

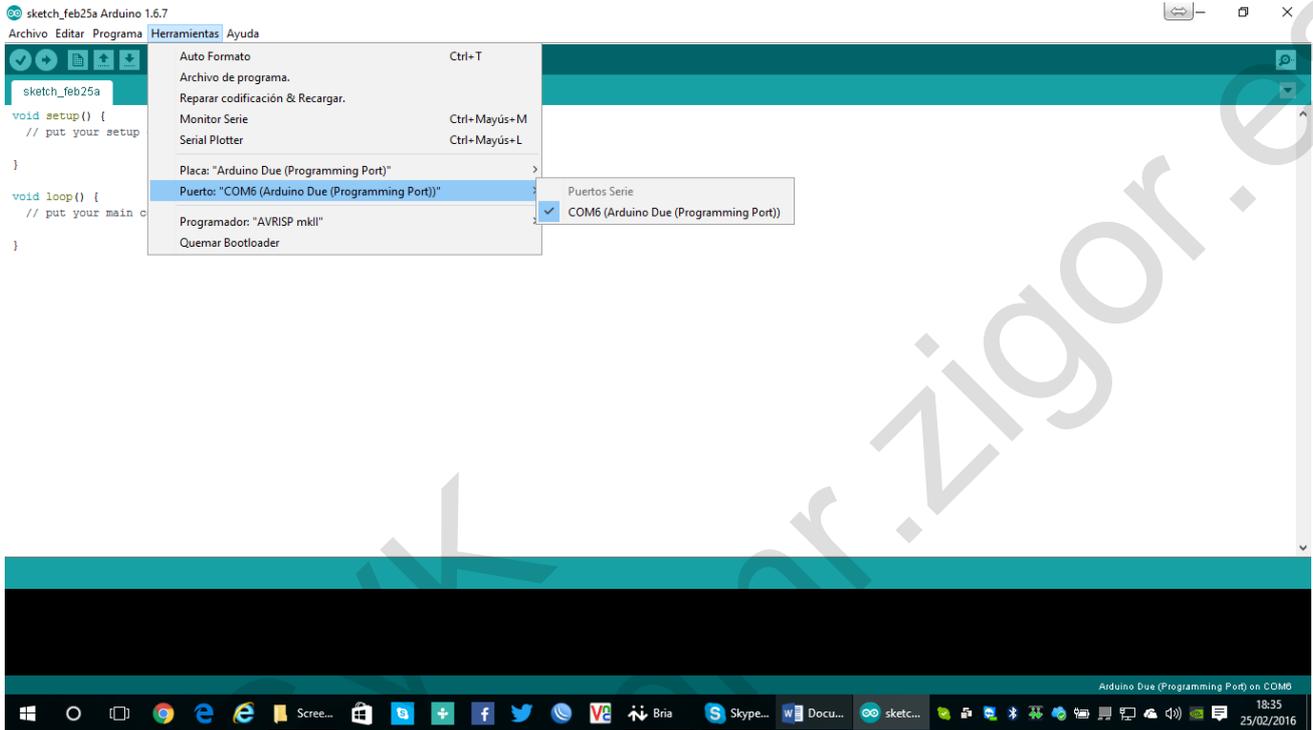
## INSTALACION Y CONFIGURACION REPETIDOR MMDVM (Multiprocol).

1.- Instalar software Arduino. Desde la web  
Htt:\www.arduino.cc -> Area download.

