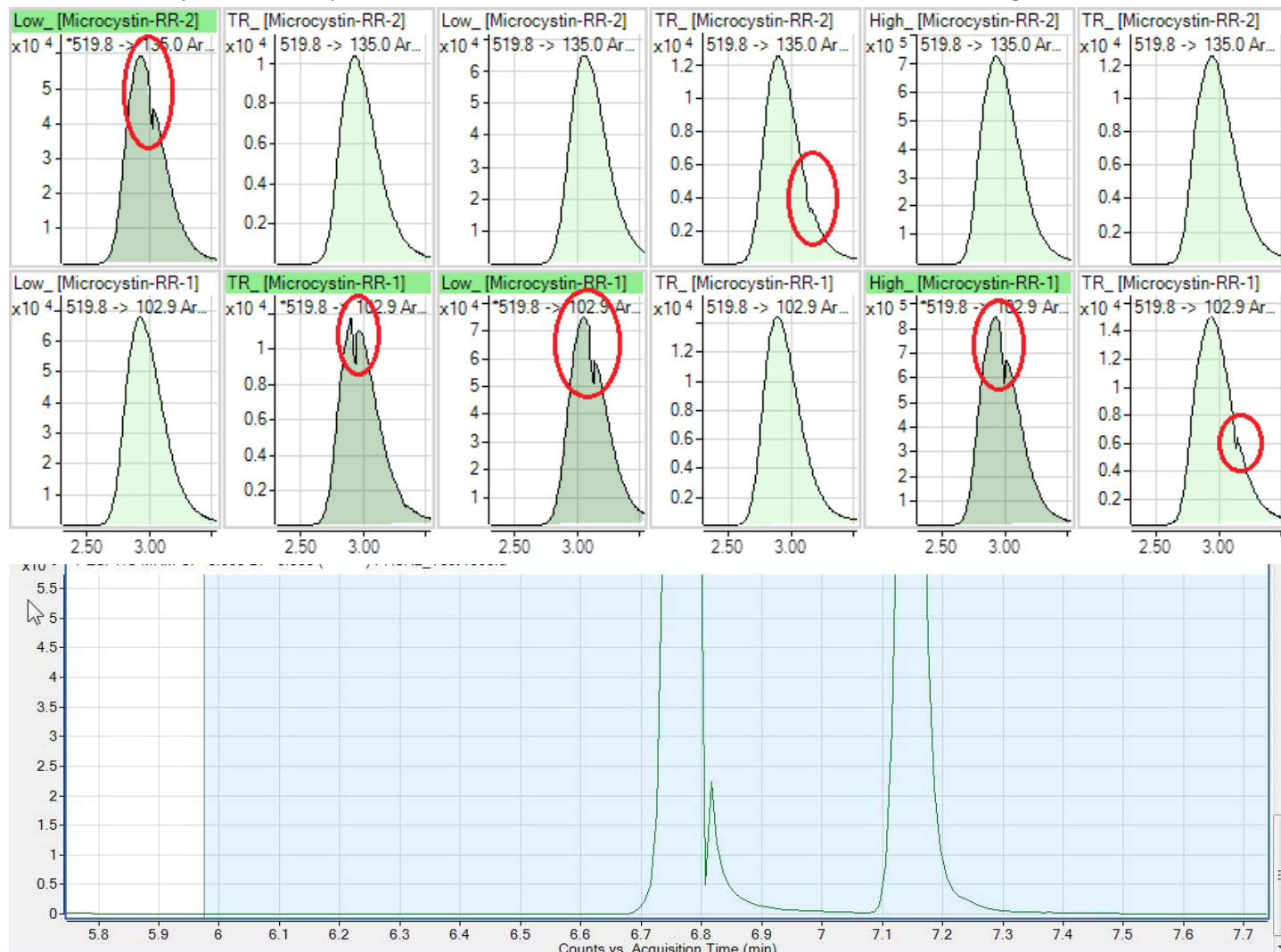




Situation and Explanation: Reports of “distorted chromatographic peaks” was resolved by updating the Mainboard FPGA version to 0x24. This FPGA Update Tool is for Agilent LCMS QQQ, G6400 series, with mainboard part number G7000-65105/61105 only – check the mainboard part number. The update tool’s algorithm checks mainboard’s FPGA version and will proceed the update if mainboard FPGA version is 0x24 or older. The tool will ignore 0x25 and later.



