



1 - The EasyCAT Gateway

The EasyCAT Gateway is a device that allows to exchange data between an EtherCAT® bus and an USB host or an Ethernet network.

It is seen by the EtherCAT® master as a slave, with two exchange data areas, one for the inputs and one for the outputs. The embedded microcontroller, an Arm M0 SAMD21G18, will transfer these data to and from a different interface, I.E. the USB or the Ethernet, thus implementing the gateway functionality.

The microcontroller firmware can be developed using the Arduino IDE and therefore you can easily customize the behaviour of the Gateway, taking advantage of the semplicity of the Arduino enviroment and of the huge amount of software libraries available.



Fig-1 The EasyCAT Gateway

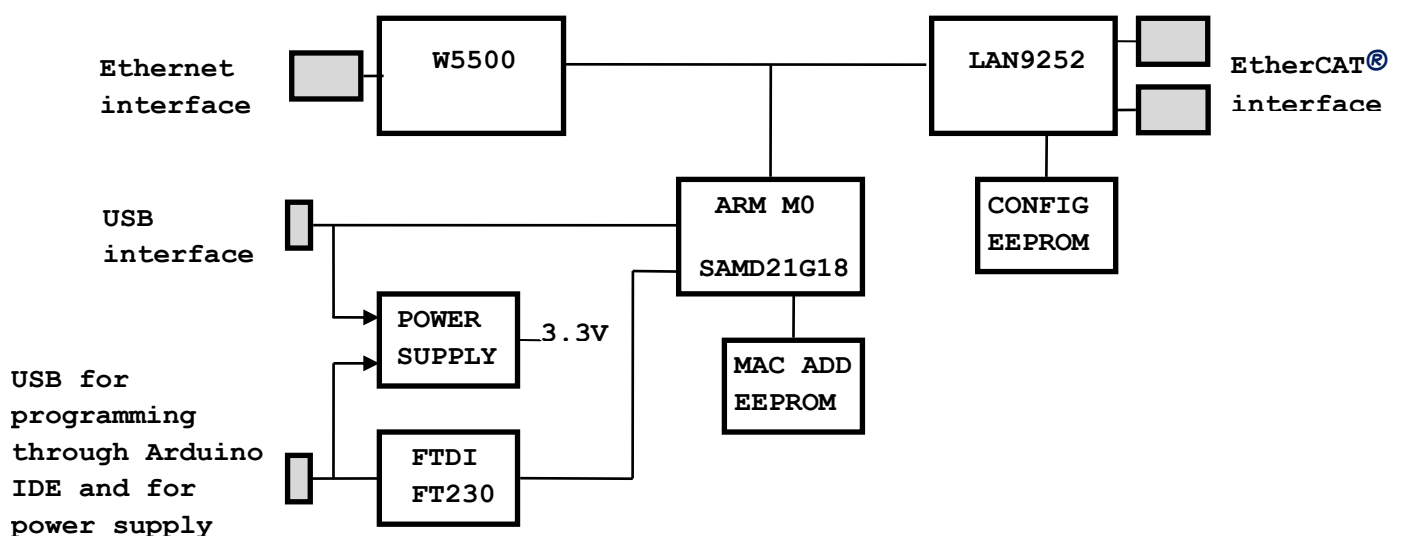


Fig-2 EasyCAT Gateway block diagram

1.1 - Data flow illustration

In the picture aside the data are exchanged between the EtherCAT® interface and an host connected to the USB interface, E.G. a PC or a tablet.

The power can be provided by the USB host, or by an external power supply connected to the Power/Programming connector. The use of an external power supply allows the USB host to go to sleep mode, or also to be physically disconnected, without disrupting the EtherCAT® network activity.

