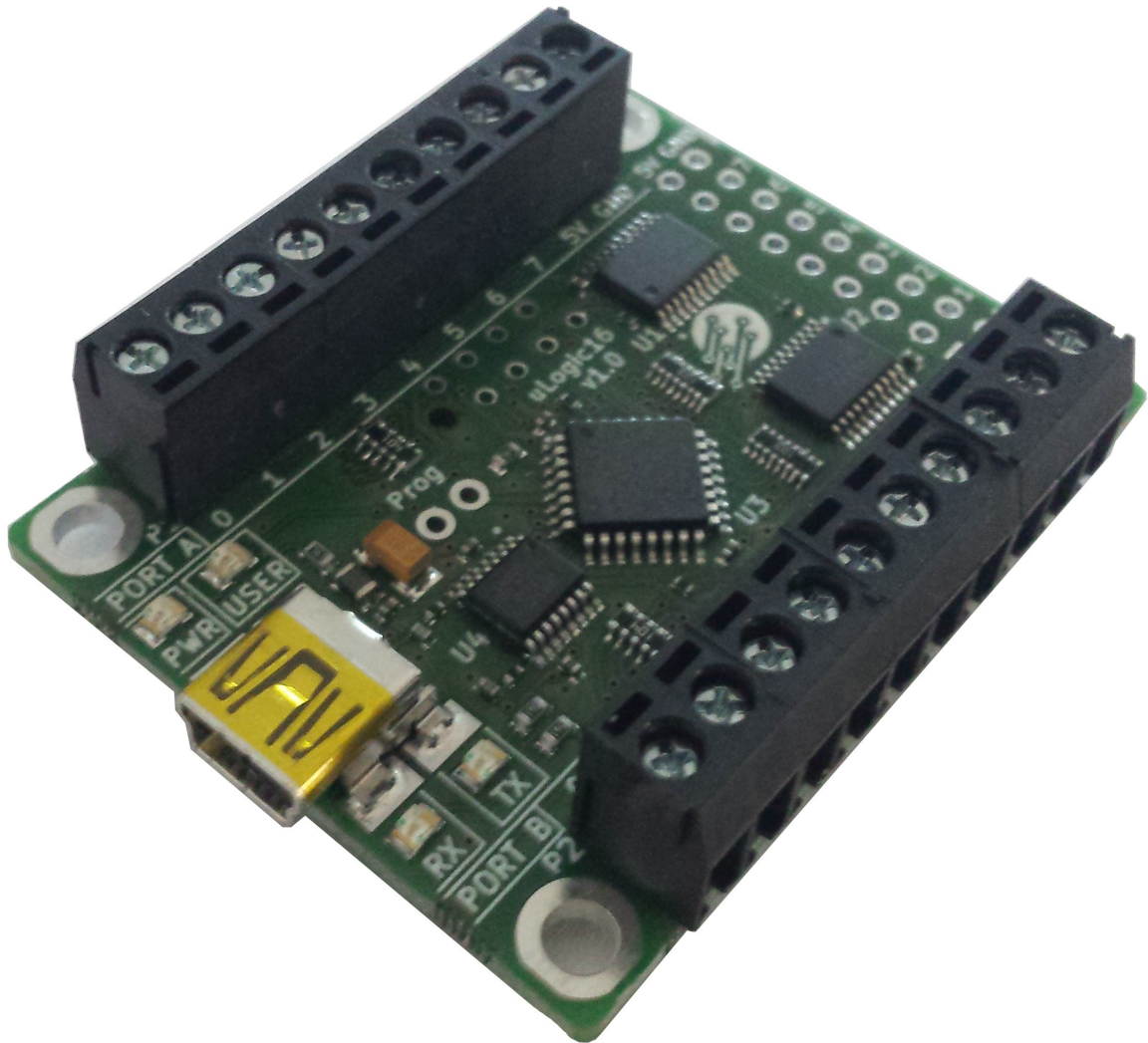




BS
Electronics

User manual Ulogic16





Contents

User manual.....	1
1 Introduction.....	3
2 Mode of operation.....	4
2.1 Mode Input-Output.....	4
2.1.1 Input.....	4
2.1.2 Output.....	5
2.2 Logic Mode.....	6
2.2.1 Description.....	6
2.2.2 Useful commands.....	6
2.2.3 Logic function.....	7
2.2.4 Internal function.....	7
3 Communication.....	9
3.1 Overview.....	9
3.2 Protocol.....	10
3.3 Comand list.....	10
3.4 Getting started.....	12
4 Hardware Features.....	13
4.1 General.....	13
4.2 Inputs/Outputs.....	13
4.3 Power supply.....	13
5 Contacts.....	14



1 Introduction

The μ Logic16 system is a board for the rapid deployment of logical functions.

With the dedicated software, it is possible to integrate your logical functions in a few clicks and make your systems autonomous without the development of microcontrollers or FPGAs that can be long and expensive.

Features:

- 2 ports configurable as input or output independently, accessible on a terminal block.
- 8 inputs or outputs by port.
- 2 operating modes: execution of logic diagrams or PC-controllable I / O.
- Simple communication protocol by ascii command compatible with any hyperterminal or specific software
- 5V power supply via USB port or terminal block.



2 Mode of operation

The system has two modes of operation selectable by a command on the serial port.

At startup, the system chooses the last mode used.

In the case of logic mode, if the memory containing the logic function information is corrupted, the system will automatically switch to the input-output mode.

2.1 Mode Input-Output

The input-output mode allows easy access to the state of the ports.

Two ports are available: PortA, PortB.

Each can be configured independently: input or output.

To select this mode, use the command "**MODE**<sp>**IO**<cr>".

2.1.1 Input

