

# Drive special inputs Limit switches

**Application Note** 

**Easy Motion Studio II** 

Next Intelligent **Move** 

Your



# Table of content

1. Introduction	3
2. Hardware limit switches	3
2.1 Limit switches inputs default behavior, configuration and usage	3
2.2 Limit switches related interrupts	6
2.3 Limit switches inputs usage as general-purpose inputs	7
3. Software limits configuring and usage	9

## 1. Introduction

This application note describes the default functionality of limit switch inputs and explains how to disable it to use the respective digital inputs as general-purpose inputs. T

The document also details how to restrict load movement using the software limit switches option.

### 2. Hardware limit switches

Limit switches are control components used to determine the presence, passing, positioning, and end of travel of an object. The simplest version is a push button, but in complex applications, optical, induction, or capacitive sensors are used.

#### 2.1 Limit switches inputs default behavior, configuration and usage

Present on all Technosoft servo drives, the "IN2/LSP" (positive limit switch) and "IN3/LSN" (negative limit switch) digital inputs allow connecting the system limit switch sensors.

When one of the limit switches became active, the drive performs the following default actions:

- stops the motion, using a QUICK STOP profile
- sets the correspondent bit in the MER error register (bit 6 LSP active / bit 7 LSN active)
- executes the code inside the correspondent limit switch interrupt routine, if it is enable and set to trigger on the inactive to active limit switch transition
- prevents any motion that moves the load against the active limit switch

The limit switches active level can be set during the drive commissioning through the "Inputs/ outputs" page in the "Setup tree".

EasyMotion Studio II - P091.APN4.1 Drive : Project Application Communication	special inputs (Limit Switches) Tools Control panels Utilitie	s Options Help			- U >
<mark>- [</mark> - <b>[</b> ] <b>•</b> •	o 🖬 🖳 🛲 👔		Work offline Set all apps offline	2 🛓 🕨 O	Z Axis
Application tree         •         #           •	Inputs / outputs Digital inputs Configure inputs as Sink (F	2NP) 🔹 🕕			
M Motion	Input	Туре	Function	Active level	
menory seeings	INO	Sink (PNP)	General purpose input		
	IN1	Sink (PNP)	General purpose input		
	IN2/LSP	Sink (PNP)	Limit switch + *	High	-
• • • • • • • • • • • • • • • • • • •	IN3/LSN	Sink (PNP)	Limit switch -	High	¥
etup tree 🔹 🔻 🕸	IN4	Sink (PNP)	General purpose input		
Quick setup Advanced setup	IN5/Enable	Sink (PNP)	Enable *	Low	Ŧ
- 🛇 Motor - 🖧 Feedback	Digital outputs Configure outputs as Sink	(NPN) - 🕐			
↓↑ Inputs / outputs	Output	Туре	Function	Active level	
	OUT0	Sink (NPN)	General purpose output		
- [1] External reference	OUT1	Sink (NPN)	General purpose output		
Application settings	OUT2/Error	Sink (NPN)	Error *	Low	
Motion settings	OUT3/Ready	Sink (NPN)	Ready *	Low	
Fieldbus settings	OUT4	Sink (NPN)	General purpose output		
Protections and limits     P Controllers     Q Automatic tuning	Analog inputs Swap Reference and Feedba	ack inputs	AD2 measurement rang	e 0.51/	-10 V
26 Manual tuning & lest	Refere	nce (AD5)	AD5 measurement rang	e 05 V ±	±10 V 🕕
00					

Figure 1 - Limit switches active level selection

<u>**Remark</u>** If the drive hardware design allows configuring the digital inputs type (NPN or PNP), it can also be done through the "Inputs/Outputs" page and applies to the limit switch inputs as well.</u>

The default behavior of the limit switches can be easily observed by plotting the position, speed, and digital inputs status (or only the used limit switch) in Logger or Scope.

For example, the drive was programmed to perform a trapezoidal position profile as in the picture below.



