Pong Game

1.0

ARQUITECTURA DE COMPUTADORES Y ENSAMBLADORES 1 SECCIÓN A

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 2

File Documentation

2.1 alphabet.h File Reference

Arrays de bytes de los numeros (0,1,2,3,4) a mostrar en la matriz de led's.

Variables

- byte **zero0** [ ] = { 0x7c, 0xc6, 0xce, 0xde, 0xf6, 0xc6, 0x7c, 0x00 }
- byte **one1** [ ] = { 0x38, 0x78, 0x18, 0x18, 0x18, 0x18, 0x7e, 0x00 }
- byte **two2** [ ] = { 0x3c, 0x66, 0x06, 0x1c, 0x30, 0x66, 0x7e, 0x00 }
- byte **three3** [ ] = { 0x3c, 0x66, 0x06, 0x1c, 0x06, 0x66, 0x3c, 0x00 }
- byte **four4** [ ] = { 0x1c, 0x3c, 0x6c, 0xcc, 0xfe, 0x0c, 0x1e, 0x00 }
- byte **zero** [ ] = { 0x00, 0x3E, 0x7F, 0x59, 0x4D, 0x7F, 0x3E, 0x00 }
- byte **one** [ ] = { 0x00, 0x42, 0x43, 0x7F, 0x7F, 0x40, 0x40, 0x00 }
- byte **two** [ ] = { 0x00, 0x62, 0x73, 0x59, 0x49, 0x6F, 0x66, 0x00 }
- byte **three** [ ] = { 0x00, 0x22, 0x63, 0x49, 0x49, 0x7F, 0x36, 0x00 }
- byte **four** [ ] = { 0x18, 0x1C, 0x16, 0x53, 0x7F, 0x7F, 0x50, 0x00 }

2.1.1 Detailed Description

Arrays de bytes de los numeros (0,1,2,3,4) a mostrar en la matriz de led's.

Version

1.0

Date

19/08/2020

Author

ARQUI 1, GRUPO 8

2.2 alphabetBanner.h File Reference

Arrays de bytes de letras, número y signos a mostrar en la matriz de led's en el banner.

Variables

- byte **S1** [ ] = {B00010010,B00001100,B00000000,B01000100,B01111110,B01000000,B00000000,B00010000 }
- byte **S2** [ ] = {B00001100,B00000000,B01000100,B01111110,B01000000,B00000000,B00010000,B00010000 }
- byte **S3** [ ] = {B00000000,B01000100,B01111110,B01000000,B00000000,B00010000,B00010000,B00010000 }
- byte **S4** [ ] = {B01000100,B01111110,B01000000,B00000000,B00010000,B00010000,B00010000,B00010000 }
- byte **S5** [ ] = {B01111110,B01000000,B00000000,B00010000,B00010000,B00010000,B00010000,B00000000 }
- byte **S6** [ ] = {B01000000,B00000000,B00010000,B00010000,B00010000,B00010000,B00000000,B01111110 }
- byte **S7** [ ] = {B00000000,B00010000,B00010000,B00010000,B00010000,B00000000,B01111110,B01000010 }
- byte **S8** [ ] = {B00010000,B00010000,B00010000,B00010000,B00000000,B01111110,B01000010,B01010010 }
- byte **S9** [ ] = {B00010000,B00010000,B00010000,B00000000,B01111110,B01000010,B01010010,B01010010 }
- byte **S10** [ ] = {B00010000,B00010000,B00000000,B01111110,B01000010,B01010010,B01010010,B01110010 }
- byte **S11** [ ] = {B00010000,B00000000,B01111110,B01000010,B01010010,B01010010,B01110010,B00000000 }
- byte **S12** [ ] = {B00000000,B01111110,B01000010,B01010010,B01010010,B01110010,B00000000,B01111110 }
- byte **S13** [ ] = {B01111110,B01000010,B01010010,B01010010,B01110010,B00000000,B01111110,B00010010 }
- byte **S14** [ ] = {B01000010,B01010010,B01010010,B01110010,B00000000,B01111110,B00010010,B00010010 }
- byte **S15** [ ] = {B01010010,B01010010,B01110010,B00000000,B01111110,B00010010,B00010010,B01101110 }
- byte **S16** [ ] = {B01010010,B01110010,B00000000,B01111110,B00010010,B00010010,B01101110,B00000000 }
- byte **S17** [ ] = {B01110010,B00000000,B01111110,B00010010,B00010010,B01101110,B00000000,B00011110 }
- byte **S18** [ ] = {B00000000,B01111110,B00010010,B00010010,B01101110,B00000000,B00011110,B00100000 }
- byte **S19** [ ] = {B01111110,B00010010,B00010010,B01101110,B00000000,B00011110,B00100000,B01000000 }
- byte **S20** [ ] = {B00010010,B00010010,B01101110,B00000000,B00011110,B00100000,B01000000,B01000000 }
- byte **S21** [ ] = {B00010010,B01101110,B00000000,B00011110,B00100000,B01000000,B01000000,B00100000 }
- byte **S22** [ ] = {B01101110,B00000000,B00011110,B00100000,B01000000,B01000000,B00100000,B00011110 }
- byte **S23** [ ] = {B00000000,B00011110,B00100000,B01000000,B01000000,B00100000,B00011110,B00000000 }
- byte **S24** [ ] = {B00011110,B00100000,B01000000,B01000000,B00100000,B00011110,B00000000,B01111110 }