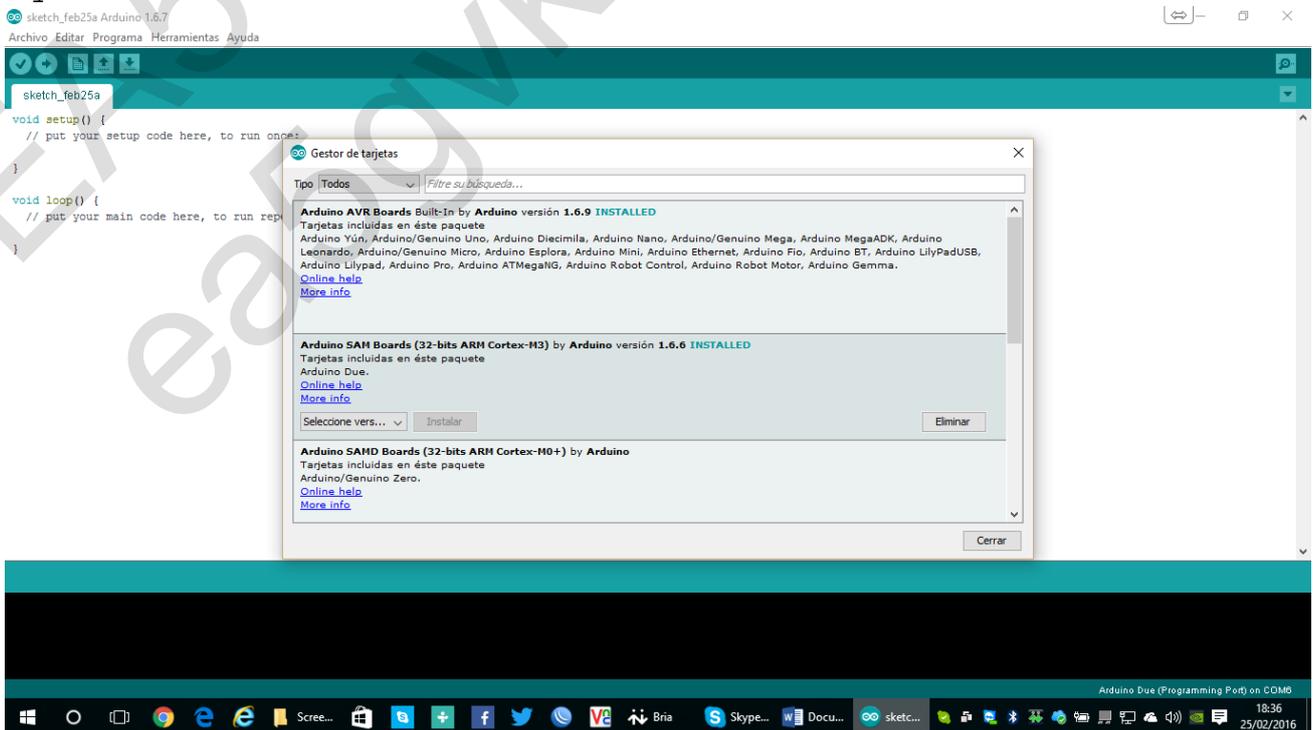
2.- Una vez instalado hay que buscar el Puerto COM asignado si es Windows, porque si fuera windows seria ttyACM0

3.- Arrancamos el software arduino, previamente ya teníamos conectado por usb el Arduino DUE (puerto físico del arduino "PROGRAMMING ->ATMEGA16U2").

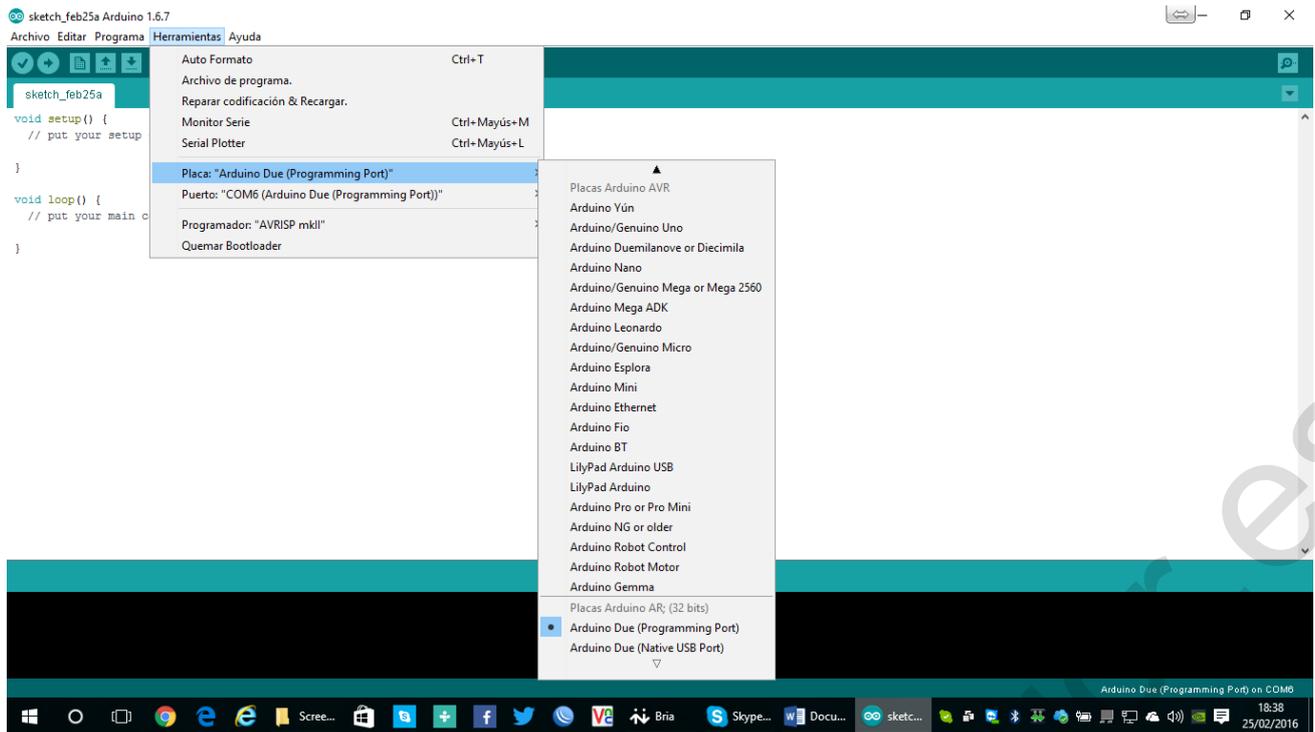
4.- Se selecciona el puerto en el programa Arduino.