Configuration of an input port:

Use the command "**PORTX**<sp>**IN**<cr>".

Replace **X** by **A** or **B**.

(Ex : "**PORTA**<sp>**IN**<cr>" this command will configure the input portA).

There are two types of reading possible:

In both cases, use the command "**READ**<sp>**XXX**<cr>".

- Reading the entire port.

Replace **XXX** by **PORTA** or **PORTB**.

(Ex : "**READ**<sp>**PORTA**<cr>" this command returns a value between 0 and 255, this value corresponds to the state of the portA pins).

- Reading a bit.

Replace **XXX** by **PA0**, **PA1 ... PA7** or **PB0**, **PB1 ... PB7**.

(Ex : "**READ**<sp>**PA2**<cr>" this command returns 0 or 1 which corresponds to the state of bit PA2).

2.1.2 Output

Configuration of an Output port:

Use the command "**PORTX**<sp>**OUT**<cr>".

Replce **X** by **A** or **B**.

(Ex : "**PORTB**<sp>**OUT**<cr>" this command will configure the output portB).

There are several types of writing possible:

- Writing the entire port

Use the command "**WRITE**<sp>**PORTX**<sp>**VAL**<cr>".

Replace **X** by **A** or **B**.

R **VAL** by the value to be written on the port, **VAL** can be written in decimal (Ex :**63**) or in hexadecimal(ex :**0X3F**).

(Ex : "**WRTIE**<sp>**PORTA**<sp>**0X3F**<cr>" this command writes the value 63 to portA).

(Ex : "**WRTIE**<sp>**PORTA**<sp>**63**<cr>" this command writes the value 63 to portA).

- Bit Reset

Use the command "**CLR**<sp>**XXX**<cr>".

Replace **XXX** by **PA0** , **PA1** ... **PA7** or **PB0** , **PB1** ... **PB7**.

(Ex : "**CLR**<sp>**PB3**<cr>" this comand reset PB3).

- Bit Set

Utiliser la commande "**SET**<sp>**XXX**<cr>".

Replace **XXX** by **PA0** , **PA1** ... **PA7** or **PB0** , **PB1** ... **PB7**.

(Ex : "**CLR**<sp>**PB3**<cr>" " this comand set PB3).

It is still possible to use the read commands to check the status of the outputs.