- byte **S25** [ ] = {B00100000,B01000000,B01000000,B00100000,B00011110,B00000000,B01111110,B00010010 }
- byte **S26** [ ] = {B01000000,B01000000,B00100000,B00011110,B00000000,B01111110,B00010010,B00010010 }
- byte **S27** [ ] = {B01000000,B00100000,B00011110,B00000000,B01111110,B00010010,B00010010,B00001100 }
- byte **S28** [ ] = {B00100000,B00011110,B00000000,B01111110,B00010010,B00010010,B00001100,B00000000 }
- byte **S29** [ ] = {B00011110,B00000000,B01111110,B00010010,B00010010,B00001100,B00000000,B00111100 }
- byte **S30** [ ] = {B00000000,B01111110,B00010010,B00010010,B00001100,B00000000,B00111100,B01000010 }
- byte **S31** [ ] = {B01111110,B00010010,B00010010,B00001100,B00000000,B00111100,B01000010,B01000010 }
- byte **S32** [ ] = {B00010010,B00010010,B00001100,B00000000,B00111100,B01000010,B01000010,B01000010 }
- byte **S33** [ ] = {B00010010,B00001100,B00000000,B00111100,B01000010,B01000010,B01000010,B01000010 }
- byte **S34** [ ] = {B00001100,B00000000,B00111100,B01000010,B01000010,B01000010,B01000010,B00111100 }
- byte **S35** [ ] = {B00000000,B00111100,B01000010,B01000010,B01000010,B01000010,B00111100,B00000000 }
- byte **S36** [ ] = {B00111100,B01000010,B01000010,B01000010,B01000010,B00111100,B00000000,B00000000 }
- byte **S37** [ ] = {B01000010,B01000010,B01000010,B01000010,B00111100,B00000000,B00000000,B00000000 }
- byte **S38** [ ] = {B01000010,B01000010,B01000010,B00111100,B00000000,B00000000,B00000000,B00110100 }
- byte **S39** [ ] = {B01000010,B01000010,B00111100,B00000000,B00000000,B00000000,B00110100,B01001010 }
- byte **S40** [ ] = {B01000010,B00111100,B00000000,B00000000,B00000000,B00110100,B01001010,B01001010 }
- byte **S41** [ ] = {B00111100,B00000000,B00000000,B00000000,B00110100,B01001010,B01001010,B01001010 }
- byte **S42** [ ] = {B00000000,B00000000,B00000000,B00110100,B01001010,B01001010,B01001010,B00110100 }
- byte **S43** [ ] = {B00000000,B00000000,B00110100,B01001010,B01001010,B01001010,B00110100,B00000000 }
- byte **S44** [ ] = {B00000000,B00110100,B01001010,B01001010,B01001010,B00110100,B00000000,B00010000 }
- byte **S45** [ ] = {B00110100,B01001010,B01001010,B01001010,B00110100,B00000000,B00010000,B00010000 }
- byte **S46** [ ] = {B01001010,B01001010,B01001010,B00110100,B00000000,B00010000,B00010000,B00010000 }
- byte **S47** [ ] = {B01001010,B01001010,B00110100,B00000000,B00010000,B00010000,B00010000,B00010000 }
- byte **S48** [ ] = {B01001010,B00110100,B00000000,B00010000,B00010000,B00010000,B00010000,B00000000 }
- byte **S49** [ ] = {B00110100,B00000000,B00010000,B00010000,B00010000,B00010000,B00000000,B01001110 }
- byte **S50** [ ] = {B00000000,B00010000,B00010000,B00010000,B00010000,B00000000,B01001110,B01001010 }
- byte **S51** [ ] = {B00010000,B00010000,B00010000,B00010000,B00000000,B01001110,B01001010,B01001010 }
- byte **S52** [ ] = {B00010000,B00010000,B00010000,B00000000,B01001110,B01001010,B01001010,B01111010 }

