ST600µNET

RMV MOTION INCORPORATED

ST600µNET SERIES

Application User’s Guide

ST600µNET

AUTHTOR: ALBERT TELLO.

User Guide Version: 7.15

3665 Kingsway • Suite 300
Vancouver, BC V5R-5W2
Phone 604.299.5173 • Fax 604.299.5174
Email: support@rmv.com

ST600µNET

Chapter 1

Introduction

Unique four axis motion control boards.

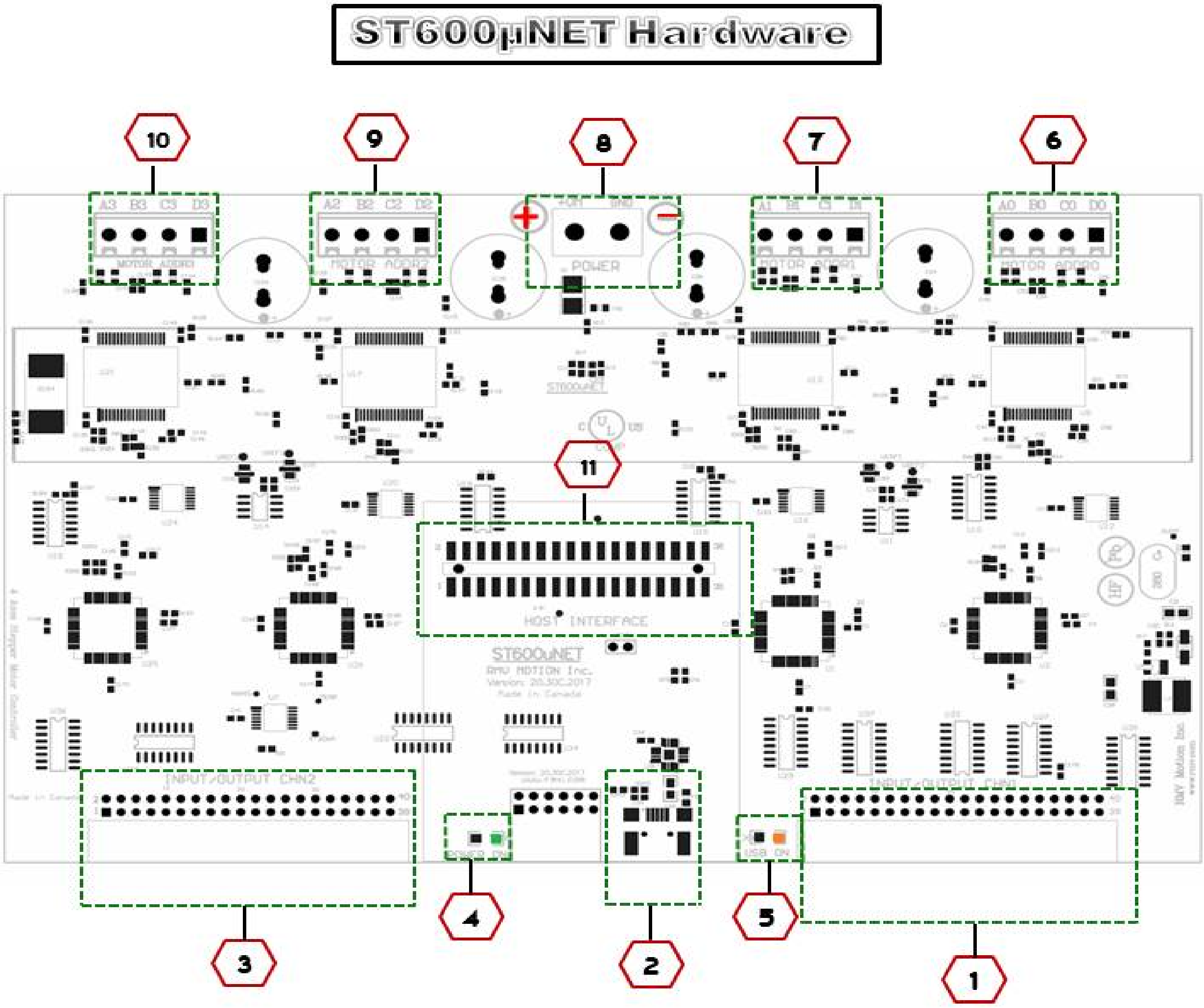
The *ST600µNET* Series has been developed with sixth mayor goals: a) very robust chopper driver design , b) kernel capabilities for different motion topologies, c) IoT full compatible, d) multi-hosts link interface, e) extended industrial multi- protocols support, and f) Open Sources application development. The *ST600µNET Series* can be connected to *rmv_Tasker Module* allowing in this way to create a motion application via HTML pages and JavaScript, through its embedded web server. These functionalities and others above described will save time in development of any motion application. In addition, *ST600µNET Series will have support* for the following ng third party boards: Raspberry PI, Arduino, Beagle-Bone, I.M.X NXP ARM 9-11.

⚠ This product is meant to be operated in industrial environment and be operated by engineers, technician familiar with risks associated with handling high voltage electrical and mechanical components, systems and subsystems.
This equipment operates at voltages and currents that can result in electrical shock, fire hazard and/or personal injury if not properly handled or applied. Equipment must be used with necessary caution and appropriate safeguards employed to avoid personal injury or property damage.
It is the user's responsibility to confirm that the voltages and isolation requirements are identified and understood, prior to energizing the board and or simulation. When energized, the ST600µNET or components connected to the ST600µNET should not be touched.

ST600µNET

Hardware Description and Connection

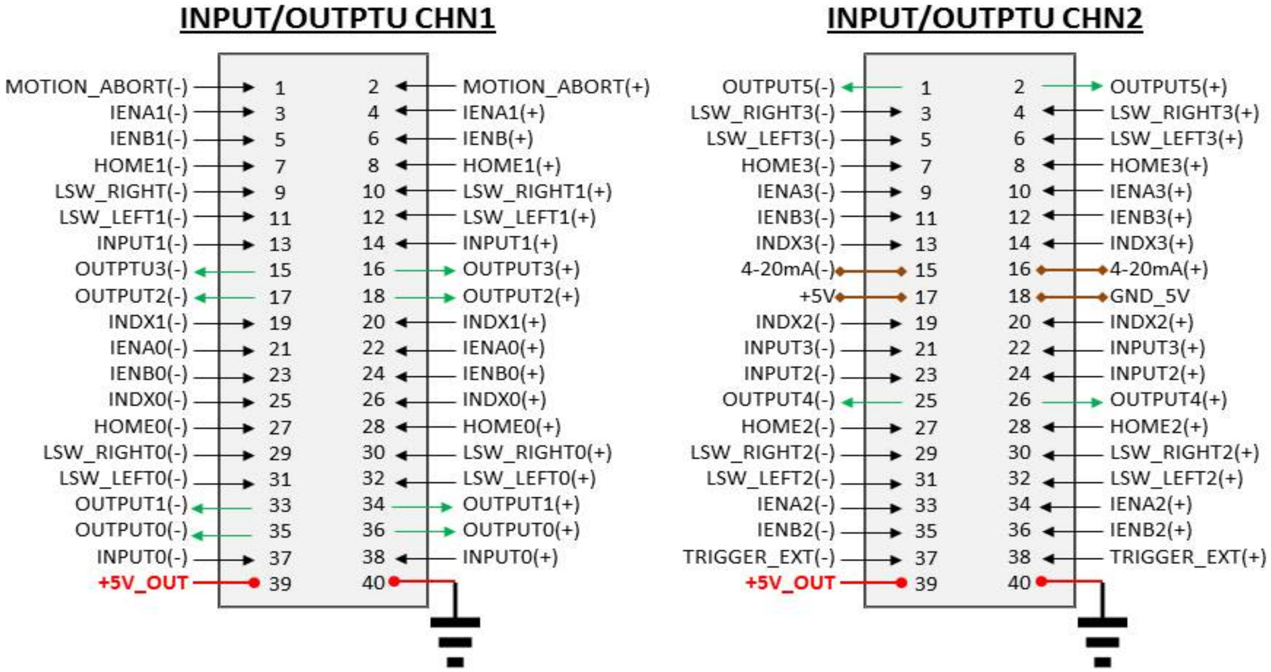
Completed board View



1. Motor Signals IO's "1"
2. Mini USB Type AB
3. Motor Signals IO's "2"
4. LED Power ON
5. LED Host USB
6. MOTOR PHASES ADDR "0"
7. MOTOR PHASES ADDR "1"
8. POWER SUPPLY INPUT
9. MOTOR PHASES ADDR "2"
10. MOTOR PHASES ADDR "3"
11. rmv_Tasker Connector

FIGURE: 1 Completed ST600µNET board at a Glance (TOP view)

Input-Output Headers 90° Connector



INPUT/OUTPTU CHN1

MOTION_ABORT(-)	1	2	MOTION_ABORT(+)
IENA1(-)	3	4	IENA1(+)
IENB1(-)	5	6	IENB(+)
HOME1(-)	7	8	HOME1(+)
LSW_RIGHT(-)	9	10	LSW_RIGHT1(+)
LSW_LEFT1(-)	11	12	LSW_LEFT1(+)
INPUT1(-)	13	14	INPUT1(+)
OUTPTU3(-)	15	16	OUTPUT3(+)
OUTPUT2(-)	17	18	OUTPUT2(+)
INDX1(-)	19	20	INDX1(+)
IENA0(-)	21	22	IENA0(+)
IENB0(-)	23	24	IENB0(+)
INDX0(-)	25	26	INDX0(+)
HOME0(-)	27	28	HOME0(+)
LSW_RIGHT0(-)	29	30	LSW_RIGHT0(+)
LSW_LEFT0(-)	31	32	LSW_LEFT0(+)
OUTPUT1(-)	33	34	OUTPUT1(+)
OUTPUT0(-)	35	36	OUTPUT0(+)
INPUT0(-)	37	38	INPUT0(+)
+5V_OUT	39	40	

