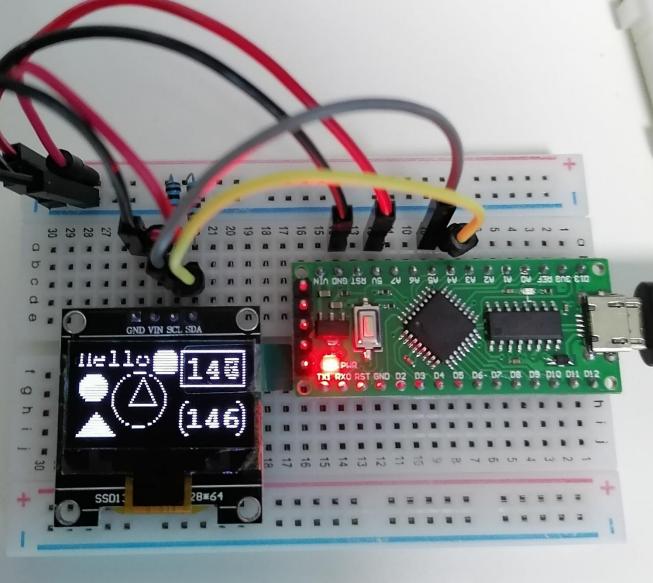


LGT8F328P

by **Ricky Gai** Revision 1.0



JANUARY 9, 2021

Nexuz Innovation, Malaysia. (MA0255412-M)

Introduction

About this Book

This is a practical information documented to illustrate the use of LGT8F328P chips that has been tested and run successfully on selected Arduino sketch projects.

This book is focusing on hands-on result merely because the actual circuitry implemented may not be the same as the original circuit diagram.

In the learning of Arduino clone chips, often came to frustration due to lack of information, online comments are not clearly stated, thus leaving me in the dark. In short, a great student not neccesary can be a great teacher right?

Therefore, I decided to write this book to document issues encountered, resolved and share to the public, it is not perfect but with necessary guidelines to get through and move on.

LGT8F328P is one of the Arduino clone chip created by Logic Green.

About the Author

Ricky Gai

is the founder and technical director of Nexuz Innovation, a small R&D IT company established in Kuala Lumpur, Malaysia.

After receiving certification from Oxford Computer Engineering discipline in 1992, my carreer was mostly exposed to C/C++ system software development for decades about 30 years since the MS-DOS time until today's Windows environments including real-time, networking, file system, 2D/3D games, software driver, application and mobile programming.

Nevertheless, much spare time devoted to further the electronics studies for two years before coming to Arduino platform, and my wife often staring at me. Arduino programming reminded me the MS-DOS season, it brought back memory of something like interrupt, vector and bootsector (eg. Bootloader in Arduino).

The reference materials and source code are available via the Github page <u>HERE</u> and you may reach me via the support email at <u>support@nexuzinnovation.com</u> for any error found, suggestions and questions, I will try to get back to you.

Well, passion is everything and the key to success, I hope you find something useful here.

DISCLAIMER

Abbrevation	Descriptions
NEXUZ INNOVATION / AUTHOR	refers to the author, Ricky Gai.
READER / READER(S) / READER'S	refers to the person who read or knowledge transferred who accessed the circuitry setup based on the contents of illustrated in this document.
COMPONENTS / EQUIPMENTS	refers to electronics components, tools, materials that used for handons to setup the circuitry.
IP / INTELLECTUAL PROPERTY / COPYRIGHT / PERMISSION	refers to the copyrighted materials (eg. Photo, Diagram, Source Code, Links) that owned by other creators.

THIS CONTENTS OF THIS DOCUMENT IS SOLELY BASED ON THE ELECTRONICS COMPONENTS OR MATERIALS ADOPTED AND TESTED BY AND AT THE AUTHOR'S PREMISES. DUE TO VOLTAGE SUPPLY VARIES FROM DIFFERENT COUNTRIES AND PHYSICAL SPECIFICATION OF COMPONENTS MAY CHANGE FROM TIME TO TIME, READER(S) TAKE OWN RESPONSIBILITY TO ACCESS THE EXPERIMENTAL SOLUTIONS BASED ON THE INFORMATION DESCRIBED IN THIS DOCUMENT.

