

## DEVELOP.INI

### - Beschreibung der Sektion [DeviceX], zuständig für die Detektoren

Detektoren werden im ini-File als Devices geführt, ebenso im weiteren Verlauf. Die im .ini-File für ein Device festzulegenden Eigenschaften und Parameter werden mit dem Sektionsnamen (in eckige Klammern gefasst) DeviceX eingeleitet. Dabei steht X für eine ganze Zahl. Die Anzahl der maximal definierbaren Devices ist mit const int nMaxDeviceAllowed in C\_Layer.cpp gegenwärtig auf 3 begrenzt, d.h. X darf maximal 2 sein. Alles was größer ist als nMaxDeviceAllowed wird stillschweigend ignoriert.

Da die einlesende Funktion GetPrivateString nicht case-sensitiv ist, ist die Groß/Kleinschreibung von Sektionsnamen und Keys irrelevant. Folgende Keys (Schlüsselworte unter einem Sektionsnamen) sind möglich (und teilweise auch erforderlich):

#### Type

<b>Beschreibung</b>	Mit diesem Key wird der Typ des Device spezifiziert, entsprechend des Typs werden unterschiedliche Parameter (Keys) benötigt und entsprechende Konstruktoren aufgerufen.
<b>mögliche Werte</b>	Generic, Stoe-PSD, Radicon, Braun-PSD, PSD, Encoder, Matrox, Test
<b>Defaultwert</b>	Generic
<b>Detektortyp</b>	alle
<b>Hinweis</b>	Der Typ Matrox ist momentan durch die Anweisung #ifdef CCD_Extension deaktiviert. Der Typ Test ist in dem Sinne nicht als Stringangabe erforderlich, vielmehr führt jeder String der nicht zu den möglichen Werten gehört (momentan also auch Matrox) zu einem Device vom Typ Test.

#### Name

<b>Beschreibung</b>	Hiermit wird dem Device ein Name zu geordnet. Dieser Name wird fortan bei der Arbeit mit dem Device angezeigt
<b>mögliche Werte</b>	Es können die üblichen alphanumerischen Zeichen verwendet werden, insgesamt maximal 80. Von der Verwendung von Semikola ist abzuraten, weil diese als Kommentareinleitung von GetPrivateString interpretiert werden, und nachfolgende Zeichen somit ignoriert würden.
<b>Defaultwert</b>	Counter
<b>Detektortyp</b>	alle

## Sound

<b>Beschreibung</b>	Damit wird die akustische Ausgabe der Messung ein/ausgeschaltet. Aufgrund der Quelltextanalyse gehen wir davon aus, daß ein Setzen dieses Flags einen Hardwarezugriff zur Folge hat, ein Speaker wird (de)aktiviert.
<b>mögliche Werte</b>	0- aus; 1- ein
<b>Defaultwert</b>	1 (außer bei TRadicon : 0)
<b>Detektortyp</b>	alle

## Debug

<b>Beschreibung</b>	Mit diesem Flag wird die Ausgabe von Debugginginformationen ein- bzw. ausgeschaltet. Es wird auf die Variable <i>bDebug</i> geschrieben, diese wird im weiteren Programmverlauf an bestimmten Stellen abgefragt. Wenn sie gesetzt ist, wird die Funktion <i>SetInfo()</i> mit einem String ausgeführt, welcher in die Statuszeile des Steuerprogramms geschrieben wird. Dieses Flag wird nur in Verbindung mit den Detektoren verwendet.
<b>mögliche Werte</b>	0- aus; 1- ein
<b>Defaultwert</b>	0
<b>Detektortyp</b>	alle
<b>Hinweis</b>	Auch dieses Flag ist unserer Meinung nach für alle Detektoren (abgesehen von Testdetektoren) gedacht worden. Es wird in der Methode <i>Initialize()</i> eingelesen und gesetzt. Jedoch hat der Entwickler im Konstruktor <i>TBraun_PSD</i> dieses Vorgehen wohl vergessen und eine neue Variable samt Key eingeführt: <i>DebugInfo</i> . Auf diese wird im weiteren Verlauf jedoch nie wieder zugegriffen, statt dessen wird wie sonst üblich auch in <i>Braun_PSD</i> Methoden auf <i>bDebug</i> zugegriffen. (siehe Key <i>DebugInfo</i> )

