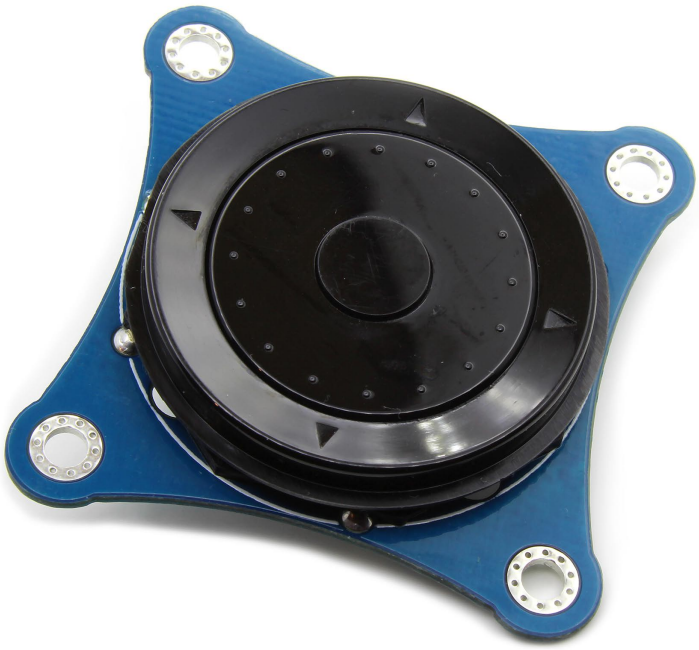


I2CNavKey

HW V1.0

Revision History

Revision	Date	Author(s)	Description
1.0	08.04.19	Simone	First version

Contents

1. Device Overview

The I2CNavKey is a small board that includes a 7 functions joypad.
The joypad has 4 directional keys plus a central button and a rotary ring. It is controllable with the I^{2}C bus, and it's possible to configure 16 different I^{2}C addresses.

The I2CNavKey has also 3 configurable GPIOs with the same footprint of RGB LED. This 3 GPIOs can be configured independently in 4 different way: input, output, PWM and analog input.

Since the I2CNavKey is a I^{2}C slave, it has a series of 8 bit registers where it's possible to configure different parameters, it's also possible to detect when the buttons are pr essed or when the ring is rotated.
It also has an open-drain output pin used as interrupt output. This pin is used to notify to the master when some event happens in the I2CNavKey, the source of interrupt can be customized.