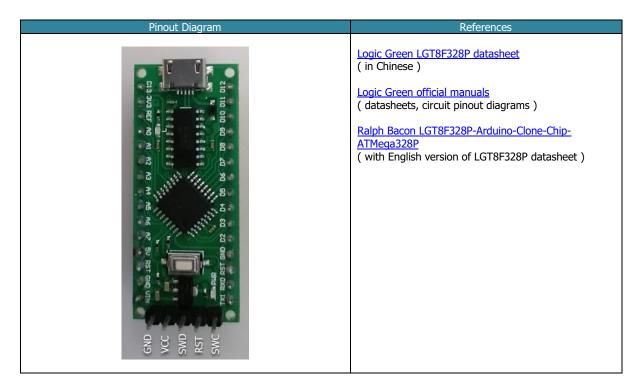
THE COMPANY NEXUZ INNOVATION AND THE AUTHOR BARES NO RESPONSIBILITY UPON ANY DAMAGE OR HARM IN CASE HAPPENS TO THE READER(S) OR READER'S SIDE, THIS INCLUDE READER'S RELATED SUCH AS HUMAN HEALTH, HARDWARE EQUIPMENTS AND OTHER LOSSES.

NEXUZ INNOVATION IS A SOLE PROPRIETORSHIP COMPANY WITH NO RELATION TO ANY HARDWARE MANUFACTURERS OR VENDORS MENTIONED IN THIS DOCUMENT, SUCH AS LOGIC GREEN, ARDUINO AND ETC. THE MENTIONED OF INTEGRATED CIRCUIT (IC) OR CHIPS PRODUCT MODEL ARE SOLELY FOR STUDY AND REFERENCE PURPOSES ONLY.

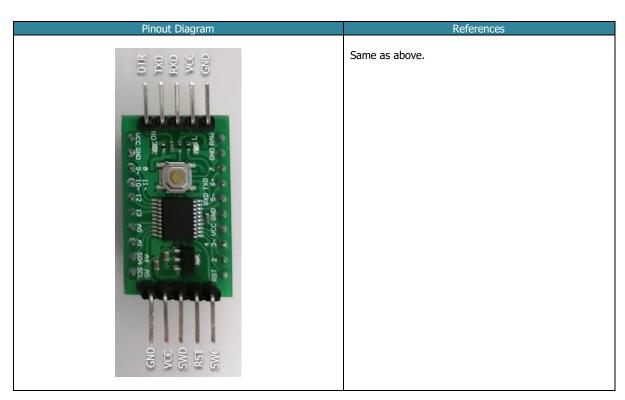
THE AUTHOR RESPECTS THE INTELLECTUAL PROPERTY FROM OTHER CREATORS, THIS DOCUMENTATION MAY SHARE SOME PARTLY EXTRACTED PORTION OF PHOTO OR DIAGRAM AS PART OF THE ILLUSTRATION USAGE. IF THERE IS VIOLATION TO YOUR IP, PLEASE DO INFORM THE AUTHOR TO EXCLUDE FROM THIS DOCUMENT.

