Quick Start
Intelligent Display CAN WVGA
(Base IronGraph)

11 avenue Henri Becquerel
33700 MERIGNAC (FRANCE)
Tel : +33 (0)5.56.13.04.68
www.clairitec.com

Test sequences CAN

This QuickStart guide enables you to familiarize yourself with our development environment and to discover very quickly the way our solution functions.
In order to do this, you can use a CAN2.0B terminal which allows you enter the hexadecimal commands described in this document.
You can, for example, use the program "PCAN-View", provided by the company PEAK-SYSTEM, which is compatible with their PCAN-USB adapter, available as an option for your StarterKit.

You will find the installation drivers for the module on the USB stick in the following directory: *Exemples\Drivers_Dongle\USB_CAN_Converter_Driver*
You will find the project file of this QuickStart with the pre-written frames for PCAN-View in the following directory: *Exemples\IronGraph\CAN\PCANView_project_example*

The Intelligent Display used is based on the ***IronGraph*** board and has thus all the characteristics of this HMI board.

The first step is to install the software program GraphConverter, available in the folder with the same name on the USB stick.

Afterwards you should connect your StarterKit to your computer and to electricity.

Connect the USB/CAN adapter for the CAN2.0B commands.
Connect the USB cable for the uploads from GraphConverter.
Connect the StarterKit to electricity.

When the StarterKit is first connected to the USB port, you are asked by Windows to install the drivers for the new device. You find the configuration file for your Intelligent Display ".inf" on the USB key in the directory "Firmwares\IntelligentDisplay\Driver_USB_CDC" (the file is the same for the **IronGraph** board).
If Windows does not detect the installation of the drivers, make use the "Device Manager" of Windows.

Do not mistake the installation of the USB/CAN adapter with the driver CDC of IronGraph.

GraphConverter is delivered with the drivers of the most frequent HMI products. These files can be found in the folder "LibraryCard" which is created in the directory of the executable file (by default C:\Program Files\Clairitec) during the installation of the software program.
It may arrive that the list of reference products has evolved since the creation of the installation package of GraphConverter. We advise you to copy the latest drivers for your Intelligent Display from the USB stick "Firmwares\IronGraph\PilotesIronGraph".

For more information about the installation of your material, consult the documentation UserManual_GraphConverter.pdf (Chapter 2).

The StarterKit is delivered with a pre-programmed graphic library (*IRG_CAN_WVGA.gxp*) , available on the USB stick in the directory *\Exemples\IronGraph\CAN\GraphConverter_Project\IRG_CAN_WVGA*

In this directory you will also find the "HMI Editor" project, linked to this graphic library. The project file *IRG_CAN_WVGA_HMI_.XML* contains all pre-defined HMI screens in their initial state.

Thanks to the "IronGraph" technology, all pre-defined HMI screens can be uploaded into the memory of the HMI board in the same way as the graphic library.
This allows to display an entire HMI screen in its initial state, with its icons, (buttons in the "non-pushed" mode, by default lightened up or not), the texts shown at start in their pre-defined color and display mode, as well as the pre-defined touch zones, by making use of only one command (GX_PutScreen).

This project is only provided as an example for the start of your development. It helps you to master very quickly the main functions of our concept.

To access the pre-programmed graphic library of your StarterKit on your computer, open the GXP file with GraphConverter.

Once opened, you can also access the "HMI Editor" project and see all the pre-defined HMI screen by clicking on the tab "HMI Editor".

This library configures your Intelligent Display in CAN mode with a speed of 500Kbds and the following identifiers:
>>> ID Command: 0x049
>>> ID Read Frame: 0x50
>>> ID After Reset frame: 0x52

Specification of the SUBD-9 connector, supplied with your StarterKit:

Pin	Type designation	Type	Voltage level	Description
1	NC	-	-	
2	CAN-L	E/S	CAN2.0B	Line CAN Low
3	GND	E	Power supply	Mass (0v)
4	NC	-	-	
5	NC	-	-	
6	NC	-	-	
7	CAN-H	E/S	CAN2.0B	Line CAN High
8	NC	-	-	
9	NC	-	-	

Note : Consult the documentation "hardware" of your Intelligent Display for more information about its specific connections.