INPUT/OUTPTU CHN2

OUTPUT5(-)	1	2	OUTPUT5(+)
LSW_RIGHT3(-)	3	4	LSW_RIGHT3(+)
LSW_LEFT3(-)	5	6	LSW_LEFT3(+)
HOME3(-)	7	8	HOME3(+)
IENA3(-)	9	10	IENA3(+)
IENB3(-)	11	12	IENB3(+)
INDX3(-)	13	14	INDX3(+)
4-20mA(-)	15	16	4-20mA(+)
+5V	17	18	GND_5V
INDX2(-)	19	20	INDX2(+)
INPUT3(-)	21	22	INPUT3(+)
INPUT2(-)	23	24	INPUT2(+)
OUTPUT4(-)	25	26	OUTPUT4(+)
HOME2(-)	27	28	HOME2(+)
LSW_RIGHT2(-)	29	30	LSW_RIGHT2(+)
LSW_LEFT2(-)	31	32	LSW_LEFT2(+)
IENA2(-)	33	34	IENA2(+)
IENB2(-)	35	36	IENB2(+)
TRIGGER_EXT(-)	37	38	TRIGGER_EXT(+)
+5V_OUT	39	40	

FIGURE: 2. ST600µNET - Input Output Signal connection

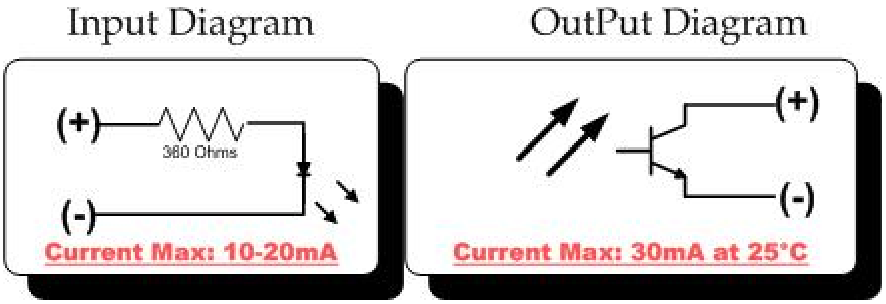


FIGURE: 3. Opto-Isolator diagram for Digital Input and Output

Input Signal Description

The ST600µNET in all digital inputs has in series a 360 Ohms in series with the anode from the optocoupler diode signed as (+) positive. The cathode from optocoupler diode is signed (-).

For The following equation can be applied to calculate "The Resistor in Series" needs to be added when an input voltage is greater than 5 or 6.5 Volts.

$$\boldsymbol{R(Ohms) = \frac{Vin - 1.2 - VCE(sat)}{0.010} - 360}$$

`Note1:` `If a TTL gate is controlling  the input diode , with a fix 360 Ohms at 5V, the IF   current fordward  will be :`

If = (5- 1.2- 0.4)/360 = 9.4 mA , which is compatible with the Opto-isolator specifications.

`Note2:` `It is assumed in all calculations for a workable Frequency of 5KHz`

Output Signal Description

The followin diagram and equations will show different basic configurations for the output phototransistor and equation to calculate "the outptu resistance " to minimized cover any applicatioon topology   tools

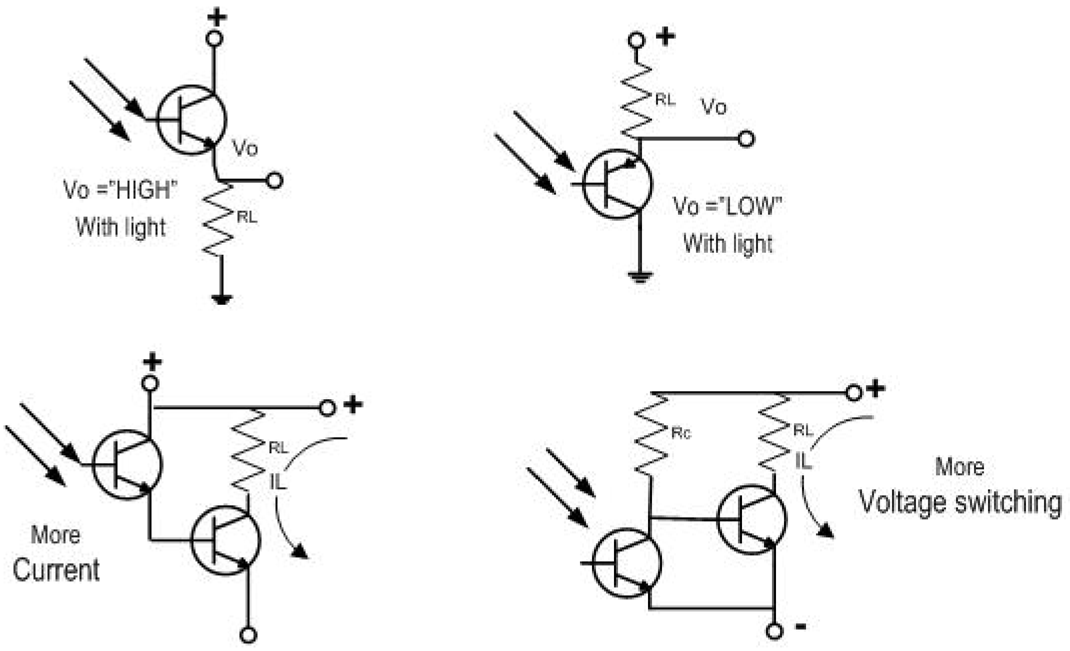


FIGURE: 4. Different Basic Phototransistor Output Configuration

Signals Description

INPUT/OUTPTUT CHN1

SIGNAL NAME	PIN Connection		Function Description
MOTION ABORT	1 (-)	2 (+)	Input Motion Abort to all axes. This signal can be connected to external push button. Voltage: 5V to 24V.
IENCA ADDR1	3 (-)	4 (+)	Input Encoder A "1". Voltage: 5V to 24V
IENCB ADDR1	5 (-)	6 (+)	Input Encoder B "1". Voltage: 5V to 24V
HOME ADDR1	7 (-)	8 (+)	Input Home Switch "1". Voltage: 5V to 24V
LIMIT SWITCH RIGHT ADDR1	9 (-)	10 (+)	Input Limit Switch Right "1". Voltage: 5V to 24V
LIMIT SWITCH LEFT ADDR1	11 (-)	12 (+)	Input Limit Switch Left "1". Voltage: 5V to 24V
INPUT ADDR1	13 (-)	14 (+)	Input Signal "1". Voltage: 5V to 24V
OUTPUT 3	15 (-)	16 (+)	Output Signal 3. Voltage: 5V to 24V. Max Current= 30 mA
OUTPUT 2	17 (-)	18 (+)	Output Signal 2. Voltage: 5V to 24V. Max Current= 30 mA
INDX ADDR1	19 (-)	20 (+)	Input Encoder INDX "1". Voltage: 5V to 24V
IENCA ADDR0	21 (-)	22 (+)	Input Encoder A "0". Voltage: 5V to 24V
IENCB ADDR0	23 (-)	24 (+)	Input Encoder B "0". Voltage: 5V to 24V
INDX ADDR0	25 (-)	26 (+)	Input Encoder INDX "0". Voltage: 5V to 24V
HOME ADDR0	27 (-)	28 (+)	Input Home Switch "0". Voltage: 5V to 24V
LIMIT SWITCH RIGHT ADDR0	29 (-)	30 (+)	Input Limit Switch Right "0". Voltage: 5V to 24V
LIMIT SWITCH LEFT ADDR0	31 (-)	32 (+)	Input Limit Switch Left "0". Voltage: 5V to 24V
OUTPUT 1	33 (-)	34 (+)	Output Signal 3. Voltage: 5V to 24V. Max Current= 30 mA
OUTPUT 0	35 (-)	36 (+)	Output Signal 2. Voltage: 5V to 24V. Max Current= 30 mA
INPUT 0	37 (-)	38 (+)	Input Signal "1". Voltage: 5V to 24V
POWER 5V_OUT	39		5V , Max Current =500 mA
POWER GND		40	5V , Digital Ground

INPUT/OUTPTUT CHN2

SIGNAL NAME	PIN Connection		Function Description
OUTPUT 5	1 (-)	2 (+)	Output Signal 5. Voltage: 5V to 24V. Max Current= 30 mA
LIMIT SWITCH RIGHT ADDR3	3 (-)	4 (+)	Input Limit Switch Right “3”. Voltage: 5V to 24V
LIMIT SWITCH LEFT ADDR3	5 (-)	6 (+)	Input Limit Switch Left “3”. Voltage: 5V to 24V
HOME ADDR3	7 (-)	8 (+)	Input Home Switch “3”. Voltage: 5V to 24V
IENCA ADDR3	9 (-)	10 (+)	Input Encoder A “3”. Voltage: 5V to 24V
IENCB ADDR3	11 (-)	12 (+)	Input Encoder B “3”. Voltage: 5V to 24V
INDX ADDR3	13 (-)	14 (+)	Input Encoder INDX “0”. Voltage: 5V to 24V
4-20 mA (-) Input	**15** **4-20mA(-)**	**16** **4-20mA(+)**	Analog Input 4-20 mA
+5 Volt Input	**17** **5+V(+)**	**18** **GND**	Analog Input: 0V to +5V
INDX ADDR2	19 (-)	20 (+)	Input Encoder INDX “2”. Voltage: 5V to 24V
INPUT 3	21 (-)	22 (+)	Input Signal “1”. Voltage: 5V to 24V
INPUT 2	23 (-)	24 (+)	Input Signal “1”. Voltage: 5V to 24V
OUTPUT 4	25 (-)	26 (+)	Output Signal 4. Voltage: 5V to 24V. Max Current= 30 mA
HOME ADDR2	27 (-)	28 (+)	Input Home Switch “2”. Voltage: 5V to 24V
LIMIT SWITCH RIGHT ADDR2	29 (-)	30 (+)	Input Limit Switch Right “2”. Voltage: 5V to 24V
LIMIT SWITCH LEFT ADDR2	31 (-)	32 (+)	Input Limit Switch Left “2”. Voltage: 5V to 24V
IENCA ADDR2	33 (-)	34 (+)	Input Encoder A “2”. Voltage: 5V to 24V
IENCB ADDR2	35 (-)	36 (+)	Input Encoder B “2”. Voltage: 5V to 24V
TRIGGER_EXT	37 (-)	38 (+)	External Trigger, this signal will trigger all controllers, via a push button for example. Voltage: 5V to 24V
POWER 5V_OUT	**39**		5V , Max Current =500 mA
POWER GND		**40**	5V , Digital Ground