2.2 Logic Mode

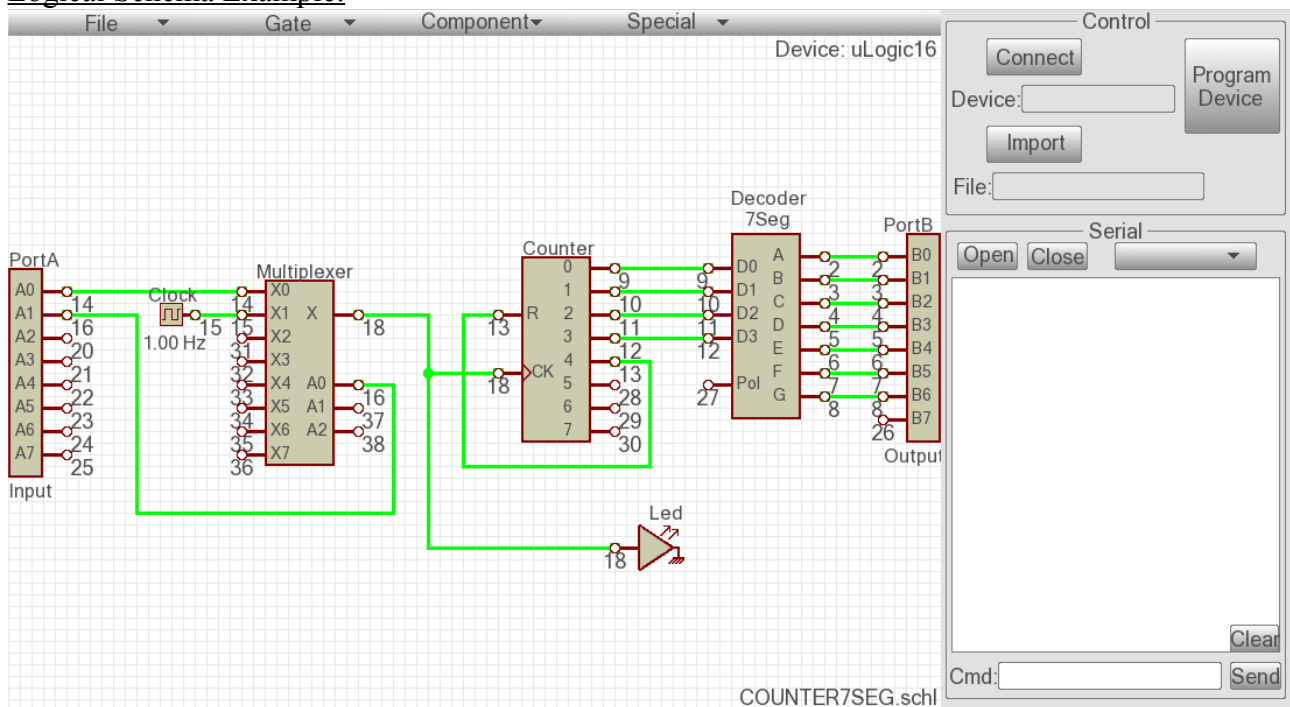
2.2.1 Description

The logical mode allows the card to become autonomous.

The inputs and outputs are controlled according to the logical diagrams defined by the user.

The logical mode is selected by default when programming by the user.

Logical Schema Example:



In the example above, portB is intended to be connected to a 7-segment display.

Still on this example, we see a counter that drives a 7-segment decoder.

This counter is clocked by an internal or external clock that can be chosen via the PA0 bit and the multiplexer.

The USER led on the card is used to view the clock.

2.2.2 Useful commands

In the logic operation mode, the write and read commands on the ports remain functional.

However, writing to a pin that is already driven by the logic diagram does not make sense and will not work properly.

It is also possible to access all the internal bits of the logic diagrams.

Use command "**READM**<sp>**XXX**<cr>".

Replace **XXX** by the number written below the pin to be analyzed.