## ExposureTime

<b>Beschreibung</b>	Hier wird die Länge eines Meßintervalls festgelegt. Der Wert wird in der Variablen <i>fExposureTime</i> gespeichert.
<b>mögliche Werte</b>	0.1- 500.0
<b>Defaultwert</b>	4.0
<b>Detektortyp</b>	alle
<b>Hinweis</b>	Für <i>fExposureTime</i> existieren die Methoden <i>TDevice::GetExposureValues</i> und <i>TDevice::SetExposureValues</i> zum Lesen und Setzen, jedoch wird mindestens dreimal aus der Klasse <i>TCommonDevParam</i> direkt auf das Attribut zugegriffen. Der Typ PSD dient unserer Meinung nach dazu einen PSD Testzähler zu erzeugen.

## ExposureCounts

<b>Beschreibung</b>	Mit diesem Key wird die maximale Anzahl der Impulse pro Meßintervall festgelegt. Der Wert wird in der Variablen <i>dwExposureCounts</i> gespeichert.
<b>mögliche Werte</b>	1 - 300000
<b>Defaultwert</b>	10000
<b>Detektortyp</b>	alle
<b>Hinweis</b>	Für <i>dwExposureCounts</i> existieren die Methoden <i>TDevice::GetExposureValues</i> und <i>TDevice::SetExposureValues</i> zum Lesen und Setzen, jedoch wird mindestens dreimal aus der Klasse <i>TCommonDevParam</i> direkt auf das Attribut zugegriffen

## OverflowIntensity

<b>Beschreibung</b>	Der hier angegebene Wert gibt an, welches der größte mögliche Meßwert ist.
<b>mögliche Werte</b>	0 - ?
<b>Defaultwert</b>	50000.0
<b>Detektortyp</b>	PSD

## BaseAddr

<b>Beschreibung</b>	Hier wird die I/O-Adresse (hexadezimal) für die Kommunikation mit der Hardware angegeben. Der Wert wird im Attribut <i>nBaseAddr</i> abgelegt.
<b>mögliche Werte</b>	0x100 – 0x3F8 in 4h Schritten (Quelle: BraunPSD Betriebsanleitung)
<b>Defaultwert</b>	0x300
<b>Detektortyp</b>	PSD
<b>Hinweis</b>	Das Programm führt an keiner Stelle eine Überprüfung des Inhaltes von <i>nBaseAddr</i> durch.

## SignalGrowUp

<b>Beschreibung</b>	Dieses Flag gibt an, ob auch vor Messungsende Zwischendaten übernommen werden sollen.
<b>mögliche Werte</b>	0- nein; 1- ja
<b>Defaultwert</b>	1
<b>Detektortyp</b>	PSD

## HVRegelung

<b>Beschreibung</b>	Mit diesem Flag gibt man an, ob die Hochspannung am Gerät gesteuert werden soll.
<b>mögliche Werte</b>	0- nein; 1- ja
<b>Defaultwert</b>	0
<b>Detektortyp</b>	PSD

## ReadLeftFirst

<b>Beschreibung</b>	Mit diesem Flag legt man fest, daß beim Auslesen der Geräte von links begonnen werden soll.
<b>mögliche Werte</b>	0- nein; 1- ja
<b>Defaultwert</b>	1
<b>Detektortyp</b>	PSD

## AngleStep

<b>Beschreibung</b>	Dieser Key gibt den Winkel an, der von einem Kanal abgedeckt werden soll.
<b>mögliche Werte</b>	0-?
<b>Defaultwert</b>	1.0
<b>Detektortyp</b>	PSD

## Unit

<b>Beschreibung</b>	Mit diesem Key wird die Einheit der Winkelangabe festgelegt.
<b>mögliche Werte</b>	Grad, Sekunden, Minuten, Minuts
<b>Defaultwert</b>	Sekunden
<b>Detektortyp</b>	PSD

## AddedChannels

<b>Beschreibung</b>	Hiermit wird die Anzahl der zu einem Meßwert zusammenzufassenden Kanäle angegeben.
<b>mögliche Werte</b>	1 - ?
<b>Defaultwert</b>	4
<b>Detektortyp</b>	PSD

## FirstChannel

<b>Beschreibung</b>	Dieser Key gibt den ersten zu benutzenden Kanal an.
<b>mögliche Werte</b>	0- 4095
<b>Defaultwert</b>	0
<b>Detektortyp</b>	PSD
<b>Hinweis</b>	Wenn der Wert von <i>FirstChannel</i> größer oder gleich <i>LastChannel</i> ist, wird für <i>FirstChannel</i> =0 und für <i>LastChannel</i> =4095 gesetzt.