Motor Connections

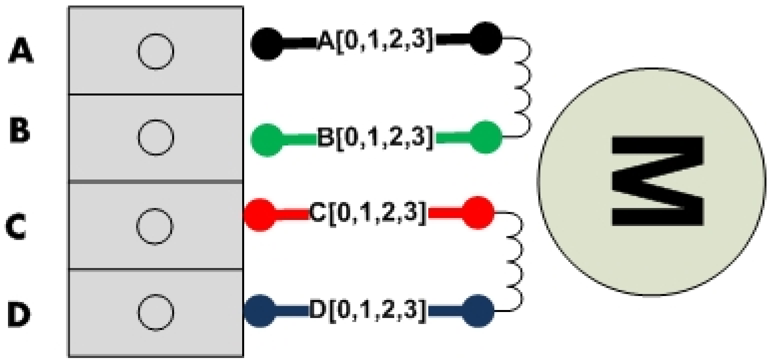


FIGURE: 3. ST600µNET – Motor Terminal Blocks Motor connection

Motor Power

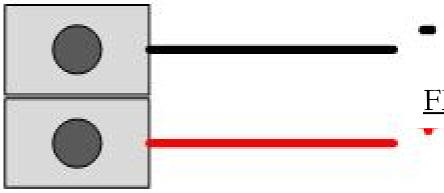


FIGURE: 4. ST600µNET –Motor Terminal Block POWER Connection

The ST600µNET Series require only one power supply regulated. In FIGURE1 "4" if the board has been powered correctly, the LED Green will be ON. If the power supply is not regulated the LED Green will flash and the board will not run properly. The voltage power supply requirements are as follows:

V(Minimum) = 18 DCV
V(Maximum) = 48 DCV
I(Idle) = 0.2 Amp

`Note3: Power supply Maximum Current depends on motor winding current, and stepper motors connected.`

Host Interface

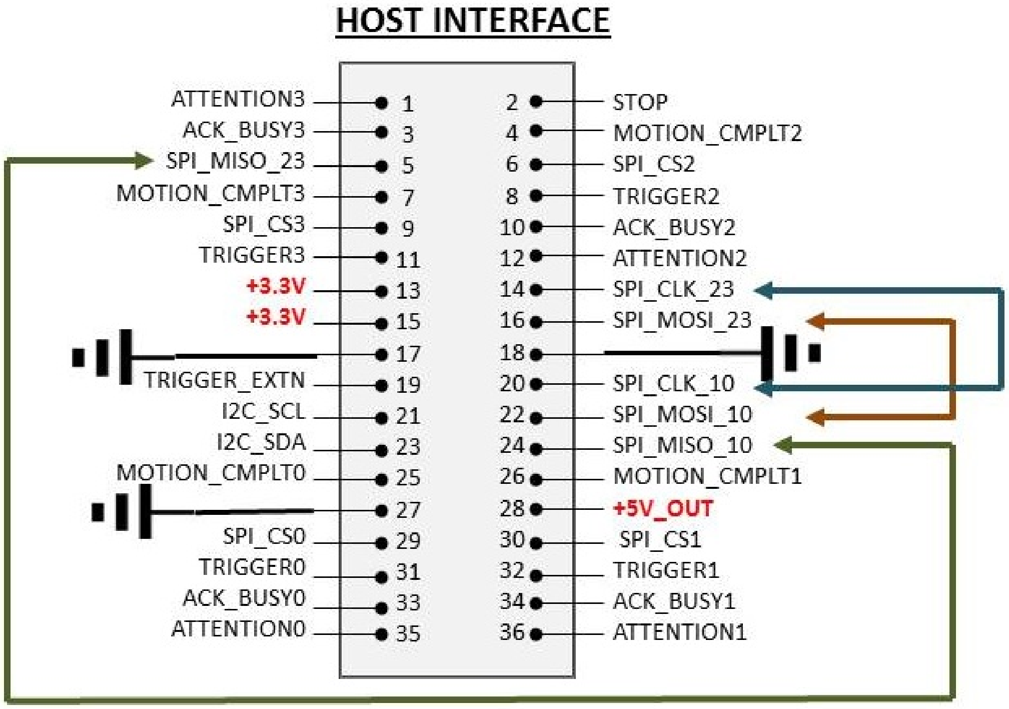


FIGURE: 4. ST600µNET – External Host Interface: rmv_Tasker, Arduino

Serial Peripheral Interface: *The following SPI lines must be connected together inside of the external third party hardware:*

SPI_CLK_10 ↔ SPI_CLK_23 ↔ **SPI_CLK**
SPI_MISO_10 ↔ SPI_MISO_23 ↔ **SPI_MISO**
SPI_MOSI_10 ↔ SPI_MOSI_23 ↔ **SPI_MOSI**

In Table1, shows the description-functionality for each signal described in "Host Interface Connectors", voltage compatibility, and direction.

Host Interface Signals Description

INPUT: Are all signals which direction goes from any RMV859 controller to the Host Processor Interface.

OUTPUT: Are all signals which direction goes from “the Host Processor Interface” to any RMV959 controller.

SIGNAL NAME	PIN	VOLTAGE	DIRECTION	FUNCTION DESCRIPTION
ATTENTION 3	**1**	**+3.3V**	INPUT	Active LOW. RMV859 Controller ADDR3 informs to” Host” a full report VIA SPI, must be request, as an example this signal is activated when a limit switch is activated.
STOP	**2**	**+3.3V**	OUTPUT	Active LOW. Emergency STOP. All controllers will stop its motion.

ACK_BUSY3	**3**	**+3.3V**	INPUT	ACK=Acknowledge Active LOW, from RMV859 Controller ADDR3
				BUSY=Busy, Active HIGH

MOTION_CMPLT2	**4**	**+3.3V**	INPUT	Active LOW, ADDR 2. Is activated every time motion has been completed. When a motion is loaded goes HIGH
SPI_MISO_23	**5**	**+3.3V**	INPUT	Master Input Slave Output. This Line is the same SPI_MISO_10, *“Both Signal Line must be connected together in the host interface board”.*
SPI_CS2	**6**	**+3.3V**	OUTPUT	Active LOW , SPI Salve chip select to RMV859 ADDR2 Controller
MOTION_CMPLT3	**7**	**+3.3V**	INPUT	Active LOW, ADDR 3. Is activated every time motion has been completed. When a motion is loaded goes HIGH
TRIGGER2	**8**	**+3.3V**	OUTPUT	Active LOW, Trigger RMV959 Controller ADDR2. A LOW must be at least 30 uSec, and after can return to +3.3V.
SPI_CS3	**9**	**+3.3V**	OUTPUT	Active LOW , SPI Salve chip select to RMV859 ADDR3 Controller .

ACK_BUSY2	**10**	**+3.3V**	INPUT	ACK=Acknowledge Active LOW, from RMV859 Controller ADDR2
				BUSY=Busy, Active HIGH

TRIGGER3	**11**	**+3.3V**	OUTPUT	Active LOW, Trigger RMV959 Controller ADDR3. A LOW must be at least 30 uSec, and after can return to +3.3V.
ATTENTION 2	**12**	**+3.3V**	INPUT	Active LOW. RMV859 Controller ADDR2 informs to” Host” a full report VIA SPI, must be request, as an example this signal is activated when a limit switch is activated.