Connect your StarterKit to a COM port of the following type, as shown below:
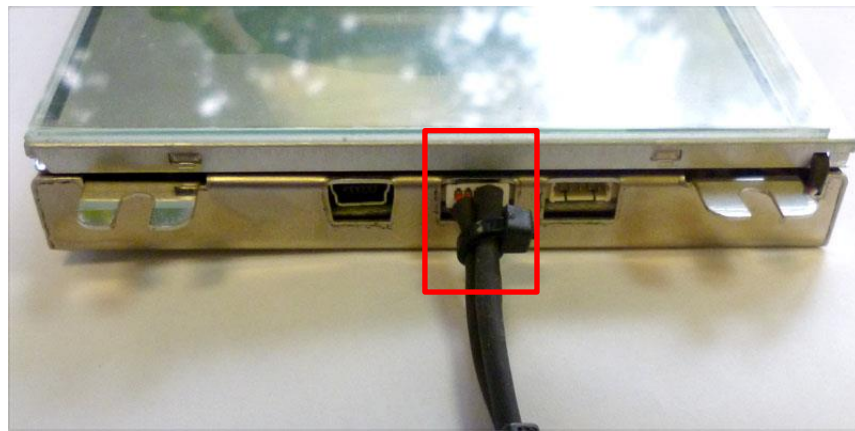
Connection example with PEAK SYSTEM, provided optionally, and the network case:
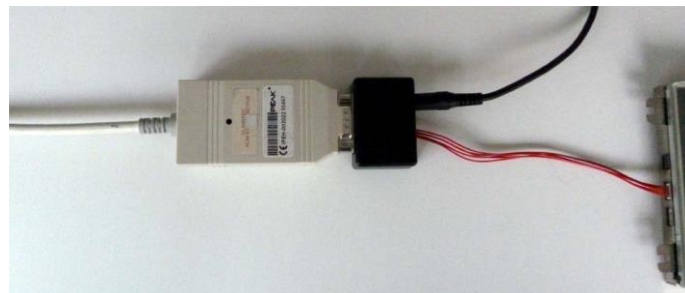
Upon powering up your Intelligent Display, the first image of the graphic library is automatically displayed in the center of the display. The StarterKit contains the logo of Clairitec as its first image.

ID 0x052 DLC : 8 - Rx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x02	0x00	0x00	0x49	0x00	0x50	0x00	0x00

You can now send the CAN sequences and observe their effects on the display

ATTENTION: It is important to send all necessary bytes of a command in subsequent order and to respect the DLC of each one.

- **Display of the first HMI screen, created with the HMI Editor of GraphConverter**