- byte **S53** [ ] = {B00010000,B00010000,B00000000,B01001110,B01001010,B01001010,B01111010,B00000000 }
- byte **S54** [ ] = {B00010000,B00000000,B01001110,B01001010,B01001010,B01111010,B00000000,B01111110 }
- byte **S55** [ ] = {B00000000,B01001110,B01001010,B01001010,B01111010,B00000000,B01111110,B01001010 }
- byte **S56** [ ] = {B01001110,B01001010,B01001010,B01111010,B00000000,B01111110,B01001010,B01001010 }
- byte **S57** [ ] = {B01001010,B01001010,B01111010,B00000000,B01111110,B01001010,B01001010,B01000010 }
- byte **S58** [ ] = {B01001010,B01111010,B00000000,B01111110,B01001010,B01001010,B01000010,B00000000 }
- byte **S59** [ ] = {B01111010,B00000000,B01111110,B01001010,B01001010,B01000010,B00000000,B01111110 }
- byte **S60** [ ] = {B00000000,B01111110,B01001010,B01001010,B01000010,B00000000,B01111110,B01000010 }
- byte **S61** [ ] = {B01111110,B01001010,B01001010,B01000010,B00000000,B01111110,B01000010,B01000010 }
- byte **S62** [ ] = {B01001010,B01001010,B01000010,B00000000,B01111110,B01000010,B01000010,B01000010 }
- byte **S63** [ ] = {B01001010,B01000010,B00000000,B01111110,B01000010,B01000010,B01000010,B00000000 }
- byte **S64** [ ] = {B01000010,B00000000,B01111110,B01000010,B01000010,B01000010,B00000000,B01111110 }
- byte **S65** [ ] = {B00000000,B01111110,B01000010,B01000010,B01000010,B00000000,B01111110,B01000010 }
- byte **S66** [ ] = {B01111110,B01000010,B01000010,B01000010,B00000000,B01111110,B01000010,B01000010 }
- byte **S67** [ ] = {B01000010,B01000010,B01000010,B00000000,B01111110,B01000010,B01000010,B01000010 }
- byte **S68** [ ] = {B01000010,B01000010,B00000000,B01111110,B01000010,B01000010,B01000010,B00000000 }
- byte **S69** [ ] = {B01000010,B00000000,B01111110,B01000010,B01000010,B01000010,B00000000,B01000010 }
- byte **S70** [ ] = {B00000000,B01111110,B01000010,B01000010,B01000010,B00000000,B01000010,B01111110 }
- byte **S71** [ ] = {B01111110,B01000010,B01000010,B01000010,B00000000,B01000010,B01111110,B01000010 }
- byte **S72** [ ] = {B01000010,B01000010,B01000010,B00000000,B01000010,B01111110,B01000010,B00000000 }
- byte **S73** [ ] = {B01000010,B01000010,B00000000,B01000010,B01111110,B01000010,B00000000,B00111100 }
- byte **S74** [ ] = {B01000010,B00000000,B01000010,B01111110,B01000010,B00000000,B00111100,B01000010 }
- byte **S75** [ ] = {B00000000,B01000010,B01111110,B01000010,B00000000,B00111100,B01000010,B01000010 }
- byte **S76** [ ] = {B01000010,B01111110,B01000010,B00000000,B00111100,B01000010,B01000010,B01000010 }
- byte **S77** [ ] = {B01111110,B01000010,B00000000,B00111100,B01000010,B01000010,B01000010,B01000010 }
- byte **S78** [ ] = {B01000010,B00000000,B00111100,B01000010,B01000010,B01000010,B01000010,B00111100 }
- byte **S79** [ ] = {B00000000,B00111100,B01000010,B01000010,B01000010,B01000010,B00111100,B00000000 }
- byte **S80** [ ] = {B00111100,B01000010,B01000010,B01000010,B01000010,B00111100,B00000000,B00111110 }