+3.3V	**13, 15**	**+3.3V**	**POWER**	**+3.3V =Out , Max current= 550 mA**
SPI_CLK_23	**14**	**+3.3V**	OUTPUT	Master Input Slave Output. This Line is the same SPI_CLK_10, *"Both Signal Line must be connected tougher in the host interface board".*
SPI_MOSI_23	**16**	**+3.3V**	OUTPUT	Master Input Slave Output. This Line is the same SPI_MOSI_10, *"Both Signal Line must be connected together in the host interface board".*
GND	**17, 18**	**GND**	**POWER**	**GND connected to Ground Plane**
TRIGGER_EXTN	**19**	**+3.3V**	OUTPTU	Active Low, this signal is connected together to al l RMV859 Controllers via INPUT/OUTPTUT CHN2
SPI_CLK_10	**20**	**+3.3V**	OUTPUT	Master Input Slave Output. This Line is the same SPI_CLK_23, *"Both Signal Line must be connected tougher in the host interface board".*
I2C_SDA	**21**	**+3.3V**	INPUT/OUTPUT	I2C SDA Master -Slave
SPI_MOSI_10	**22**	**+3.3V**	OUTPUT	Master Input Slave Output. This Line is the same SPI_MOSI_23, *"Both Signal Line must be connected together in the host interface board".*
I2C_SCL	**23**	**+3.3V**	INPUT/OUTPUT	I2C SCL Clock Master-Slave
SPI_MISO_10	**24**	**+3.3V**	INPUT	Master Input Slave Output. This Line is the same SPI_MISO_23, *"Both Signal Line must be connected together in the host interface board".*
MOTION_CMPLT0	**25**	**+3.3V**	INPUT	Active LOW, ADDR 0. Is activated every time motion has been completed. When a motion is loaded goes HIGH
MOTION_CMPLT1	**26**	**+3.3V**	INPUT	Active LOW, ADDR 1. Is activated every time motion has been completed. When a motion is loaded goes HIGH
GND	**27**	**GND**	**POWER**	**GND connected to Ground Plane**
+5V	**28**	**+5V**	**POWER**	**+5V=Out , Max current= 100 mA**
SPI_CS0	**29**	**+3.3V**	OUTPUT	Active LOW , SPI Salve chip select to RMV859 ADDR0 Controller .
SPI_CS0	**30**	**+3.3V**	OUTPUT	Active LOW , SPI Salve chip select to RMV859 ADDR1 Controller .
TRIGGER0	**31**	**+3.3V**	OUTPUT	Active LOW, Trigger RMV959 Controller ADDR0. A LOW must be at least 30 uSec, and after can return to +3.3V.
TRIGGER1	**32**	**+3.3V**	OUTPUT	Active LOW, Trigger RMV959 Controller ADDR1. A LOW must be at least 30 uSec, and after can return to +3.3V.

ACK_BUSY0	**33**	**+3.3V**	INPUT	ACK=Acknowledge Active LOW, from RMV859 Controller ADDR0
				BUSY=Busy, Active HIGH

ACK_BUSY1	**34**	**+3.3V**	INPUT	ACK=Acknowledge Active LOW, from RMV859 Controller ADDR1
				BUSY=Busy, Active HIGH

ATTENTION 0	**12**	**+3.3V**	INPUT	Active LOW. RMV859 Controller ADDR0 informs to" Host" a full report

				VIA SPI, must be request, as an example this signal is activated when a limit switch is activated.
ATTENTION 1	**36**	**+3.3V**	INPUT	Active LOW. RMV859 Controller ADDR1 informs to” Host” a full report VIA SPI, must be request, as an example this signal is activated when a limit switch is activated.

Chapter 2

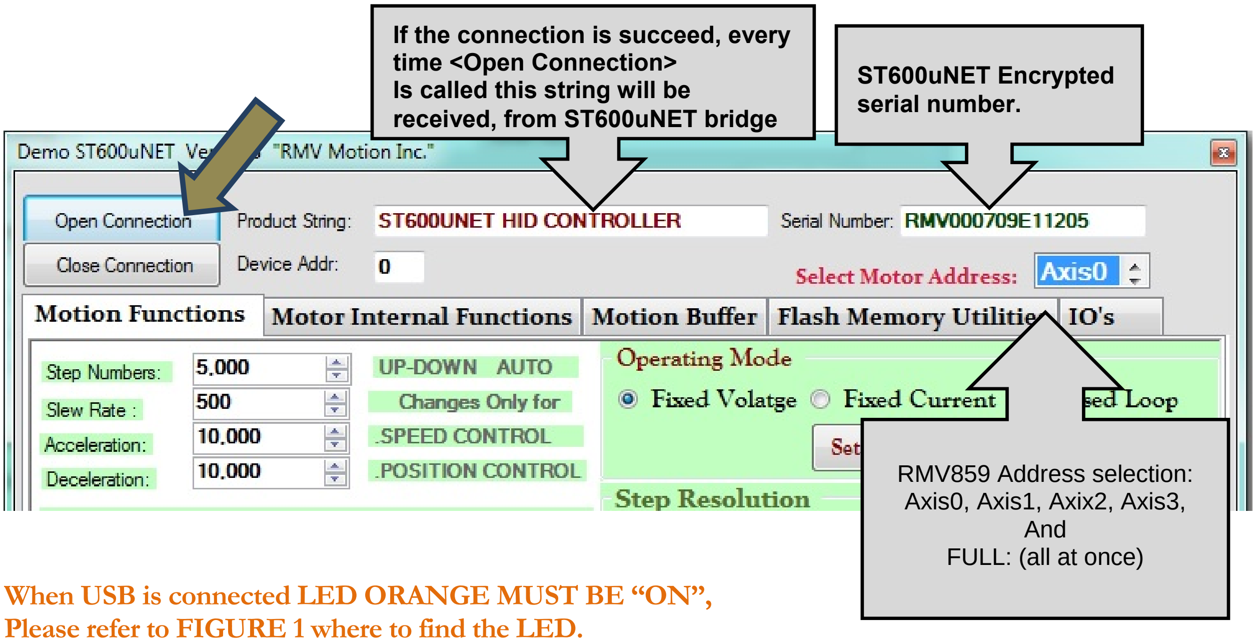
WINDOWS GUI APPLICATION DEMO

A free application that run in Windows xx O.S. and it can be downloaded from GitHub, please follows this link: https://github.com/rmvmotion/ST600uNET.git .

A full C++ Visual Studio project has been developed and published as open source as Apache II License. In the Demo mostly all mayor functionalities from ST600µNET Series has been developed, and can be recompiled and tested.

After the project has been downloaded and unzipped, under the directory /Executable the following files will be found: APP_ST600uNet.exe, SLABHIDDevice.dll, and ST600uNet.dll. The application can be run from this directory, before a re-compiled version is done.

Connect Host PC to ST600µNET



If the connection is succeed, every time <Open Connection> Is called this string will be received, from ST600uNET bridge

ST600uNET Encrypted serial number.

RMV859 Address selection: Axis0, Axis1, Axix2, Axis3, And FULL: (all at once)

Demo ST600uNET Ve... "RMV Motion Inc."

Product String: ST600UNET HID CONTROLLER

Serial Number: RMV000709E11205

Device Addr: 0

Select Motor Address: Axis0

Step Numbers: 5,000

Slew Rate : 500

Acceleration: 10,000

Deceleration: 10,000

Operating Mode: ● Fixed Volatge ○ Fixed Current ...sed Loop

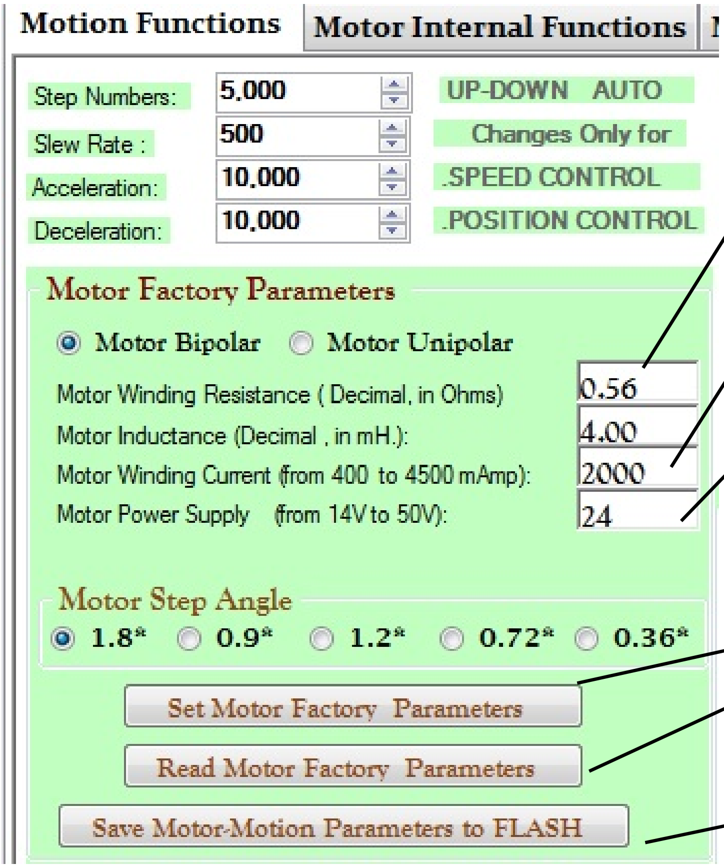
When USB is connected LED ORANGE MUST BE "ON", Please refer to FIGURE 1 where to find the LED.

Note4: When "FULL" address is selected the command sent, will be processed by all four RMV859 controllers, listening as broadcasting mode. The command is "acknowledged" by the address "0". Please refer to ST600µNet library reference.

Setting Motor Parameters

After ST600µNet has been connected to the host PC , the motor parapets "must" be set in order to RMV859 Controller can adjust its algorithms for current calculation , over current protection, current calculation maximum-minimum , chopper setup internal parameters, close loop parameters, and PI parameters from motor voltage supply .

After all motor parameters have been entered they can be <sent> to RMV859 Controller which address has been selected. All parameters can read from RMV859 in order to verify if any mistake occurred
When for certain all motor parameters are correct, they can be saved to RMV859 Internal FLASH Memory,
This process need to be done only one time, until a new stepper motor is connected to the board, or different motor power supply voltage power the board.



Step Numbers: 5,000
Slew Rate : 500
Acceleration: 10,000
Deceleration: 10,000

Motor Factory Parameters

● Motor Bipolar ○ Motor Unipolar

Motor Winding Resistance (Decimal, in Ohms) 0.56
Motor Inductance (Decimal , in mH.): 4.00
Motor Winding Current (from 400 to 4500 mAmp): 2000
Motor Power Supply (from 14V to 50V): 24

Motor Step Angle

● 1.8* ○ 0.9* ○ 1.2* ○ 0.72* ○ 0.36*

Motor Resistance Limit Values:
Minimal Resistance = 0.19 Ω
Maximum Resistance = 63.00 Ω

Motor Current Limit Values:
Current Minimal = 400 mA
Current Maximum = 4500 mA

Motor POWER Limit Values:
VOLTAGE Minimal = 18 DCV
VOLTAGE Maximum = 50 DCV

Set Parameters:
Motors Parameters are updated, and RMV859 will re-calculate motion equations

Read Motor Parameters:
RMV859 will send all motor parameters from internal memory

Save Motor Parameters to FLASH:
All motor parameters will be saved into Flash Memory, and every time Power ON the parameters will be read.