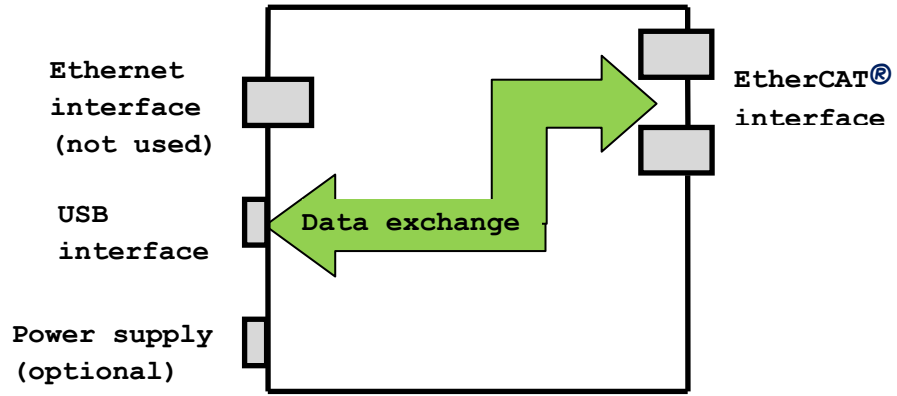


Fig-3 Gateway EtherCAT® - USB

Instead, here the data are exchanged between the EtherCAT® interface and the Ethernet network.

In this case it is mandatory to have an external power supply connected to the Power/Programming connector.

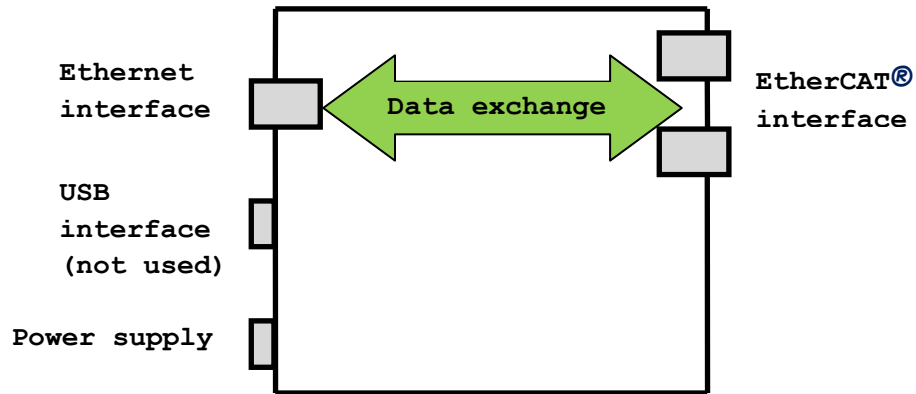


Fig-4 Gateway EtherCAT® - Ethernet

1.2 - The EtherCAT® interface

The EasyCAT Gateway must be connected to a standard EtherCAT® network using the two RJ45 connectors marked "IN" and "OUT". The EasyCAT Gateway is shipped with the exchange area sized as 32 bytes in input and 32 bytes in output, but this can be customized, to satisfy particular needs, using the Easy Configurator, the free software tool downloadable from the AB&T website.

The access to the EtherCAT® network data is particularly simple as it is achieved through the EasyCAT library, the same used by the EasyCAT shield and downloadable from the AB&T website.

Also the slave description .XML file is the same used by the EasyCAT shield.

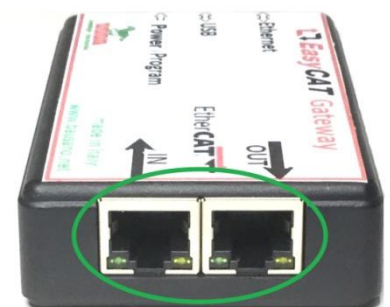


Fig-5 EtherCAT® connectors

1.3 - The Ethernet interface

The Ethernet interface is implemented with a WIZnet W5500, that is an hardwired TCP/IP embedded Ethernet controller. This avoids the burden of writing a TCP/IP stack and, most of all, the W5500 is widely supported by the standard Arduino Ethernet library.