5.- Seleccionamos Herramientas - Placa -> Gestor de tarjetas.  
Aquí instalamos "ARDUINO SAM BOARDS"



Una vez instalado ya nos reconoce nuestra ARDUINO DUE, por lo que abra que se seleccionar "Arduino DUE Programming Port"



6.- Hay que modificar el archivo "platform.txt", el cual se encontrará en plataforma WINDOWS, y con versión soft arduino 1.6.7 con SAM 1.6.6

C:\Users\usuario\AppData\local\Arduino15\packages\arduino\hardware\sam\1.6.6

ABRIMOS DICHO archivo platform.txt con nuestro editor de texto, y habrá que localizar la cadena que empieza con el siguiente encabezado

```
## Combine gc-section, archives, adn objests
```

**Habria que suplantarla por esta otra**

```
## Combine gc-sections, archives, and objects
recipe.c.combine.pattern="{compiler.path}{compiler.c.elf.cmd}" -
mcpu={build.mcu} -mthumb {compiler.c.elf.flags} "-
T{build.variant.path}/{build.ldscript}" "-Wl,-
Map,{build.path}/{build.project_name}.map" {compiler.c.elf.extra_flags} -
o "{build.path}/{build.project_name}.elf" "-L{build.path}" -Wl,--cref -
Wl,--check-sections -Wl,--gc-sections -Wl,--entry=Reset_Handler -Wl,--
unresolved-symbols=report-all -Wl,--warn-common -Wl,--warn-section-align
-Wl,--start-group "{build.path}/core/syscalls_sam3.c.o" {object_files}
"{build.variant.path}/{build.variant_system_lib}"
"{build.system.path}/CMSIS/CMSIS/Lib/ARM/arm_cortexM31_math.lib"
"{build.path}/{archive_file}" -Wl,--end-group -lm -gcc
```

Una vez hecho esto. Guardamos y salimos del archivo txt.

7.- Tenemos que tener descargados previamente la carpeta "MMDVM" y por otro lado "MMDVMHost".

.- Bien vamos por partes. La carpeta "MMDVM" es el firmware que hay que cargar a la DUE. Para ello iremos a la carpeta MMDVM y localizamos el archivo MMDVM.ino y lo ejecutamos abriéndose nuestra aplicación Arduino.

```

MMDVM Arduino 1.6.7
Archivo Editar Programa Herramientas Ayuda

MMDVM CalRX.cpp CalRX.h CalTX.cpp CalTX.h Config.h DMRDefines.h DMRIdleRX.cpp DMRIdleRX.h DMRRX.cpp DMRRX.h DMRSlotRX.cpp DMRSlotRX.h DMRSlotType.cpp DMRSlotType.h

/*
 * Copyright (C) 2015 by Jonathan Naylor G4KLX
 *
 * This program is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 2 of the License, or
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software
 * Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
 */

#include "Config.h"
#include "Globals.h"

// Global variables
MMDVM_STATE m_modemState = STATE_IDLE;

bool m_dstarEnable = true;
bool m_dmrEnable = true;
bool m_yefEnable = true;

```

Ahora una cosa muy importante y es en el apartado de "Config.H", que se puede ver en las pestañas, aquí deberemos de habilitar la placa MMDVM, que estemos trabajando.