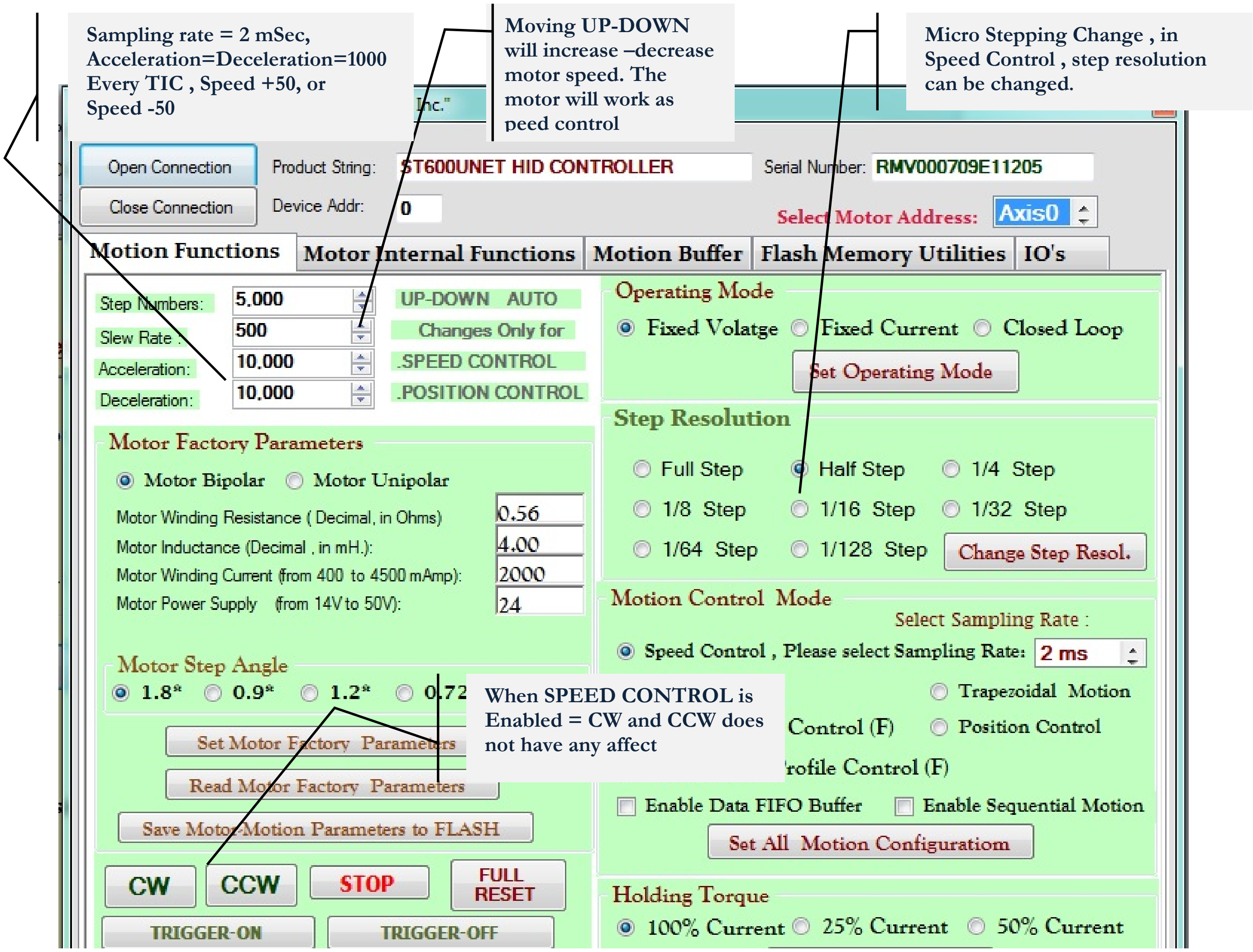
Move the motor by Speed Control

After the board is connected successful with host computer, and motor parameters have been set is time move, the motor can be move!!! It is Time to get some Fun!!!.

After POWER On, "RMV859" has as default the following motion parameters:

Slew Rate = 500 Steps/sec,
Acceleration = 10000 Steps/sec^2,
Deceleration = 10000 Steps/ sec^2

If Motor Power is OK (LED Green ON), USB Connection (LED Orange ON), motor parameters SET, and Motor phases well connected (please refer to FIGURE3)
Pressing <TRIGGER-ON>, MOTOR WILL START TURNING.



Sampling rate = 2 mSec, Acceleration=Deceleration=1000 Every TIC , Speed +50, or Speed -50

Moving UP-DOWN will increase –decrease motor speed. The motor will work as peed control

Micro Stepping Change , in Speed Control , step resolution can be changed.

When SPEED CONTROL is Enabled = CW and CCW does not have any affect

Product String: ST600UNET HID CONTROLLER
Serial Number: RMV000709E11205
Device Addr: 0
Select Motor Address: Axis0

Step Numbers: 5,000
Slew Rate : 500
Acceleration: 10,000
Deceleration: 10,000

UP-DOWN AUTO Changes Only for .SPEED CONTROL .POSITION CONTROL

Operating Mode: ● Fixed Volatge ○ Fixed Current ○ Closed Loop

Step Resolution: ○ Full Step ● Half Step ○ 1/4 Step ○ 1/8 Step ○ 1/16 Step ○ 1/32 Step ○ 1/64 Step ○ 1/128 Step

Motor Factory Parameters: ● Motor Bipolar ○ Motor Unipolar

Motor Winding Resistance (Decimal, in Ohms): 0.56
Motor Inductance (Decimal , in mH.): 4.00
Motor Winding Current (from 400 to 4500 mAmp): 2000
Motor Power Supply (from 14V to 50V): 24

Motor Step Angle: ● 1.8° ○ 0.9° ○ 1.2° ○ 0.72

Motion Control Mode: Select Sampling Rate :
● Speed Control , Please select Sampling Rate: 2 ms
○ Trapezoidal Motion
Control (F) ○ Position Control
rofile Control (F)
□ Enable Data FIFO Buffer □ Enable Sequential Motion

Holding Torque: ● 100% Current ○ 25% Current ○ 50% Current

FIGURE:5. ST600µNET – Demo principal desktop

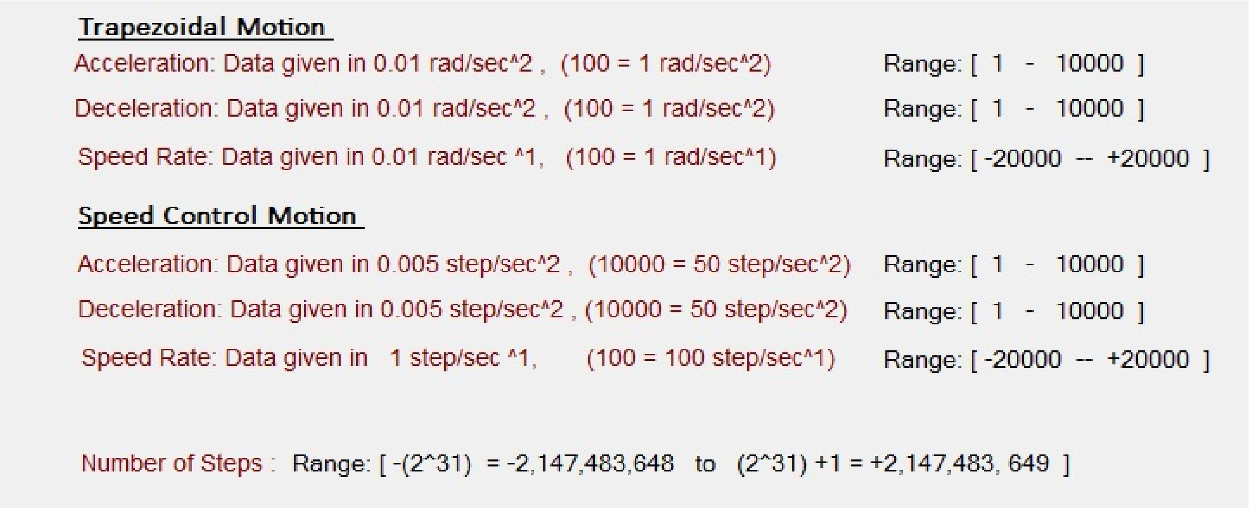
<u>Holding Torque:</u> The holding current after motor has been stopped can be set to: 100%, 75%, 50%., or 25%
<u>Operating Mode Selection:</u> *Voltage Constant*, *Current calculated* in all quadrants, *Closed Loop* : PI control.

<u>Motion Control Mode:</u>

- <u>Speed Control:</u> Has three sampling rate: 2 ms (default) , 1 ms, and 5 ms. When this method is used two parameters are used : Slew Rate and Acceleration/Deceleration, these two parameter are calculated as "absolute value" for example an Acceleration=Deceleration = 10000, it will increment or decrease the motor speed in every TIC. Speed(k) = Speed(k-1)+ 50; or Speed(k) = Speed(k-1)- 50;

- <u>Position Control:</u> In this method the motor will move to the position designated in Step Numbers, at the speed designated in Slew Rate. The motor speed need to be careful chosen in conjunction to step size.

