|  |  |
| --- | --- |
| **Title:** APDTest program log file | |
| **Project:** APDCAM-10G | |
| **Version:** 1.0 | **Date:** 2015 március 29 |
| **Written by:**  S. Zoletnik | **Checked by:** |
| **Document history:**  Original | |
| **To be done:**  See list at end | |
| **Document access:** Confidental document  **Can be shared with:** | |

Copyright © Fusion Instruments Kft. 2015

**Log file az APDCAM-10G APDTest programhoz**

# 2014.01.04 Zoletnik S.

A program utolsó változata a 140710 verzió. Ebben a Löki Krisztián utolsó verziójához képest saját javítás, hogy a trigger bit-et nem írja a program. Ebből kiindulva elkezdetem módosítani a programokat, hogy egy FLREAD utasítás legyen ami a flash memóriát olvassa ki. Itt van néhány támpont:

APDTest.cpp detektálja FLREAD parancsot

Hívja a ReadFlashPage függvényt az APDTest.cpp-ben

Ez hívja az APCAM\_ReadFlashPage függvényt a HighLevelFunctions.cpp-ben, ez kerül be a könyvtárba.

Ez hívja a GetFlashPage függvényt az InternalFunctions.cpp-ben.

Ez hívja a ReadFlashPage függvényt (ez egy másik mint az APDTest.cpp-beli) a LowLevelFunctions.cpp-ben.

Az OP\_FLREAD parancsot fel kellett sorolni a GECCommands.cpp file tetején a választ adó parancsok listájában.

int CReplyCmdSet::g\_CmdSet[] = {OP\_SENDACK, OP\_READPDI, OP\_FLREAD}; Meg kell növelni ennek az array-nek a méretét is és a kereső ciklusban a GECCommands.h-ban a class definiíciójánál az In() függvényben.

A kamera válaszát CCamClient::OnFlashData() függvény dolgozza fel a CamClient.cpp-ben.

A GECCommands.h-ban definiálni kell az utasítást: FLREAD\_I és FLREAD

Ezekkel a módosításokkal a program lefordul, de nem jön válasz a flash olvasásra, sőt a kamera kommunikáció bedugul utána, csak ki-be kapcsolással lehet feléleszteni.

A flash memória mindig foglaltat jelez (a 215-ös byte mindig 2-t ad). Lehet, hogy nincs flash memória a kártyán.

# 2015.01.09 Zoletnik S.

Kiderült, hogy tényleg nincs flash memória a kártyákon. Egy példányra felraktam próbaképpen egyet. Ezzel a busy flag már 0-t jelez és ment a flash erase utasítás is. Az FLREAD utasítás hibakeresését befejeztem és leteszteltem 5 byte írásával a 0. lapra. Utána megint töröltem a memóriát, tehát ez rendben van. Ebből az APDTest programból csináltam egy új verziót 150109 néven és az exe-t bemásoltam az aktuális IDL csomagba.

# 2015.01.14 Zoletnik S.

Elkezdtem a startFUP utasítás kezelését. A LowLevelFunctions.cpp-ben meg kell csinálni még a StartFUP függvényt és az acknowledge kezelését a megfelelő függvényekben. Az e feletti hívások megvannak.

2015.01.20 Zoletnik S.

Befejeztem a STARTFUP utasítást, de probléma volt, a program nem várt a válaszra. Hosszas tesztelés után kiderült, hogy a GECClient.h-ban volt egy bug, az utasítás kódot unsigned char-ként kezelte. Ez érdekes módon eddig nem jött elő. Ezután a program lefut, megjön a FUP checksum válasz, de nem 0 a checksum.

01.22-én sikerült megoldani a problémát és megy a FUP. (a MAX\_RW\_BYTES értékét állítottam át 2000-re). Viszont nem megy a mérés, nem fogad adatot.

# 2015.03.20 Zoletnik S.

Most tudok visszatérni az APDTest mérés problémára. A január 19-i verzió mér, a 20-i nem, ugyanazzal a parancs file-al. Csak 1-1 csomag jön ADCnként, úgy néz ki valamiért az adat mintaszám nem jó.