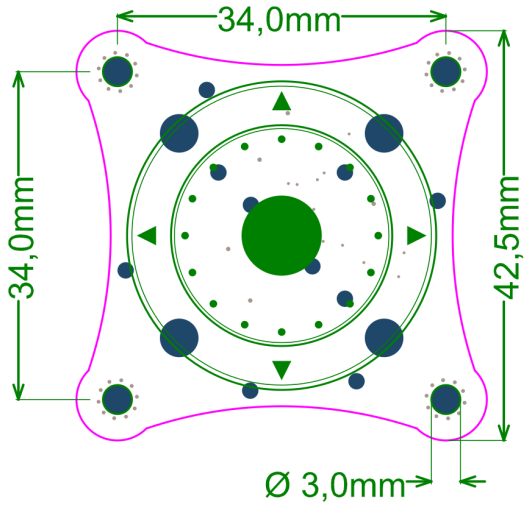


Figure 1.1: Dimension

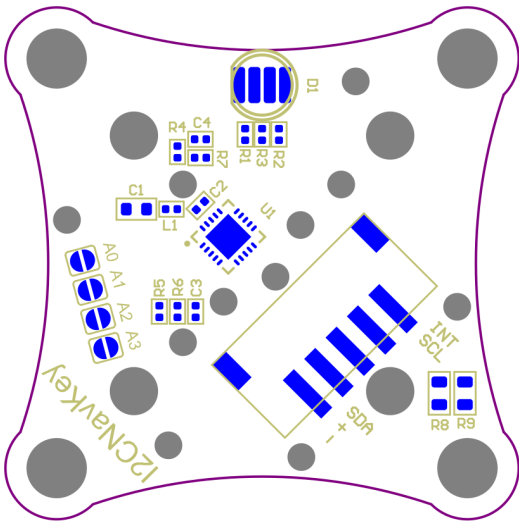


Figure 1.2: Bottom view of the board

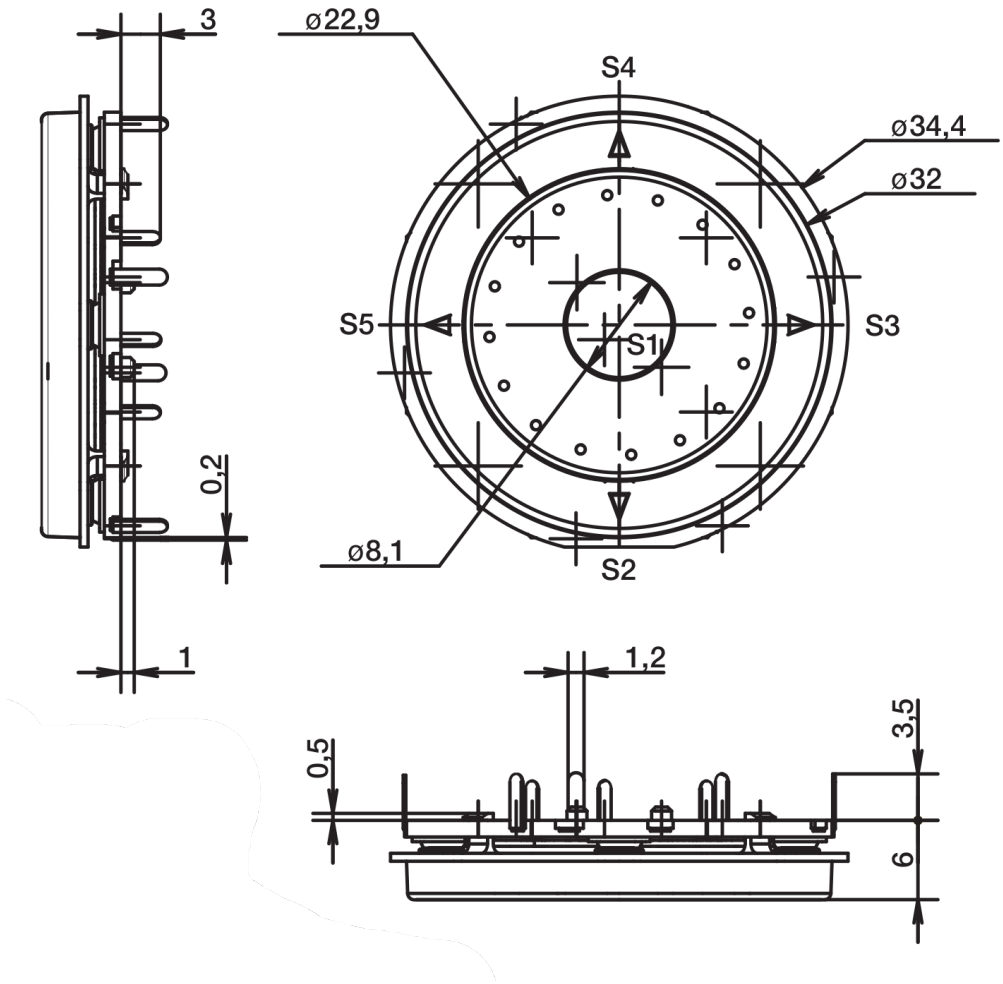


Figure 1.3: Dimension of the joypad: TSWB-3N-CB222 LFS from C&K

1.1 Electrical characteristics

Parameter	Symbol	Min	Max
Supply voltage	V_{DD}	3V	5V
I^2C input-low level	V_{IL}	0	0.3 * V_{DD}
I^2C input-high level	V_{IH}	0.8 * V_{DD}	V_{DD}
I^2C clock input frequency	f_{SCL}		400kHz
Supply current (LEDs off)	I_{DD}		1.8mA
Interrupt pull-up resistor	R_{INT}	15kΩ	120kΩ

1.2 Connection

Figure 1.4 shows the pin-out of the I2CNavKey.

Pin	I/O Type	Function
GND	Power	Ground reference for logic
Vcc	Power	Positive supply for logic
SDA	I/O	I^2C data
SCL	I	I^2C clock
INT	OD	Open-drain interrupt output

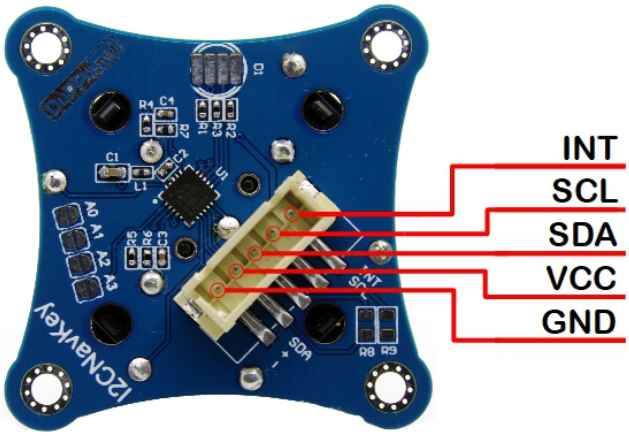


Figure 1.4: Pinout of the board

There are a 5 pins JST XH connector where it is possible to connect the I2CNavKey to the master. In order to avoid I^2C address conflict with other devices on the bus, the address of each device must be different. In the section 1.3, it is described how to set the address.

1.3 I^2C interface

The I2CNavKey is a I^2C slave, it's possible to the set 16 different addresses. All of the 7-bit address can be customized by soldering the jumpers A0 - A3 on the bottom of the board. When the jumper is open, it means a logic 0. if jumper is shorted it means a logic 1.

I^2C address							
7	6	5	4	3	2	1	0
0	0	1	A3	A2	A1	A0	R/W

The I2CNavKey has I^2C pull-up resistors, by default they are not soldered. It's possible to solder two resistors **R8** and **R9**. This must be done in case that the master doesn't have these resistors and must be enabled only one I2CNavKey in a chain. A typical value of this resistors are 4.7kΩ.

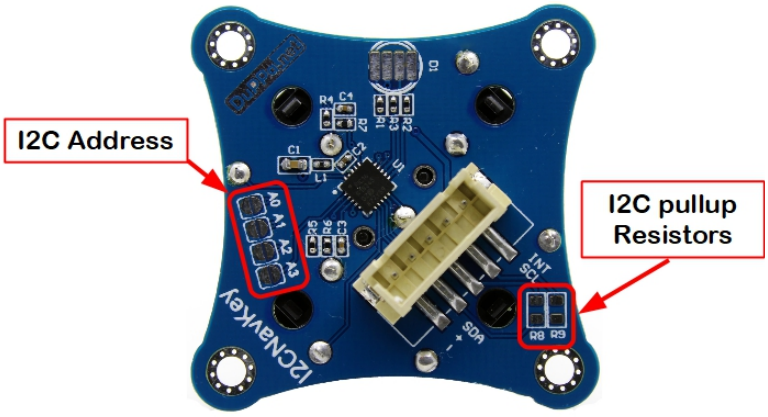


Figure 1.5: Pull-up resistors and jumper location

1.4 Joypad

On the I2CNavKey, there is a joypad with 7 functions:

- 4 directional buttons: up, down, left and right
- 1 central button

- Central wheel that can be rotated left and right, and it has 12 pulses per rotation.

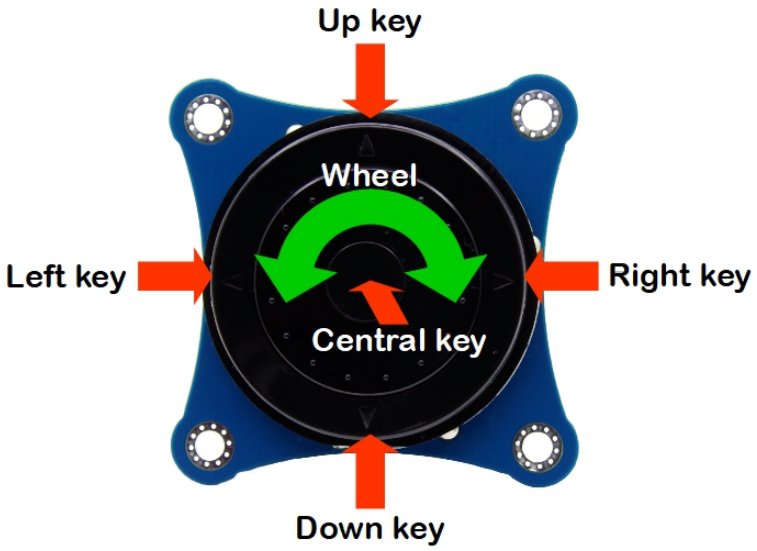


Figure 1.6: 7 function of the joypad

1.4.1 Buttons

When a button is pressed, the specific bit is set in the register **ISTATUS**. Every button has 2 specific bits because it's possible to detect when the button is pressed and when is released.
The central button has also the possibility to read a fast double push by setting a window time in the register **DPPERIOD**. When a double push is made inside of the **DPPERIOD** window, the bit **SCTRDP** is set and it is possible to read in the **ISTATUS** register.
If the **DPPERIOD** is 0, the double push function is disabled.

