

MMB CAN V1.0

User Manual



Revision Log

Version	Date	Revisions
v1.00	23rd August 2023	Initial Version

CONTENTS

Revision Log
Product Profile
Feature Highlights
Specifications
Firmware Support
Product Dimensions
Peripheral Interfaces
Pin Description
Interface Introduction
USB Power Supply
Servo Wiring
RGB-WS2812 Wiring 7
Sensor (e.g., CRT5000 infrared sensor) Wiring7
I2C (e.g., AHT10 temperature and humidity sensor) Wiring
Endstop (e.g., Hall sensor) Wiring8
Klipper Firmware
Flashing CANBOOT
Compiling Klipper Firmware10
Firmware Update via CANBOOT12
Firmware Update via DFU13
CAN Bus Configuration
Configuring Klipper15

Product Profile

BIGTREETECH MMB CAN V1.0 is a control board for multi-material 3D printing

system, e.g., ERCF. It simplifies wiring by communicating via USB or CAN.

Feature Highlights

- The board has BOOT and RESET buttons, allowing users to update firmware in DFU mode via USB.
- I2C interface reserved for filament run out/clogging detection, or other DIY functions.
- The power interface has reverse polarity protection to prevent the board from being damaged if power cables are connected incorrectly during DIY.
- Supports CAN or USB communication, with selectable 120R terminal resistance for CAN and reserved CAN expansion interface.
- USB port has ESD protection to prevent controller damage from electrostatic discharge.
- Uses XT30 interface for CAN communication and board power supply, simplifying wiring.
- Stepper motor driver supports high and low voltage selection for DIY use.

Specifications

125mm x 54mm
Refer to
BIGTREETECH MMB CAN V1.0-SIZE.pdf
ARM Cortex-M0+ STM32G0B1CBT6 64MHz
DC12V-DC24V 9A
DC 3.3V
5V 2A, peak 2.5A
STP1-STP11, I2C, RGB, Sensor (infrared sensor
interface), USB, CAN
EZ Drive (supports voltage selection)
STEP/DIR, UART, SPI
M1, M2, M3, M4
USB Type-C
2.04
3.0A

Firmware Support

Currently, MMB CAN V1.0 only supports Klipper firmware.

Product Dimensions



Peripheral Interfaces

Pin Description



Interface Introduction

USB Power Supply

After the board is powered on, the power indicator light will turn on, indicating that the power supply is normal. The VUSB label on the board is the power selection terminal, and a jumper is needed to short VUSB only when using USB to power the board.



Servo Wiring





RGB-WS2812 Wiring

Sensor (e.g., CRT5000 infrared sensor) Wiring





I2C (e.g., AHT10 temperature and humidity sensor) Wiring

Endstop (e.g., Hall sensor) Wiring



Klipper Firmware

Flashing CANBOOT

Note: CanBoot is for updating MCU firmware directly via CAN bus. If you prefer DFU, skip this step.

To flash CanBoot on Raspberry Pi or CB1, follow the instructions at https://github.com/Arksine/CanBoot

1. Enter

cd ~
Navigate to the main directory and input
git clone <u>https://github.com/Arksine/CanBoot</u>
to download the CanBoot project. Then enter

cd CanBoot

navigate to the CanBoot directory.

2. Enter

make menuconfig

configure as shown in the provided image:

(10)	p)
	Katapult Configuration v0.0.1-57-gabd1545
	Micro-controller Architecture (STMicroelectronics STM32)>
	Processor model (STM32G0B1)> Build Katapult deployment application (Do not build)>
	Clock Reference (8 MHz crystal)>
	Communication interface (CAN bus (on PB0/PB1))>
(10)	Application start offset (okib offset)
() [*] []	GPIO pins to set on bootloader entry Support bootloader entry on rapid double click of reset button Enable bootloader entry on button (or gpio) state Enable Status LED
[Spa [Q]	ace/Enter] Toggle/enter [?] Help [/] Search Quit (prompts for save) [ESC] Leave menu

3. Enter **make** to compile the firmware. The resulting **canboot.bin** file will be in the **home/biqu/CanBoot/out** folder. This can be directly downloaded to