**C function**:

GX_PutScreen (NumMSG, 0x01);   // Display HMI screen n° 1, (_NumMsg = number of the CAN mailbox_)
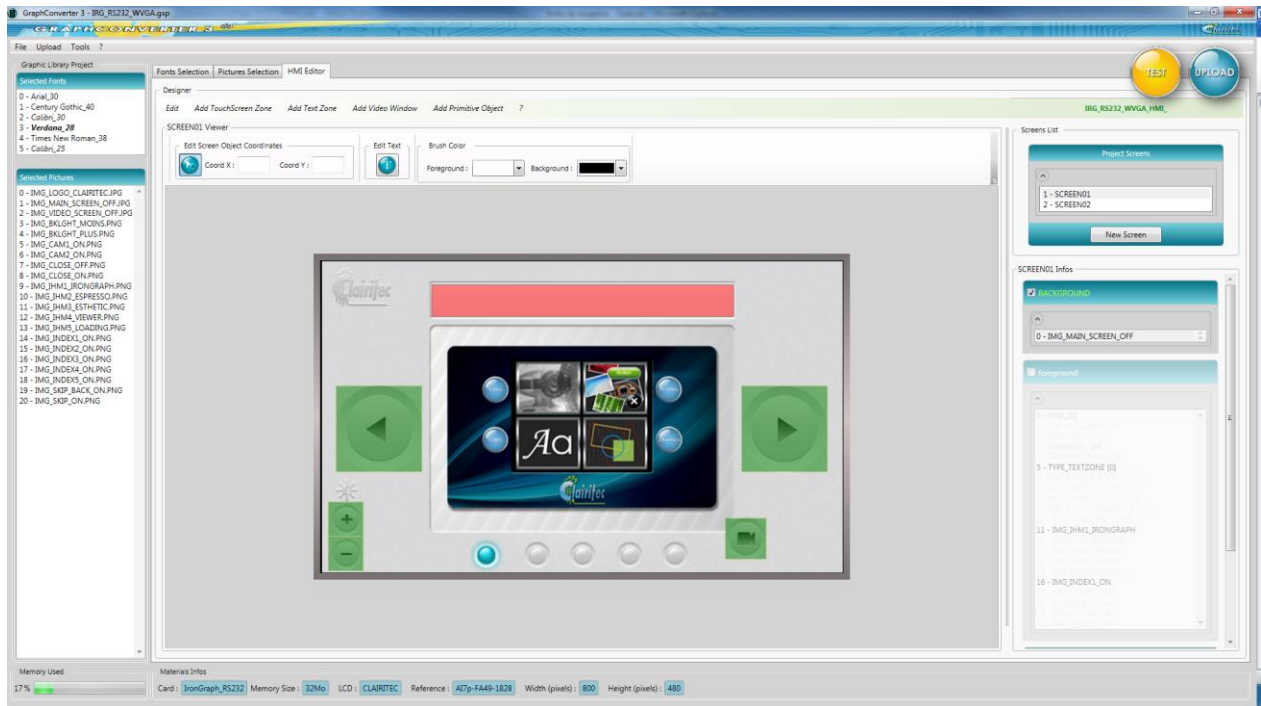
**CAN frame**: (1 frame for this command)

ID 0x049 DLC: 3 - Tx

Byte1	Byte2	Byte3
0x00	0x64	0x01

The HMI screen is completely displayed:

On the display you can see a gray background with 5 types of buttons ("forward", "back", "+", "-", "camera"), a lightened-up, blue indicator "lamp", and a dashboard image in the center.
This HMI screen is displayed the following way in GraphConverter:

Our composition contains:

- • Background layer:
- Background image *0-IMG_MAIN_SCREEN_OFF*

- • Foreground layer:
- A text zone *5-TYPE_TEXTZONE[0]*
- 2 hidden images of « lit-up » buttons, *IMG_SKIP_BACK_ON* and *IMG_SKIP_ON*, both linked to a touch zone ([T]-1 et [T]-2).
- An image of a blue "lamp" *16-IMG_INDEX_1_ON*
- A centered image *11-IMG_IHM1_IRONGRAPH*

- • Touch panel layer
- A touch zone *0-TYPE_TOUCHSCREENZONE[1]*
- A touch zone *1-TYPE_TOUCHSCREENZONE[2]*
- A touch zone *2-TYPE_TOUCHSCREENZONE[3]*
- A touch zone *3-TYPE_TOUCHSCREENZONE[4]*
- A touch zone *4-TYPE_TOUCHSCREENZONE[5]*

The hidden elements do not get displayed automatically upon sending the command GX_PutScreen. However, their characteristics, dimension and coordinates, are saved in a text file, type .h (well known for developers in C language). This file helps you to write the code for the commands GX_PutImage and GX_PutString, allowing you to display the hidden elements at the right place and at the right moment of your program.

We will see below how to do this.

When using the command GX_PutScreen to display a pre-defined HMI screen, the two graphic layers of your Intelligent Display are entirely initialized with all elements which have previously been positioned and set into visible mode in the HMI Editor. The foreground layer is activated in writing and display mode.