Each EasyCAT Gateway has a unique Ethernet MAC address that is preprogrammed in a serial EEPROM. The same chip provides also 128 bytes available for the user application.

1.4 - The USB interface

The USB interface is embedded in the SAMD21G18 and it is accessed using the standard Arduino library. This connector can also be used to supply the power to the Gateway, even if it is recommended to use an external power supply connected to the Power/Programming connector.

1.5 - The Power/Programming connector

This is another USB connector and has a double function. It can be used to program the SAMD21G18 through an FT230X USB to serial converter. More on this in paragraph [2 - Using the Arduino IDE with the EasyCAT Gateway](#).

The second function is to supply the power to the Gateway when an USB host is not connected or if we don't want to draw current from it: note that if a power supply is connected to this port and an USB host to the USB interface, the current will be draw from the power supply.

The EasyCAT Gateway supply requirement is 5V max. 350mA.

1.6 - Controls and indicators

The general purpose led is under the microcontroller control and then its behavior is defined by the firmware. The power on led simply indicates that the Gateway is powered.

The reset button is accessed through a pin hole on one side of the box. Normally it is not needed but it can be useful during the development of custom firmware.

Note that pressing the reset will interrupt the EtherCAT® communication, and to restart it the master must rescan the network.



Fig-6 Ethernet connector



Fig-7 USB connector



Fig-8 Power/Programming connector

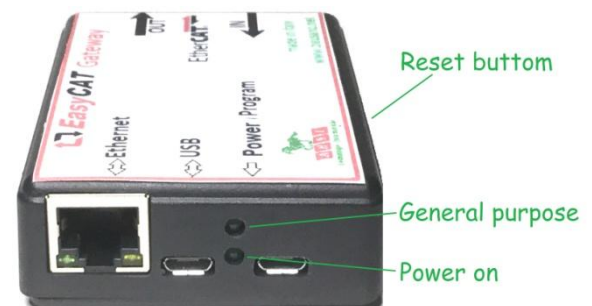


Fig-8 Controls and indicators

2 - Using the Arduino IDE with the EasyCAT Gateway

The EasyCAT Gateway is fully compatible with the Arduino IDE and then you can customize the firmware in a simple way, taking advantage of this widely spread development environment and of the large amount of libraries and programming examples available on the internet.

The board must be setted as "Arduino/GenuinoZero (Native USB Port)" and must be connected to the Gateway through the Power Programming connector.

The programming serial line will appear with the label 'COMXXX' and no other captions. Note that the actual COM number varies from one PC to another.

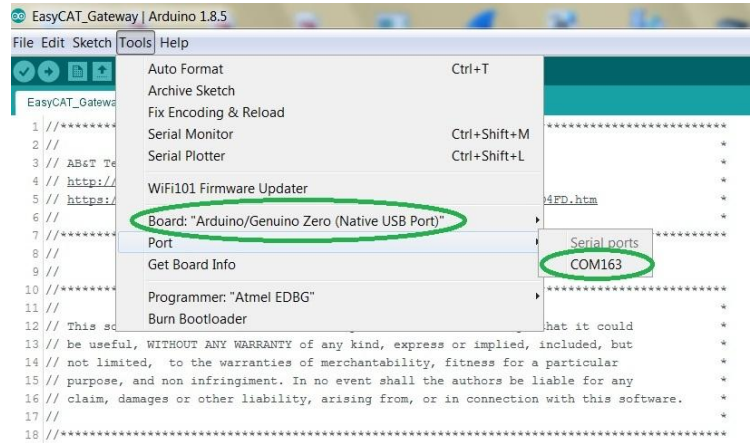


Fig - 9 Setting the Gateway in the Arduino IDE

3 - The Firmware example

The EasyCAT Gateway is shipped programmed with an firmware example that you can use as a starting point to develop your own application. The source code is available on the AB&T website.

The example provides a double Gateway function as it exchange data between the Ethercat® bus and an USB host and, at the same time, between the Ethercat® bus and the Ethernet network.