- <u>Trapezoidal Motion:</u> In this method the speed trajectory is trapezoidal. All dimension are specified in 0.01 rad/sec. The following Figure shows dimensions and difference between Speed Control and Trapezoidal.



<u>Trapezoidal Motion</u>

Acceleration: Data given in 0.01 rad/sec^2 , (100 = 1 rad/sec^2) Range: [1 - 10000]

Deceleration: Data given in 0.01 rad/sec^2 , (100 = 1 rad/sec^2) Range: [1 - 10000]

Speed Rate: Data given in 0.01 rad/sec ^1, (100 = 1 rad/sec^1) Range: [-20000 -- +20000]

<u>Speed Control Motion</u>

Acceleration: Data given in 0.005 step/sec^2 , (10000 = 50 step/sec^2) Range: [1 - 10000]

Deceleration: Data given in 0.005 step/sec^2 , (10000 = 50 step/sec^2) Range: [1 - 10000]

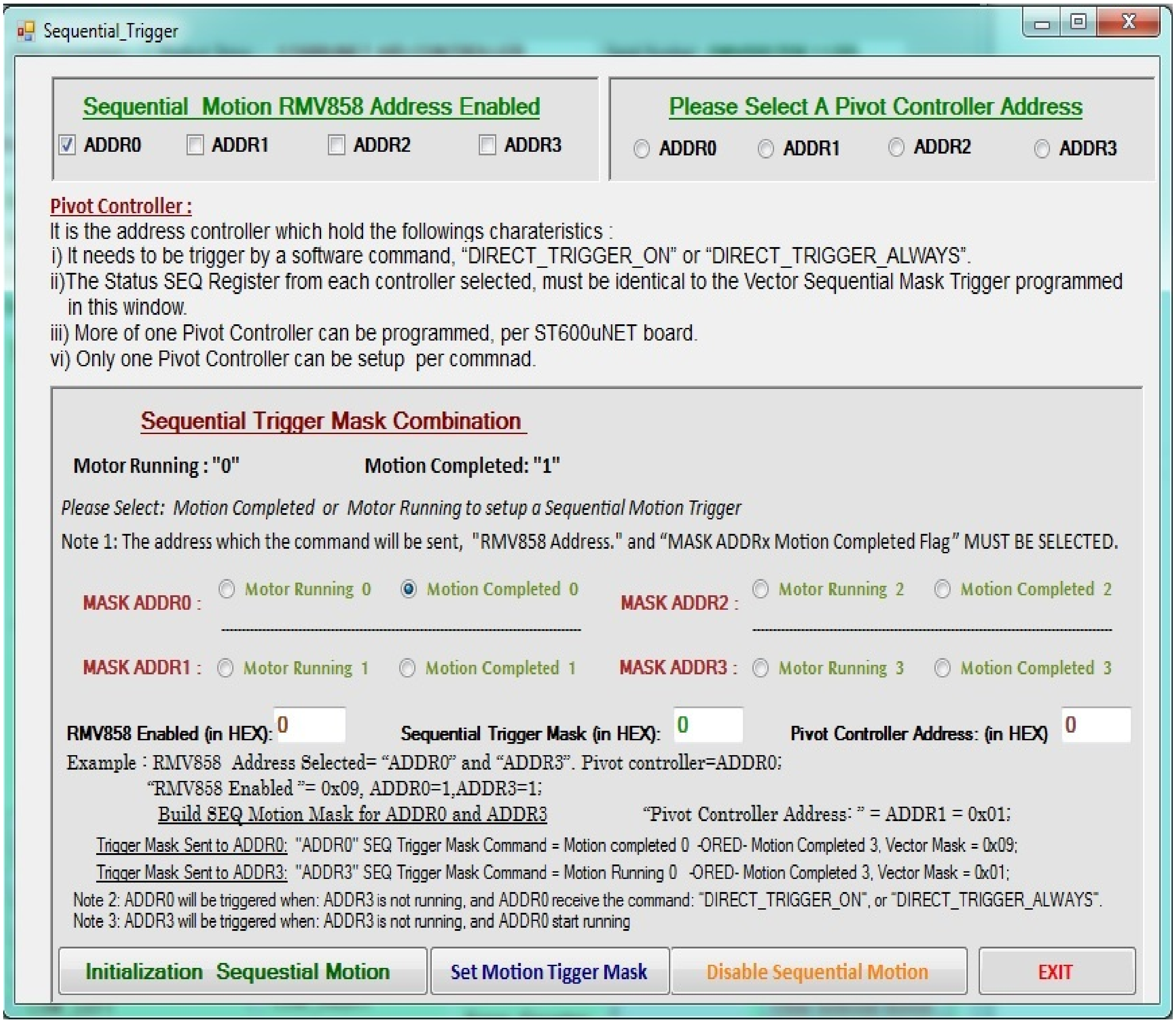
Speed Rate: Data given in 1 step/sec ^1, (100 = 100 step/sec^1) Range: [-20000 -- +20000]

Number of Steps : Range: [-(2^31) = -2,147,483,648 to (2^31) +1 = +2,147,483, 649]

- <u>Speed Profile and Trapezoidal Profile Control FIFO:</u> Speed Profile has the same properties and parameters than Speed control , and Trapezoidal Profile same than Trapezoidal Motion. All motion are written the RMV859 internal FIFO (700 words depth). In order to download an external file to download RMV859 internal FIFO, please refer to this manual in Section FIFO Program.

- <u>Enable Sequential Motion:</u> This function allow to trigger "By Hardware " any RMV959 controller sequence via a pre-programmed "motion completed sequence". For example in order to start running controller address "0", the sequence has been programmed to controllers address "1" and "3" must have "motion completed FLAG in High for less of 1-3 µsec. This function works in all kernel steps mode.

Enabling sequential trigger will allow to perform any kind of motion with "no motor jitter", in all four axes. The following figure explains how to setup this functionality.



Sequential Motion RMV858 Address Enabled

☑ ADDR0 □ ADDR1 □ ADDR2 □ ADDR3

Please Select A Pivot Controller Address

○ ADDR0 ○ ADDR1 ○ ADDR2 ○ ADDR3

Pivot Controller :

It is the address controller which hold the followings charateristics :

i) It needs to be trigger by a software command, "DIRECT_TRIGGER_ON" or "DIRECT_TRIGGER_ALWAYS".

ii)The Status SEQ Register from each controller selected, must be identical to the Vector Sequential Mask Trigger programmed in this window.

iii) More of one Pivot Controller can be programmed, per ST600uNET board.

vi) Only one Pivot Controller can be setup per commnad.

Sequential Trigger Mask Combination

Motor Running : "0" Motion Completed: "1"

Please Select: Motion Completed or Motor Running to setup a Sequential Motion Trigger

Note 1: The address which the command will be sent, "RMV858 Address." and "MASK ADDRx Motion Completed Flag" MUST BE SELECTED.

MASK ADDR0 : ○ Motor Running 0 ● Motion Completed 0

MASK ADDR2 : ○ Motor Running 2 ○ Motion Completed 2

MASK ADDR1 : ○ Motor Running 1 ○ Motion Completed 1

MASK ADDR3 : ○ Motor Running 3 ○ Motion Completed 3

RMV858 Enabled (in HEX): 0 Sequential Trigger Mask (in HEX): 0 Pivot Controller Address: (in HEX) 0

Example : RMV858 Address Selected= "ADDR0" and "ADDR3". Pivot controller=ADDR0;
"RMV858 Enabled "= 0x09, ADDR0=1,ADDR3=1;
Build SEQ Motion Mask for ADDR0 and ADDR3 "Pivot Controller Address: " = ADDR1 = 0x01;

Trigger Mask Sent to ADDR0: "ADDR0" SEQ Trigger Mask Command = Motion completed 0 -ORED- Motion Completed 3, Vector Mask = 0x09;

Trigger Mask Sent to ADDR3: "ADDR3" SEQ Trigger Mask Command = Motion Running 0 -ORED- Motion Completed 3, Vector Mask = 0x01;

Note 2: ADDR0 will be triggered when: ADDR3 is not running, and ADDR0 receive the command: "DIRECT_TRIGGER_ON", or "DIRECT_TRIGGER_ALWAYS".
Note 3: ADDR3 will be triggered when: ADDR3 is not running, and ADDR0 start running

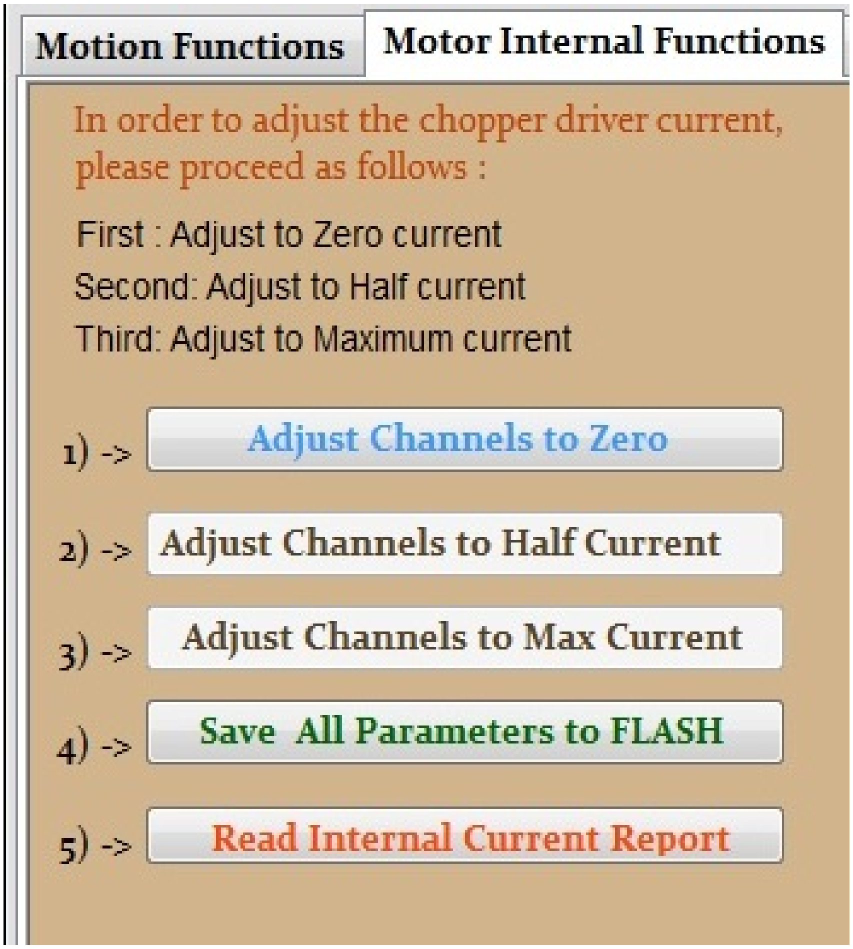
FIGURE:6. ST600µNET –Sequential Trigger desktop

Setting as Pivot Controller: Is the controller address which will receive the first trigger sent by a software command. Immediately the software trigger is received "Motion Completed Flag" will go LOW and will start the motion. The application software will trigger only Pivot Controller address, and the rest of the controllers will be triggered automatically via hardware.