Description of correction for data dropout problem

“When MRM transitions are made that change collision energy, the electrical potentials in elements prior to the collision cell are modified to achieve the desired collision energy. Due to a problem in the FPGA (Field Programmable Gate Array) logic on the system’s Main Board, occasionally two of the potentials that set the collision energy can be left at their previous values because of a transition, resulting in collision energy different that is desired.

The reason why this happens is due to how the voltages are set. The voltages are set using a serial Digital to Analog Converter (DAC) with multiple outputs. The embedded control processor of the system sends the voltage settings to the main board in parallel fashion. These setting are converted to serial streams by the FPGA. Because it takes some time to

send the serial stream to the DAC the settings are placed in a First in First Out (FIFO) memory while they are being written. The FIFO is implemented as a circular buffer where two pointers are incremented or decremented as information is written to or read from the memory. A comparator that looks at the pointers is used by DAC writing logic to determine whether there are values that need to be written. Because it takes a finite amount of time for the FIFO pointers to change their values, we determined that an extra glitch can appear on the comparator that on certain rare occasions causes the logic that writes to the DACs to not update properly. A gate has been added to the DAC writing logic to eliminate the glitches that would cause the DACs not to be written.

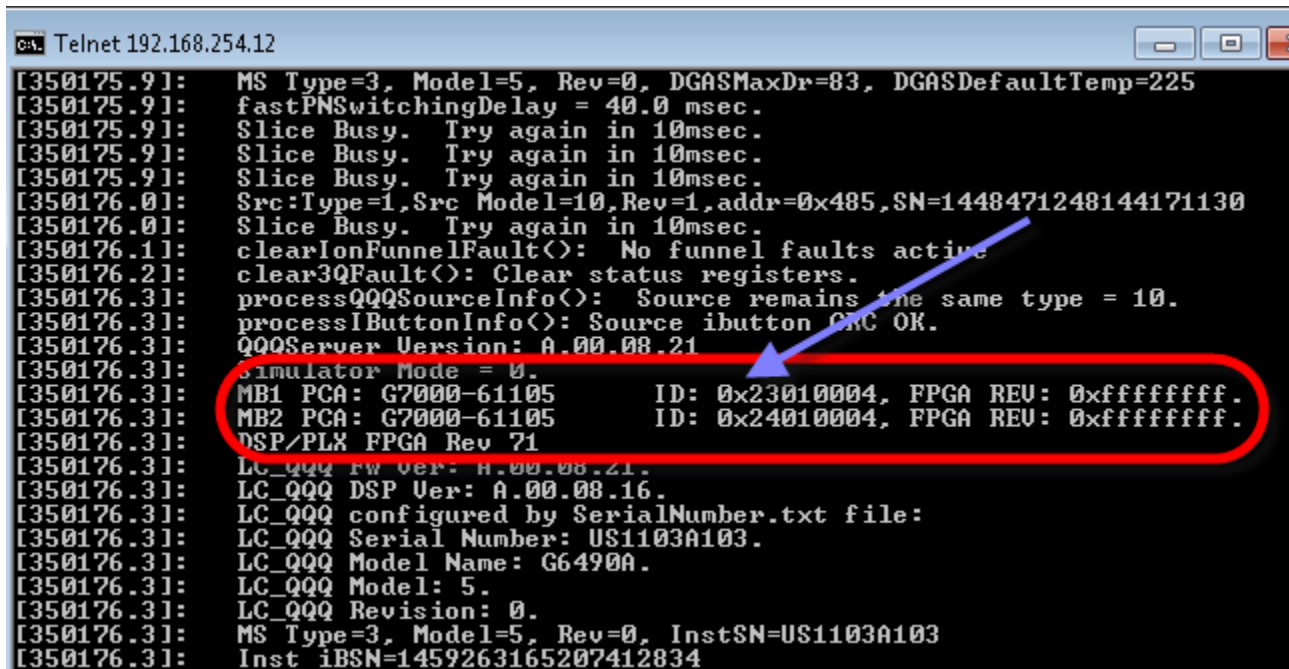
We have extensively tested the FPGA logic change on multiple systems both to be sure that the drop outs are eliminated and to be sure that no unintended artifacts result. We have run the pesticide method from Japan on multiple systems as long as four days without seeing any dropouts. We have run on systems for days using methods developed to induce dropouts every few minutes, and have seen no dropouts. We have also run a series of “regression” tests used by our software team to check various elements of system functionality and have not seen any unexpected artifacts.”