Figure 2 – Motion profile example

After running the program, the positive limit switch (IN2/LSP) was manually triggered. As a result, the motion stopped. The position, target speed and digital outputs status was plotted using the Logger tool.



Figure 3 – Positive limit switch default behavior

**<u>Remark</u>** The Logger setup and usage is detailed in the EasyMotion Studio II help topics. The related topic can be open by pressing the "?" sign in the Logger window (upper left corner).

The first plot shows the motor positon (red line) starting from 0 IU and following the imposed reference. When the positive limit switch (blue line) became active, the drive automatically executed a quick stop profile, stopping the motion.

The second plot shows the target speed (orange line) that goes to 0 IU, when the limit switch (blue line) triggers, with the set quick stop deceleration.

<u>**Remark**</u> If needed, the quick stop deceleration value can be set / changed through the "Motor Commands" wizard dialogue.

	e 🖳 🚥 👔 🖬 🖻	🟦 🔹 🗔 🗋 Work offline Set all apps offline 🥕 🛓 🕨 🛛 🍕 🕺 🗶 🖕
ication tree 🔹 🔹 🖛		
P091.APN4.1 Drive special inputs (Li	Trapezoidal Profiles	Search Data K + +
LSP and LSN - [2]	. S Curve	
- S Setup	PT 🔁	Motor Commands ×
M Motion	PVT	<ul> <li>Activate the control loops and PWM outputs (AXISON)</li> </ul>
🖟 Homings 🕴	External	<ul> <li>Deactivate the control loops and PWM outputs (AXISOFF)</li> </ul>
- 🖟 Functions 🛛 🕅 🕅	Electronic Gearing	O Stop motion with acceleration / deceleration set
- 🔃 Interrupts	Electronic Camming	O Update immediately
CAM Tables	Motor Commands	O Update on event
Memory settings	Position Triggers	Set actual position to
命	Homing	Variable
M	Contouring	Sat current limit
T∕-	Test	Variable
I C*	Events	Set quick step deceleration rate     Value
LJ	Jumps and Function Calls	Variable
1	Inputs / outputs	OK Cancel Help
<u>16</u>	16-bit Integer	
32	32-bit Long or Fixed	
*-	Arithmetic	
°‡₫	Data Transfer Between Axes	
	Send Data To Host	
MISC	Miscellaneous	
4	Interrupt Settings	
txt	Free Text	

Figure 4 – Quick stop deceleration rate adjustment

The status of the limit switches is included in the MER error register and can be visualized through the "Drive Status" control panel. The screenshot below was taken while performing the above example.

SRH - Status Register High		SRL - Status Regis	ter Lo	w	MER - Error Register		DER - Detail Error Register		DER2 - Detail Error Registe	er 2
15 - Fault	0	15 - Axis is ON		1	15 - Enable is inactive	0	15 - EEPROM Locked	0	15 - Output frequency	
14 - In Cam	0	14 - Event set has oc	cured	0	14 - Command error	0	14 - ENA hardware error	0	14 - Reserved	
13 - In freeze control	0	10 - Motion is comple	eted	1	13 - Under voltage	0	13 - Self check error	0	13 - Reserved	
12 - In Gear	0	8 - Homing/CALLS a	active	0	12 - Over voltage	0	12 - TML Heartbeat ignored	0	12 - Reserved	
11 - I2t warning - Drive	0	7 - Homing/CALLS w	varning	ı 0	11 - Over temp Drive	0	11 - Start mode failed	0	11 - Reserved	
10 - I2t warning - Motor	0	Register lagend:			10 - Over temp Motor	0	10 - Encoder broken wire	0	10 - Reserved	
9 - Target reached	1	1 - Yes / True		ENDINIT	9 - I2t	0	9 - UPD ignored for S-curve	0	9 - Reserved	
8 - Capture event/interrupt	0	o norraise			8 - Over current	0	8 - Invalid S-curve profile	0	8 - Reserved	
7 - LSN event/interrupt	0				7 - LSN (limit -) active	0	7 - Software LSN active	0	7 - Reserved	
6 - LSP event/interrupt	0	IMPORTANT! Ch Supply voltage and	neck SRH d some :	H.O ! status	6 - LSP (limit +) active	1	6 - Software LSP active	0	6 - Position wraparound	
5 - Autorun enabled	1	or error bits are se ENDINIT is e	et ONLY xecuted.	after	5 - Feedback error	0	5 - Cancelable call ignored	0	5 - Hall sensor missing	
4 - Over position trigger 4	1	If SRH.0 = 0, down reset the drive ar	nload a s nd press	setup, the	4 - Serial comm. error	0	4 - UPD ignored	0	4 - AEI interface error	
3 - Over position trigger 3	1	command. You can	also run	a TML	3 - Control error	0	3 - Function not available	0	3 - BiSS sensor missing	
2 - Over position trigger 2	1	of ENDIN	lides ex IIT.	ecution	2 - Invalid setup data	0	2 - Homing not available	0	2 - BiSS data error	
1 - Over position trigger 1	1				1 - Short-circuit	0	1 - TML stack underflow	0	1 - BiSS data warning	
0 - ENDINIT executed	1				0 - CANbus error	0	0 - TML stack overflow	0	0 - BISS CRC error	
Supply voltage [V] 24	.007	Drive temperature [°C	C]	49.443						