Only one controller can be set as Pivot controller from address 0 to 3. The rest controllers (No Pivot) which address have been selected, in order to trigger a new motion, RMV859 will wait until the "four" bits mask are matched (pre-programmed) with the "motion completed bus [0,1,2,3].

Motor Internal Functions

This Demo allows setup the chopper driver internal parameters for the motor connected and motor factory parameters.



In order to adjust the chopper driver current, please proceed as follows :

First : Adjust to Zero current
Second: Adjust to Half current
Third: Adjust to Maximum current

1) -> Adjust Channels to Zero

2) -> Adjust Channels to Half Current

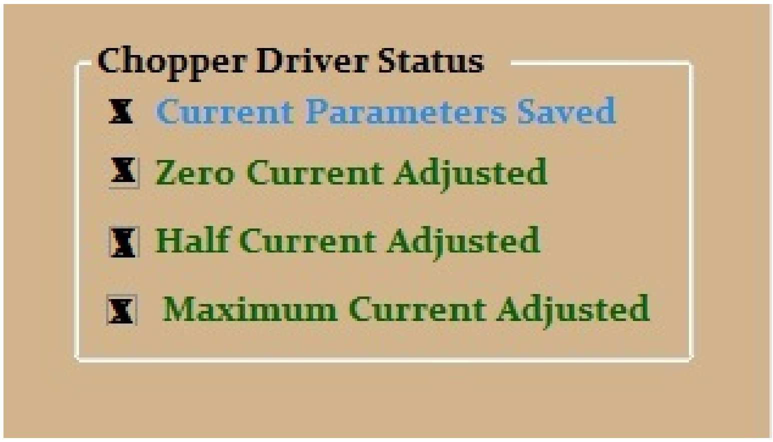
3) -> Adjust Channels to Max Current

4) -> Save All Parameters to FLASH

5) -> Read Internal Current Report

FIGURE7: ST600µNET – Chopper Driver Adjustment

After all parameters has been saved to RMV859 internal FLASH Memory, pressing the button " Read Internal Current Report", all status of the internal chopper parameters can be seen with the flags checked.



Chopper Driver Status

☒ Current Parameters Saved

☒ Zero Current Adjusted

☒ Half Current Adjusted

☒ Maximum Current Adjusted

FIGURE8: ST600µNET – Status Chopper Driver Adjustment

Seek Home, and Move Out from a Limit Switch

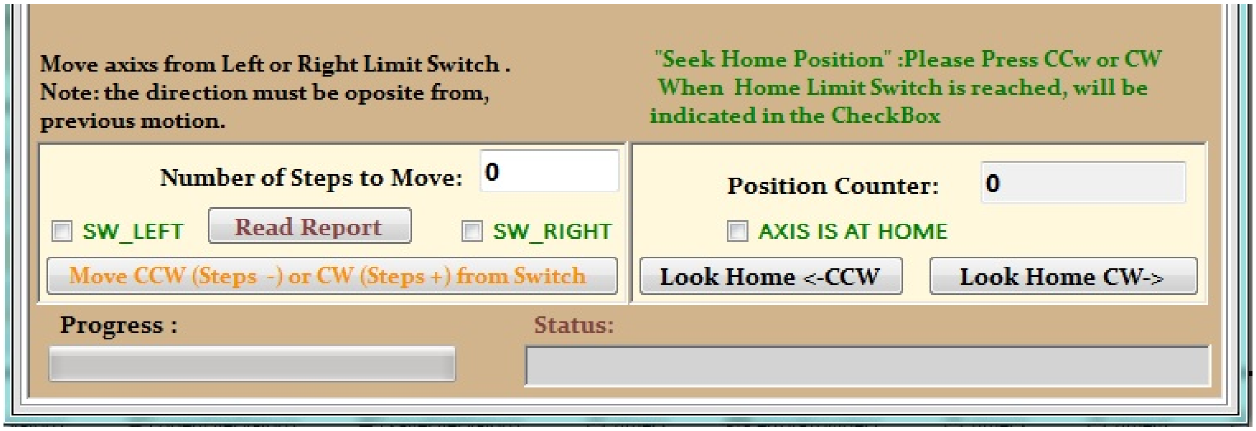
The following Figure9 shows an axis taken from a Limit Switch, and an axis seeking Home Switch.
Both tasks have different setting:

i) Seek Home: RMV859 Kernel must be enabled as "Speed control"

ii) Take from Limit Switch: RMV859 Kernel must be enabled as "Trapezoidal control"

SEEK HOME: Setting as Speed control the motor will move CW or CCW depending on the sign of the parameter Speed

Take Axis From Limit Switch: this function will move the axis to a maximum of 1000 steps, to an inverse direction from previous motion. Motion direction is based on the sign of step numbers, positive: CW, Negative: CCW.
If the direction is the same from the previous, RMV859 will inform an error ocurred



Move axixs from Left or Right Limit Switch .
Note: the direction must be oposite from, previous motion.

"Seek Home Position" :Please Press CCw or CW When Home Limit Switch is reached, will be indicated in the CheckBox