- byte **S81** [ ] = {B01000010,B01000010,B01000010,B01000010,B00111100,B00000000,B00111110,B00000100 }
- byte **S82** [ ] = {B01000010,B01000010,B01000010,B00111100,B00000000,B00111110,B00000100,B00001000 }
- byte **S83** [ ] = {B01000010,B01000010,B00111100,B00000000,B00111110,B00000100,B00001000,B00010000 }
- byte **S84** [ ] = {B01000010,B00111100,B00000000,B00111110,B00000100,B00001000,B00010000,B00111110 }
- byte **S85** [ ] = {B00111100,B00000000,B00111110,B00000100,B00001000,B00010000,B00111110,B00000000 }
- byte **S86** [ ] = {B00000000,B00111110,B00000100,B00001000,B00010000,B00111110,B00000000,B00000000 }
- byte **S87** [ ] = {B00111110,B00000100,B00001000,B00010000,B00111110,B00000000,B00000000,B00000000 }
- byte **S88** [ ] = {B00000100,B00001000,B00010000,B00111110,B00000000,B00000000,B00000000,B01111100 }
- byte **S89** [ ] = {B00001000,B00010000,B00111110,B00000000,B00000000,B00000000,B01111100,B00010010 }
- byte **S90** [ ] = {B00010000,B00111110,B00000000,B00000000,B00000000,B01111100,B00010010,B00010010 }
- byte **S91** [ ] = {B00111110,B00000000,B00000000,B00000000,B01111100,B00010010,B00010010,B01111100 }
- byte **S92** [ ] = {B00000000,B00000000,B00000000,B01111100,B00010010,B00010010,B01111100,B00000000 }
- byte **S93** [ ] = {B00000000,B00000000,B01111100,B00010010,B00010010,B01111100,B00000000,B00000000 }
- byte **S94** [ ] = {B00000000,B01111100,B00010010,B00010010,B01111100,B00000000,B00000000,B00000000 }
- byte **S95** [ ] = {B01111100,B00010010,B00010010,B01111100,B00000000,B00000000,B00000000,B00000010 }
- byte **S96** [ ] = {B00010010,B00010010,B01111100,B00000000,B00000000,B00000000,B00000010,B00000010 }
- byte **S97** [ ] = {B00010010,B01111100,B00000000,B00000000,B00000000,B00000010,B00000010,B01111110 }
- byte **S98** [ ] = {B01111100,B00000000,B00000000,B00000000,B00000010,B00000010,B01111110,B00000010 }
- byte **S99** [ ] = {B00000000,B00000000,B00000000,B00000010,B00000010,B01111110,B00000010,B00000010 }
- byte **S100** [ ] = {B00000000,B00000000,B00000010,B00000010,B01111110,B00000010,B00000010,B00000000 }
- byte **S101** [ ] = {B00000000,B00000010,B00000010,B01111110,B00000010,B00000010,B00000000,B01111110 }
- byte **S102** [ ] = {B00000010,B00000010,B01111110,B00000010,B00000010,B00000000,B01111110,B00010010 }
- byte **S103** [ ] = {B00000010,B01111110,B00000010,B00000010,B00000000,B01111110,B00010010,B00010010 }
- byte **S104** [ ] = {B01111110,B00000010,B00000010,B00000000,B01111110,B00010010,B00000010,B00001100 }
- byte **S105** [ ] = {B00000010,B00000010,B00000000,B01111110,B00010010,B00010010,B00001100,B00000000 }
- byte **S106** [ ] = {B00000010,B00000000,B01111110,B00010010,B00010010,B00001100,B00000000,B01000100 }
- byte **S107** [ ] = {B00000000,B01111110,B00010010,B00010010,B00001100,B00000000,B01000100,B01111110 }
- byte **S108** [ ] = {B01111110,B00010010,B00010010,B00001100,B00000000,B01000100,B01111110,B01000000 }