Figure 5 - Drive Status control panel that shows the MER error register

**<u>Remark</u>** As a default drive behavior, while the positive limit switch is active, the drive accepts only negative position or speed commands.

#### 2.2 Limit switches related interrupts

The TML interrupts are special routines that the drive can execute automatically when the triggering condition (continuously monitored by the drive) occurs. A detailed description of the TML interrupts can be found in the related EasyMotion Studio II help topic or in the "<u>TML interrupts usage</u>" application note.

The available interrupts are displayed when selecting the "Interrupts" subtree item. The "Interrupts" page also allows setting one or more interrupts as "user-defined" or reloading the default code.

For limit switch inputs, the associated interrupts are: "*int6 – LSP programmed transition detected*" and "*int7 – LSN programmed transition detected*".

Project Application Communication To	ols Control panels Utilities Options	Help	😡 🗌 Work o	ffline Set all apps	offline
pplication tree 🔹 🔹 🖡	Interrupts 🔞				
4- 📃 LSP and LSN - [2]	TML Interrupt	Default	User defined		
- S Setup	int0 - STO/Enable input has changed	۲	0		Go to code
Motion	int1 - Short-circuit	۲	0		Go to code
- Functions	int2 - Software protections	۲	0		Go to code
Interrupts	int3 - Control error	۲	0		Go to code
int6 - LSP programm	int4 - Communication error	۲	0		Go to code
int7 - LSN programm	int5 - Position wraparound	۲	0		Go to code
CAM Tables	int6 - LSP programmed transition detected	0	۲	Reload default	Go to code
Memory settings	int7 - LSN programmed transition detected	0	۲	Reload default	Go to code
	int8 - Capture input transition detected	۲	0		Go to code
	int9 - Motion is completed / in velocity	۲	0		Go to code
	int10 - Time period has elapsed	۲	0		Go to code
	int11 - Event set has occurred	۲	0		Go to code
	int12 - Position trigger 14 change detected	۲	0		Go to code
	int13 - Digital Input X programmed transition	۲	0		Go to code

**Figure 6** – *Limit switches related interrupt routines* 

When selecting an interrupts from the list, the related code page will open, allowing to modify the existent code (if any) or to implement the needed custom functionality.