(Ex : "**READM**<sp>**18**<cr>" this command retrieves the state of the clock bit that is clocking the counter).

By following the same method, it is possible to write on any internal bit.

On the other hand, the problem is the same as for the ports, to write on a bit which is already driven will not work or will generate an unexpected operation.

(Ex : "**CLRM**<sp>**27**<cr>" this command reset the input POL of decoder).

(Ex : "**SETM**<sp>**27**<cr>" this command set the input POL of decoder).

2.2.3 Logic function

List of logical functions:

- AND, OR, XOR, NAND, NOR, XNOR, NOT.
- Flip flop D(Set,Reset), Flip flop RS(NOR)
- Multiplexer 8bit, Démultiplexer 8bit
- Shift register 8bit
- Latch8bit
- Compteur 8bit
- Décodeur Hexa → 7segment
- Monostable, Debounceu, Clock
- Logic state

2.2.4 Internal function

It is important to understand that the uLogic simulates all logical functions.

The logic functions are not physically present in comparison with an FPGA where the programming allows the wiring of gate that actually exists internally.

The treatment process works in 5 steps:

- Retrieving and storing the status of the inputs.
- Execution of all internal logical functions.
- Reading the memory to drive on the outputs and the led.
- Serial link management (if needed).
- Calculation of the treatment time.



BS
Electronics

All of these tasks require a processing time that can become problematic for fast signals.

The simplest program (16 input outputs, 1 LED) will take about 4 μ S.

Each logical function will take between 0.5 μ S to 3 μ S on average. This time depends on the complexity of the logical function used.

Ex: A program that requires 10 μ S of processing will not work properly if incoming signals change states every 9 μ S.

Using the serial link is a time-consuming process, so avoid using it too often for fast signal processing.

It is possible to get the processing time with the command: *"READ<sp>TIME<cr>"*.

3 Communication

3.1 Overview

The communication is established via a USB port, thanks to the USB driver → UART integrated on the card (FT230) communication is simple, it comes down to a simple serial link.

Communication is possible via any specific terminal or software.

Just install the appropriate driver.

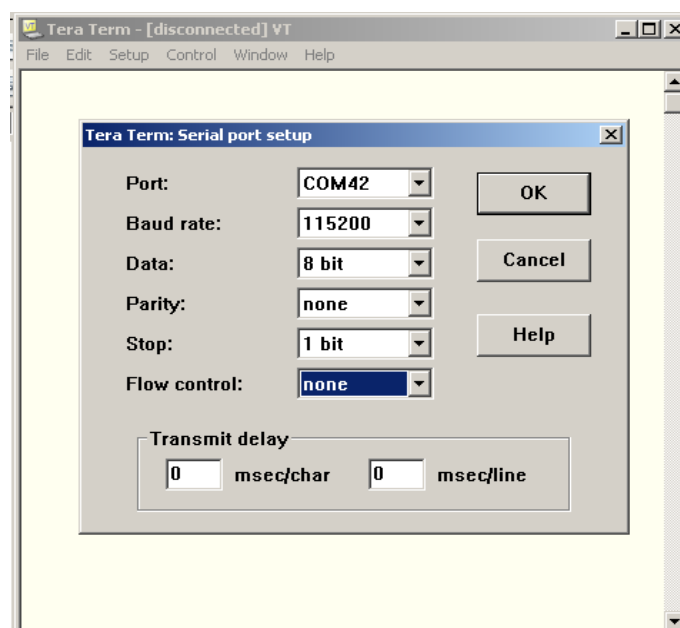
Under windows, the driver should be found automatically.

If not, you can download it from the FTDI website: <http://www.ftdichip.com/Drivers/VCP.htm>

Serial link configuration:

- 115200 Baud.
- 8 Bits Data width.
- 1 stop bit.
- No parity.
- No hardware flow control.

Example of configuration under Tera Term.



3.2 Protocol

All command have the same construction and must be written in ascii and capital letters imperatively.

The construction is as follows. Everything in red must be sent via the serial link.