1.4.2 Encoder

For reading the rotary encoder movement, there are 4 of 32bit registers: **CVAL**, **CMAX**, **CMIN** and **ISTEP**. All of these 4 registers can be configured to work as 32bit int or as IEEE 754 floating number, this format can be set in the **GCONF** register.
The counter limits are the following:

- **32bit INT:** from -2147483648 to $+2147483647$
- **IEEE 754 float:** from -126.0 to $+127.0$

In case of 32bit int, it is not necessary to read all the 4 bytes, you can read only the first 8 bit or the first 16 bit. For example, if you want to count between 0 and 10, you can read only the first byte of the **CVAL** register. In this way, you can save I2C transactions.
Every time the encoder moves one step, the value of the **CVAL** register is increased or decreased of the value of **ISTEP**. The direction of the rotation decides if **ISTEP** is added or subtracted from **CVAL**.
CMAX and **CMIN** are used for setting a minimum and maximum thresholds of **CVAL**. In the **GCONF** register, there is **WRAPE** bit. This bit is used to enable or disable a wrap functionality of **CVAL** when it exceeds from the thresholds.
For example, if i configure the I2CNavKey as following:

- **CVAL**$= 0$
- **CMAX** $= 5$
- **CMIN** $=$ -5
- **ISTEP**$= 1$

I will have **CVAL** is incremented of 1 at each rotation step of the encoder. The maximum value that **CVAL** can reach will be 5 while the minimum is -5. In the figure 1.7 shows the value of **CVAL**.
As showed in the figure 1.8, when **WRAPE** is set to 1 when **CVAL** reaches the value of 5, at the next increment **CVAL** it will be wrapped to -5.
Every time when the encoder is rotated one step and when **CVAL** reaches the thresholds, an interrupt is generated and it is possible to read in the register **ISTATUS**.

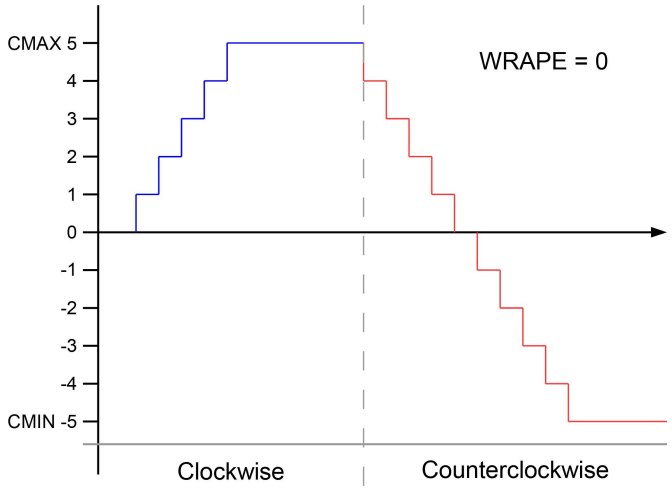


Figure 1.7: Blue and red lines are the **CVAL** values when the encoder is rotated and the **WRAPE** is disabled

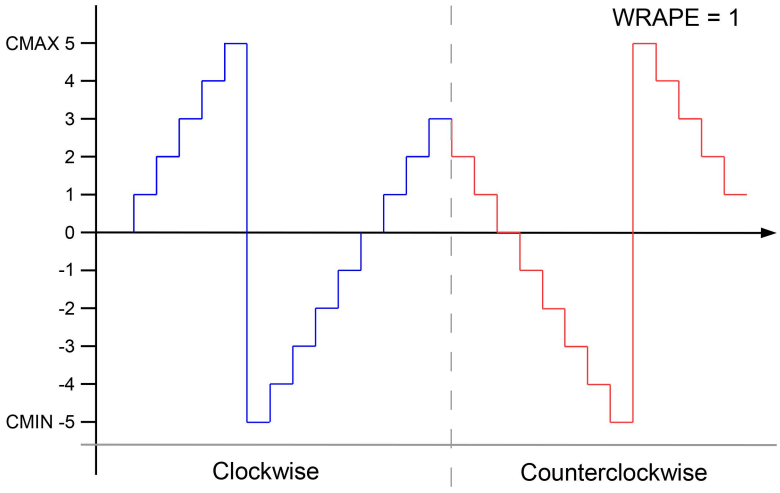


Figure 1.8: Blue and red line are the **CVAL** values when the encoder is rotate and the **WRAPE** is enabled

1.5 GP pins

There are 3 configurable GP pins, they are called GP1, GP2 and GP3. They have the same footprint of an 5mm RGB led. In figure 1.9, it is shown the position. Each pin can be configured to be:

➤ **PWM:** The pins is configured to be a PWM output.

➤ **Analog input:** The pins is configured to be an input of the internal ADC.

➤ **GPIO output:** The pins is configured to be a digital output.

➤ **GPIO Input:** The pin is configured to be a digital input.