- byte **S109** [ ] = {B00010010,B00010010,B00001100,B00000000,B01000100,B01111110,B01000000,B00000000 }
- byte **T1** [8] = { B00000000, B11111011, B00100010, B00100010, B00100011, B00100010, B00100010, B00000000}
- byte **T2** [8] = { B00000000, B11110111, B01000100, B01000100, B01000111, B01000100, B01000100, B00000000}
- byte **T3** [8] = { B00000000, B11101110, B10001001, B10001001, B10001100, B10001000, B10001000, B00000000}
- byte **T4** [8] = { B00000000, B11011100, B00010010, B00010010, B00011100, B00010000, B00010000, B00000000}
- byte **T5** [8] = { B00000000, B10111000, B00100101, B00100100, B00111000, B00100000, B00100001, B00000000}
- byte **T6** [8] = { B00000000, B01110001, B01001011, B01001001, B01110001, B01000001, B01000011, B00000000}
- byte **T7** [8] = { B00000000, B11100010, B10010110, B10010010, B11100010, B10000010, B10000111, B00000000}
- byte **T8** [8] = { B00000000, B11000100, B00101100, B00100100, B11000100, B00000100, B00001110, B00000000}
- byte **T9** [8] = { B00000000, B10001000, B01011000, B01001000, B10001001, B00001000, B00011100, B00000000}
- byte **T10** [8] = { B00000000, B00010000, B10110000, B10010000, B00010011, B00010000, B00111000, B00000000}
- byte **T11** [8] = { B00000000, B00100000, B01100000, B00100000, B00100111, B00100000, B01110000, B00000000}
- byte **T12** [8] = { B00000000, B01000000, B11000000, B01000000, B01001111, B01000000, B11100000, B00000000}
- byte **T13** [8] = { B00000000, B10000000, B10000000, B10000000, B10011110, B10000000, B11000000, B00000000}
- byte **T14** [8] = { B00000000, B00000001, B00000001, B00000001, B00111101, B00000001, B10000001, B00000000}
- byte **T15** [8] = { B00000000, B00000011, B00000010, B00000010, B01111010, B00000010, B00000011, B00000000}
- byte **T16** [8] = { B00000000, B00000111, B00000100, B00000100, B11110101, B00000100, B00000111, B00000000}
- byte **T17** [8] = { B00000000, B00001111, B00001000, B00001000, B11101011, B00001000, B00001111, B00000000}
- byte **T18** [8] = { B00000000, B00011111, B00010000, B00010000, B11010111, B00010001, B00011111, B00000000}
- byte **T19** [8] = { B00000000, B00111110, B00100000, B00100000, B10101110, B00100010, B00111110, B00000000}
- byte **T20** [8] = { B00000000, B01111101, B01000001, B01000001, B01011101, B01000101, B01111101, B00000000}
- byte **T21** [8] = { B00000000, B11111011, B10000010, B10000010, B10111011, B10001010, B11111010, B00000000}
- byte **T22** [8] = { B00000000, B11110111, B00000100, B00000100, B01110111, B00010100, B11110100, B00000000}
- byte **T23** [8] = { B00000000, B11101111, B00001001, B00001001, B11101110, B00101001, B11101001, B00000000}
- byte **T24** [8] = { B00000000, B11011110, B00010010, B00010010, B11011100, B01010010, B11010010, B00000000}
- byte **T25** [8] = { B00000000, B10111101, B00100101, B00100101, B10111001, B10100100, B10100100, B00000000}
- byte **T26** [8] = { B00000000, B01111010, B01001010, B01001010, B01110010, B01001001, B01001000, B00000000}
- byte **T27** [8] = { B00000000, B11110100, B10010100, B10010100, B11100100, B10010010, B10010001, B00000000}

- byte **T28** [8] = { B00000000, B11101000, B00101000, B00101000, B11001000, B00100100, B00100011, B00000000}
- byte **T29** [8] = { B00000000, B11010000, B01010000, B01010000, B10010000, B01001001, B01000110, B00000000}
- byte **T30** [8] = { B00000000, B10100001, B10100001, B10100001, B00100001, B10010010, B10001100, B00000000}
- byte **T31** [8] = { B00000000, B01000010, B01000010, B01000010, B01000010, B00100100, B00011000, B00000000}
- byte **T32** [8] = { B00000000, B10000101, B10000101, B10000101, B10000101, B01001001, B00110001, B00000000}
- byte **T33** [8] = { B00000000, B00001011, B00001010, B00001010, B00001011, B10010010, B01100010, B00000000}
- byte **T34** [8] = { B00000000, B00010111, B00010100, B00010100, B00010111, B00100100, B11000100, B00000000}
- byte **T35** [8] = { B00000000, B00101110, B00101001, B00101001, B00101110, B01001000, B10001000, B00000000}
- byte **T36** [8] = { B00000000, B01011100, B01010010, B01010010, B01011100, B10010000, B00010000, B00000000}
- byte **T37** [8] = { B00000000, B10111000, B10100101, B10100101, B10111001, B00100001, B00100000, B00000000}
- byte **T38** [8] = { B00000000, B01110001, B01001010, B01001010, B01110010, B01000010, B01000001, B00000000}
- byte **T39** [8] = { B00000000, B11100011, B10010100, B10010100, B11100100, B10000100, B10000011, B00000000}
- byte **T40** [8] = { B00000000, B11000111, B00101000, B00101000, B11001000, B00001000, B00000111, B00000000}
- byte **T41** [8] = { B00000000, B10001111, B01010000, B01010000, B10010000, B00010000, B00001111, B00000000}
- byte **T42** [8] = { B00000000, B00011110, B10100001, B10100001, B00100001, B00100001, B00011110, B00000000}
- byte **T43** [8] = { B00000000, B00111100, B01000010, B01000010, B01000010, B01000010, B00111100, B00000000}
- byte **T44** [8] = { B00000000, B01111000, B10000100, B10000100, B10000100, B10000100, B01111000, B00000000}
- byte **T45** [8] = { B00000000, B11110000, B00001000, B00001000, B00001000, B00001000, B11110000, B00000000}
- byte **T46** [8] = { B00000000, B11100000, B00010001, B00010000, B00010001, B00010001, B11100000, B00000000}
- byte **T47** [8] = { B00000000, B11000001, B00100010, B00100001, B00100010, B00100010, B11000001, B00000000}
- byte **T48** [8] = { B00000000, B10000011, B01000100, B01000011, B01000100, B01000100, B10000011, B00000000}
- byte **T49** [8] = { B00000000, B00000111, B10001000, B10000111, B10001000, B10001000, B00000111, B00000000}
- byte **T50** [8] = { B00000000, B00001110, B00010001, B00001110, B00010001, B00010001, B00001110, B00000000}
- byte **T51** [8] = { B00000000, B00011100, B00100010, B00011100, B00100010, B00100010, B00011100, B00000000}
- byte **T52** [8] = { B00000000, B00111000, B01000100, B00111000, B01000101, B01000100, B00111000, B00000000}
- byte **T53** [8] = { B00000000, B01110000, B10001000, B01110000, B10001011, B10001000, B01110000, B00000000}
- byte **T54** [8] = { B00000000, B11100000, B00010000, B11100000, B00010111, B00010000, B11100000, B00000000}
- byte **T55** [8] = { B00000000, B11000000, B00100000, B11000000, B00101111, B00100000, B11000000, B00000000}