3.1 - The example EtherCAT®-USB Gateway

The part EtherCAT®-USB of the example establishes a bidirectional communication between the two interfaces.

The data are encapsulated in a Slip frame, to guarantee the recognition of the boundaries of the packet.

In the picture aside we can see a screenshot of a test program that we wrote on a Linux machine, used as an USB host.

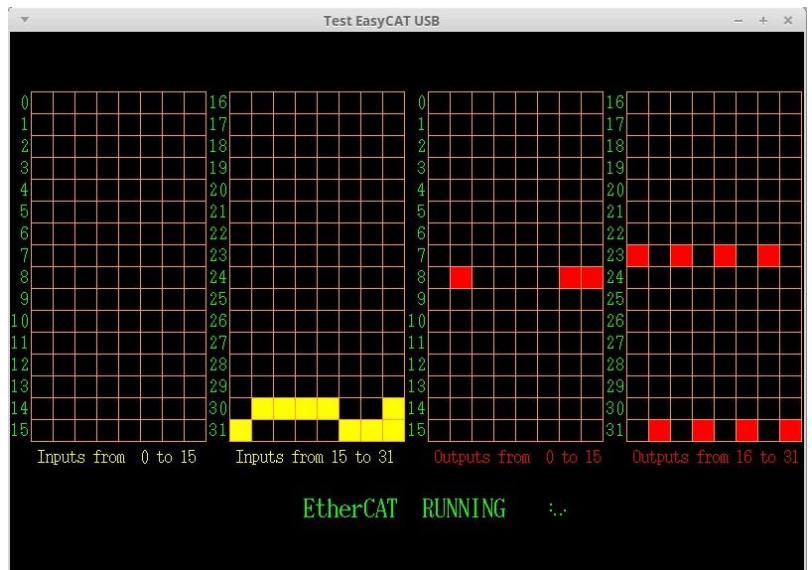
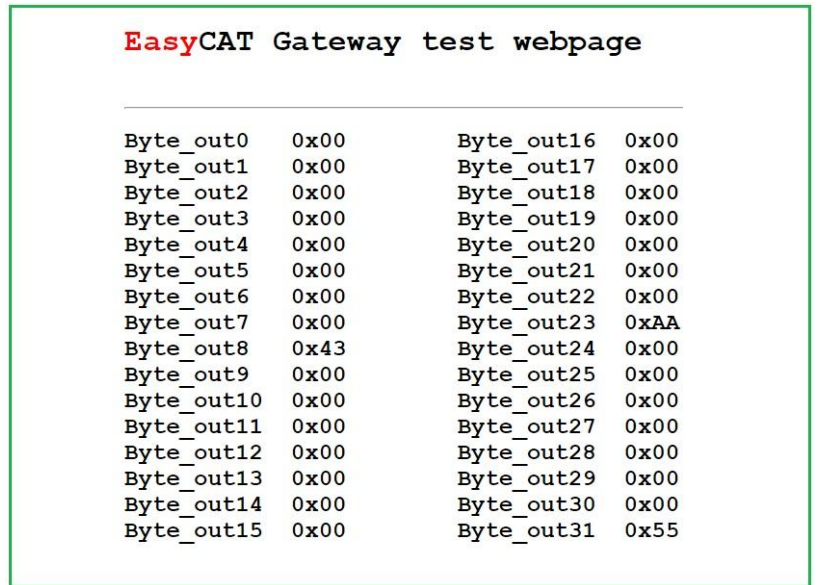


Fig - 10 Test program on a Linux USB host

3.2 - The example EtherCAT®-Ethernet Gateway

The part EtherCAT®-Ethernet of the example establishes a one way data flow, between the two interfaces, as only the Ethercat® data are forwarded to the Ethernet network, in order to not create conflicts with the USB section of the Gateway.

To communicate with the Ethernet network the Gateway acts as a very basic web server, and so the data coming from the EtherCAT® bus can be visualized using a standard web browser.