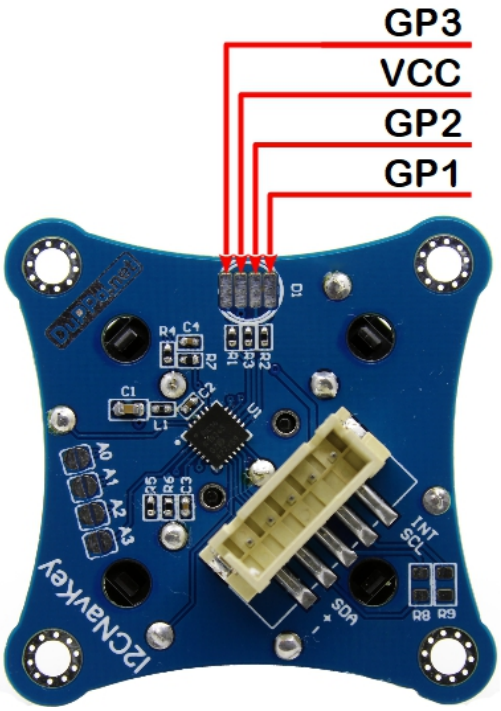


Figure 1.9: GP pins pinout

1.5.1 PWM mode

A GP pins can be configured in PWM mode by configuring the **GPxMODE** [1] to 0x00. The default configuration of the GP pins at the start up is PWM. Once the GP pin is configure in PWM mode, it is possible to set the PWM value by writing the corresponding **GPxREG**.
The PWM is resolution 10bit with a frequency of 31.25kHz, but the user can not access to the full resolution. It is possible to set a PWM value between 0% to 100% by writing 0 to 100 to the **GPxREG**.
The PWM is inverted, this means that when the PWM is 0% the GP pin voltage is VCC while when it is 100% the GP pin is GND.
In this way it is possible to connect a common anode RGB LED and have the LED off when the PWM is 0 and full on when the PWM is 100.
To each PWM outputs it is possible to configure a specific gamma correction by writing the register **GAMMAGPx**. There are 7 type of gamma correction:

- Gamma 1 (linear)
- Gamma 1.8
- Gamma 2.0
- Gamma 2.2
- Gamma 2.4
- Gamma 2.6
- Gamma 2.8

In the figure 1.10 is showed the relationship between the PWM at the full resolution and the PWM settable from the user.

[1] In a register name the **x** correspond to 1, 2 or 3

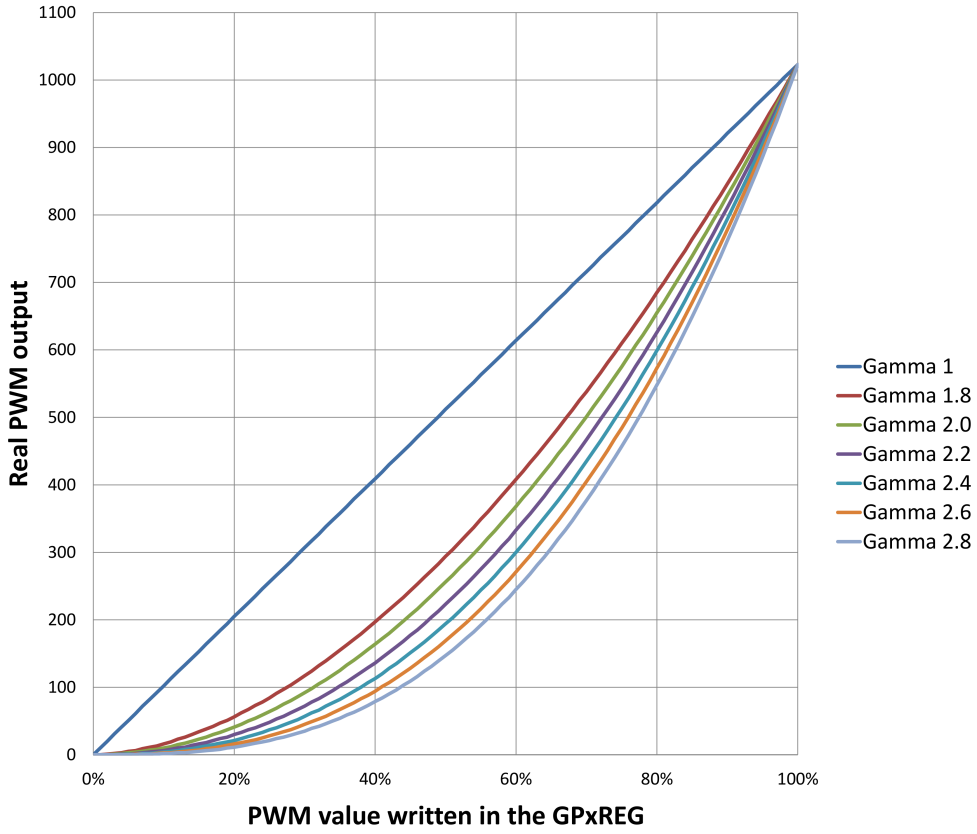


Figure 1.10: Possible gamma option

Fade function

The I2CNavKey has an auto fading features.
There is the register **FADEGP**, the value you write inside these registers is the step speed of the fading ramp, and it's in *ms*. The fading will follow the selected gamma correction. If the value is 0, the fade feature is disabled. It means that when the new value of PWM is written, it is immediately updated to the output.
The fading process starts when a PWM value is written. Fading process is complete according to the PWM value you have set. When the internal fade PWM value is the same of the PWM value (it means that when fading is complete), an interrupt will be generated. Let's make an example:

At the startup, i write as:

- **GP1REG** = 0
- **GP2REG** = 0
- **GP3REG** = 0
- **FADEGP** = 1

In above case, the LEDs of the encoder are off and the fade has 1ms step size.
Now i write as: *GP1REG=100*.
This moment, the ramp of the **GP1REG** starts and reaches the value 0x100 in 100ms. (the other LEDs remain OFF)
When the ramp reaches 100, i get an interrupt from the **INT** pin.
After this, i write as: *GP1REG=0x00*. At this point, the fade ramp starts again and it will turn off the LED in 100ms, and an interrupt will be generated at the end.
The figure 1.11 is showing an example of the fade output when a PWM value is written in different moments, as well as when the interrupt of the fade is generated.