Number of Steps to Move: 0

Position Counter: 0

FIGURE9: ST600µNET – Seek Home and Move Axis from Limit Switch

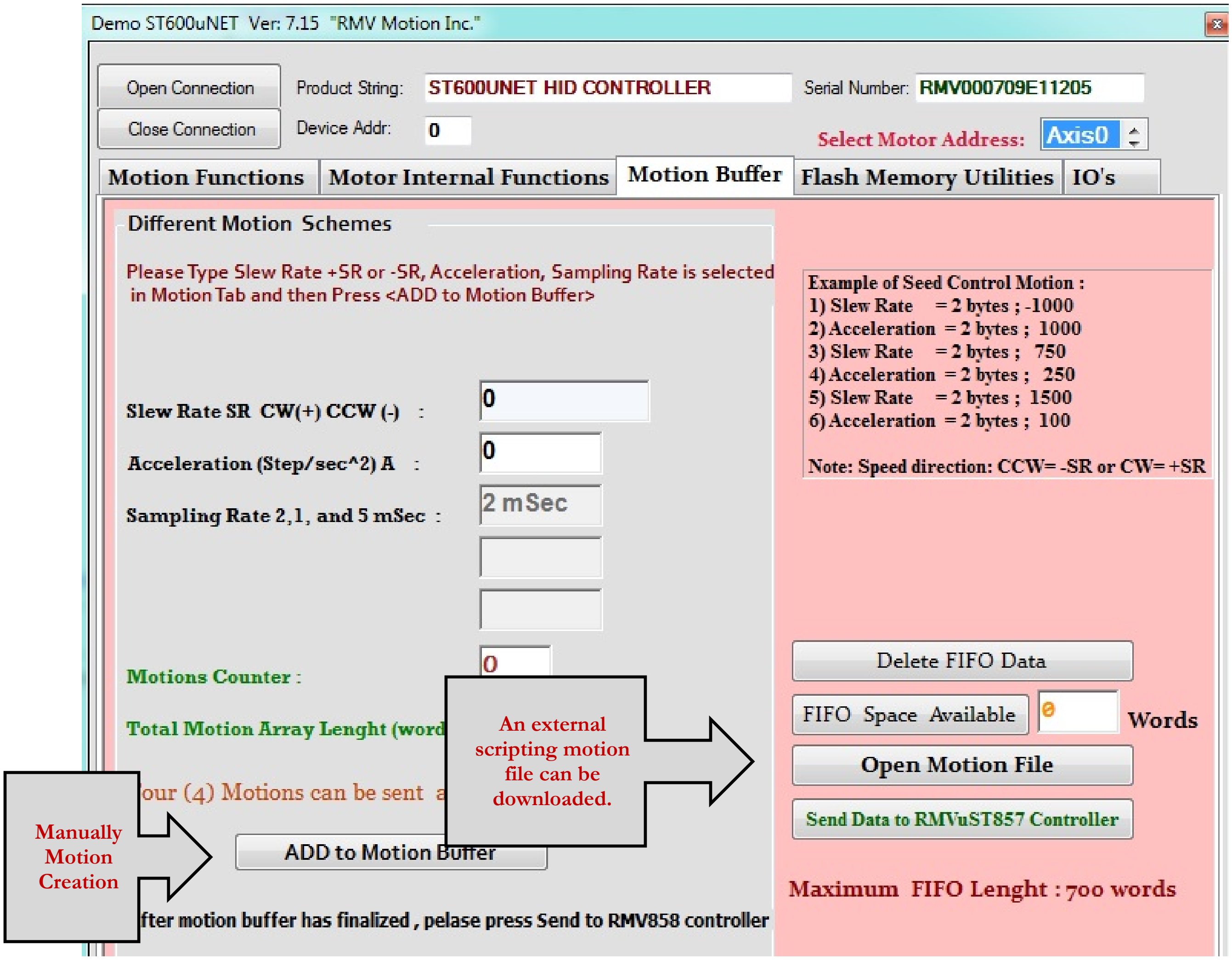
Chapter 3

FIFO BUFFER OPERATIONS

The RMV859 controllers have an internal FIFO to store commands in sequential mode. The maximum capacity for the FIFO is 700 words. Two kind of motion are support in this firmware version: Profile Trapezoidal Control and Profile Speed Control .

```
#define FIFO_PROFILE_TRAPEZOIDAL_CONTROL   'FT'

Struck MANDATORY COMMANDS:
        Command:   'N'              2 bytes
        Total Steps Number:         4 bytes (signed)
        Command:     'A'            2 bytes
        Acceleration:   (unsigned)  2 bytes
        Command:     'D'            2 bytes
        Deceleration:   (unsigned)  2 bytes
        Command 'SR'(slew rate)     2 bytes
        Slew Rate:   (unsigned)     2 bytes
                                  ____________
                       TOTAL      18 bytes; 9 = words


#define TRAPEZOIDAL_ONE_SET_WORD              9 = 18 bytes
#define TRAPEZOIDAL_CONTROL_MAX_ITERATIONS    3   Numbers of commands motion   in one 
time to RMV859 inetrnal FIFO RAM


#define FIFO_PROFILE_SPEED_CONTROL     'FE'   // Set Profile Speed control
       Command 'SR' (unsigned)      2 bytes; ‘SR’
       Slew Rate:   (unsigned)      2 bytes; SR=1000
       Command  'A'                 2 bytes  ; 'A'
       Acceleration (unsigned)      2 bytes  ; A=10000
       Command  'n'                 2 byes   ; 'n'
       Number of Steps (signed)     4 bytes  ; n=18500 CW(+), CCW(-)
                                ____________
                       TOTAL      14 bytes  = 7 words

#define SPEED_CONTROL_ONE_SET_WORD      7 = 14 bytes
#define SPEED_CONTROL_MAX_ITERATIONS    4 Numbers of commands motion in one time to 
RMV859 internal FIFO RAM
```

FIFO Demo Desktop



Demo ST600uNET Ver: 7.15 "RMV Motion Inc."

Product String: ST600UNET HID CONTROLLER

Serial Number: RMV000709E11205

Device Addr: 0

Select Motor Address: Axis0

Different Motion Schemes

Please Type Slew Rate +SR or -SR, Acceleration, Sampling Rate is selected in Motion Tab and then Press <ADD to Motion Buffer>

Slew Rate SR CW(+) CCW (-) : 0

Acceleration (Step/sec^2) A : 0

Sampling Rate 2,1, and 5 mSec : 2 mSec

Motions Counter : 0

Total Motion Array Lenght (word

our (4) Motions can be sent a

fter motion buffer has finalized , pelase press Send to RMV858 controller

Example of Seed Control Motion :
1) Slew Rate = 2 bytes ; -1000
2) Acceleration = 2 bytes ; 1000
3) Slew Rate = 2 bytes ; 750
4) Acceleration = 2 bytes ; 250
5) Slew Rate = 2 bytes ; 1500
6) Acceleration = 2 bytes ; 100

Note: Speed direction: CCW= -SR or CW= +SR

FIFO Space Available 0 Words

Maximum FIFO Lenght : 700 words

NOTE5: The Flag "*Enable Data FIFO Buffer*", must be checked otherwise an error will be issued

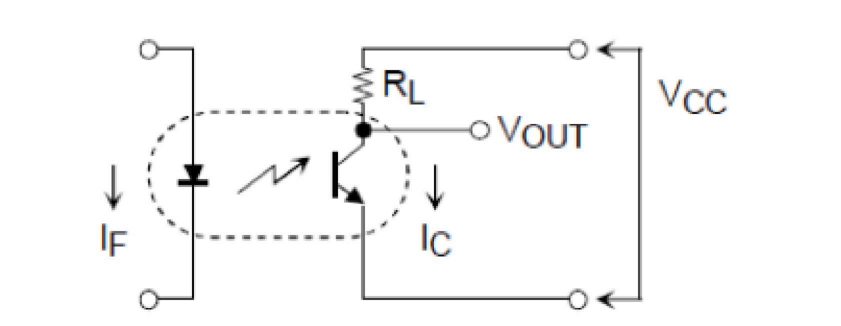
Two samples files are provide as example in this demo application for both kind of motion supported for this version: *RMV_Profile_Speed_Control.txt*, *RMV_Profile_Trapezoidal_Control.txt* .

Chapter 4

DIGITAL IO's

The ST600µNET Series has Six Isolated Outputs from Output[0,,5], and four Isolated Input from Input[0,,3]. In addition, three analog inputs with signal conditioning are included, 0V to 5V, 4-20mA, and on board real time heatsink temperature measurements, in order the application monitoring heat transfer to ambient.

Digital Output: Please refer to FIGURE 3 and FIGURE 4 to see different output configurations .



Transistor photocoupler

$$If = \frac{3.3 - 1.2 - 0.4}{200} = 8.5\ mA$$

$$CRT = \left(\frac{Ic}{If}\right)\ 100[\%]\ ,$$

When the collector current IC (ON) flows on the output side of the photocoupler, output VOUT (ON) has to be less than VL (the required low level voltage) as follows:

VCC – (IC (ON) x RL) = VOUT (ON) < VL

Therefore: $$RL > \frac{Vcc-Vl}{IC(ON)}$$

Generally when the RL value is large, the switching response time increases, so the RL value should be kept as small as possible.

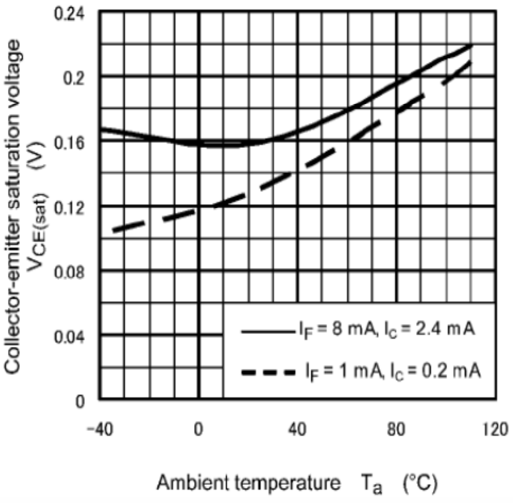


Fig. 11.12 $V_{CE(sat)}$ - T_a

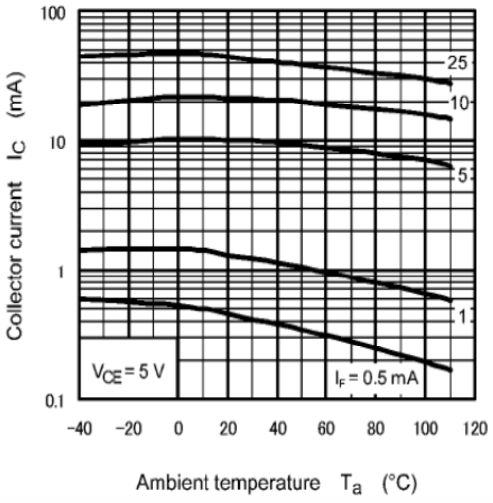


Fig. 11.13 I_C - T_a

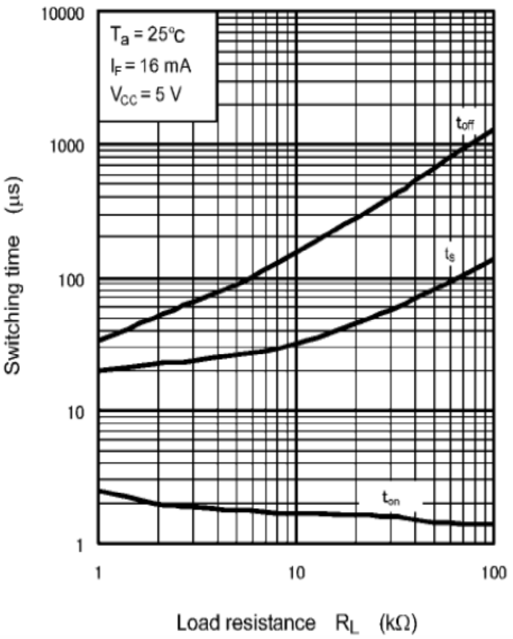


Fig. 11.14 Switching Time - R_L

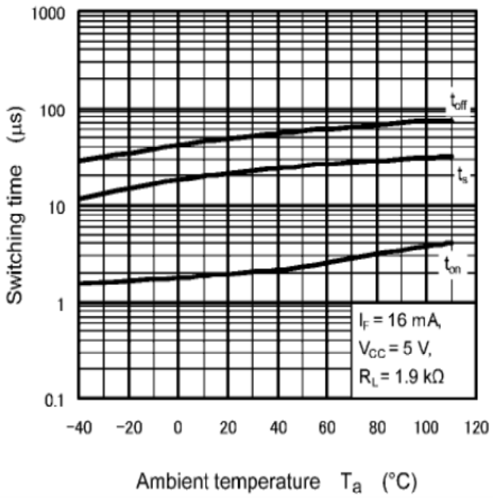


Fig. 11.15 Switching Time - T_a

Index