- byte **T56** [8] = { B00000000, B10000000, B01000000, B10000000, B01011110, B01000000, B10000000, B00000000}
- byte **T57** [8] = { B00000000, B00000001, B10000001, B00000001, B10111100, B10000000, B00000001, B00000000}
- byte **T58** [8] = { B00000000, B00000011, B00000010, B00000011, B01111000, B00000000, B00000011, B00000000}
- byte **T59** [8] = { B00000000, B00000111, B00000100, B00000111, B11110000, B00000000, B00000111, B00000000}
- byte **T60** [8] = { B00000000, B00001111, B00001000, B00001111, B11100001, B00000001, B00001111, B00000000}
- byte **T61** [8] = { B00000000, B00011110, B00010000, B00011110, B11000010, B00000010, B00011110, B00000000}
- byte **T62** [8] = { B00000000, B00111101, B00100001, B00111101, B10000101, B00000101, B00111101, B00000000}
- byte **T63** [8] = { B00000000, B01111011, B01000010, B01111011, B00001010, B00001010, B01111011, B00000000}
- byte **T64** [8] = { B00000000, B11110111, B10000100, B11110111, B00010100, B00010100, B11110111, B00000000}
- byte **T65** [8] = { B00000000, B11101111, B00001000, B11101110, B00101000, B00101000, B11101111, B00000000}
- byte **T66** [8] = { B00000000, B11011110, B00010000, B11011100, B01010000, B01010000, B11011110, B00000000}
- byte **T67** [8] = { B00000000, B10111101, B00100001, B10111001, B10100001, B10100001, B10111101, B00000000}
- byte **T68** [8] = { B00000000, B01111011, B01000010, B01110010, B01000010, B01000010, B01111011, B00000000}
- byte **T69** [8] = { B00000000, B11110111, B10000100, B11100100, B10000100, B10000100, B11110111, B00000000}
- byte **T70** [8] = { B00000000, B11101111, B00001000, B11001000, B00001000, B00001000, B11101111, B00000000}
- byte **T71** [8] = { B00000000, B11011110, B00010000, B10010000, B00010000, B00010000, B11011110, B00000000}
- byte **T72** [8] = { B00000000, B10111101, B00100001, B00100001, B00100001, B00100001, B10111101, B00000000}
- byte **T73** [8] = { B00000000, B01111011, B01000010, B01000010, B01000010, B01000010, B01111011, B00000000}
- byte **T74** [8] = { B00000000, B11110111, B10000100, B10000100, B10000100, B10000100, B11110111, B00000000}
- byte **T75** [8] = { B00000000, B11101111, B00001000, B00001000, B00001000, B00001000, B11101111, B00000000}
- byte **T76** [8] = { B00000000, B11011110, B00010000, B00010000, B00010000, B00010000, B11011110, B00000000}
- byte **T77** [8] = { B00000000, B10111101, B00100000, B00100000, B00100000, B00100000, B10111101, B00000000}
- byte **T78** [8] = { B00000000, B01111011, B01000001, B01000001, B01000001, B01000001, B01111011, B00000000}
- byte **T79** [8] = { B00000000, B11110111, B10000010, B10000010, B10000010, B10000010, B11110111, B00000000}
- byte **T80** [8] = { B00000000, B11101110, B00000100, B00000100, B00000100, B00000100, B11101110, B00000000}
- byte **T81** [8] = { B00000000, B11011100, B00001001, B00001001, B00001001, B00001001, B11011100, B00000000}
- byte **T82** [8] = { B00000000, B10111001, B00010010, B00010010, B00010010, B00010010, B10111001, B00000000}
- byte **T83** [8] = { B00000000, B01110011, B00100100, B00100100, B00100100, B00100100, B01110011, B00000000}