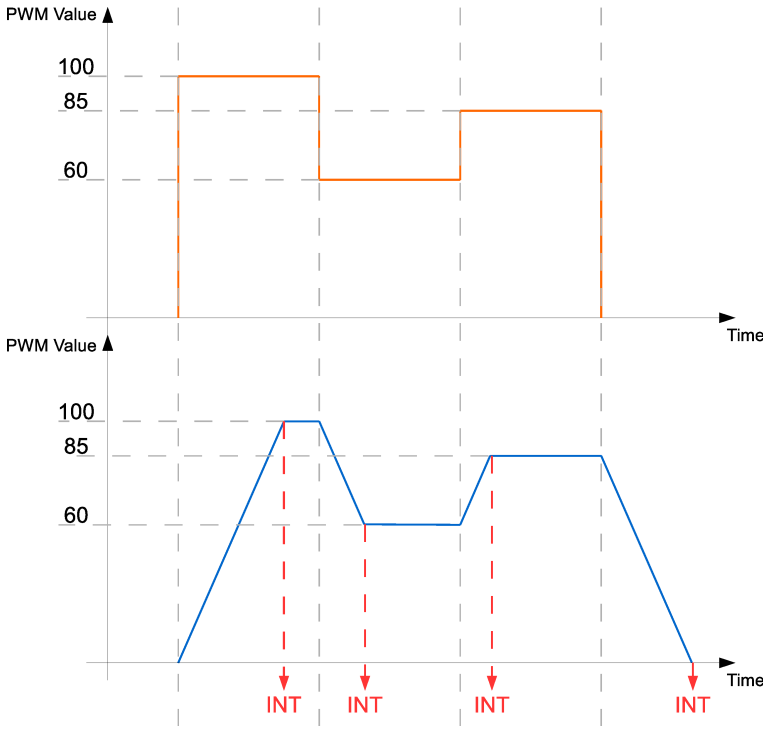


Figure 1.11: Time diagram of the fade output when writing on the PWM register

1.5.2 OUT mode

A GP pin can be configured as output mode by configuring the **GPxMODE** to 0x01. When the pin is configured as output writing 1 to **GPxREG** it make the GP pin high, while writing 0 make the pin low.

1.5.3 ADC mode

A GP pin can be configured as ADC mode by configuring the **GPxMODE** to 0x02. It is possible to read the ADC value by reading the register **GPxREG**. The content of the register updates every 10ms. When the GP pin is tied to 0 the ADC value is 0, while when the GP pin is tied to VCC the ADC value is 255. In this case it is also possible to enable the internal pull-up by writing 1 in the **GPxPUL** field.

1.5.4 IN mode

A GP pin can be configured as input mode by configuring the **GPxMODE** to 0x03. In this case, it is also possible to enable the internal pull-up by writing 1 in the **GPxPUL** field. It is also possible to configure the GP pin to generate an interrupt when the signal change. In the field **GPxINT** it possible to enable the interrupt when there is a positive edge, negative edge or both. The interrupt can be read in the register **I2STATUS**.It can be also reported to the **INT** pin by enabling the **SINT2L** bit.

1.6 EEPROM

The I2CNavKey has 256 bytes of EEPROM.
This memory is divided in two banks of 128 bytes. With the bit **MBANK** in the **GCONF** register, it is possible to choose either the bank 1 or the bank 2.
The memory area is between 0x80 and 0xFF address. To use the EEPROM, user only needs to perform reading or a writing in these address areas.
The writing time takes 4 - 5ms to be executed. Wait this time before sending other commands.

1.7 Interrupt

The I2CNavKey has multiple interrupt source previously described. When an interrupt is generated, the **INT** pin is tied low. By reading the register **ISTATUS**, the interrupts are cleared and the **INT** pin returns high.

The INT pin is open-drain output. Hence it requires an external pull-up resistor, or internal pull-up resistor can be enabled by setting the bit **IPUD** to 1.
In a chain of I2CNavKey all the **INT** pins can be connected together, like the pin of the I^2C. When an interrupt occurs, user has to scan the boards in the chain to find who generates the interrupt.
With the register **INTCONF**, it is possible to enable or disable interrupt. When an interrupt is disabled, the corresponding bit is set, but the **INT** pin is not affected.

2. Registers

In this section, the internal registers of I2CNavKey is described.
The I2CNavKey has the *auto increment* feature. This means that after writing or reading a register, the internal address pointer is automatically incremented by one.

Address range	Name	Description	Dimension	Default value
0x00	GCONF	General Configuration	1 Byte	0
0x01	GP1CONF	GP 1 Configuration	1 Byte	0
0x02	GP2CONF	GP 2 Configuration	1 Byte	0
0x03	GP3CONF	GP 3 Configuration	1 Byte	0
0x04 0x05	INTCONF	INT pin Configuration	2 Byte	0
0x06 - 0x07	ISTATUS	Interrupt Status	2 Byte	0
0x08	I2STATUS	Secondary interrupt status	1 Byte	0
0x09	FSTATUS	Fade process status	1 Byte	0
0x0A - 0x0D	CVAL	Counter Value	4 Byte	0
0x0E - 0x11	CMAX	Counter Max value	4 Byte	0
0x12 - 0x15	CMIN	Counter Min value	4 Byte	0
0x16 - 0x19	ISTEP	Increment step value	4 Byte	1
0x1A	GP1REG	I/O GP1 register	1 Byte	0
0x1B	GP2REG	I/O GP2 register	1 Byte	0
0x1C	GP3REG	I/O GP3 register	1 Byte	0
0x1D	DPPERIOD	Double push period	1 Byte	0
0x1E	FADEGP	Fade timer step of GPs ports	1 Byte	0
0x1F	GAMMAGP1	Gamma correction on PWM of GP1	1 Byte	0
0x20	GAMMAGP2	Gamma correction on PWM of GP2	1 Byte	0
0x21	GAMMAGP3	Gamma correction on PWM of GP3	1 Byte	0
0x70	IDCODE	I2CNavKey unique code	1 Byte	0x5B
0x71	VERSION	I2CNavKey version	1 Byte	0x11
0x80 - 0xFF	EEPROM	EEPROM memory	128 Byte	0

2.1 Configuration

2.1.1 General Configuration

GCONF Address: **0x00**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RESET		MBANK	CKSRC	IPUD	DIRE	WRAPE	DTYPE

❖ **DTYPE** Data type of the register: **CVAL**, **CMAX**, **CMIN** and **ISTEP**.

1: The registers are considered float numbers IEEE 754

0: The registers are considered int 32bit

❖ **WRAPE** Enable counter wrap.

1: Wrap enable. When the counter value reaches the **CMAX**+1, restart to the **CMIN** and vice versa

0: Wrap disable. When the counter value reaches the **CMAX** or **CMIN**, the counter stops to increasing or decreasing

❖ **DIRE** Direction of the encoder when increment.