- Pasasystem
- ZUM
- NTH

```

MMDVM - Config.h | Arduino 1.6.7
Archivo Editar Programa Herramientas Ayuda

MMDVM CalRX.cpp CalRX.h CalTX.cpp CalTX.h Config.h DMRDefines.h DMRIdleRX.cpp DMRIdleRX.h DMRRX.cpp DMRRX.h DMRSlotRX.cpp DMRSlotRX.h DMRSlotType.cpp DMRSlotType.h

/*
 * (at your option) any later version.
 *
 * This program is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with this program; if not, write to the Free Software
 * Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.
 */

#ifndef CONFIG_H
#define CONFIG_H

#define WANT_DEBUG

// For the original Arduino Due pin layout
#define ARDUINO_DUE_PAPA

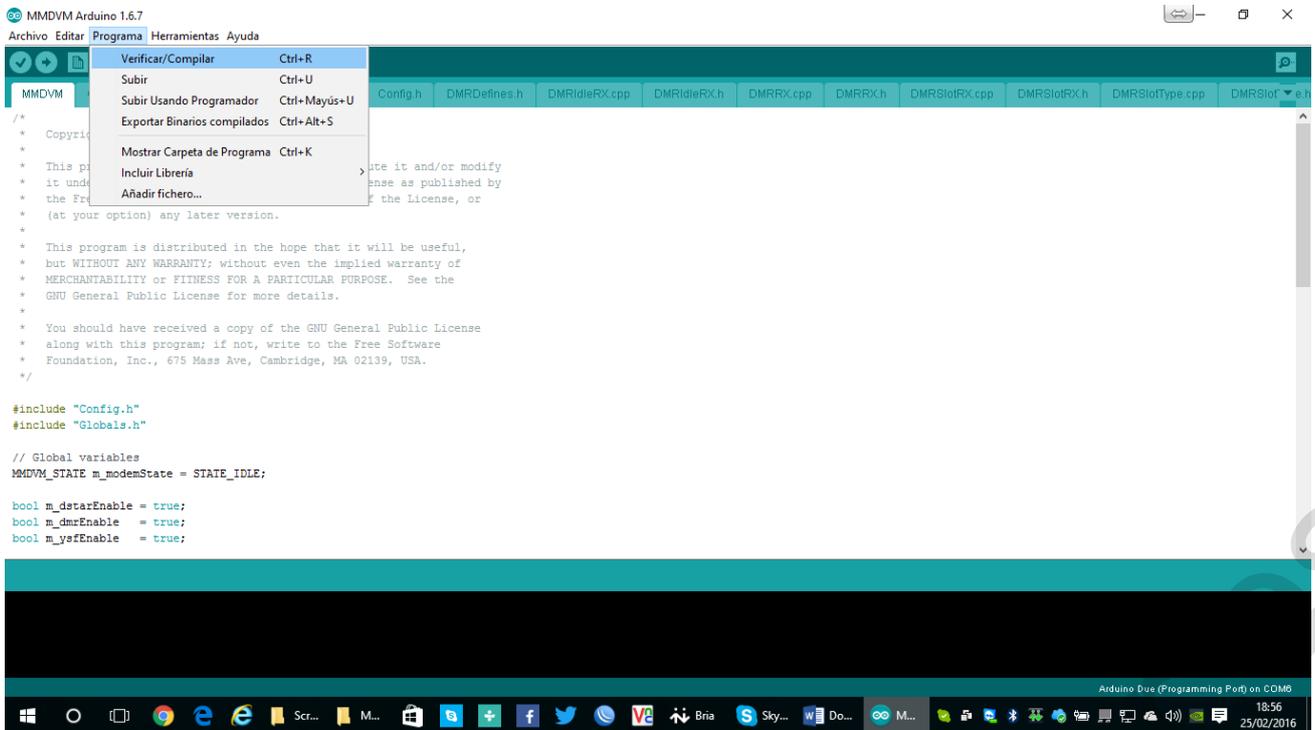
// For the new Arduino Due pin layout
// #define ARDUINO_DUE_ZUM

// For the SP8NTH board
// #define ARDUINO_DUE_NTH

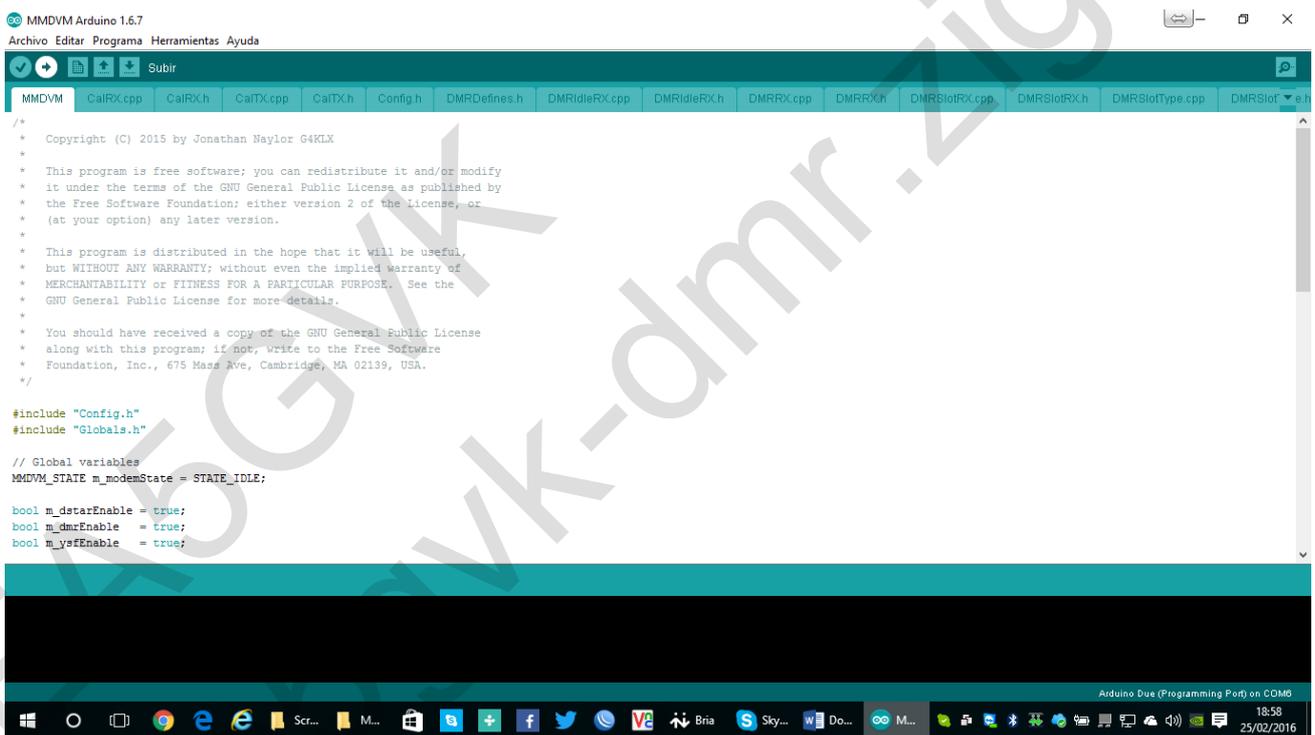
#endif