## LastChannel

<b>Beschreibung</b>	Hier wird der letzte zu benutzende Kanal angegeben.
<b>mögliche Werte</b>	0- 4095
<b>Defaultwert</b>	4095
<b>Detektortyp</b>	PSD
<b>Hinweis</b>	Wenn der Wert von <i>FirstChannel</i> größer oder gleich <i>LastChannel</i> ist wird für <i>FirstChannel</i> =0 und für <i>LastChannel</i> =4095 gesetzt.

## IOAddr

<b>Beschreibung</b>	Hier wird die I/O-Adresse für die Hardwarekommunikation mit dem Radicon-Device angegeben, sie wird in <i>Rdd</i> gespeichert.
<b>mögliche Werte</b>	0x100 – 0x160 in 0x10 Schritten, 0x180 – 0x1E0 in 0x10 Schritten, 0x300, 0x310, 0x330 – 0x360 in 0x10 Schritten, 0x390, 0x3E0 (Quelle: SCSCS Operator's Manual)
<b>Defaultwert</b>	0x100
<b>Detektortyp</b>	Radicon
<b>Hinweis</b>	Das Programm führt an keiner Stelle eine Überprüfung des Inhaltes von <i>Rdd</i> durch. Des weiteren wäre anzumerken, daß es unnötig ist, zwei Bezeichnungen für den gleichen Sachverhalt zu vergeben. Es handelt sich zwar bei <i>IOAddr</i> und <i>BaseAddr</i> nicht um den gleichen Inhalt ( <i>IOAddr</i> ist für Radicon, <i>BaseAddr</i> für PSD), da sie aber nie in der gleichen Sektion zugleich vorkommen und sowieso getrennte Methoden zum Einlesen dieser Werte existieren, wäre es sinnvoll und ein weiterer Schritt in Richtung Benutzerfreundlichkeit, sich auf eine Bezeichnung zu einigen.

## UpperThresh

<b>Beschreibung</b>	Dieser Wert repräsentiert die obere Schranke für die Hochspannung, mit der das Gerät betrieben werden soll.
<b>mögliche Werte</b>	1 – 1023
<b>Defaultwert</b>	950
<b>Detektortyp</b>	Radicon

## LowerThresh

<b>Beschreibung</b>	Dieser Wert repräsentiert die untere Schranke für die Hochspannung, mit der das Gerät betrieben werden soll.
<b>mögliche Werte</b>	1 – 1023
<b>Defaultwert</b>	150
<b>Detektortyp</b>	Radicon

## HighVoltage

<b>Beschreibung</b>	Dieser Key gibt den Wert der Hochspannung bei Initialisierung an.
<b>mögliche Werte</b>	1 - 900
<b>Defaultwert</b>	640
<b>Detektortyp</b>	Radicon

## EnergyScale

<b>Beschreibung</b>	Dieser Key bestimmt die Energieskalierung.
<b>mögliche Werte</b>	0,1,2,3
<b>Defaultwert</b>	2
<b>Detektortyp</b>	Braun

## AbbruchMitShutter

<b>Beschreibung</b>	Dieses Flag setzt die Option: Messung abbrechen mit Shutter.
<b>mögliche Werte</b>	0- nein; 1- ja
<b>Defaultwert</b>	0
<b>Detektortyp</b>	Braun

## PositionScale

<b>Beschreibung</b>	Dieser Key bestimmt die Positionsskalierung.
<b>mögliche Werte</b>	0,1,2,3
<b>Defaultwert</b>	2
<b>Detektortyp</b>	Braun

## EnergyHigh

<b>Beschreibung</b>	Dieser Key legt die obere Grenze für das Energiefenster fest.
<b>mögliche Werte</b>	0(?) bis $(0xFFF / (EnergyScale+1))$
<b>Defaultwert</b>	870
<b>Detektortyp</b>	Braun

## EnergyLow

<b>Beschreibung</b>	Hiermit wird die untere Grenze für das Energiefenster festgelegt.
<b>mögliche Werte</b>	0(?) bis $(0xFFF / (EnergyScale+1))$
<b>Defaultwert</b>	526
<b>Detektortyp</b>	Braun