- byte **T84** [8] = { B00000000, B11100111, B01001000, B01001000, B01001000, B01001000, B11100111, B00000000}
- byte **T85** [8] = { B00000000, B11001111, B10010000, B10010000, B10010000, B10010000, B11001111, B00000000}
- byte **T86** [8] = { B00000000, B10011110, B00100001, B00100001, B00100001, B00100001, B10011110, B00000000}
- byte **T87** [8] = { B00000000, B00111100, B01000010, B01000010, B01000010, B01000010, B00111100, B00000000}
- byte **T88** [8] = { B00000000, B01111001, B10000101, B10000101, B10000101, B10000101, B01111000, B00000000}
- byte **T89** [8] = { B00000000, B11110010, B00001011, B00001010, B00001010, B00001010, B11110000, B00000000}
- byte **T90** [8] = { B00000000, B11100100, B00010110, B00010101, B00010100, B00010100, B11100000, B00000000}
- byte **T91** [8] = { B00000000, B11001000, B00101100, B00101010, B00101001, B00101000, B11000000, B00000000}
- byte **T92** [8] = { B00000000, B10010001, B01011001, B01010101, B01010011, B01010001, B10000000, B00000000}
- byte **T93** [8] = { B00000000, B00100010, B10110010, B10101010, B10100110, B10100010, B00000000, B00000000}
- byte **T94** [8] = { B00000000, B01000100, B01100100, B01010100, B01001100, B01000100, B00000000, B00000000}
- byte **T95** [8] = { B00000000, B10001000, B11001000, B10101000, B10011000, B10001000, B00000000, B00000000}
- byte **T96** [8] = { B00000000, B00010000, B10010001, B01010001, B00110001, B00010001, B00000001, B00000000}
- byte **T97** [8] = { B00000000, B00100001, B00100010, B10100010, B01100011, B00100010, B00000010, B00000000}
- byte **T98** [8] = { B00000000, B01000011, B01000100, B01000100, B11000111, B01000100, B00000100, B00000000}
- byte **T99** [8] = { B00000000, B10000110, B10001001, B10001001, B10001111, B10001001, B00001001, B00000000}
- byte **T100** [8] = { B00000000, B00001100, B00010010, B00010010, B00011110, B00010010, B00010010, B00000000}
- byte **T101** [8] = { B00000000, B00011000, B00100100, B00100100, B00111100, B00100100, B00100100, B00000000}
- byte **T102** [8] = { B00000000, B00110000, B01001000, B01001000, B01111000, B01001000, B01001000, B00000000}
- byte **T103** [8] = { B00000000, B01100001, B10010000, B10010000, B11110000, B10010000, B10010000, B00000000}
- byte **T104** [8] = { B00000000, B11000011, B00100000, B00100000, B11100000, B00100000, B00100000, B00000000}
- byte **T105** [8] = { B00000000, B10000111, B01000001, B01000001, B11000001, B01000001, B01000001, B00000000}
- byte **T106** [8] = { B00000000, B00001111, B10000010, B10000010, B10000010, B10000010, B10000010, B00000000}
- byte **T107** [8] = { B00000000, B00011111, B00000100, B00000100, B00000100, B00000100, B00000100, B00000000}
- byte **T108** [8] = { B00000000, B00111110, B00001000, B00001000, B00001000, B00001000, B00001000, B00000000}
- byte **T109** [8] = { B00000000, B01111101, B00010001, B00010001, B00010001, B00010001, B00010001, B00000000}

2.2.1 Detailed Description

Arrays de bytes de letras, número y signos a mostrar en la matriz de led's en el banner.

Version

1.0

Date

19/08/2020

Author

ARQUI 1, GRUPO 8

2.3 banner.h File Reference

Funciones para mostrar el mensaje inicial solicitado.

Functions

- void setup_banner ()
  setup_banner Se ejecuta para inicializar todas las variables y pines necesarios para mostrar el mensaje.
- void showBanner ()
  showBanner Mostrara el mensaje en en la matriz de led´s.
- void dir ()
  dir Controla la dirrección(iquierda o derecha) en la que se movera el mensaje.
- void DefinirDireccion (int var)
  Se manda a imprimir las letras en la matriz de led's.
- void drawScreen (byte buffer2[ ])
  drawScreen Se manda a imprimir las letras en la matriz de led's sin driver.
- void setColumns (byte b)
  setColumns Se selección la columna que encendera.

2.3.1 Detailed Description

Funciones para mostrar el mensaje inicial solicitado.

Version

1.0

Date

19/08/2020

Author

ARQUI 1, GRUPO 8

2.3.2 Function Documentation

2.3.2.1 DefinirDireccion()

```
void DefinirDireccion (
            int var )
```

Se manda a imprimir las letras en la matriz de led's.