Types of LGT8F328P chips

LGT8F328P LQFP32 MiniEVB



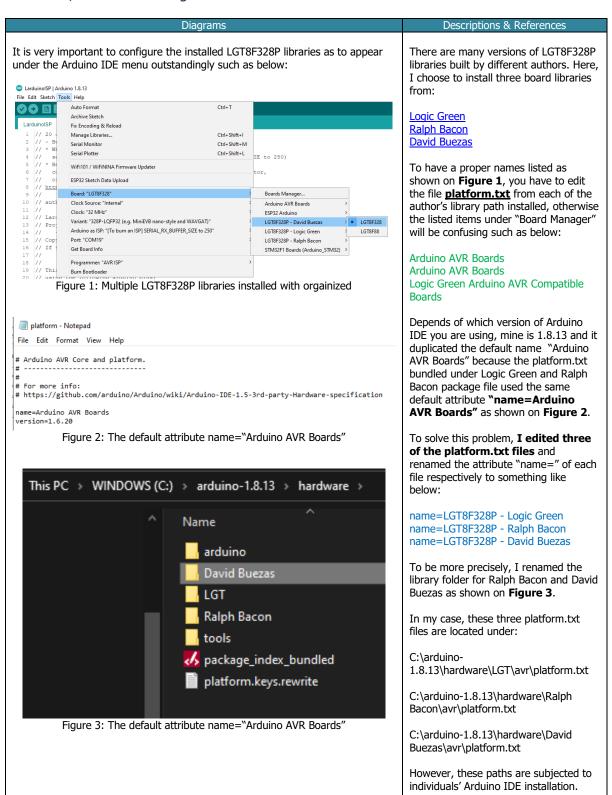
LGT8F328P SSOP20 MiniEVB



Arduino Sketch Projects

Arduino IDE - Setup the LGT8F328P library

Download, install and configure the LGT8F328P libraries



LarduinoISP - How to burn the BOOTLOADER

Introduction

ISP Programmer 10 km Target to burn BOOTLOADER to.

Figure 4: LarduinoISP working circuitry setup

Descriptions & References

LarduinoISP is the use of Arduino ATMega328P (NANO) or LGT8F328P (LQFP32 MiniEVB) as **ISP Programmer** to burn BOOTLOADER to the Target LGT8F328P SSOP20 MiniEVB as shown at **Figure 1**.

Here, I want to share some painful experiences that went through in dealing with the LarduinoISP circuitry setup.

I started by referring to <u>LGTMCU/LarduinoISP</u>, at that time I was doubted about the short-circuit of the $10k\Omega$ resistor between VCC and SWD pin as shown at **Figure 5**, with two green highlighted circles below:

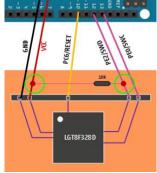


Figure 5: LGT/LarduinoISP extracted diagram

Really wasted much time on it, no one from the <u>Arduino Forum</u> can gives me at least a snapshot photo of how to setup the physical circuitry correctly.

Based on **Figure 4**, I decided to remove the SWD (blue cable) from connecting to the $10k\Omega$ pull-up resistor and connect it straight to pin 12 of the ISP Programmer.

Next, I used <u>David Buezas's LarduinoISP</u> sketch to upload to the ISP Programmer because his sketch is more recently updated.

Somehow, still failed to upload the sketch (may be that time Arduino Nano was used as ISP Programmer). Later, I found a site called SuperUserNameMan/LGTISP, after reading its **usage** section, I begin to realize the whole story and quickly edit the **HardwareSerial.h** to to change the following instruction from:

#define SERIAL RX BUFFER SIZE 64

to:

#define SERIAL_RX_BUFFER_SIZE 250

Eventually, LarduinoISP successfully burned the BOOTLOADER to the Target LGT8F328P SSOP20 MiniEVB.

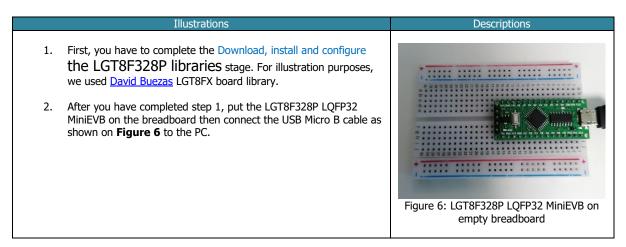
The entire circuitry a very straight forward actually but took me sometime to accomplish.

We will cover more details about the hands-on process on next page.

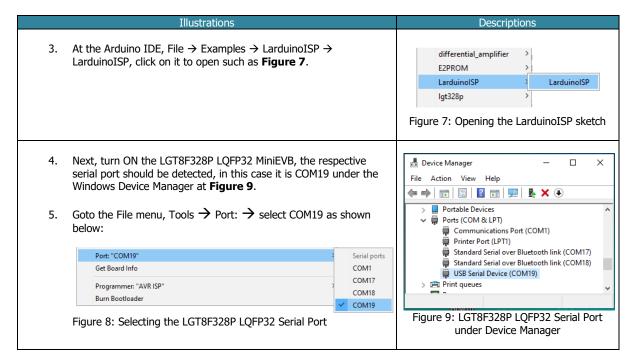
The Parts list

Components	Quantity
LGT8F328P LQFP32 MiniEVB	1
LGT8F328P SSOP20 MiniEVB	1
FTDI FT232RL	1
RESISTOR 10kΩ	1
10 cm Female-to-Male Jumper Wire	5
10 cm Female-to-Female Jumper Wire	5
10 cm Male-to-Male Jumper Wire	1
USB 2.0 Micro B cable connector	1
USB 2.0 Mini B cable connector	1
MB102 Mini Breadboard 8.5CM x 5.5CM 400 Holes	2

