|  |
| --- |
| Predictive Health Notification  Alinity c Cuvette Integrity |
| ALINITY c Clinical Chemistry Analyzer  Nate Parks  November 18, 2018 |

**PHN - Alinity c Cuvette Integrity Spec Sheet for Apollo**

**References**

D000062792/A Cuvette Status Alert Algorithm Design Document ALINITY c Clinical Chemistry Analyzer

**Summary**

To implement a Predictive Health Notification (PHN) algorithm for Alinity c Analyzers that will detect broken or cracked cuvettes.

|  |  |
| --- | --- |
| PHN Descriptor | Alinity CC Cuvette Integrity |
| PHN Experience Code / Name | PHN\_Alinity\_CC;PHN\_Cuvette; Integrity Segment X  B972(1), B973(2), B974(3), B975(4), B976(5), B978(6), B979(7), B97A(8), B97B(9), B97C(10), B97D(11), B97E(12), B97F(13), B97G(14), B97H(15), B97I(16), B97J(17) |
| PHN KM Article Number/ Name | K31460679 PHN\_Alinity\_CC: Cuvette Integrity |
| Service Level | 1 - Basic Service/Ambassador |
| Always On Package | Always On 01DP5- 01, 02, 03, 09, 70, 79, 80, 83, 84, 89 |
| IDA Table(s) | IDAQOWNER.ICQ\_CCSAMPLEDISPCI SDP  IDAQOWNER.ICQ\_CCDISPENSEPM DPM  IDAQOWNER.ICQ\_RESULTS R |
| IDA Table Fields | SDP.SYSTEMSN,  SDP.LOGDATE\_LOCAL,  SDP.DISPENSEBEGINAVERAGE,  SDP.SAMPLEKEY,  SDP.TESTNUMBER,  SDP.REPLICATESTART,  SDP.REPLICATENUMBER,  DPM.MODULESN,  DPM.SYSTEMSN,  DPM.LOGDATE\_LOCAL,  DPM.SAMPLEKEY,  DPM.TOSHIBATESTNUMBER,  DPM.STARTINGREPLICATENUMBER,  DPM.REPLICATENUMBER,  DPM.TESTID,  R.SYSTEMSN,  R.TESTID,  R.CUVETTENUMBER,  SDP.DISPENSEBEGINAVERAGE |
| Analysis Frequency | Daily |
| Data Required | Previous 7 days |
| Data Aggregation | None |
| Run Time Estimate | N/A |
| Flag Criteria | **From Architect PHN00065**: DisBeginAve > 20,000 for> 20% of all sampling events for a total of no more than 4 (c4000) or 7 (c16000, A- and B-lines are treated separately) cuvettes\*. Note: total number of sampling events for each suspected cuvette > 20.  **For Alinity**: DISPENSEBEGINAVERAGE > 20,000 for > 20% of all sampling events for individual cuvette(s) when <= 8\* cuvettes are triggered. The desirable number of sampling events for each cuvette > 20.  \*Note that for Alinity, “for a total of no more than 4 (c4000) or 7 (c16000, A- and B-lines are treated separately) cuvettes” translates to “when <= 8 cuvettes are triggered”. For Architect, 4 cuvettes on a c4000 and 7 cuvettes on a c16000 account for a bit over 4% of the total number of cuvettes. Translating to Alinity, 4% of 187 cuvettes is about 8. |
| Probable Failure Modes | Replace cuvette segment with broken/cracked cuvettes. |
| Suppression Experience Codes | Z5O0, FA7K, Z3UO |
| Applicable Work Done Codes (WDC) | F7\*\* Optics/Detector Cuvette |

**Data Processing Steps**

|  |  |
| --- | --- |
| Data Processing Steps | |
| 1 | Query all data from the previous 7 days for each instrument (IDA Tables: IDAQOWNER.ICQ\_CCSAMPLEDISPCI, IDAQOWNER.ICQ\_CCDISPENSEPM, IDAQOWNER.ICQ\_RESULTS) |
| 2 | Exclude cuvettes where the number of sampling events is less than 20. |
| 3 | Exclude rows where the cuvette number is null. |
| 4 | Identify cuvettes with sampling events having DISPENSEBEGINAVERAGE > 20,000. |
| 5 | For cuvettes with sampling events having DISPENSEBEGINAVERAGE > 20,000, calculate the percent of sampling events for individual cuvettes having DISPENSEBEGINAVERAGE > 20,0000. |
| 6 | Calculate the number of cuvettes with > 20% of all sampling events having DISPENSEBEGINAVERAGE > 20,000. |
| 7 | Flag any instrument where DISPENSEBEGINAVERAGE > 20,000 for > 20% of all sampling events for individual cuvette(s) when <= 8 cuvettes are triggered. |