Process: “To be performed ONLY by LCMS QQQ Trained Field Service Engineer”

Download a copy of the FPGA Update Program - QQQ_0x24_FPGA_Uploader.zip – from the Support Portal. Unzip and copy this in a temp directory in the PC (connected to the QQQ).

Please follow the procedure below:

- A. Mainboard FPGA Version
 - a. Open ‘backdoor’ with telnet
 - b. Type ‘get3qverinfo’
 - c. Example screenshot below:



```

Telnet 192.168.254.12
[350175.91]: MS Type=3, Model=5, Rev=0, DGASMaxDr=83, DGASDefaultTemp=225
[350175.91]: fastPNSwitchingDelay = 40.0 msec.
[350175.91]: Slice Busy. Try again in 10msec.
[350175.91]: Slice Busy. Try again in 10msec.
[350175.91]: Slice Busy. Try again in 10msec.
[350176.01]: Src:Type=1,Src Model=10,Rev=1,addr=0x485,SN=1448471248144171130
[350176.01]: Slice Busy. Try again in 10msec.
[350176.11]: clearIonFunnelFault(): No funnel faults active
[350176.21]: clear3QFault(): Clear status registers.
[350176.31]: processQQQSourceInfo(): Source remains the same type = 10.
[350176.31]: processIButtonInfo(): Source ibutton OK.
[350176.31]: QQQServer Version: A.00.08.21
[350176.31]: Simulator Mode = 0.
[350176.31]: MB1 PCA: G7000-61105 ID: 0x23010004, FPGA REV: 0xffffffff.
[350176.31]: MB2 PCA: G7000-61105 ID: 0x24010004, FPGA REV: 0xffffffff.
[350176.31]: DSP/PLX FPGA Rev 71
[350176.31]: LC_QQQ FW Ver: H.00.08.21.
[350176.31]: LC_QQQ DSP Ver: A.00.08.16.
[350176.31]: LC_QQQ configured by SerialNumber.txt file:
[350176.31]: LC_QQQ Serial Number: US1103A103.
[350176.31]: LC_QQQ Model Name: G6490A.
[350176.31]: LC_QQQ Model: 5.
[350176.31]: LC_QQQ Revision: 0.
[350176.31]: MS Type=3, Model=5, Rev=0, InstSN=US1103A103
[350176.31]: Inst iBSN=1459263165207412834
    
```

In this example above: MB1 PCA is 0x23, MB2 PCA is 0x24



B. FPGA Upload Process:

- Using Windows Explorer, execute QQQ_0x24_FPGA_Uploader.exe ('Run as Administrator')
- Please see below:

```

C:\Windows\system32\cmd.exe
Run this Program as Administrator!
Windows PC batch Program to Update LC-QQQ FPGA Logic
0x24 and earlier.(Program will fail if updating 0x25+
Main Boards)
Close Mass Hunter and Remove Mass Hunter Processes.
Windows PC "Telnet" must be enabled.
Close any open Telnet Windows from PC's to LC-QQQ.
Verify PC running this Program is configured
with LAN connected to LC-QQQ Firewalls and Virus Protection OFF.
(i.e.: Standard PC configuration for LC-QQQ)
This is a multi-step process:
1) Enter LC-QQQ IP Address, FPGA files are copied to LC-QQQ SC4.
2) Enter LC-QQQ IP Address, SC4 Telnet login, password, command: ./fpga
3) FPGA logic update progress will be visible in Telnet Window, DO NOT close
Telnet window, Update could take 10+ minutes.
4) Verify Telnet FPGA Update Program completes successfully.
5) Enter "exit" to close Telnet Window.
6) POWER CYCLE LC-QQQ instrument to complete update.
7) If this is a Main Board FPGA Logic Update, use LC-QQQ SC4 "backdoor"
command "get3qverinfo" and verify Main Board revision matches
revision of this LC-QQQ FPGA Logic Updater.
Press Enter to copy files to LC-QQQ:
Copying Files to LC-QQQ SC4 to Update FPGA Logic.
QqQFPGAUpdater version 1.00
Enter instrument ip (192.168.254.12 default):
192.168.254.12 communications check PASSED
SmartcardIU detected.
NewArch_0x24010004_15Khz.xsvf <- This is FPGA Logic File to be used, OK??
FPGA logic @ 192.168.254.12 will be updated
!!>>>DO NOT DISRUPT POWER TO LCMS DURING THE FPGA UPDATE.<<!!
OK to proceed (Y=YES): y
Clearing old PlxCmDir.
Sending 192.168.254.12:/home/sc4/PlxCmDir/PlxCm
Sending 192.168.254.12:/home/sc4/PlxCmDir/CheckPlxRev
Sending 192.168.254.12:/home/sc4/qqq/killfwupdate
Sending 192.168.254.12:/home/sc4/fpga
NEW Logic: srcname=NewArch_0x24010004_15Khz.xsvf dstname=NewArch_0x240100
04_15Khz.xsvf
Sending 192.168.254.12:NewArch_0x24010004_15Khz.xsvf
Aborting qqqServer FW.
Waiting for 0 seconds, press CTRL+C to quit ...
Ready to start LC-QQQ Telnet session to update FPGA logic!
Enter LC-QQQ Telnet: IP address, Login, Password and Command ./fpga.
Enter IP Address(Default 192.168.254.12):