```

8.- Ahora, con todo conectado arduino DUE y la placa MMDVM ligada, deberemos de ir en el software de arduino a PROGRAMA -> VERIFICAR Y COMPILAR



Bien una vez realizado la verificación y compilación tendremos que enviar a la DUE, como se muestra mas abajo.



Si sale todo OK, dira "VERIFY SUCCESSFUL".

Bien la placa y arduino due ya tiene cargado su Firmware para trabajar.

9.- Ahora vamos al apartado del software que estaría en la CARPETA "MMDVMHost",

Bien allí, encontraremos una subcarpeta llamada "RELEASE" y dentro de ella un archivo llamado MMDVMHost.exe, este lo copiamos a la carpeta raíz de MMDVMHost.

Una vez hecho esto, en la carpeta raíz, localizamos otro archivo llamado "MMDVM.ini",

(aquí esta la madre del cordero), deberemos editarlo con alguna editor de texto, y donde tendremos que configurar, frecuencias, color del repe dmr, activar o desactivar, si solo lo queremos para DMR, o multiprocolo, pudiendo tener DSTAR, C4FM también.

Cuando dice 0= Desactivado y cuando pone 1=Activado. Esa opción.

Bien me centrare en el DMR. En el apartador DMR

```
[DMR Network]
Enable=1
Address=master.spain-dmr.es
Port=62031
Password=passw0rd
Slot1=1
Slot2=1
Debug=1
```

Esta seria la configuración para BRANDMEISTER en España.

Y para DMR Plus en España, sería:

```
[DMR Network]
Enable=1
Address=51.254.214.66
Port=55555
Password=passw0rd
Slot1=1
Slot2=1
Debug=1
```

IMPORTANTE PARA NO CONFUNDIR EN EL APARTADO SIGUIENTE DEL MISMO ARCHIVO MMDVM.INI

```
[Modem]
# Port=/dev/ttyACM0 → Esto sería para LINUX y abrir que quitarle #
Port=\\.COM6 -> Ahora mismo asi estaría habilitado para Windows
TXInvert=1
RXInvert=0
PTTInvert=0
TXDelay=100
RXLevel=50
TXLevel=50
Debug=1
```

**Bien ahora para ejecutarlo en MSDOS - Windows seria, dentro de la Carpeta MMDVMHost**

MMDVMHost.exe MMDVM.ini

**Y en LINUX seria**

Sudo ./MMDVMHost MMDVM.ini