APDCAM 1G Windows API   
Change Log  
Last modified: 21.04.2017.

**2014.12.11.**

***V1.2 -> V1.3 changes***

* ADC lib unchanged
* APD lib function added to APDLib.h
  + APDCAM\_GetHW
  + APDCAM\_SetOverLoad
  + APDCAM\_GetOverLoad
* common\WinClasses.cpp
  + CwinWaitForEvents function changed
* libcommon\DataEvaluation class
  + Reset function was added
* libcommon\HighlevelFunction.cpp
  + APDCAM\_GetBuffers function modified
  + APDCAM\_Arm function modified
  + APDCAM\_GetHW added
  + APDCAM\_SetOverLoad added
  + APDCAM\_GetOverLoad added
* libcommon\TypeDefs.h
  + ADT\_OVERLOADINFO structure was added

**2014.12.30**

***V1.3 -> V1.3.1 changes***

* APDLib.h changed
  + APDCAM\_SetHWTriggerDelay deleted
  + APDCAM\_SWTrigger deleted
  + APDCAM\_HVMonitor deleted
  + APDCAM\_GetHW changed to GetHV
* HighLevelFunctions.cpp modified
  + ADPCAM\_SetHWTriggerDelay deleted
  + ADPCAM\_SWTrigger deleted
  + APDCAM\_HVMonitor deleted
  + APDCAM\_Calibrate modified
    - Replaced APDCAM\_HVMonitor with ADPCAM\_GetHV
  + APDCAM\_ReadPDI modified
    - Error handling changed
  + APDCAM\_WritePDI modified
    - Error handling changed
  + APDCAM\_GetInfo modified
    - Replaced HVMonitor with GetHV
  + APDCAM\_GetHW changed to APDCAM\_GetHV
* TypeDefs.h changed
  + ADT\_SYSTEM\_STATUS changed
    - Firmaware -> Firmware

**2015.01.05**

***V1.3 -> V1.3.1 changes***

* APDLib.h changed
  + APDCAM\_Open modified
    - openMode parameter added, the user can select normal or force open modes(normal default)
    - ADPCAM\_SetRingBufferSize and APDCAM\_GetRingBufferSize declared
* InternalFunction class modified
  + GetControlBoardVersion function defined
  + Theres were two SetRingBufferSize function, one was change to GetRingBufferSize
* HighLevelFunctions.cpp changed
  + ADPCAM\_Open function changed
    - openMode parameter added
    - error handling done according to different openModes
  + APDCAM\_SetTiming modified
    - clkSorce -> clkSource
    - external clock select implemented
  + APDCAM\_Sampling
    - error handling changed sampleDiv can’t be negative,
    - sampleSrc implemented
  + APDCAM\_Filter changed
    - Filter enabling/disabling implemented
    - if all parameters are zero, filter disables
  + APDCAM\_Allocate changed
    - if primary buffer size less than 10 or bigger than 100 the function will returns an error code
  + APDCAM\_SetRingBufferSize and APDCAM\_GetRingBufferSize functioan implemented
* APDTest.cpp changed
  + ForceOpen function defined
  + APDTest application now recognize ForceOpen command

**2015.01.06**

***V1.3 -> V1.3.1 changes***

* APDLib.h changed
  + Get/SetShutterMode name changed to APDCAM\_Get/SetShutterMode
* HighLevelFunctions.cpp changed
  + APDCAM\_Trigger function changed
    - HW trigger delay wired
  + APDCAM\_GetInfo upgraded
    - Added shutter state reading
    - Added caliblight status reading
  + APDCAM\_SetTiming
    - clear bit bug fix at clk settings
    - check external PLL is locked or not
  + APDCAM\_Sampling
    - clear bit bug fix at sampleSrc settings
  + APDCAM\_Filter
    - clear bit bug fix at filter enable/disable
  + APDCAM\_Caliblight
    - boundary check implemented

**2015.01.10.**

***V1.3.1 changes***

* APDLib.h changed
  + ADPCAM\_CalculateFilterParams defined
  + APDCAM\_SelfTest defined
* HighLevelFunctions.cpp changed
  + APDCAM\_CalculateFilterParams implemented
  + APDCAM\_SelfTest implemented

**2015.01.12.**

***V1.3.1 changes***

* HighLevelFunctions.cpp changed
  + APDCAM\_SelfTest implemented

**2015.01.13.**

***V1.3.1 changes***

* APDLib.h changed
  + ADPCAM\_Trigger changed
    - preTriggerSampleCount parameter added
* HighLevelFunctions.cpp changed
  + APDCAM\_Trigger changed
    - preTriggerSampleCount added
  + APDCAM\_SelfTest changed
    - implementing ADC and Control card check
    - implementing factory table check
    - implementing ADC PLL lock check
    - implementing temperature check
* InternalFunctions class changed
  + defined PC\_REG\_ERROR\_CODE register
  + declared GetControlBoardError function
  + defined ADC\_REG\_ERROR\_CODE register
  + defined PC\_REG\_HV1MAX register
  + declared GetADCBoardError function
  + declared GetHV1Max function
  + declared GetHV2Max function