_																	
192.10	58.1.113																
erminal	Sessions	View	X server	Tools	Games	Settings	Macros	Help									
	+2.		-	+			11		181	.0	0						
-	- CC	1	19.00			-	T	-		Φ.	•						
ession	Servers	Tools	Games	Sessions	View	Split	MultiExec	Tunneling	Packages	Settings	Help						
Quick o	connect.							1	3.1	92.168.1.11	3	×	4				
-	1 7 0		0.5					C	ompiling	out/sr	Viniti	al nins o					
-	<u>• • •</u>	0	U A					č	ompiling	out/sr	/led.o	rc_p charte					
/hom	e/biqu/CanB	oot/out/						🖌 с	ompiling	out/sr	/gener	ic/armcm_c	anboot.o				
	Name			Size (KE) La	t modified	Owner	C	ompiling	out/sr	:/stm32,	/gpio.o					
	board-oe	neric			20	22-12-28	bigu	C	ompiling	g out/sr	:/stm32,	flash.o					
	compile t	ime requi	ect tyt	1	20	77-17-78	bigu		ompiling	out/sr	:/stm32,	clockline	.0				
	comple_t	ime requ	esto	64	20	22-12-20	bigu		ompiling	out/sr	/stm32	atu_reboo	t.0				
-	comple_t	ime_requ	est d	1	20	22.12.20	bigu	č	ompiling	out/sr	/gener		citt o				
	a comple_t	ime_requ	est c	2	20	22.12.29	bigu	č	ompiling	out/sr	/stm32	/stm32f0_t	imer.o				
	canhoot	navload /	Colic.	22	20	22-12-20	bigu	c	ompiling	out/sr	/stm32	/stm32g0.0					
	canboot	work		4	20	22-12-20	bigu	c	ompiling	out/sr	/stm32,	gpioperip	h.o				
	canboot	alf		707	20	22-12-20.11	bigu	C	ompiling	g out/sr	/gener	ic/canseri	al.o				
	canboot	hin		4	20	22-12-20	bigu	C	ompiling	j out/sr	://lu	p/fast-has	h/fastha	sh.o			
_	board-lin								ompliing	j out/sr	/stm32,	rdcan.o					
	board	Dpe	en					l c	ompiling	out/sr	/gener	chipid o					
	b autoconf	Dpe	en with def	fault text	editor			i i	uilding	out/com	ile ti	ne request					
_	n autocom	One	on with					P	reproces	sing ou	/src/a	eneric/arm	icm link.	ld			
_		112 000	a with dat	fault areas				1 1	inking o	ut/canb	not elf						
		Ope	en wich dei	aun prog	ditter			C	reating	bin file	out/ci	anboot.bin	1				
		Con	npare file	with				biq	u@BTT-CE	1:~/Can	Boot\$						
		👲 Dov	vnload					100	PTT CD1			0.01.00.40	06 CR -	0.0116/2	0.00000	[77] . 24 min	-
		E a l						193	B11-C81	1%		0.21 GB / 0.	30.00	0.01 MD/S	• 0.00 MD/s	1920 34 min	E DIQ

your computer from the SSH software's left panel.

- 4. Hold the **Boot** button and connect the board to Raspberry Pi/CB1 via Type-C cable to enter DFU mode.
- 5. Enter **Isusb** in the SSH terminal to query the DFU device ID.

pi@1		ldpi:~\$	lsusb				
Bus	001	Device	005:	ID	0483:df11	STHicroolactronics STM Device in DFU Mode	
Bus	001	Device	004:	ID	1d50:6061	OpenMoko, Inc. Geschwister Schneider CAN adapter	
Bus	001	Device	003:	ID	0424:0c00	Microchip Technology, Inc. (formerly SMSC) SMC9512/9514 Fast Ethernet A	Adapter
Bus	001	Device	002:	ID	0424:9514	Microchip Technology, Inc. (formerly SMSC) SMC9514 Hub	
Bus	001	Device	001:	ID	1d6b:0002	Linux Foundation 2.0 root hub	
pi@1	Fluic	ldpi:~\$					

- Enter the following command to flash CanBoot make flash FLASH_DEVICE=0483:df11 replacing 0483:df11 with the actual device ID found in the previous step.
- 7. After flashing, disconnect the Type-C cable.

Compiling Klipper Firmware

 Connect to CB1/Raspberry Pi via SSH and enter the following commands: cd ~/klipper/ make menuconfig Configure the firmware as shown in the provided image (update)

Configure the firmware as shown in the provided image (update Klipper firmware to the latest version if options are not available).