# 2015.03.22 Zoletnik S.

Megpróbálom a január 9-i működő verzióba egyenként belerakni a változásokat a test\_150322 directoryban.

GECCommands.h és cpp lefordul, mérés OK.

GECClient.h és cpp lefordul, mérés OK.

LowlevelFunctions.h és cpp lefordul, mérés OK

InternalFunctions.h és cpp lefordul, mérés OK

HighlevelFunctions.cpp lefodul, mérés OK

APDLib.h átrakva, nincs változás

APDTest lefordul, de a mérés nem megy végig

Próbálom a január 9-i APDTest-cpp-ben csak a max regiszter számot átíłrni 1024-ról 2000-re. Ettől nem megy a mérés.

Átállítom a MAX\_WRITE\_SIZE paramétert a LowlevelFunctions.cpp.ben, de nincs változás. Szintén állítom a recBuffer-t a GECClient.cpp-ben.

A mérés során ténylegesen csak 1-1 csomag jön ADC kártyánként. Ha egyszer a rossz programot futtatom, akkor nem jön több adat a kamerából míg ki-be nem kapcsolom, vagy a hálózati interface-en egy down-up ciklust csinálok.

Átállítom a MAX\_WRITE\_SIZE paramétert 1030-ra, ekkor megy a mérés.

Átmásolom a forrásnyelvet a Work directoryba, megy a mérés.

# 2015.03.23 Zoletnik S.

A flash page olvasást próbálom. Nem kapok jó Pg címet, pedig Wireshark-al látom, hogy az UDP csomagban benne van a cím.

# 2015.03.24 Zoletnik S.

A flash page olvasást megjavítottam a page address volt fordított byte sorrendben. Kisebb javítások után ez OK.

# 2015.03.25 Zoletnik S.

Miután a flash memória írás-olvasás is ment a Work verziót átneveztem 150325 verziónak.

# 2015.03.29 Zoletnik S.

A március 25-i verzióban benne maradt egy teszt kiírás, ezt kiszedtem és az utolsó változatot Actual directoryba raktam. Az IDL programokhoz hasonlóan ez lesz a mindenkori aktuálisan működő verzió.

# 2015.03.30 Zoletnik S.

Az APDTest.cpp-ben a main elején volt egy tipus méret kiírás, ezt kiszedtem. Ez a verziót el van mentve 150330 néven, és ez van az Actual-ban is.

# 2015.07.21 Zoletnik S.

Az EAST BES mérésben az APDTest port száma ütközött a shot number olvasás port számával. A HighLevelFunctions.cpp-ben a STREAM\_PORT\_BASE parametert 10000-ről 10001-re állítottam. Jó lenne ha ezt valami környezeti változóval lehetne állítani.

# 2015.10.19 Zoletnik S.

Kiderült, hogy az Actual-ban az APDTest program nem tudja a STARTFUP utasítást. Valahogyan elkeveredtek a verziók. Megtaláltam, hogy csak az APDTest programban hiányzott ez a rész két helyen, ahol a függvény van és ahol a parancs kód le van válogatva. Ezeket a részeket megtaláltam a 2015 január 19-i kimentésben és átmásoltam az APDTest.cpp-be. Ettől megy a firmware upgrade. Még azt is látom, hogy a július 21-i módosítás sem volt benne, ezt is megcsinálom az Actual verzióban. Ez a verzió van elmentve Backup\_20151019a néven. Ez nem megy Dual Sata módban két ADC-vel.

# 2015.10.20 Zoletnik S.

A DualSATA mód azért nem ment két ADC-vel, mert a 2. ADC adatai a 3. stream-en jönnek és nem a 2-on. A HighLevelFunctions.cpp-be beleírtam, hogy olvassa el a CC kártyáról a dualSATA beállítást és annak függvényében programozza fel a 2. vagy a 3. stream-et a 2. ADC-hez. Ez működik mind DualSATA mind nem dualSATA módban. Ezt a verziót teszem el Actualként.