BasyMotion Studio II - P091.APN4.1 Drive	special inputs (Limit Switches)	_	
Project Application Communication	Tools Control panels Utilities Options Help		
<b>. . . . .</b>	💿 📟 📟 👔 👔 🖪 🚖 📩 🗔 🗆 Work offline Set all apps	offline 🎤	Axis off <b>F</b> 3
Application tree			
P091.APN4.1 Drive special inputs (Li	Search Search		0/0
LSP default behavior - [2]	$\land$		1
— S Setup	M		ter l
- M Motion			1
— 🕞 Homings	<b>₽</b> −		Ţ
- 🕞 Functions	Pr 8		×
- 🕞 Interrupts			1
int6 - LSP programm			
int7 - LSN programm	<b>*</b>		
CAM Tables			{ <del>}</del> }
Memory settings	Ţ•\		{1}

**Figure 7** – LSP limit switches interrupt routine body

For limit switch interrupts, there is no code inside the interrupt body, as the related functionality is implemented at the firmware level.

Project	Application	Com	munic	4.1 Drive	special Tools	Cont	(Limit	inels	utilit	ies	Optic	ons	Help										_		
					0					1	400	\$	*	Ģ	<b>N</b>	ork off	line S	et all ap	ps offlin	2	*	Þ	0	Ŧ	Axis F
pplicatio	n tree	ive spec behavio tions rupts Tables tings	cial inp r - [2]	т म puts (Lii	LL 다 한 것 첫 10 % 여 명 % 우 송 > > >	Trape S Cur PT Extern Electr Moto Positi Homi Conto Test Event Jump Input	rzoidal ve nal onic ( on Tri ing ouring s and s and s / ou	Profile Gearing Cammin Imands ggers I Function	ng i i i i i i i i i i i i i i i i i i i			Intern TML Int0 - Int1 - Int2 - Int3 - Int5 - Gener When CAPP	Interru Enable Interru STO/E Short Softwa Contro Comm Positic LSP pr rate lim n transii	ings pts e globa pt nable in circuit are pro ol error uunicati on wrap ogram it switc tion occ t oositi	nput I tectio	Disal Disal nas chai ns ror nd errupt o positior	nged	bally teted ow → H digh →	Enable	Disable	Rese inter	t prev rupt r	ious TM equests ] ] ] ] ] ] ] ] ] ] ] ] ] ] ] ] ] ]		etails
					16	16-bi	t Integ	ger			(	Int7 -	LSN p	rogram	med	transitio	on dete	ected	✓			V	]		÷
					 	32-br	t Long	) or Fix	ed			Int8 -	Captu	re inpu	t tran	sition d	letecte	d					]		+
					7-	Arithr	metic					Int9 -	Motio	n is cor	mplet	ed / in v	velocit	/					]		
					ŤŌ	Data	Transf	er Betv	veen A	xes		Int10	- Time	period	has e	lapsed							]		+
						Send	Data	To Hos	t			Int11	- Event	set ha	s occu	irred							]		
					MISC Miscellaneous					_		Int12 - Position trigger 14 change detected					ected					]			
					ź	Interr	upt Se	ettings				Int13	- Digita	al Input	t X pro	ogramn	ned tra	nsition					]		
					t≺t	Free	Text			-											OK	C	ancel	He	elp

The interrupts can be enabled and configured through the "Interrupts Settings" wizard.

Figure 8 – Interrupts settings wizard dialogue

For limit switch interrupts, the wizard also allows setting the input transition that will generate the related interrupt.

#### 2.3 Limit switches inputs usage as general-purpose inputs

The functionality of the limit switch inputs ("IN2/LSP" and "IN3/LSN") can be set during drive commissioning through the "Inputs/Outputs" page in the "Setup" tree.

BasyMotion Studio II - P091.APN4.1 Drive spo	oecial inputs (Limit Switches)				- 🗆 X
Project Application Communication To	ools Control panels Utilities	Options Help			
	) 🔤 🛄 📖 👔	🖻 🟦 🖄 🗔 🗆 W	/ork offline Set all apps offline	2 🛓 🕨	O
Application tree • • • • • • • • • • • • • • • • • •	🗅 Inputs / outputs 🔞				< D B
LSP default behavior - [2] S Setup	Digital inputs Configure inputs as Sink (PNP)	• 0			Î
- 🕞 Homings	Input	Туре	Function	Active level	
- Functions	IN0	Sink (PNP)	General purpose input		
≻ 🕞 Interrupts 🚽	IN1	Sink (PNP)	General purpose input		
4	IN2/LSP	Sink (PNP)	Limit switch + *	High	Ŧ
Setup tree	IN3/LSN	Sink (PNP)	Limit switch +	High	*
	IN4	Sink (PNP)	General purpose input		
Mechanical configuration	IN5/Enable	Sink (PNP)	Enable *	Low	*
- 🛇 Motor	Digital outputs				
ິ່ເງີ Feedback		n – –			
↓↑ Inputs / outputs	Soundare sochus as Suuk (INPI				
* Control cottings	0.4	T	F	A - 1 1 1	