- 2. After configuration, press **q** to exit, and select **Yes** when prompted to save.
- 3. Enter **make** to compile the firmware. The resulting **klipper.bin** file will be in the **home/pi/klipper/out** folder. This can be directly downloaded to your computer from the SSH software's left panel.

W 192.1	68.1.107										
Terminal	Sessions	View	X server	Tools	Games	Settings	Macros	Help			
	181	1		*		=	Y	••	14	*	0
Session	Servers	Tools	Games	Sessions	View	Split	MultiExec	Tunneling	Packages	Settings	Help
Quick	connect.								1	4.1	192.168.1.107 × 🚱
	+ TO		0 .						Co	mpiling	g out/src/buttons.o
/hor	ne/pi/klipper/or	ut/			-					mpiling mpiling	g out/src/tmcuart.o
1 -	Name			Size (K	l) La	st modified	Owner		Grt Co	mpiling	g out/src/pulse_counter.o g out/src/stm32/watchdog.o
1	src				20	22-03-08	pi		co Co	mpiling mpiling	g out/src/stm32/gpio.o g out/src/stm32/clockline.o
44	lb				20	22-03-08	pi	1	pi Co	mpiling	out/src/generic/crc16_ccitt.o
	board-ger	neric			20	22-03-08	pi	1	co Co	mpiling	g out/src/generic/armcm_boot.o
•	klipper.elf			1635	20	22-03-08	pi	1	co	mpiling	g out/src/generic/armcm_irq.o
_	kloper.dc	t		6	20	22-03-08	pi	1		mpiling	g out/src/generic/armcm_reset.o
	klipper.bin			23	20	22-03-08	pi	1		mpiling	out/src/stm32/stm32h7.0
	compile_ti compile_ti compile_ti compile_ti board-link board h autoconf.	me_requ me_requ me_requ h	est.bxt est.o est.d est.c	Open wit Open wit Open wit Compare Downloa	th defaul th th defaul th defaul d	t text edito t program.	r			mpiling mpiling mpiling mpiling mpiling mpiling mpiling	<pre>out/src/generic/armcm_timer.o out/src/stm32/gpioperiph.o out/src/stm32/stm32h7_adc.o out/src/stm32/stm32h7_spi.o out/src/stm32/usbotg.o out/src/stm32/chipid.o out/src/generic/usb_cdc.o out/src/stm32/hard_pwm.o</pre>
<			2	Delete Rename Copy file	path		-		ers Pr Li	ion: v0 eproces nking o eating	out/compile_time_request.o 0.10.0-278-g7c964e5f ssing out/src/generic/armcm_link.ld nut/klipper.elf hex_file_out/klipper.bin

Firmware Update via CANBOOT

- 1. Connect the CAN bus cable and plug a jumper at the 120R terminal resistor.
- 2. Enter

cd ~/CanBoot/scripts
then enter
python3 flash_can.py -i can0 -q
query the canbus ID (make sure the CAN cable is connected and
powered on) as shown in the figure below, the UUID of the device has
been found:
biqu@BTT-CB1:~/CanBoot/scripts\$ python3 flash_can.py -i can0 -q
Resetting all bootloader node IDs...
Checking for canboot nodes
Detected UUID: be69315a613c, Application: CanBoot
Query Complete
biqu@BTT-CB1:~/CanBoot/scripts\$

3. Enter

python3 flash_can.py -i can0 -f ~/klipper/out/klipper.bin -u be69315a613c

The **be69315a613c** is replaced with the actual UUID. Note: **klipper.bin** needs to be generated in advance using the **make** command, and the **application start offset** of CanBoot is **8KiB offset**, so **Klipper's menuconfig Bootloader offset** should also be **8KiB bootloader**, as shown in the following figure.



 Query again with python3 flash_can.py -i can0 -q The Application should now show Klipper, indicating it is running correctly.

```
biqu@BTT-CB1:~/CanBoot/scripts$ python3 flash_can.py -i can0 -q
Resetting all bootloader node IDs...
Checking for canboot nodes...
Detected UUID: be69315a613c, Application: Klipper
Query Complete
biqu@BTT-CB1:~/CanBoot/scripts$
```

Firmware Update via DFU

Raspberry Pi or CB1 update via DFU.