"**COMMAND**<sp>**PARAMETER**(optional)<cr>"

<sp> Corresponds to the space character (ascii value : 0x20).

<cr> Corresponds to the carriage return character (ascii value : 0x0D), indicates the end of the command. It is often generated automatically on the terminal when pressing Enter.

When receiving a command, the card returns OK or a value according to the command.

If the command is not correct an error message is returned.

3.3 Comand list

Reading commands :

Command	Parameter	Description	Reply	Operating mode
READ	PORTA, PORTB	Returns the decimal value of the port.	Value between 0 and 255. Ex : "128<cr>"	Logic mode. I/O mode.
READ	PA0...PA7, PB0...PB7	Returns the state of the selected bit.	Value between 0 and 255. Ex : "0<cr>"	Logic mode. I/O mode.
READ	MODE	Returns operating mode.	"Mode Logic<cr>" "Mode IO<cr>"	Logic mode. I/O mode.
READ	TIME	Returns the processing time.	Ex : "10,2<sp>uS<cr>" Logic mode.	Logic mode.
READM	XXX (valeur decimale)	Returns the state of internal bit. (XXX). Ex : "READM<sp>18<cr>"	Value between 0 and 1. EX : "1<cr>"	Logic mode.



Commandes d'écriture :

Command	Parameter	Description	Reply	Operating mode
SETM	XXX(valeur decimale)	Set the internal bit (XXX). Ex : "SETM<sp>11<cr>"	"OK<cr>"	Logic mode.
CLRM	XXX(valeur decimale)	Reset the internal bit (XXX). Ex : "CLRM<sp>14<cr>"	"OK<cr>"	Logic mode.
SET	PA0...PA7, PB0...PB7	Set the external bit. Ex : "SET<sp>PA0<cr>"	"OK<cr>"	Logic mode. I/O mode.
CLR	PA0...PA7, PB0...PB7	Reset the external bit. Ex : "CLR<sp>PB2<cr>"	"OK<cr>"	Logic mode. I/O mode.
WRITE	P1 :PORTA, PORTB. P2 :VAL.	Write on port (P1) the value on(P2). Ex : "WRITE<sp>PORTA<sp>44<cr>" Ex : "WRITE<sp>PORTA<sp>0X2C<cr>" P2 can be decimal or hexadecimal	"OK<cr>"	Logic mode. I/O mode.
LEDON	N/A	Turn on the led user Ex:"LEDON<cr>"	"OK<cr>"	Logic mode. I/O mode.
LEDOFF	N/A	Turn off the led user Ex:"LEDOFF<cr>"	"OK<cr>"	Logic mode. I/O mode.

Other commands :

Command	Parameter	Description	Reply	Operating mode
PORTA	IN, OUT	Configures the direction of portA. Ex:"PORTA<sp>IN<cr>"	"OK<cr>"	I/O mode.
PORTB	IN, OUT	Configures the direction of portB. Ex:"PORTB<sp>OUT<cr>"	"OK<cr>"	I/O mode.
MODE	LOGIC, IO	Select the operating mode Ex"MODE<sp>LOGIC<cr>"	"OK<cr>"	Logic mode. I/O mode.
DEVICE	N/A	Returns the type of card.	Ex : "uLogic16<cr>"	Logic mode. I/O mode.
VERSION	N/A	Returns the version of the internal software	Ex : "v1.0<cr>"	Logic mode. I/O mode.
ERASE FLASH	N/A	Clears the memory of logic mode. Ex : "ERASE<sp>FLASH<cr>"	"OK<cr>"	Logic mode. I/O mode.

3.4 Getting started

Connect the PC to the uLogic card.

Windows must detect the device, it will look for the driver of the USB interface alone.