**2015.10.08.**

***V1.3.2 changes***

* LowLevelFunctions class changed
  + Implemented the same UPD packet lost corrention as on linux version

**2015.10.21.**

* InternalFunction class changed
  + Register address definitions moved to header file
  + Unnecessary Sleeps were deleted due to UDP packet lost correction

**2015.11.02.**

* APDLib.h
  + Removed unnecessary comments
* LowLevelFunctions.cpp
  + bug fix for more than 1 byte register reading and writing
* TypeDefs.h
  + Removed unnecessary „typedef” statements to clear warnings on compile

**2015.12.30**

* Continue testing APDLib, small changes
  + ARM –> Arm

**2016.01.06**

* Changed APDCAM\_Trigger
  + Implemented parameters boundary check for trigger, mode and edge parameters

**2016.01.07**

* + APDCAM\_CalibLight was changed to APDCAM\_SetCalibLight
  + APDCAM\_Filter was changed to APDCAM\_SetFilter
  + APDCAM\_Gain was changed to APDCAM\_SetHV
  + APDCAM\_GetInfo was changed to APDCAM\_GetStatus

**2016.01.19**

* + APDCAM\_SetAnalogPower defined
  + APDCAM\_GetAnalogPower defined
  + APDCAM\_GetInfo defined
  + APDInfo structure defined in TypeDefs.h
  + GetInfo defined in InternalFunctions class

**2016.01.21**

* + defined PC\_REG\_IRQ\_POWER\_PID\_ENABLE register in InternalFunctions.h
  + defined PC\_REG\_IRQ\_ENABLE\_HV register in InternalFunctions.h
  + defined PC\_REG\_PELT\_DAC\_REG register in InternalFunctions.h
  + defined PC\_REG\_HV\_IRQ\_LEVEL register in InternalFunctions.h
  + defined PC\_REG\_HV\_FAILURE register in InternalFunctions.h
  + defined PC\_REG\_CALIB\_LIGHT\_UPDATE\_TIME register in InternalFunctions.h
  + defined PC\_REG\_IRQ\_STATUS register in InternalFunctions.h
  + defined PC\_REG\_RESET\_FACTORY register in InternalFunctions.h
  + defined and implemented SetAnalogPower and GetAnalogPower functions in InternalFunction class
  + implemented GetPeltierOutPutVoltage
  + implemented SetPeltierIndirectControl
  + implemented GetPeltierIndirectControl
  + implemented SetTempInterruptEnable
  + implemented GetTempInterruptEnable
  + implemented SetDisablePIDControl
  + implemented GetDisablePIDControl
  + defined APDStatus structure in TypeDefs.h
  + implemented GetControlBoardFWVersion
  + implemented GetInfo

**2016.01.22**

* + implemented Set/GetFan1Speed
  + implemented Set/GetFan2Speed
  + implemented Set/GetFan3Speed
  + implemented GetStatus
  + defined Status1 structure in TypeDefs.h
  + defined Status2 structure in TypeDefs.h
  + defined Control structure in TypeDefs.h

**2016.01.23**

* + defined APDCAM\_GetConfig in APDLib.h
  + defined APDCAMStr structure in TypeDefs.h
  + implemented GetStatus
  + modified APDCAM\_GetStatus
  + defined APDPeltierControl structure
  + implemented APDCAM\_SetPeltierControl
  + implemented APDCAM\_GetPeltierControl
  + implemented Set/GetPGain
  + implemented Set/GetIGain
  + implemented Set/GetDGain

**2016.01.27**

* + added sleep for ARM after settings sampleCount and before status readings.

**2016.01.28**

* + there was a bug in GetStatus, it always return false
  + In GetHV voltage calibration factors were declared to 0.12 as Factory table is not availabel in the current camera
  + APDCAM\_GetHV error handling was wrong in APDCAM\_GetConfig so it always returned false
  + Added 20ms sleep between two register reading in GetExtClkPLL functions in InternalFunctions.cpp
  + Defined new members for APDStr in TypeDefs.h
    - calibLightOut
    - fanSpeed[3]
    - shutterState
    - shutterMode
  + Added them to read in APDCAM\_GetConfig

**2016.01.29**

* + Defined new members for APDStr in TypeDefs.h
    - peltierIndirectControl
    - pGain,iGain,dGain
    - disablePIDControl
  + Added them to read in APDCAM\_GetConfig
  + There was a bug in GetDisablePIDControl function in InternalFunctions.cpp which crashed the API
  + Implemented GetEnableHV function in InternalFunctions class
  + added reading HV enable bit in APDCAM\_GetConfig
  + There was a bug in GetEnableHV
  + There was a bug in GetChannels, it returned chennel mask wrong

**2016.02.01**

* + Defined constant values for outputHVCalibs because of the missing factory calibration table of the current camera.