The screenshot shows a webpage titled "EasyCAT Gateway test webpage". Below the title is a table with two columns of data. The first column lists "Byte_out" from 0 to 15, and the second column lists "Byte_out" from 16 to 31. Each entry has a corresponding hexadecimal value. For example, Byte_out0 is 0x00, Byte_out8 is 0x43, Byte_out7 is 0x00, and Byte_out23 is 0xAA. Byte_out31 is 0x55.

Byte_out	Value	Byte_out	Value
Byte_out0	0x00	Byte_out16	0x00
Byte_out1	0x00	Byte_out17	0x00
Byte_out2	0x00	Byte_out18	0x00
Byte_out3	0x00	Byte_out19	0x00
Byte_out4	0x00	Byte_out20	0x00
Byte_out5	0x00	Byte_out21	0x00
Byte_out6	0x00	Byte_out22	0x00
Byte_out7	0x00	Byte_out23	0xAA
Byte_out8	0x43	Byte_out24	0x00
Byte_out9	0x00	Byte_out25	0x00
Byte_out10	0x00	Byte_out26	0x00
Byte_out11	0x00	Byte_out27	0x00
Byte_out12	0x00	Byte_out28	0x00
Byte_out13	0x00	Byte_out29	0x00
Byte_out14	0x00	Byte_out30	0x00
Byte_out15	0x00	Byte_out31	0x55

Fig - 11 Test page shown in a web browser

The IP address to be inserted in the browser is the one that the Gateway obtains from the Ethernet network DHCP, at startup.

It can be visualized using the Arduino monitor as shown in the picture aside.

Note that opening the Arduino monitor resets the Gateway and then the EtherCAT® communication will be interrupted. To restart it, the master must rescan the network.



The screenshot shows the output of the EasyCAT Gateway basic example. The text is as follows: "EasyCAT Gateway basic example", "Firmware version V_1.0", "STANDARD MODE", "32 Byte Out", "32 Byte In", "Sync = ASYNC", "Detected chip 9252 Rev 1", "EtherCAT interface initialized", "Ethernet interface Mac Address 54:10:EC:F5:7A:64", "Ethernet cable is connected", "Ethernet interface IP address 192.168.178.119", and "Web server started". The IP address "192.168.178.119" is circled in red.

```
EasyCAT Gateway basic example
Firmware version V_1.0

STANDARD MODE
32 Byte Out
32 Byte In
Sync = ASYNC
Detected chip 9252 Rev 1
EtherCAT interface initialized

Ethernet interface Mac Address
54:10:EC:F5:7A:64
Ethernet cable is connected
Ethernet interface IP address
192.168.178.119
Web server started
```

Fig - 12 Web server IP address

Specification

EtherCAT® interface	LAN9252 with 4K bytes configuration EEPROM
Ethernet interface	W5500 with unique MAC address preprogrammed in external EEPROM (128 bytes available for general purpose)
USB interface	Native Samd21 USB 2.0
USB programming	Through FTDI FX230 USB-serial converter
Microcontroller	SAMD21G18, 32-Bit ARM Cortex M0+ Clock speed 48 MHz Flash 256 KB Sram 32KB Fully compatible with the Arduino IDE
Power Supply	5V max. 350mA

----- Useful links -----

AB&T website

<https://www.bausano.net/en/>

EasyCAT Gateway page on the AB&T website

<https://www.bausano.net/en/hardware/ethercat-e-arduino/easycat-usb.html>

Easy Configurator user manual

https://www.bausano.net/images/arduino-easycat/EasyConfigurator_UserManual.pdf

Easy Configurator download

https://www.bausano.net/images/arduino-easycat/EasyConfigurator_V3_0.zip

EtherCAT Technology Group ETG

<https://www.ethercat.org/default.htm>

Basic Arduino Web server example

<https://www.arduino.cc/en/Tutorial/WebServer>

----- Document history -----

Rev 1.0 First release