|  |  |
| --- | --- |
| **AD7147 slider position calculation**  **Version v1** | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **Project Number** |  |
| **Project Name** |  |
|  |  |
| **Creation Author** | BAERT Wouter |
| **Creation Date** |  |
|  |  |
|  |  |
| **Document Number** |  |
| **File Location** |  |
|  |  |
|  |  |
|  |  |
| **Document Status** | **Draft** / Approved |
| **Approved By** |  |
| **Approved On** |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | |
| **Last Saved By** | PAUWELS Brecht |
| **Last Saved On** | 04/11/2014 16:44 |
|  |  |
|  |  |
|  |  |
|  |  |
| **PsiControl NV**  Steverlyncklaan 15  B-8900 Ieper (Belgium)  [www.psicontrol.com](http://www.psicontrol.com) Member of the PICANOL group | |

**Revision history**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **By** | **Description** |
| 1 | 27/01/16 | WBRT | Initial version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Input documents**

|  |  |
| --- | --- |
| **Document name** | **Version** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Table of Contents**

[1 Heading 1 3](#_Toc402882787)

[1.1 Heading 1.1 3](#_Toc402882788)

[1.1.1 Heading 1.1.1 3](#_Toc402882789)

[2 Heading 2 4](#_Toc402882790)

[2.1 Heading 2.1 4](#_Toc402882791)

**Table of Figures**

**No table of figures entries found.**

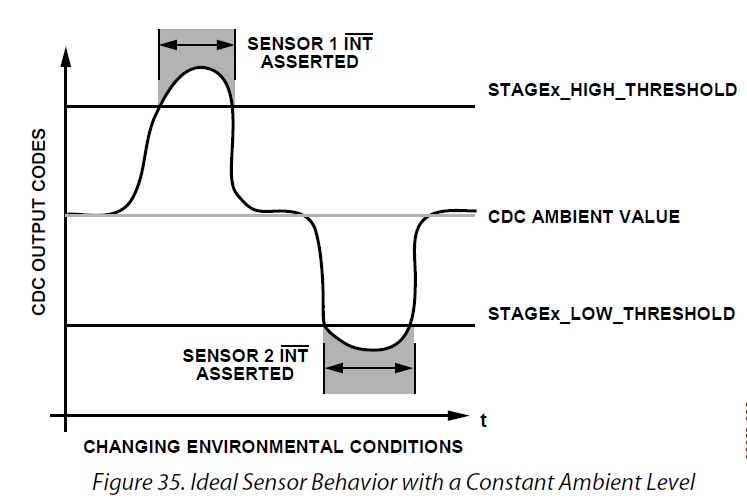
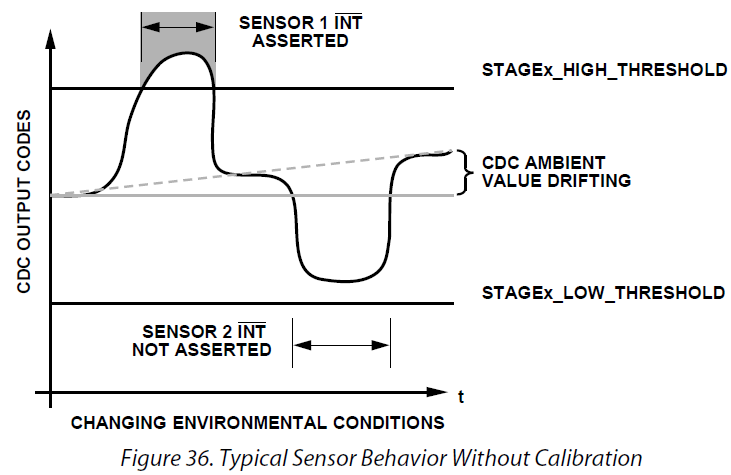
**Table of Tables**

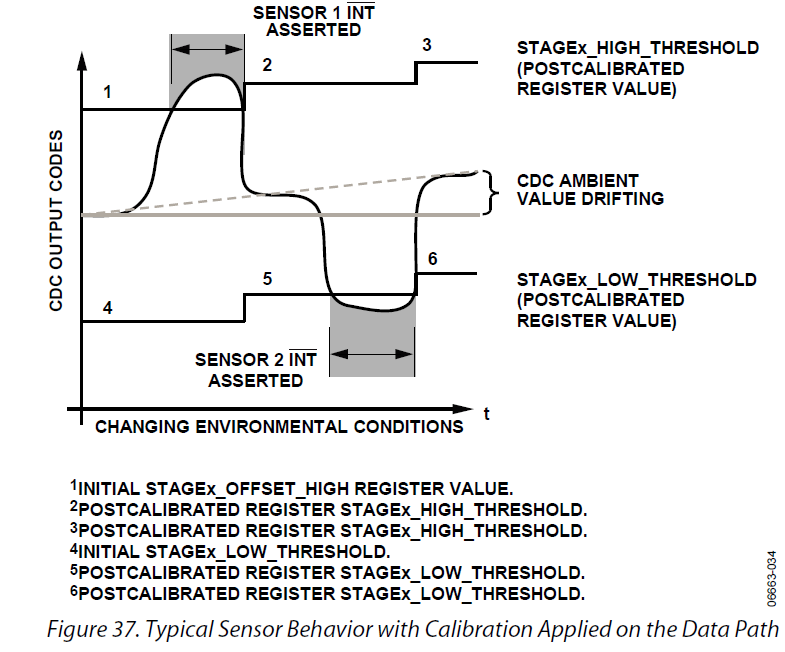
**No table of figures entries found.**

# AD7147 Slider values and theoretical operation

De AD7147 berekent constant nieuwe “ambient” values die een gemiddelde waarde weergeven wanneer er geen touch is. Op basis van die waarden en twee treshold levels wordt er al dan niet een touch geregistreerd. Zowel de ambient als threshold waarden worden constant herberekend door de chip.