**2016.02.02**

* + Implemented APDCAM\_SetFansSpeed function
  + Added ADC offset reading to APDCAM\_GetConfig
  + Defined offsets member in APDStr structure

**2016.02.03**

* + Changed APDCAM\_GetStatus error handling, it always return ADT\_OK which was wrong

**2016.03.05**

* Temporary fix of Continuity conunter error issue

**2016.04.06**

* DataEvaluation class
  + ProcessFrame(\*pFrame, packetNo) function
    - Added else statement when calling FindPreamb
    - Added test for m\_StreamNo variable
  + ThreadHandler
    - m\_StreamNo was initialized to zero
  + Added stream data dump to file debug functionality

**2016.04.08**

* HighLevelFunctions class
  + Arm
    - Multiple bugs were found, mistiping of different stream settings, etc

**2016.04.14**

* APDCAM\_Allocate primary buffer size parameter changed from default 10 to 100.

**2016.04.19**

* TypeDefs.h
  + defined detectorReferenceTemp field for apdStr struct
* APDCAM\_GetConfig
  + added detector reference temp reading
* InternalFunctions
  + Added Set/GetReferenceTemp function

**2016.04.19**

* APDLib.h
  + Defined APDCAM\_SetDetectorReferenceTemp
  + Defined APDCAM\_GetDetectorReferenceTemp
* HighLevelFunctions.cpp
  + Implemented APDCAM\_SetDetectorReferenceTemp
  + Implemented APDCAM\_GetDetectorReferenceTemp

**2016.05.10**

* Working on continuity counter error issue
  + First I restored the continuity counter cheking to the original code
  + Continuity counter error showdn ow I trying to find the solution to search for the next packet start

**2016.05.11**

* Added m\_PreambSize member variable to DataEvaluation class
* On my PC continuity counter error disappeared but it needs some more inverstigations in the future
* I deleted uneccessary debug file writes.

**2016.05.24**

* Added 100 ms Sleep in Arm function when setting resolution(it did not set before)

**2016.05.26**

* Working on to fix false register writings
  + In the current version if I disconnecting the ethernet cable fro the camera during setup, the functions will fail so there is status check
  + In the next test I will remove the read instruction from the write register function and see what happens
    - The same happened as earlier, I make some more tests with debugger
  + If the read request is there after the write it will gives and error if the network is not OK, but if it is not there there is no return value which says the network is off, although it is still not enough to check if the register writing was successful and the registers are containing the right value.
* Defined APDFactory structure in TypeDefs.h
* Reimplementing GetFactoryData function
  + Done
* Removed the support for writing factory are from the API
* Replacing old factory getter function with the newer one
  + Open done
  + SelfTest done
  + Calibrate
  + SetHV
  + GetHV
  + HVMonitor
* Implemented APDCAM\_Set/GetAnalogPower functions

**2016.05.28**

* Working on to handle wrong register readings when they are too fast
  + Adding a ReadPDI after every WritePDI in a loop, where we are reading back the register value and check if they are the same with the written value, if they are not after 10 tries, we give back an error.
  + Implemented a new WritePDISafe function in LowLevelFunction class
    - With this, all the checking codes are located in one function and only the called function names have to be replaced elsewhere
    - Implemented, tested, still there is uncertainty when we are triing to set some registers, Sleep is still needed in some places, now the system seems working

**2017.04.07.**

* TypeDefs.h
  + Appliing changes made in APDFactory structure according to APDWriteFactoryData software.
* InternalFunctions class
  + Reimplemented GetFactoryData
* APDLib.h
  + implemented APDCAM\_GetFactoryConfig
* HighlevelFunctions
  + Fixed APDCAM\_GetHV
    - changet outputHVCalibs to inputHVCalibs
* **Version number is now v1.5**

**2017.04.10.**

* HighlevelFunctions
  + Fixed APDCAM\_GetConfig
    - Reimplemented voltage readings
* There was a bug in GetFactoryData function with detectorIDList reading
* **Version number is now v1.5.1**

**2017.04.11.**

* Fixing GetTestMode functions
* Added test pattern reading to APDCAM\_GetConfig
* **Version number is now v1.5.2**

**2017.04.21.**

* Implementing Network speed checking function
  + Implemented APDCAM\_GetNetworkConfig in HighLevelFunctions class
  + Defined NetConfStr structure
  + Implemented GetNetworkConfiguration in InternalFunctions class
  + Corrected SENDACK struct naming in GECCommands.h
  + Implemented SendAck function in LowLevelFunctions class
    - The implementation seems rignt because depending on to the ackType the correct byte of data was received according to the GEC card documentation
  + Tested, it works
* **Version number is now v1.6**