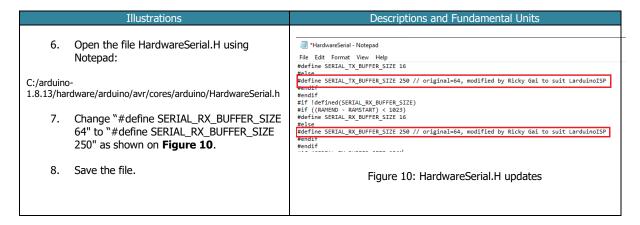
Installing the LGT8F328P board library



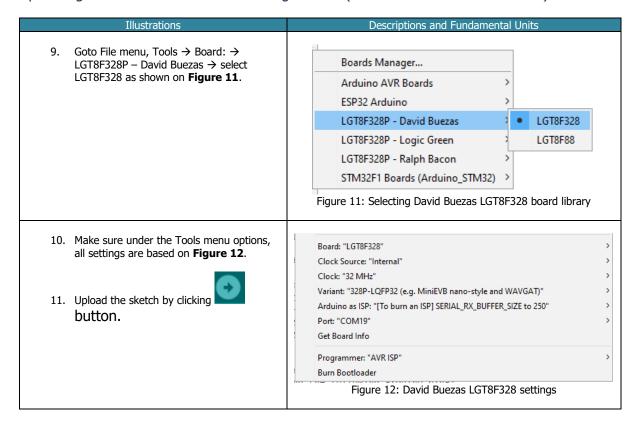
Opening the LarduinoISP sketch



Updating HardwareSerial.H



Uploading LarduinoISP sketch to ISP Programmer (LGT8F328P LGFP32 MiniEVB)



The success message of LarduinoISP sketch uploaded to ISP Programmer.

Descriptions

12. If the LarduinoISP sketch is successfully uploaded to ISP Programmer (LGT8F328P LQFP32 MiniEVB), you should be seeing messages below:

Wiring up the LarduinoISP circuitry

Illustrations

- 13. Now, the ISP Programmer is ready.
- 14. Turn off the ISP Programmer.
- 15. We are going to wire up the LarduinoISP circuitry just like **Figure 14**.
- 16. LarduinoISP wire connection table:

LGT8F32P SSOP20	LGT8F328 LQFP32
(Target)	(ISP Programmer)
GND	GND
VCC	5V
SWD	D12
RST	D10
SWC	D13

Figure 13: LarduinoISP wire connection

Descriptions

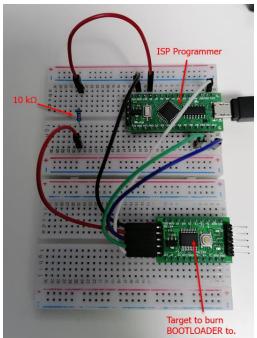
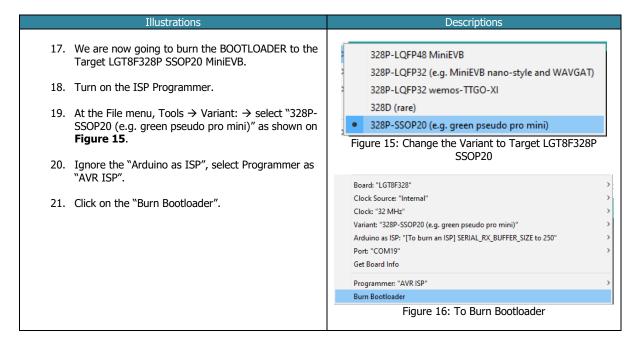
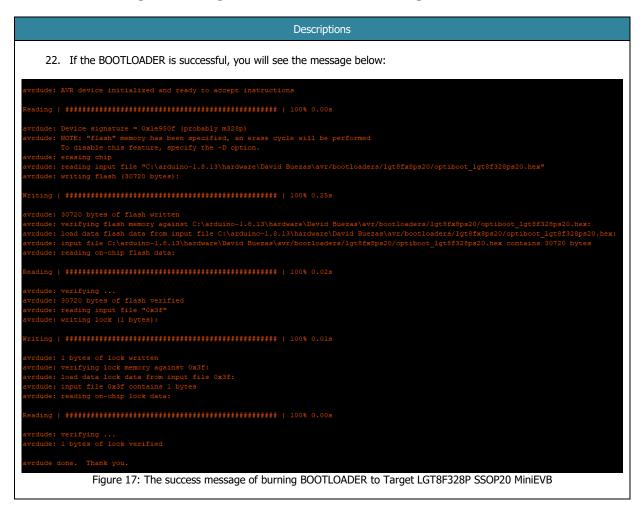