Figure 9 – Limit switches functionality selection

**<u>Remark</u>** When set as "general purpose inputs", the limit switches default behavior is disabled but the related interrupt routines are still available. They can be enabled and used as when the limit switches are used with their default function.

If the application requires temporarily disabling the limit switch functionality (e.g. during the homing procedure), the "LSACTIVE = 1;" TML instruction can be used. This instruction disables the default functionality of the limit switches and can be inserted using the "Inputs/Outputs" or "16-bit Integer" wizard dialogues.

This instruction can also be sent via the communication line or write as a free text.



Figure 10 – Limit switches default functionality disabling

The limit switches default functionality can be restored using the "LSACTIVE = 0;" TML instruction, which can be generated using "Inputs / outputs" or "16-bit Integer" wizard dialogues.

If need, it can also be sent via the communication line or write in the TML application as fee text.

# 3. Software limits configuring and usage

Software limits function similarly to hardware limit switches, restricting movement within set boundaries. This option can be enabled and configured through the "Protections and Limits" subtree item.

EasyMotion Studio II - P091.APN4.1 Drive Project Application Communication	ve spec Too	cial inputs (L	limit Swi	tches) Utilities	Opti	ons Help										-	- [		×
	D			•		1	😡 🗌 Work offline S	et all apps offlin	e Z 2	≛ ▶	0	¥	<u>K</u>	x				Ах	is off ⊓3
Application tree • # • P091.APN4.1 Drive special inputs	Δ	Protections	and lim	its 🔞													•	0	62
► Deactivate LS function - [2]		Drive op	eration	paramete	rs														
▷ □ LSP default behavior - [2]							Current li	nit 7.06	A	*									
- Software limits - [2]							Load speed li	nit 5820	rpm	*									
- S Setup							Motor sup	ply 1	V	Ŧ									
M Motion		Protecti	ons																
Mamony rattingr				Overcurrer	nt prote	ction 🗸	Trigger if motor curren	t > 8.472	Α	Ŧ		for mor	re than	(	0.01	s		Ŧ	
			Position	control erro	or prote	ction 🗸	Trigger if position erro	> 0.6	rot	*		for mor	re than		3	s		*	
Setup tree + #			Speed	control erro	or prote	ction 🖌	Trigger if speed erro	> 600	rpm	*		for mor	re than		3	s		Ŧ	
Quick setup Advanced setup				12	t prote	ction 🗌	I2t Overcurren	t is 7.766		*			for		15			*	
Mechanical configuration		E-starmal			_														
- 🛇 Motor		External	cnoppi	Is conne	r rted to	drive 🗔													
- 🖏 Feedback			ls conne	ted to BC90	100 mc	odule 🗆													
- ↓↑ Inputs / outputs																			
- 💸 Control settings		Limits		_										_		_		_	
- 🕄 External reference				So	ftware	limits 🔽	Negative li	nit -10	rot	*		Positiv	/e limit		10	rot		*	
<ul> <li>Application settings</li> </ul>				Position	range	limits 🔄													
- 🖉 Motion settings		Drive f	Data																
- 🚎 Fieldbus settings																			
→ Protections and limits																			
- / Controllers																			
- 🖄 Automatic tuning																			
<b>G D</b>																			
									SetupID 849	97BL   iPC	DS4808	MY-CAN	N S/N	V ON50	10   F,	/W F514L	Axis	2 0	line

Figure 8 - Software limit switches configuration

When a software limit is reached, the drive stops the motion, using a quick stop profile, and sets the correspondent bit in the DER register.