Er worden maximaal 12 “stages” sequentieel berekend door de chip. Aan elke stage kunnen er 1 of meerdere sensor input gehangen worden, die de stage-output waarde (=CDC value) positief of negatief beïnvloeden. Voor een slider wordt er telkens 1 sensor input aan 1 stage positief gekoppeld. Op deze manier kan er telkens gescand worden op een HIGH\_TRESHOLD interrupt om te zien of er een touch is geweest.





In het swipestat algoritme wordt eerst het HIGH\_INTERRUPT register uitgelezen. Als dit een interrupt geeft op 1 van de stages van de slider (register 0x0009). Daarna worden alle stage CDC values en stage ambient values uitgelezen en van elkaar afgetrokken (met een minimum van 0 aangezien er theoretisch geen negatieve waardes kunnen zijn).

Dit gebeurt met volgende methode:

*static U16 Get\_Stage\_Diff\_Values(U16 \* diff\_values\_ptr);*

De *diff\_values\_ptr* is de array die gebruikt wordt om de verschilwaarden door te geven. De return waarde is de positie in de array die de maximale waarde heeft.

HIGH\_INT STATUS = 0

STAGEx\_CDC – STAGEx\_AMBIENT

(x is number of sensor inputs)

NO

YES

Return previous position and touch = FALSE

Calculate and return position on slider and touch = TRUE

GetValue()

Schematic overview of position calculation overview on AD7147

# Position calculation

Wanneer de array met gemeten CDC-values en de maximum positie in deze array gevonden zijn, wordt de exacte positiebepaling gedaan.

De posities worden verdeeld tussen 0x0000 en 0xFFFF. In geval van 8 slider posities ligt elk midden van een vlak op positie **[(i\*0x2000)+0x1000]**, i=0..7.

## Geval 1: de maximale waarde bevindt zich NIET aan 1 van de uiteindes

* 2 samples uit de array worden gebruik: degene die zich net naast de maximale waarde bevinden (**Max\_pos** is array positie met maximale waarde)
* Eerst wordt uitgemaakt aan welke kant van de positie met maximale waar de touch zich bevindt. Ten tweede wordt aan de hand van de verhouding tussen de twee aanliggende samples berekend hoe ver van de centrale positie de touch zich bevindt.
* Hiervoor wordt de **relative\_position** berekend, dit is de offset ten opzicht van de centrale waarde.

0x3000

Example:

0x1000

0x0000

….

0x2000

0x5000

0x4000

0x6000

0x10000

….

Positie zou dus **tussen 0x3000 en 0x4000** moeten liggen

MAX\_POS

  = **1**

MAX\_POS-1

= **0** in het vb

MAX\_POS+1

= **2**

Mogelijke touch area

De berekening van de relatieve positie tussen de twee gebeurt als volgt (wanneer val[max\_pos+1] groter is dan val[max\_pos-1]):

*Relative\_position = ( ( value[max\_pos+1] - values [max\_pos-1] ) \* 0x1000 ) / value [max\_pos+1] )*

De vermenigvuldiging \*0x1000 gebeurt hier in de noemer al om te voorkomen dat de positive altijd 0 geeft. Hierdoor ligt de relative\_postion tussen 0x0 en 0x1000, de offeset van de maximale positie.

De exacte positie wordt als volgt vastgesteld (altijd uitgaand van de max\_pos locatie):

S*lider\_pos = ( max\_pos \* 0x2000 ) + 0x1000 + relative\_position*

*= ( 1 \* 0x2000 ) + 0x1000 + relative\_postion*

*= 0x3000 + relative postion*

* Slider\_pos ligt hier dus tussen 0x3000 en 0x4000, afhankelijk van de grootte van de sliders naast sensor 1 (die de grootste is)

## Geval 2: de maximale waarde bevindt zich WEL aan 1 van de uiteindes

In dit geval wordt gekeken naar de waardes op de maximale positie en ernaast. De mogelijke touch area is hier dubbel zo groot aangezien de uiterste kant anders niet bereikt kan worden.

0x3000

Example:

0x1000

0x0000

….

0x2000

0x4000

Positie zou dus **tussen 0x0000 en 0x2000** moeten liggen

MAX\_POS

  = **0**

MAX\_POS+1

= **1**

Mogelijke touch area

Gelijkaardig wordt de relatieve positie bepaald door de verhouding van de twee samples, de waarde 0x2600 is groter dan wat theoretisch 0x2000 moet zijn, omdat anders de uiterste slider positie anders nooit kan bereikt worden.

*Relative\_position = ( ( value[0] - values [1] ) \** ***0x2600*** *) / value [0] )*

S*lider\_pos = 0x2000 - relative\_position*

Daarna wordt ervoor gezorgd dan de slider positie niet onder 0x0 of boven 0xFFFF kan gaan.