- **Activation of the touchscreen layer in the « On Click » mode**

C function:

GX_SetTouchScreen(NumMSG, 1, 0, 2, 1, 0, 0) ;   //Activate the touchscreen layer in the "Zones" mode, "On Click" mode, with a fixed period of repetition of 100 mS during the time the touch zone is touched. (*NumMsg = number of the CAN mailbox*)

CAN frame: (1 frame for this command)

ID 0x049 DLC: 7 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x00	0x49	0x74	0x01	0x02	0x81	0x00	0x00

As soon as this command is send to the HMI board, and when you touch on one of the two buttons "forward" "backward", represented in green in the HMI Editor, you will receive the feedback frame with a DLC of 5 bytes (ID: 0x50 in our example).

The frame sent out as a feedback by the Intelligent Display when the touchscreen is touched:

Byte1	Byte2	Byte3	Byte4	Byte5
Number of zone touched or released	Coordinate X where the touchscreen has been touched (MSB)	Coordinate X where the touchscreen has been touched (LSB)	Coordinate Y where the touchscreen has been touched (MSB)	Coordinate Y where the touchscreen has been touched (LSB)

When, in our example, the button "forward" is touched, the byte1 of the frame received has the hexadecimal value 0x8. The 4 other bytes contain the specific coordinates of the point touched. The frame will be sent out every 100 mS as long as the touch zone keeps getting touched.

As soon as the touch zone is released, the frame is no longer sent out as a feedback. If a non-defined zone of the touchscreen is touched, the byte1 of the frame will be 0x80, indicating that the zone n°0 (=no specific zone) has been touched.

The fact of having linked these two zones to the 2 images of "lit-up" buttons allows you to display the corresponding "lit-up" button during the time you receive the code 0x81 or 0x82. When the zone is no longer touched, you can replace the "lit-up" image by a transparent rectangle. All of this is possible by making reference to the information described in the text file *IRG_CAN_WVGA_HMI_.h.*

In this file you therefore find the definition of the necessary arguments for the GX_PutImage command (for those working in C language) or for the Escape+'I' command of the CAN protocol.

Extract of the file *IRG_CAN_WVGA_HMI_.h*, defining the lit-up button "backward", linked to the touch zone 1 (0-TYPE_TOUCHSCREENZONE[1]) :

```
/* --- PICTURE IMG_SKIP_BACK_ON  - [T] 1 N° 6 --- */
#define SCREEN01_FOREGROUND_IMG_6        19, 0
#define XY_SCREEN01_FOREGROUND_IMG_6     21, 191
```

The first line defines the number of the image in the graphic library, "19", followed by the option of the function GX_PutImage (always 0).

The second line defines the coordinates of the image as shown in the HMI Editor, this means x=21, Y=191.

- **Displaying an image on the foreground layer**

C function:

GX_PutImage(NumMSG, 0x0013,0, 0x0015, 0x00BF);   //Display image 2 at the coordinates x=21, Y=191
 (*NumMsg = number of the CAN mailbox*).

CAN frame: (2 frame for this command)

ID 0x049 DLC: 8 - Tx

Bytet1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x00	0x49	0x00	0x13	0x00	0x00	0x15	0x00

ID 0x049 DLC: 2 - Tx

Byte1	Byte2
0x01	0xBF

The image is displayed at the coordinates x=21, y =191:

The button « backward » appears lit-up.

- **Definition of the text color**

  ▪ **Definition of the background color of the text**

C function :

GX_SetColor(NumMSG, BACK_COLOR, 0, 0, 0, 0);   //Background color of the text = transparent (Alpha =0),
(*NumMsg = number of the CAN mailbox*).

CAN frame: (1 frame for this command)

ID 0x049 DLC: 7 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x00	0x43	0x01	0x00	0x00	0x00	0x00

  ▪ **Definition of the text color**

C function:

GX_SetColor(NumMSG, FORE_COLOR, 255, 255, 0, 0) ;//   Foreground color of the text (Alpha=255, R=255, G=0,
B=0, (0xFFFF0000))
(*NumMsg = number of the CAN mailbox*).