Figure 14: LarduinoISP circuitry setup

To burn BOOTLOADER to Target LGT8F328P SSOP20 MiniEVB



The success message of burning the BOOTLOADER to the Target.



Testing the LGT8F328P SSOP20 MiniEVB with new BOOTLOADER burned.

23. Congratulations, the Target LGT8F328P SSOP20 MiniEVB is burned with new BOOTLOADER image.

Illustrations

- 24. Here, we are going to do a test on it by uploading a fast blinking sketch to see the significant changing effects. If it is uploaded successfully, the LGT8F328P SSOP20 MiniEVB should be blinking faster at 64 ms interval.
- 25. Turn off the ISP Programmer, connect the LGT8F328P SSOP20 MiniEVB to FTDI FT232RL as shown on **Figure 19**.
- 26. The FTDI FT232RL connection, its CTS pin is not required here:

LGT8F32P SSOP20	FTDI FT232RL
GND	GND
VCC	5V
RXD	TXD
TXD	RXD
DTR	DTR

Figure 18: The standalone LGT8F328P SSOP20 MiniEVB pin connection with FTDI FT232RL

- 27. Turn on the FTDI FT232RL that will power up LGT8F328P SSOP20 MiniEVB.
- 28. NOTE: This will create a new serial port COM22 interface as shown on **Figure 20**.

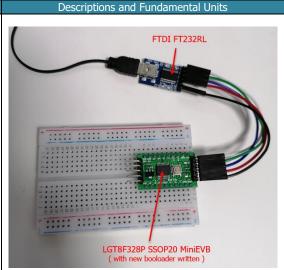


Figure 19: The standalone LGT8F328P SSOP20 MiniEVB with FTDI FT232RL interface

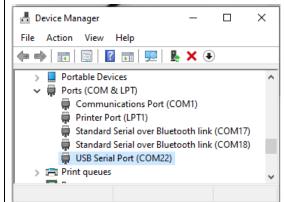


Figure 20: FTDI FT232RL new serial port interface

29. Take a look at the LGT8F328P SSOP20 MiniEVB by default itself, it is blinking at internal roughly two seconds as shown on **Figure 21**.

To see the different, later when the new sketch blink64ms.ino is uploaded to it, the blue LED should be blinking faster at 64 ms interval.

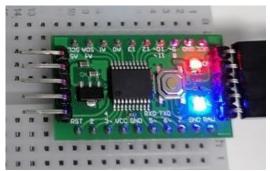


Figure 21: FTDI FT232RL default blinking (blue light) two seconds interval

Uploading blink64ms.ino sketch to LGT8F328P SSOP20 MiniEVB via FT232RL

Descriptions

- 30. Here, we are to upload a sketch called blink64ms.ino to the LGT8F328P SSOP20 MiniEVB since the Bootloader already written, if it is working normally after a reset, the Bootloader should starts and jumps to the program address where the blink64ms.ino program will be loaded and run.
- 31. Select the correct Port COM22 (FT232RL) from the menu as shown below:



Figure 22: Selecting the FTDI FT232RL serial port COM22

- 32. At Arduino IDE, open the blink64ms.ino sketch then click the button to upload sketch.
- 33. If it is uploaded successfully you will notice the blue led will blink faster continuously.

The success message of uploading blink64ms.ino to LGT8F328 SSOP20 MiniEVB

#