```

Above, follow the instruction.



c. sc4-linux login: sc4 and password: sc4demo (small letters)

```

CA: Telnet 192.168.254.12

sc4-linux login: sc4
Password: _
sc4demo
  
```

d. next screen, type ./fpga

```

CA: Telnet 192.168.254.12

sc4-linux login: sc4
Password:
Login incorrect

sc4-linux login: sc4
Password:
Linux 2.4.31.
1 failure since last login. Last was 07:05:56 on pts/0.
Last login: Tue Mar  8 07:05:04 -0800 2016 on pts/0 from 192.168.254.1.
No mail.
Starting /home/sc4/.profile
LS_OPTIONS changed to: --color=none -F -b -T 0
Tue Mar  8 07:06:05 PST 2016
Done /home/sc4/.profile
sc4@sc4-linux:~$ ./fpga_
  
```

e. You should see below:

```

CA: Telnet 192.168.254.12

FPGA Firmware filename = NewArch_0x24010004_15Khz.xsvf.
Updating FPGA Firmware...
NewArch_0x24010004_15Khz.xsvf opened successfully.
setFWUpdateEnv: set global fil
e pointer and enable JTAG.
XREPEAT
XSTATE
XSTATE
XRUNTEST
XSIR
XSDRSIZE
XSDRSIZE = 37, totalbitswritten = 37
XIDOMASK
XSDRDO
XRUNTEST
XSIR
XRUNTEST
XSIR
XSTATE
XSIR
XSDRSIZE
XSDRSIZE = 11, totalbitswritten = 48
XIDOMASK
XSDRDO
XSIR
XRUNTEST
XSDRSIZE
XSDRSIZE = 21, totalbitswritten = 69
XIDOMASK
XSDRDO
XRUNTEST
XSIR
  
```

***be patient, it might
take 10 or more minutes
to complete update***



f. At its completion:

```
XSIR
XTDOMASK
XSDRTDO
XRUNTEST
XSIR
XRUNTEST
XSIR
XSDRSIZE
XTDOMASK
XSDRTDO
XRUNTEST
XSIR
XREPEAT
XREPEAT
XSTATE
XSTATE
XRUNTEST
XSIR
XSDRSIZE
XTDOMASK
XSDRTDO 6, totalbitswritten = 6903700
XCOMPLETE
SUCCESS - Completed XSUF execution.
Execution Time = 346.910 seconds
Update FPGA Firmware finished.
Disable JTAG now.
sc4@sc4-linux:~$
```

successful completion
of the FPGA update

g. Type exit and "POWER CYCLE" the QQQ.

```
XREPEAT
XREPEAT
XSTATE
XSTATE
XRUNTEST
XSIR
XSDRSIZE
XTDOMASK
XSDRTDO 6, totalbitswritten = 6903700
XCOMPLETE
SUCCESS - Completed XSUF execution.
Execution Time = 346.910 seconds
Update FPGA Firmware finished.
Disable JTAG now.
sc4@sc4-linux:~$ exit
Connection to host lost.
Done FPGA Logic Update, POWER CYCLE LC-XXX MS.
After LC-XXX MS restart, open Mass Hunter
and verify Mass Spectrometer instrument runs correctly.
Waiting for 25 seconds, press CTRL+C to quit ...
```