# 2016 március 24 Zoletnik S.

Az APDTest programban módosítottam, hogy a különböző kiírások után legyen egy fflush. Ez azért kellett, hogy a programot a háttrében futtatva és kimenetét egy file-ba irányítva látni lehessen hol tart. Ez az APDTest build 2.1a

# 2018.09.03 Zoletnik S.

2016-18-ban Szabolics Tamás dolgozott a programon. 2018 júliusban találtam meg, hogy az aktuális verzió nem kezeli a dualSATA beállítást és időnként teljesen elrontja a kamera beállításait, nem tartalmazza a firmware upgrade-et és még „segmentation fault’ hibát is tud csinálni. Ezért visszatértem a 2016 március 23-i állapothoz. Ezt most elneveztem 2.2 verziónak.

Beírtam az fflush()-okat, hogy minden üzenet azonnal kimenjen.

Átírtam a DataEvaluation.cpp-ben a ProcessData rutint:

* Bevezettem egy MAX\_PACKET\_LOSS konstanst (DataEvaluation.h) ami megmondja, hogy mennyi elveszett csomagot tolerál a program.
* Ha ugrik a csomag számláló, akkor betesz megfelelő számú 0 tartalmú csomagokt.
* Kivettem minden egyéb átvitel javítási próbálkozást.
* A CDataEvaluation::Handler() rutinban leállítottam a programot, ha MAX\_PACKET\_LOSS feletti csomag vesztés van.

Ez a verzió még nem jól kezeli az adatokat, ha vannak csatornák letiltva.

Tesztelem mi történik, ha triggerelt mérés nem kap triggert. Ha nem jön trigger, nem jön adat. Amikor minden trigger forrás tiltok, akkor elindul az adat és nem áll le. Ilyenkor a ampleNumber már 0 a kamerában és ezért nyilván nem áll le a mérés. Egy stop (CCCOMMAND) utasítás hatására megáll a mérés. Próbáltam az APDTest Stop parancsát, de ez nem állította le. Azért mert hibát adott ha nem volt mérési állapotban a program. Javítottam, a Stop leállítja a kamerát akkor is ha nincs a program mérési állapotban.

# 2018.09.04 Zoletnik S.

A programot módosítottam:

* Ha a csomag számláló visszafele ugrik, akkor hibával leáll.
* Az APDTEST\_ARM hívás elejére tettem egy APDTEST\_Stop-ot. Ennek hatására leáll a mérés a kamerában ha esetleg menne. A fenti trigger hiba így más akkor is megoldódik, ha nem ír valaki Stop marancsot az APDTest scriptjébe.
* Kiszedtem az időmérő kiírásokat és egyebeket.
* Minden olyan kiírás ami hiba az Error-al kezdődik.
* Ha MAX\_PACKET\_LOSS-nál kevesebb csomag veszik el, akkor Warning-ot ír és beilleszti az üres adatot.

Az alábbiakat teszteltem:

* Kidobtam a DataEvaluation-ban a 100-es csomagot. (Kikommentelve benne hagytam, lehet látni hol volt ez.) Ez 1 csomag vesztést szimulál. Valóban megjelenik egy Warning mindkét stream-ben. A mérés végigmegy és az apdcam10g\_control is jól mér, de látszik az egy csomagnyi 0 adat.
* Csináltam triggerelt mérést, de nem adtam triggert. Utána a nem-triggerelt mérés lemegy.
* Az APDTest kimenetét egy file-ba irányítva folyamatosan látható mit csinál. A Start success-re kellene várni az IDL/Python programokban.

# 2018.09.07 Zoletnik S.

A kikapcsolt csatornák kezelése nem volt jó az utolsó verzióban. A DataEvaluation.cpp file-ban a ProcessBlock és ProcessFrame rutinokat áttettem Szabolics Tamás utolsó verziójából. Azt hiszem ezt a kódrészletet én csináltam még valamikor. A class név més volt mint ebben a program verzióban (benne volt egy 10G), ezt javítottam. Ezután lefordult a program. A csatorna kiválasztás viszont nem egészen jó, úgy néz ki a csatorna engedélyező bit maszk ki van fordítva bytonként. (Az IDL programban látható, hogy fordítva jön vissza.) Megnéztem az IDL kódban, valóban úgy kell beadni a kamerába. Csináltam a DataEvaluation.cpp-be egy reverseBits függvényt és betettem ahol beírja a csatorna maszkokat. Ezzel jó a csatorna kezelés.