- 1. Hold the **Boot** button and connect the board to Raspberry Pi/CB1 via Type-C cable to enter DFU mode.
- 2. Enter Isusb in the SSH terminal to query the DFU device ID.

pi@i	fluid	dpi:~ \$	lsusb			
Bus	001	Device	005:	ID	0483:df11	STHicroolactronics STM Device in DFU Mode
Bus	001	Device	004:	ID	1d50:6061	OpenMoko, Inc. Geschwister Schneider CAN adapter
Bus	001	Device	003:	ID	0424:0c00	Microchip Technology, Inc. (formerly SMSC) SMC9512/9514 Fast Ethernet Adapter
Bus	001	Device	002:	ID	0424:9514	Microchip Technology, Inc. (formerly SMSC) SMC9514 Hub
Bus	001	Device	001:	ID	1d6b:0002	Linux Foundation 2.0 root hub
	67	doni - C				

3. Enter

```
cd klipper
navigate to the Klipper directory, and enter
make flash FLASH_DEVICE=0483:df11
start flashing the firmware (Note: Replace 0483: df11 with the actual
device ID found in the previous step.)
```

- After flashing, enter
 Is /dev/serial/by-id/
 to query the device's Serial ID (only applicable for USB communication, not for CANBus).
- 5. For USB communication, you don't need to press the Boot button for subsequent updates. Enter the following command to flash the firmware

make flash FLASH_DEVICE=/dev/serial/by-id/usb-Klipper_stm32g0b1xx_4550357128922FC8-if00 (Note: replacing /dev/serial/by-id/xxx with the actual ID found in the previous step).

6. For CAN bus communication, disconnect the Type-C cable after flashing.

CAN bus Configuration

Use with BIGTREETECH U2C module.



1. Enter the following command in the SSH terminal: sudo nano /etc/network/interfaces.d/can0 Add the following content: allow-hotplug can0 iface can0 can static bitrate 1000000 up ifconfig \$IFACE txqueuelen 1024 Set the CAN bus speed to **1M** (must match the firmware setting of

1000000 CAN bus speed). Save (Ctrl + S) and exit (Ctrl + X), then enter sudo reboot to restart Raspberry Pi.

- 2. Each device on CAN bus will generate a **canbus_uuid** according to the UID of MCU, to find each microcontroller device ID, ensure the hardware is powered on and connected correctly, then run: ~/klippy-env/bin/python ~/klipper/scripts/canbus query.py can0
- 3. If an uninitialized CAN device is detected, the command will report the canbus uuid: Found canbus_uuid=0e0d81e4210c
- 4. If Klipper is running and connected to the device, the canbus_uuid will not be reported, which is normal.

Configuring Klipper

 In your computer's web browser, enter the Raspberry Pi's IP address, and as shown in the path below, download the reference configuration file sample-bigtreetech-mmb-canbus.cfg. If the file is not available, update Klipper firmware to the latest version or download from: https://github.com/bigtreetech/MMB

≡ 🚉 BTT-CB1			
DASHBOARD			
>_ CONSOLE	Config Files		*
G-CODE FILES	config_examples		- C 🗱
3D G-CODE VIEWER	Current path: /config_examples		Free disk: 25.1 GB
	Name 🛧	Filesize	Last modified
	printer-wanhao-duplicator-i3-v2.1-2017.cfg	5.0 kB	2023年1月12日 11:15
<u> </u>	sample-aliases.cfg	5.8 kB	2023年1月12日 11:15
	sample-bigtreetech-ebb-canbus-v1.0.cfg	1.4 kB	2023年1月12日 11:15
	sample-bigtreetech-ebb-canbus-v1.1.cfg	1.5 kB	2023年1月12日 11:15

2. Upload the configuration file to Configuration Files.

≡ 🚉 BTT-CB1			
E DASHBOARD	i Config Files		~
>_ CONSOLE	Root Sonfig (3)	2 💽 💽	C 🗘
30 G-CODE VIEWER	Current path: /config		Free disk: 25.1 GB
	□ Name ↑	Filesize	Last modified
	.theme		1970年1月20日 16:51
	.moonraker.conf.bkp	1.5 kB	2023年1月12日 11:07
	Crowsnest.conf	1.8 kB	2023年1月4日 13:07
	generic-bigtreetech-manta-m5p.cfg	3.5 kB	2023年1月12日 11:13

- 3. Add the board configuration to the "printer.cfg" file: [include sample-bigtreetech-mmb-canbus.cfg]
- 4. Modify the ID number in the configuration file to match the actual ID of the board (USB serial or canbus).
- 5. Configure the module's specific functions according to the instructions at <u>https://www.klipper3d.org/Overview.html</u>