## MuxTimeDet1

<b>Beschreibung</b>	Dieser Key dient der Einstellung der Multiplexerzeit für den Detektor 1. Unklar ist was genau damit gemeint ist. Wir konnten diese Information lediglich aus einem String ziehen.
<b>mögliche Werte</b>	0 – 61439(0xEFFF)
<b>Defaultwert</b>	40
<b>Detektortyp</b>	Braun

## Ratometer

<b>Beschreibung</b>	Über dieses Flag wird das externe Ratometer selektiert.
<b>mögliche Werte</b>	0- aus; 1- ein
<b>Defaultwert</b>	0
<b>Detektortyp</b>	Braun

## RealLifeTime

<b>Beschreibung</b>	Mit diesem Flag kann zwischen Real- und Lifetimeberechnung umgeschaltet werden.
<b>mögliche Werte</b>	0- Realtime, 1-Lifetime
<b>Defaultwert</b>	0
<b>Detektortyp</b>	Braun

## DeathTime

<b>Beschreibung</b>	Dieser Wert gibt den notwendigen Mindestimpulsabstand an.
<b>mögliche Werte</b>	0- 99
<b>Defaultwert</b>	10
<b>Detektortyp</b>	Braun

## DelayFast

<b>Beschreibung</b>	Die Bedeutung dieses Key konnten wir nicht endgültig klären, es handelt sich jedoch vermutlich um einen Verzögerungswert, welcher bei der Kommunikation mit der Hardware einen Timeout-Wert darstellt.
<b>mögliche Werte</b>	Ebenso fanden wir keine Anhaltspunkte für die möglichen Werte.
<b>Defaultwert</b>	2
<b>Detektortyp</b>	Braun

## DelaySlow

<b>Beschreibung</b>	Die Bedeutung dieses Key konnten wir ebenfalls nicht endgültig klären, es handelt sich jedoch vermutlich auch hier um einen Verzögerungswert, welcher bei der Kommunikation mit der Hardware einen Timeout-Wert darstellt.
<b>mögliche Werte</b>	Ebenso fanden wir keine Anhaltspunkte über die möglichen Werte.
<b>Defaultwert</b>	4
<b>Detektortyp</b>	Braun

## DebugInfo

<b>Beschreibung</b>	Dieser Key soll vermutlich die Ausgabe von Debugging-Informationen aktivieren, jedoch wird er nur einmal gesetzt und im weiteren Verlauf nie wieder verwendet, was vermutlich damit zu erklären ist, daß der bereits weiter oben erläuterte Key <i>Debug</i> diese Aufgabe übernimmt. Vermutlich stellt <i>DebugInfo</i> einen Moment der Unklarheit des Entwicklers über die bereits implementierten Attribute dar, mit anderen Worten: <i>DebugInfo</i> ist überflüssig und kann bedenkenlos weggelassen werden, da auch die Routinen der PSD-Klasse mit dem Attribut <i>bDebug</i> (entsprechende Variable für den Key <i>Debug</i> ) arbeiten.
<b>mögliche Werte</b>	0- aus; 1- ja
<b>Defaultwert</b>	0
<b>Detektortyp</b>	Braun
<b>Hinweis</b>	siehe Information weiter oben über Key <i>Debug</i>

## IOAddr

<b>Beschreibung</b>	Dieser Key dient ebenfalls der Festlegung der I/O-Kommunikationsadresse. Er wird von <i>TAm9513a::LoockUp</i> eingelesen.
<b>mögliche Werte</b>	?
<b>Defaultwert</b>	0x230
<b>Detektortyp</b>	AM9513 (Generic)
<b>Hinweis</b>	Siehe Hinweise oben unter BaseAddr und IOAddr

## TimeCorrection

<b>Beschreibung</b>	Die Bedeutung dieses Key konnte von uns nicht eindeutig geklärt werden. Es bleibt uns lediglich vom Namen auf die Bedeutung zu schließen, was in diesem Fall auf die Funktion einer Zeitkorrekturkonstante deutet.
<b>mögliche Werte</b>	Ebenfalls gab es keine Anhaltspunkte für die hier möglichen Werte.
<b>Defaultwert</b>	1.0
<b>Detektortyp</b>	AM9513 (Generic)