Amit teszteltem:

* 8,12,14 bit felbontással egyik blokkban a 2-es, másikban a 3-as csatornát kitiltva az ADC teszt módjában jól jönnek az adatok. Ilyenkor 8-asával minden ADC állandó, de különböző adatot küld.

Amit csinálni kell:

A TRIGGER hívásra azt mondja, hogy not implemented.

A leírásba bele kell tenni a TRIGGERT, a SAMPLING-ot, TIMING-ot

# 2018.09.08 Zoletnik S.

Megpróbáltam a TRIGGER hívást feléleszteni, de ezzel sok baj van. A programban vannak valami nyomok, hogy elindult az átírása 10G-re az 1G-ről, de ez csak félig van megcsinálva. A csatornaszám mindenhol 32, az APCAM\_Trigger() függvényben a setInternalTriggerLevels hívásnak még az argumentum típusai sem jók. Az external és internal trigger mintha kizárólagosan lenne engedélyezve. Ezért ezt most abbahagyom és a TRIGGER funkció csak az external triggerre lesz engedélyezve. A software trigger most ayt jelenti, hogy a hardware nincs engedélyezve. Eredetileg egy a PC-ben levő bufferen megvalósított software triggerre gondoltunk. Ennek vannak nyomai a programban.

Megcsináltam a TRIGGER parancsot és teszteltem az IDL GUI-ban, hogy a trigger biteket és a késleltetést megfelelően állítja be

TRIGGER source (0: SOFTWARE, 1: HARDWARE) mode (0: EXTERNAL, 1: INTERNAL) edge (0: RISING, 1: FALLING) delay (microsec)

Őszerevettem hogy ha nincs megnyitva a kamera akkor segmentation fault-ot csinál a program pld. a trigger esetén. Erre betettem egy védelmet minden olyan parancsnál ahol a kamerával csinál alamit.

Kiszedtem a SETPLL parancsot, mert nem tudom mit csinál.

Kiszedtem az SWOPTIONS parancsot, mert nem világos, hogy jót ír-e ki. Pld. a continuity counter méretét miért kell kiírni.

Módosítottam a SETTIMING-ot, hogy a DEV0 ne legyen benne, mert nem állítható amúgy sem. Ellenőriztem a működését, az external clock.ot egyáltalán nem állította be és a paraméter ellenőrzés is hibás volt a hívásnál.

A SAMPLING nem állítja az external sample bitet ez ki is van commentelve. A doc-ban ki kell szedni.

Végignéztem az összes parancs hívását. Mindenhol ahol hiba van „Error....”-t írtam ki, hogy könnyebb legyen a kimenetet ellenőrizni.

Csináltam egy Sync parancsot az APDTest-be, mert nagyon macerás egyébként az ADC-ket szinkronizálni. Ezt be lehetne tenni az Open-be is, de azt gondoltam, hogy az időigénye (~8 ms egy 4 ADCs kamerára) miatt nem tettem bele.

Megcsináltam a kamera User Guide-ban az APDTest leírás aktualizálását.

Készítettem egy SET\_OFFSET parancsot az APDTest-be és leteszteltem.

Észrevett problémák a programmal:

* A script file beírásnál nem látom, hogy lenne valami hibakezelés, ha kevés a paraméter egy sorban.
* Az APDCAM\_xxx rutinok visszaadnak egy hibakódot, de az APDTest csak annyit mond már hogy hiba.

# 2023.01.16

Elkezdtem egy primitív adat vevő programot, hogy megmérjem real time hogy tudnak bejönni az UDP csomagok. Az APDTest-el az a gond, hogy nagyon bonyolult és nem egyesével vesz az UDP csomagokat, hanem egyben többet.