1: Rotate left side to increase the counter value

0: Rotate right side to increase the counter value

❖ **IPUD** Interrupt Pull-UP disable.

1: Disable

0: Enable

❖ **CKSRC** Enable or disable I^2C clock stretch

1: Enabled

0: Disabled

❖ **MBANK** Select the EEPROM memory bank. Each bank are 128 byte wide

1: Second memory bank

0: First memory bank

❖ **RST** Reset of the I2CNavKey

1: Reset of the I2CNavKey. The RESET command takes 400us to be executed.

0: No reset

2.1.2 GP1 Configuration

GP1CONF Address: **0x01**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
-	-	-	GP1INT		GP1PUL	GP1MODE	

❖ **GP1MODE** Configuration of the pin GP1

00: PWM output
01: GPIO output Push-Pull
10: Analog input
11: GPIO Input

❖ **GP1PUL** Enable or disable the internal pull-up.

0: Pull-UP disable
1: Pull-UP enabled

❖ **GP1INT** Configuration of the interrupt, available only when the pin is configured as input

00: Interrupt disabled
01: Interrupt on positive edge
10: Interrupt on negative edge
11: Interrupt on both edges

2.1.3 GP2 Configuration

GP2CONF Address: **0x02**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
-	-	-	GP2INT		GP2PUL	GP2MODE	

❖ **GP2MODE** Configuration of the pin GP2

00: PWM output
01: GPIO output Push-Pull
10: Analog input
11: GPIO Input

❖ **GP2PUL** Enable or disable the internal pull-up.

0: Pull-UP disable
1: Pull-UP enabled

❖ **GP2INT** Configuration of the interrupt, available only when the pin is configured as input

00: Interrupt disabled
01: Interrupt on positive edge
10: Interrupt on negative edge
11: Interrupt on both edges

2.1.4 GP3 Configuration

GP3CONF Address: **0x03**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
-	-	-	GP3INT		GP3PUL	GP3MODE	

❖ **GP3MODE** Configuration of the pin GP3

00: PWM output

01: GPIO output Push-Pull

10: Analog input

11: GPIO Input

❖ **GP3PUL** Enable or disable the internal pull-up.

0: Pull-UP disable

1: Pull-UP enabled

❖ **GP3INT** Configuration of the interrupt, available only when the pin is configured as input

00: Interrupt disabled

01: Interrupt on positive edge

10: Interrupt on negative edge

11: Interrupt on both edges

2.2 Interrupt output Configuration

This register is used to enable or disable the interrupt source selectively. When an interrupt event occurs, the **INT** pin goes low and the event is stored in the status register.

INTCONF Address: **0x04**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT2	IRMIN	IRMAX	IRDEC	IRINC	ICTRDP	ICTRP	ICTRR
Address: **0x05**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ILTP	ILTR	IRTP	IRTR	IDNP	IDNR	IUPP	IUPR

❖ **IUPR** Button UP release bit

1: Interrupt enabled when the UP button is released.

0: Interrupt disabled

❖ **IUPP** Button UP press bit

1: Interrupt enabled when the UP button is pressed.

0: Interrupt disabled

❖ **IDNR** Button DOWN release bit

1: Interrupt enabled when the DOWN button is released.

0: Interrupt disabled

❖ **IDNP** Button DOWN press bit

1: Interrupt enabled when the DOWN button is pressed.

0: Interrupt disabled

❖ **IRTR** Button RIGHT release bit

1: Interrupt enabled when the RIGHT button is released.

0: Interrupt disabled

❖ **IRTP** Button RIGHT press bit

1: Interrupt enabled when the RIGHT button is pressed.

0: Interrupt disabled

❖ **ILTR** Button LEFT release bit

1: Interrupt enabled when the LEFT button is released.

0: Interrupt disabled

❖ **ILTP** Button LEFT press bit

1: Interrupt enabled when the LEFT button is pressed.

0: Interrupt disabled

❖ **ICTRR** CENTRAL button release bit

1: Interrupt enabled when the CENTRAL button is released.

0: Interrupt disabled

❖ **ICTRP** Push button press bit

1: Interrupt enabled when the CENTRAL button is pressed.

0: Interrupt disabled

❖ **ICTRDP** CENTRAL button double press

1: Interrupt enabled when the CENTRAL button is double pressed.

0: Interrupt disabled

❖ **IRINC** Rotary encoder direction of increase

1: Interrupt enabled when the encoder is rotated in the direction of increase

0: Interrupt disabled

❖ **IRDEC** Rotary encoder direction of decrease

1: Interrupt enabled when the encoder is rotated in the direction of decrease

0: Interrupt disabled

❖ **IRMAX** **CVAL** reaches **CMAX** bit

1: Interrupt enabled when **CVAL** reaches **CMAX**

0: Interrupt disabled

❖ **IRMIN** **CVAL** reaches **CMIN** bit

1: Interrupt enabled when **CVAL** reaches **CMIN**

0: Interrupt disabled

❖ **INT2** Enable the secondary interrupts

1: Secondary interrupt enabled

0: Secondary interrupt disabled

2.3 Status

2.3.1 I2CNavKey Status

This register is only readable, when is read is automatically cleared.

ISTATUS Address: **0x06**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SINT2	SRMIN	SRMAX	SRDEC	SRINC	SCTRDP	SCTRP	SCTRR
Address: **0x07**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SLTP	SLTR	SRTP	SRTR	SDNP	SDNR	SUPP	SUPR