Parameters

var	Dirección en la que se movera el mensaje.

2.3.2.2 drawScreen()

```
void drawScreen (
            byte buffer2[] )
```

drawScreen Se manda a imprimir las letras en la matriz de led's sin driver.

Parameters

buffer2[]	Vector a imprimir.

2.3.2.3 setColumns()

```
void setColumns (
            byte b )
```

setColumns Se selección la columna que encendera.

Parameters

b	Columna donde se desplazara

2.4 game.h File Reference

Funciones para hacer mover a los jugadores y la pelota, pausar, reanudar y terminar el juego.

Functions

- void setup_game ()

  setup_game Se ejecuta para inicializar todas las variables y pines necesarios el juego.
- void imprimir_tablero (int pos_x, int pos_y, byte player, byte player2)

  imprimir_tablero Imprime en la matriz correspondiente la posición del jugador 1 y jugador 2.
- void gameOver ()

  gameOver Finaliza el juego, reinicará los marcadores a 0 y apagara todas las leds de la matriz para que nuevamente se cargue el mensaje inicial.
- bool validateScore ()

  validateScore Verifica y valida si el jugador que anoto un punto aún no sea el ganador, cualquier jugador ganará al llegar a 4 putnos.
- void pause ()

  pause Pausará o reanudará el juego.
- void playGame ()

  playGame Iniciará el juego.
- void counterThreeSecond ()

  counterThreeSecond Evento al pulsar el botón de start, que decidirá si el juego empieza o finaliza al mantener presionado por lo menos tres segundos.

Variables

- bool **flagGame**
- bool **flagPause**

2.4.1 Detailed Description

Funciones para hacer mover a los jugadores y la pelota, pausar, reanudar y terminar el juego.

Version

1.0

Date

19/08/2020

Author

ARQUI 1, GRUPO 8

2.4.2 Function Documentation

2.4.2.1 imprimir_tablero()

```
void imprimir_tablero (
            int pos_x,
            int pos_y,
            byte player,
            byte player2 )
```

imprimir_tablero Imprime en la matriz correspondiente la posición del jugador 1 y jugador 2.

Parameters

pos_x	Posición en X.
pos_y	Posición en Y.
player	Posición del jugador 1.
player2	Posición del jugador 2.

2.4.2.2 validateScore()

```
bool validateScore ( )
```

validateScore Verifica y valida si el jugador que anoto un punto aún no sea el ganador, cualquier jugador ganará al llegar a 4 putnos.

Returns

Resultado de la verificación.

2.5 scoreBoard.h File Reference

Funciones para mostrar el marcador durante el juego y al pausar.

Functions

- void setup_scoreBoard ()
  setup_scoreBoard Se ejecuta para inicializar todas las variables y pines necesarios para mostrar el marcador.
- void clean ()
  clean Apaga todos los leds en las dos matrices de led.
- byte * getArray (char c)
  getArray Obtiene la matriz correspondiente al marcador del jugador 1.
- byte * getArray2 (char c)
  getArray2 Obtiene la matriz correspondiente al marcador del jugador 2.
- void showScoreNumber (byte *arr, byte *arr2)
  showScoreNumber Muestra el marcador en la matriz.
- void showScore (char playerScore1, char playerScore2)
  showScore Muestra el marcador en la matriz durante 3 segundos, luego de algún jugador anote un punto.

2.5.1 Detailed Description

Funciones para mostrar el marcador durante el juego y al pausar.

Version

1.0

Date

19/08/2020

Author

ARQUI 1, GRUPO 8

2.5.2 Function Documentation

2.5.2.1 getArray()

```
byte* getArray (
            char c )
```

getArray Obtiene la matriz correspondiente al marcador del jugador 1.

Parameters

c	Marcador del jugador 1.

Returns

La el vector correspondiente al marcador de 0 a 4.

2.5.2.2 getArray2()

```
byte* getArray2 (
            char c )
```

getArray2 Obtiene la matriz correspondiente al marcador del jugador 2.

Parameters

c	Marcador del jugador 2.

Returns

La el vector correspondiente al marcador de 0 a 4.

2.5.2.3 showScore()

```
void showScore (
            char playerScore1,
            char playerScore2 )
```

showScore Muestra el marcador en la matriz durante 3 segundos, luego de algún jugador anote un punto.

Parameters

playerScore1	Marcador del jugador 1.
playerScore2	Marcador del jugador 2.

2.5.2.4 showScoreNumber()

```
void showScoreNumber (
            byte * arr,
            byte * arr2 )
```

showScoreNumber Muestra el marcador en la matriz.

Parameters

arr	Vector del marcador del jugador 1.
arr2	Vector del marcador del jugador 2.

Index