmp and it's related
  'data array
  Local FileBMP$,FileBrick$
  Local Input_Line$
  Local a,b

  FileBMP$ = "brick"
  If Number < 10 Then FileBMP$ = FileBMP$ + "0" + Str$(Number) + ".bmp"
  If Number >= 10 Then FileBMP$ = FileBMP$ + Str$(Number) + ".bmp"
  FileBrick$ = "brick"
  If Number < 10 Then FileBrick$ = FileBrick$ + "0" + Str$(Number) + ".lvl"
  If Number >= 10 Then FileBrick$ = FileBrick$ + Str$(Number) + ".lvl"
  'We load the Wall BMP
  LoadBMP FileBMP$,0,0
  'We load the data array for this level
  Open FileBrick$ For input As #1
  For a = 0 To 7
    Line Input #1,Input_Line$
    For b = 0 To 14
      If Mid$(Input_line$,b+1,1) <> " " Then
        Brick(b,a) = 1
      Else
        Brick(b,a) = 0
      EndIf
    Next b
  Next a
End Sub
```

```
Next b
Next a
Close #1
End Sub

Function Pot_Read(ADC_Pin)
'Read the potentiometer value for place
'the X coordinate from the Ship
'ADC_Pin is the pin number we have define for ADC input
Local CalcPot , tmppos
CalcPot = 3 / MM.HRes
tmppos = Int((Pin(ADC_Pin) - 0.1) / CalcPot)
If tmppos > (MM.HRes - 32) Then tmppos = MM.HRes - 32
Pot_Read = tmppos
End Function
```

From:

<https://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:

[https://fruitoftheshed.com/wiki/doku.php?id=mmbasic\\_original:sprite\\_demo\\_breakout](https://fruitoftheshed.com/wiki/doku.php?id=mmbasic_original:sprite_demo_breakout)

Last update: **2024/01/19 09:39**