❑ **SUPR** Button UP release bit

1: UP button is released

0: UP button is not released

❑ **SUPP** Button UP press bit

1: UP button is pressed

0: UP button is not pressed

❑ **SDNR** Button DOWN release bit

1: DOWN button is released

0: DOWN button is not released

❑ **SDNP** Button DOWN press bit

1: DOWN button is pressed

0: DOWN button is not pressed

❑ **SRTR** Button RIGHT release bit

1: RIGHT button is released

0: RIGHT button is not released

❑ **SRTP** Button RIGHT press bit

1: RIGHT button is pressed

0: RIGHT button is not pressed

❑ **SLTR** Button LEFT release bit

1: LEFT button is released

0: LEFT button is not released

❑ **SLTP** Button LEFT press bit

1: LEFT button is pressed

0: LEFT button is not released

❑ **SCTRR** CENTRAL button release bit

1: CENTRAL button is released

0: CENTRAL button is not released

❑ **SCTRP** CENTRAL button press bit

1: CENTRAL button is pressed

0: CENTRAL button is not pressed

❑ **SCTRDP** CENTRAL button double press

1: CENTRAL button is double pressed

0: CENTRAL button is not double pressed

❑ **RINC** Rotary encoder is rotated in the increase direction

1: Encoder is rotated

0: Encoder is not rotated

❑ **RDEC** Rotary encoder is rotated in the decrease direction

1: Encoder is rotated

0: Encoder is not rotated

❑ **RMAX** Status of the counter value

1: **CVAL** reaches the **CMAX** value

0: **CVAL** is below the **CMAX** value

❑ **RMIN** Status of the counter value

1: **CVAL** reaches the **CMIN** value

0: **CVAL** is above the **CMIN** value

❑ **SINT2** Secondary interrupt status

1: Secondary interrupt event occurs

0: No secondary event occurs

2.3.2 Secondary interrupt status

In this register is possible to check the event of the fade process as well as the GP pins when is configured as input. This register if only readable, when is read is automatically cleared.

I2STATUS Address: **0x08**							
7	6	5	4	3	2	1	0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
-	FADE	EGP3		EGP2		EGP1	

❑ **GP1** Interrupt on the GP1 port.

00: No event occurs

01: Positive edge event

10: Negative edge event

❑ **GP2** Interrupt on the GP2 port.

00: No event occurs

01: Positive edge event

10: Negative edge event

❑ **GP3** Interrupt on the GP3 port.

00: No event occurs

01: Positive edge event

10: Negative edge event

❑ **FADE** Fade process interrupt.

1: Fade process event

0: No event in the fade process

2.3.3 Fade process status

With this register is possible to check the status of the PWM channel during the fade process. This register is only readable and the bits are set or cleared automatically during the fade process.

FSTATUS Address: **0x09**							
7	6	5	4	3	2	1	0
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
-	-	-	-	-	FGP3	FGP2	FGP1

❑ **FGP1** Fade status process of the GP1

1: Fade process is running

0: Fade process terminated

❑ **FGP2** Fade status process of the GP2

1: Fade process is running

0: Fade process terminated

❑ **FGP3** Fade status process of the GP3

1: Fade process is running

0: Fade process terminated

2.4 Encoder registers

These 4 registers are in 32bit format. It is possible to read/write a single byte independently, no need to read/write all the 4 bytes at time.
They can be also used in float mode IEEE 754, but the **DTYPE** bit must be set.

2.4.1 Counter Value

CVAL Address: **0x0A**							
31	30	29	28	27	26	25	24
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVAL BYTE 4 <31 - 24>							
Address: **0x0B**							
23	22	21	20	19	18	17	16
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVAL BYTE 3 <23 - 16>							
Address: **0x0C**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVAL BYTE 2 <15 - 8>							
Address: **0x0D**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVAL BYTE 1 <7 - 0>							

2.4.2 Counter Max

CMAX Address: **0x0E**							
31	30	29	28	27	26	25	24
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMAX BYTE 4 <31 - 24>							
Address: **0x0F**							
23	22	21	20	19	18	17	16
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMAX BYTE 3 <23 - 16>							
Address: **0x10**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMAX BYTE 2 <15 - 8>							
Address: **0x11**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMAX BYTE 1 <7 - 0>							

2.4.3 Counter Min

CMIN Address: **0x12**							
31	30	29	28	27	26	25	24
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMIN BYTE 4 <15 - 8>							
Address: **0x13**							
23	22	21	20	19	18	17	16
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMIN BYTE 3 <7 - 0>							
Address: **0x14**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMIN BYTE 2 <15 - 8>							
Address: **0x15**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CMIN BYTE 1 <7 - 0>							

2.4.4 Increment step

ISTEP Address: **0x16**							
31	30	29	28	27	26	25	24
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ISTEP BYTE 4 <15 - 8>							
Address: **0x17**							
23	22	21	20	19	18	17	16
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ISTEP BYTE 3 <7 - 0>							
Address: **0x18**							
15	14	13	12	11	10	9	8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ISTEP BYTE 2 <15 - 8>							
Address: **0x19**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ISTEP BYTE 1 <7 - 0>							

2.5 GPs registers

The usage of these register depends of the configuration of the GP pins:

➤ **PWM** Is possible to write the PWM value: 0 to 100.

➤ **GPIO output:** Is possible to write 1 or 0 for setting the output logic level.

➤ **Analog input:** Is possible to read the ADC value: 8 bit format.

➤ **GPIO Input:** Is possible to read the logic level of the output.

2.5.1 GP1 register

GP1REG Address: **0x1A**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GP1 Value <7 - 0>							

2.5.2 GP2 register

GP2REG Address: **0x1B**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GP2 Value <7 - 0>							

2.5.3 GP3 register

GP3REG Address: **0x1C**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GP3 Value <7 - 0>							

2.6 Timing registers

This register are used for changing some timing parameter

2.6.1 Double push time

This register is used for setting the interval for the detection of the double push of the central button. The value is in ms $\times$ 10. When this register is 0 this function is disabled.

DPPERIOD Address: **0x1D**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PDPUSH <7 - 0>							

2.6.2 Fade GP ports

This register is used for setting the fade speed of the GP ports. The value is in ms and indicate the time of one step. This register is working only if the GP port are configured in PWM

FADEGP Address: **0x1E**							
7	6	5	4	3	2	1	0
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
FADEGP <7 - 0>							

2.7 Gamma correction

This register is used for applying a gamma correction in case the GP port is used as PWM. It can be very useful when a LED is connected.

GAMMAGP1 Address: **0x1F**							
7	6	5	4	3	2	1	0
-	-	-	-	-	R/W-0	R/W-0	R/W-0
					GAMMAGP1 <2 - 0>		

GAMMAGP2 Address: **0x20**							
7	6	5	4	3	2	1	0
-	-	-	-	-	R/W-0	R/W-0	R/W-0
					GAMMAGP2 <2 - 0>		

GAMMAGP3 Address: **0x21**							
7	6	5	4	3	2	1	0
-	-	-	-	-	R/W-0	R/W-0	R/W-0
					GAMMAGP3 <2 - 0>		

❑ **GAMMAGPx** Fade status process of the GP1

000: Gamma 1

010: Gamma 1.8

011: Gamma 2.0

100: Gamma 2.2

101: Gamma 2.4

110: Gamma 2.6

111: Gamma 2.8

2.8 I2CNavKey unique code

This register contains an unique code that it's used to identify the I2CNavKey.This register is only readable and not writable.