type exit

However, in step e. above, if the resulting screen is:

```

C:\> Telnet 192.168.254.12

FPGA Firmware filename = NewArch_0x24010004_15Khz.xsvf.
Updating FPGA Firmware...
NewArch_0x24010004_15Khz.xsvf opened successfully.
setFWUpdateEnv: set global file
e pointer and enable JTAG.
XREPEAT
XSTATE
XSTATE
XRUNTEST
XSIR
XSDRSIZE
XSDRSIZE = 37, totalbitswritten = 37
XTDOMASK
XSDRTDO
TDO Expected = 0x07a81a8498
TDO Captured = 0x0000000000
TDO Mask = 0x007ff7fff8
ERROR: TDO mismatch
ERROR at or near XSUF command #8. See line #8 in the XSUF ASCII file.
Execution Time = 0.040 seconds
Update FPGA Firmware finished.
Disable JTAG now.
  
```

Unfortunately, the uploader did NOT complete its purpose of updating the mainboard's FPGA.

Background Information: The PLX board (pictured below) has a jumper J13 that needs to be set correctly for the FPGA uploader program to work correctly. That jumper is part of the JTAG chain. The only purpose of that JTAG chain is to program the FPGAs. There are 3 devices in this specific JTAG chain: FPGA on the PLX board, FPGA on the main board 1 and FPGA on the main board 2. In some cases, such as the simulator used by software developer, there is no main board, so the JTAG chain is broken. The jumper is a way to loop back the JTAG signal in absence of the main board. These jumps (include the ones on the main board) are only used when updating the FPGA bits. They have no effect on the normal operation. This jumper J13 on the PLX board has to be set to 'Global' position. See pictures below:





Next steps to follow:

1. Install Vacuum Keep Alive Switch Box (part number 1960-67001)
2. Turn off front switch
3. Remove smartcard kit (smartcard with PLX board) from card cage
4. Remove smartcard PCA from the kit
5. Check jumper J13 on PLX board and switch jumper from 'local' to 'global' position



6. Re-install smartcard, smartcard kit back to card cage
7. Turn power back on
8. Remove Vacuum Keep Alive Switch Box
9. QQQ will complete smartcard initialization, wait 5 minutes
10. Go back to Step B, above, FPGA Upload Process
11. After successful FPGA upload, re-verify by performing Step A, above, Mainboard FPGA Version

```
[ 68.81: Slice Busy. Try again in 10msec.
[ 68.81: Slice Busy. Try again in 10msec.
[ 68.91: Src:Type=1,Src Model=10,Rev=1,addr=0x485,SN=1448471248144171130
[ 68.91: Slice Busy. Try again in 10msec.
[ 69.01: clearIonFunnelFault(): No funnel faults active
[ 69.11: clear3QFault(): Clear status registers.
[ 69.21: processQQQSourceInfo(): Source remains the same type = 10.
[ 69.21: processIButtonInfo(): Source ibutton CRC OK.
[ 69.21: QQQServer Version: A.00.00.01
[ 69.21: Simulator Mode = 0.
[ 69.21: MB1 PC#: G7000-61105 ID: 0x24010004, FPGA REV: 0xbfffffff.
[ 69.21: MB2 PC#: G7000-61105 ID: 0x24010004, FPGA REV: 0xbfffffff.
[ 69.21: DSP/PLX FPGA Rev 71
[ 69.21: LC_QQQ FW Ver: A.00.08.21.
[ 69.21: LC_QQQ DSP Ver: A.00.08.16.
[ 69.21: LC_QQQ configured by SerialNumber.txt file.
[ 69.21: LC_QQQ Serial Number: US1103A103.
[ 69.21: LC_QQQ Model Name: G6490A.
[ 69.21: LC_QQQ Model: 5.
[ 69.21: LC_QQQ Revision: 0.
[ 69.21: MS Type=3, Model=5, Rev=0, InstSN=US1103A103
[ 69.21: Inst iBSN=1459263165207412834
[ 69.21: Source Model=10, iBSN=1448471248144171130
[ 69.21: LCF MB1: Implement Read Values = TRUE
[ 69.21: LCF MB2: Implement Read Values = TRUE
[ 69.21: MSE_WRITE(): address = 1090, value = 1800.
[ 69.21: MSE_WRITE(): address = 1266, value = 211.
[ 69.21: MSE_WRITE(): address = 1281, value = 0.
[ 69.21: MSE_WRITE(): address = 1024, value = 4284867295
```

success