CAN frame : (1 frame for this command)

ID 0x049 DLC: 7 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
0x00	0x43	0x00	0xFF	0xFF	0x00	0x00

- **Writing a text with defined alignment**

C function:

GX_PutString (NumMSG, STYLE_SCREEN01_FOREGROUND_TXT_3, 0, 0, (unsigned char*)«MyText») ;   // Writing
"MyText" on the FOREGROUND LAYER in the TEXT ZONE n°0 with the TEXT FONT n°3,
(*NumMsg = number of the CAN mailbox*).

CAN frame: (2 frames for this command)

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x00	0x53	0x82	0x03	0x00	0x00	0x00	0x00

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x01	0x4D	0x79	0x54	0x65	0x78	0x74	0x00

The text "MyText" is displayed, centered, in the white zone above the dashboard image (or rather in the text zone N°0 defined in the HMI Editor):

(*See next page…*)

Never forget the character ZERO (0) to end the chain of characters.

Send a second frame to write a text:

CAN frame: (2 frames for this command)

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x00	0x53	0x82	0x03	0x00	0x00	0x00	0x00

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x01	0x4D	0x79	0x20	0x73	0x65	0x63	0x6F

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x02	0x6E	0x64	0x20	0x54	0x65	0x78	0x74

ID 0x049 DLC: 2 - Tx

Byte1	Byte2
0x03	0x00

The text "My Text" gets automatically replaced by "My second text", still centered in the text zone N°0 which has been defined in the HMI Editor.

- **A last example**

To change the HMI screen, send the command PutScreen n°2

C function:

GX_PutScreen(NumMSG, 0x02);   // Display HMI screen n° 2
(*NumMsg = number of the CAN mailbox*).

CAN frame: (1 frame for this command)

ID 0x049 DLC: 3 - Tx

Byte1	Byte2	Byte3
0x00	0x64	0x02

The HMI screen is completely displayed:

The HMI screen is composed out of 6 touchable buttons, 2 text zones, a red rectangle, and two lines (green and blue). All of these elements have been placed with the HMI Editor of GraphConverter.

When you click on the buttons, it gets clear that the touch panel is still active but that the touch zones have been modified.

This HMI screen has been created to display the real-time video of the 2 cameras which are part of the StarterKit (only one video can be displayed at once).

In order to display camera n°1, (in NTSC mode), send the following command:

C function:

GX_PutVideo(NumMSG,CAMERA_1_NTSC, VIDEO_NORMAL, 458, 309, 152, 66);//   Draw a video window of 458x309 pixels at the coordinates 152, 66 with camera n°1 activated in NTSC mode (*NumMsg = number of the CAN mailbox*).

CAN frame: (2 frames for this command)

ID 0x049 DLC: 8 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
0x00	0x66	0x02	0x00	0x01	0xCA	0x01	0x35

ID 0x049 DLC: 5 - Tx

Byte1	Byte2	Byte3	Byte4	Byte5
0x01	0x00	0x98	0x00	0x42

For more information, consult the following documents:

- [DOC-20140527-2F-FR]-CommandesEscapeIronGraph_CAN.pdf for the plain code of the CAN commands.
- [DOC-20140527-3A-FR]-FonctionsHautNiveauIronGraph.pdf for the C functions.
- [DOC-20121220-2D-FR]-Manuel_Utilisation_GraphConverter.pdf for the way of using GraphConverter and the HMI editor.
- [DOC-20160224-1A-FR]-Formation IRG_CAN.pdf to learn how to use our system.

Clairitec

CLAIRITEC
11 avenue Henri Becquerel
33700 Mérignac
FRANCE

Web site: www.clairitec.com

Clairitec's services

Customer relation service: contact@clairitec.com

Technical support service: support@clairitec.com

Non-contractual pictures