IDCODE Address: **0x70**							
7	6	5	4	3	2	1	0
R-0	R-1	R-0	R-1	R-1	R-0	R-1	R-1
IDCODE = 0x5B							

2.9 I2CNavKey version

This register contains the version of the I2CNavKey.This value will change in case of a new hardware or firmware release. This register is only readable and not writable.

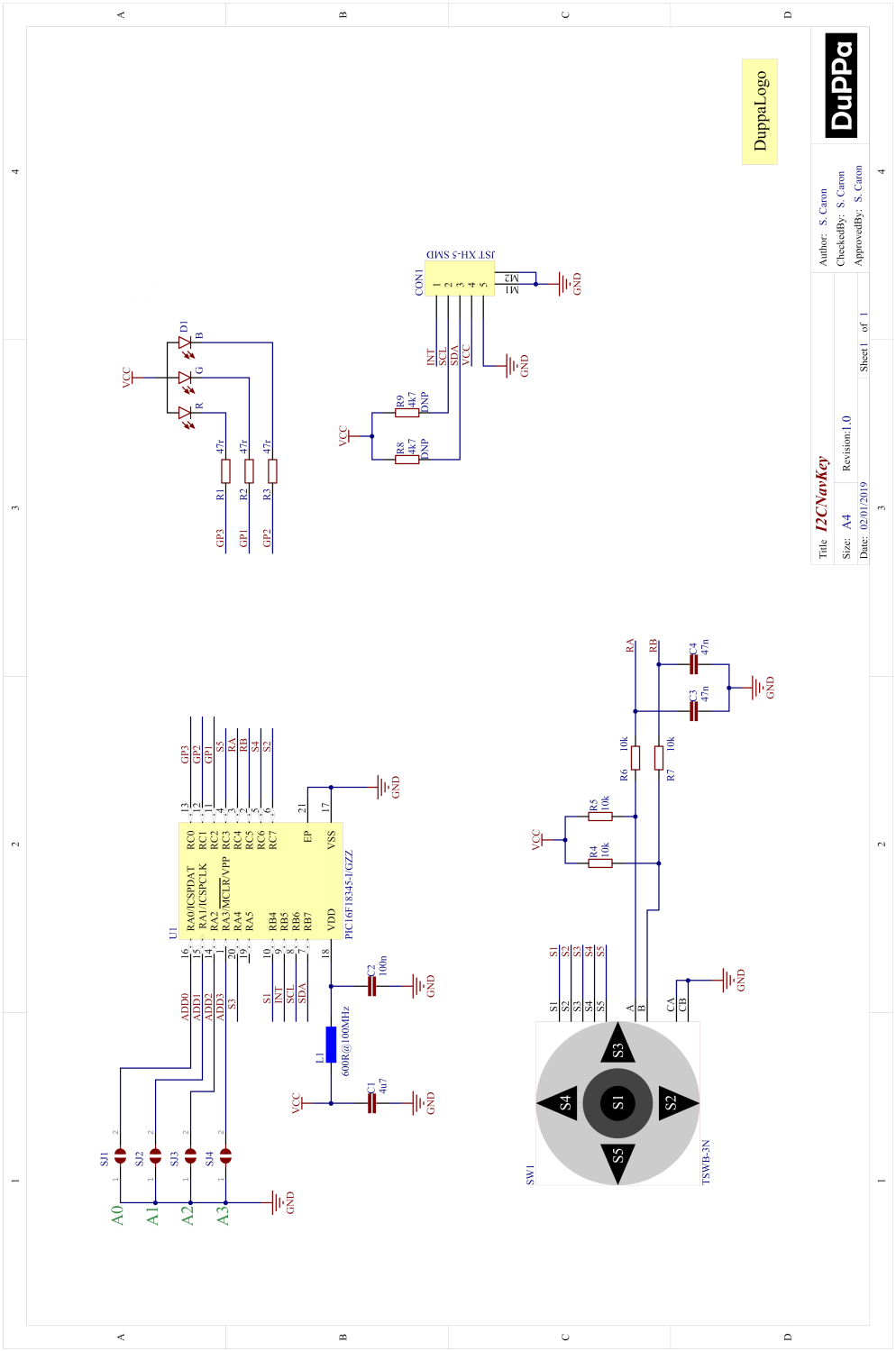
VERSION Address: **0x71**							
7	6	5	4	3	2	1	0
R-0	R-0	R-0	R-1	R-0	R-0	R-0	R-1
Version = 0x11							

3. Reference

This project is open source, the HW and the FW as well as some examples can be found on GitHub:
`https://github.com/Fattoresaimon/I2CNavKey`

4. Issues

5. Schematic


DuppaLogo
DuPPa
Author: S. Caron
CheckedBy: S. Caron
ApprovedBy: S. Caron
Title I2CNavKey
Size: A4
Revision:1.0
Date: 02/01/2019
Sheet1 of 1
JST XH-5 SMD
CON1
GND
INT
SCL
SDA
VCC
R9 4k7 DNP
R8 4k7 DNP
D1
R
G
B
R1 47r
R2 47r
R3 47r
GP3
GP1
GP2
U1
PIC16F18345-I/GZZ
RA0/ICSPDAT
RA1/ICSPCLK
RA2
RA3/MCLR/VPP
RA4
RA5
RB4
RB5
RB6
RB7
RC0
RC1
RC2
RC3
RC4
RC5
RC6
RC7
VDD
VSS
EP
ADD0
ADD1
ADD2
ADD3
S1
S2
S3
S4
S5
RA
RB
INT
SCL
SDA
SJ1
SJ2
SJ3
SJ4
A0
A1
A2
A3
L1
600R@100MHz
C1 4u7
C2 100n
R4 10k
R5 10k
R6 10k
R7 10k
C3 47n
C4 47n
CA
CB
SW1
TSWB-3N