If not, or if you are using another operating system, you can download it at the following address : <http://www.ftdichip.com/Drivers/VCP.htm>

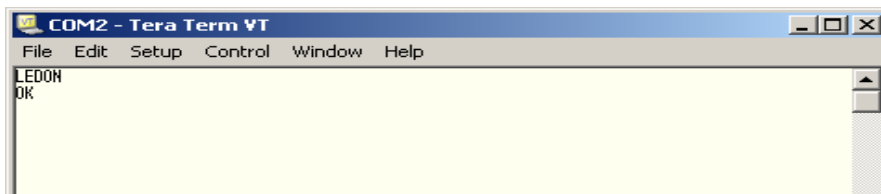
Once the driver is correctly installed there are two possibilities:

- Use a terminal (ex:TeraTerm).

Configure the terminal as explained in the communication chapter.

Once the port is correctly configured, enter the <LEDON> command and push enter, the TX and RX LEDs flash and the USER LED will light up.

Below is a command example.



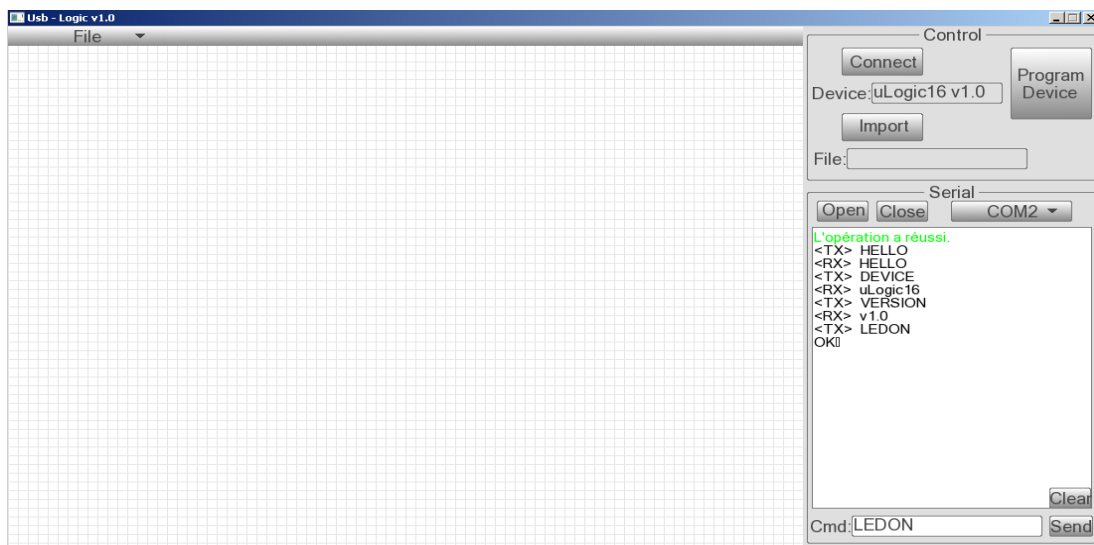
- Use the LogiControl software available at the following address.

Click on connect, the software should automatically detect the uLogic card.

The card must return its name and software version.

Enter the commands in the bar at the bottom right then click on send.

Below is a command example.



4 Hardware Features

4.1 General

The uLogic card includes:

- A mini USB connector for communication with the PC.
- A led indicating the state of the power supply (PWR).
- Two LEDs indicating data transfers between the card and the PC (TX and RX).
- A led usable directly by the user (USER).
- Two rows of terminal blocks to access the ports.
- A microcontroller STM32F33.

4.2 Inputs/Outputs

The card has a buffer (74ACT245MTCX) per port with the following characteristics.

current sink/source MAX	50mA
Input voltage MAX	0V to VCC (5V)
Low input voltage MAX	1,5V
Hight input voltage MIN	2V

For more information, refer to the manufacturer's documentation at the following address:

<http://www.onsemi.com/pub/Collateral/74ACT245-D.pdf>

4.3 Power supply.

The board can be supplied by usb port or terminal blocks.

Supply voltag MAX	5,5V
Supply voltag MIN	4,5V
Current consumption	~40mA



BS
Electronics

5 Contacts

Feel free to share your impressions, problems and ideas for improvement.

Mail : bastien@sclafer.com

Web site : <http://www.bselectronics.fr/>