For example purpose, the drive was set to perform 20 rot (using a relative trapezoidal positon profile) while the software limits were set to -10 rot. .... +10 rot.



Figure 9 - Software limits default behavior

The above picture shows the motor position (red curve) that starts form 0 rot and increase according to the set profile. When reaching 10 rot, the motion is stopped as a result of reaching the positive software limit, and bit 6 in the DER register is set. The quick stop can also be observed in the second plot where the speed (orange line) drops to 0 IU, with the set quick stop deceleration value.

If the application requires enabling or disabling the software limits on the fly, it can be done through bit 12 in the ASR register:

- if **ASR.12 = 1**, the software limits are enabled
- if ASR.12 = 0, the software limits are disabled

The ASR register value can be changed through the "16-bit Integer" wizard, using the mask option.

16-bit Integer X	16-bit Integer X
Set 16-bit variable	Set 16-bit variable     ASR     T
○ With value / 16-bit variable / label v	O With value / 16-bit variable / label
Data memory contents located at address set in pointer variable     With Program     E2ROM     increment the pointer variable	Data     memory contents located at address set in pointer variable     E2ROM     increment the pointer variable
O With Oracle Part of 32-bit variable ▼	○ With ○ Low ○ High Part of 32-bit variable ▼
O With inverse (-) of variable	O With inverse (-) of variable - Enable the software limits
Using masks AND mask EFFF h OR mask 0000 h	Using masks AND mask FFFF h OR mask 1000 h
Data     Data     With checksum of data located in     Program     memory between adresses	Data     Data     With checksum of data located in     Program     memory between adresses     h     E2ROM
Data memory contents located at address set in pointer variable     The set of the program with value / variable     E2ROM     increment the pointer variable	Data memory contents located at address set in pointer variable     Set Program     With value / variable     E2ROM     increment the pointer variable
OK Cancel Help	OK Cancel Help

Figure 10 – On the fly software limits disabling/enabling

**<u>Remark</u>** The ASR register bits description is present in the EasyMotion Studio II Help topics.

If the software limits values should also be set on the fly, this can be done through the following TML parameters:

- SWNEGLS the negative position limit value, in internal units
- SWPOSLS the positive position limit value, in internal units

The software limits related parameter values can be set through the "32-bit Long or Fixed" wizard dialogue or via the communication line.

Application Communication Tools Control par	els Utilities Options Help
	🕴 🛄 🔳 🚊 🚊 🗔 🗋 Work offline Set all apps offline 🥕 🛓 🕨 💽 🖉 🕺 🗶 📖 🥤
plication tree 🔹 🔹	
P091.APN4.1 Drive special inputs (Li	rofiles Search $\mathcal{D} \times \mathcal{C} \to 0$
> Deactivate LS function - [2] ∧ S Curve	SWPOSLS = 20000;
LSP default behavior - [2]     M PT	
Software limits - [2]	32-bit Long or Fixed X
S Setup ↓ External	Set 32-bit variable SWPOSLS
M Motion	aring
- IP Homings In Electronic Ca	mming
- IF Functions Determined Motor Comm	ands Data memory contents located at address set in pointer variable
- 🕼 Interrupts 🌮 Position Trig	ers O With Program
CAM Tables	E2KOM
Memory settings	O With inverse (-) of variable
T/⁻ Test	With 16-bit value of variable
!⊶ Events	
ិរ Jumps and F	Inction Calls
≓] Inputs / out	uts High
16 16-bit Integr	r memory contents located at address set in pointer variable
32 32-bit Long	or Fixed
* Arithmetic	© E2ROM
fo Data Transfe	Between Axes
Send Data Te	Host OK Cancel Help
MISC Miscellaneou	5
	ings

**<u>Remark</u>** The position internal units are described in the "Internal units and Scaling factors" help topic.