**Define Reusable Group Routine**

To run all the Cuvvette Integrity algorithms as a single group routine (to avoid performance issues), define a reusable group routine with the following parameters.

|  |  |
| --- | --- |
| **Routine Details** |  |
| Routine Source | Define Reusable Group Routine |
| Routine Type | Oracle Procedure |
| Run Mode | Batch |
| Routine Invoke Command | PHM\_ICQ\_Cuv\_Integrity\_Grp\_Alg |
| Status | Enable |
| ODS Routine Name | PHM\_ODS\_ICQ\_CC\_RESULTS\_PROC |

|  |  |
| --- | --- |
| **Apollo Details** |  |
| Algorithm ID \* | ALINITYC\_CUVETTE\_INTEGRITY\_GROUP |
| Algorithm Name \* | Alinity CC Cuvette Integrity Group Algorithm |
| Algorithm Description \* | Group algorithms to run all the Cuvette Integrity algorithms which are part of this group |
| Product Family \* | Alinity CC |
| Algorithm Group \* | Alinity Cuvette |
| Functional Area | N/A |
| Algorithm Category 1 | N/A |
| Algorithm Category 2 | N/A |
| Algorithm Category 3 | N/A |
| Remaining Useful Life Value | N/A |
| Remaining Useful Life Unit | N/A |
| Keep Results Num Days | N/A |
| **Routine Details** |  |
| Routine Source | Define Reusable Group Routine |
| Routine Type | Oracle Procedure |
| Run Mode | Batch |
| Routine Invoke Command | PHM\_ICQ\_Cuv\_Integrity\_Grp\_Alg |
| **ODS Routine Details** |  |
| ODS Routine Name | PHM\_ODS\_ICQ\_CC\_RESULTS\_PROC |

**Define Reusable Routine**

|  |  |
| --- | --- |
| **Routine Details** |  |
| Routine Source | Define Reusable Group Routine |
| Routine Type | Oracle Procedure |
| Run Mode | Batch |
| Routine Invoke Command | PHM\_ICQ\_Cuvette\_Integrity\_PROC |
| Status | Disable |

|  |  |
| --- | --- |
| **Apollo Details** |  |
| Algorithm ID \* | Alinity CC Cuvette Integrity - Generic |
| Algorithm Name \* | Alinity CC Cuvette Integrity - Generic |
| Algorithm Description \* | To detect broken or cracked cuvettes. |
| Product Family \* | Alinity CC |
| Algorithm Group \* | Alinity Cuvette |
| Functional Area | N/A |
| Algorithm Category 1 | N/A |
| Algorithm Category 2 | N/A |
| Algorithm Category 3 | N/A |
| Remaining Useful Life Value | 7 |
| Remaining Useful Life Unit | Day |
| Keep Results Num Days | 14 |
| **Routine Details** |  |
| Routine Source | Define Reusable Routine |
| Routine Type | Oracle Procedure |
| Run Mode | Batch |
| Routine Invoke Command | PHM\_ICQ\_Cuvette\_Integrity\_PROC |
| **Parameters** |  |
| Parameter Group Name | ICQ\_Cuvette\_Integrity |
| **Parameter Name** | **Parameter Values** |
| IHN\_LEVEL3\_DESC | Alinity CC Cuvette Integrity Segment X |
| CUVETTEINTEGRITY\_SEGMENT1 | 1 |
| CUVETTEINTEGRITY\_SEGMENT2 | 11 |
| CUVETTEINTEGRITY\_DISBEGAVG\_MIN | 20000 |
| CUVETTEINTEGRITY\_SAMPEVENTS\_MIN | 20 |
| CUVETTEINTEGRITY\_PERCSAMPEVENTS\_MIN | 0.2 |
| CUVETTEINTEGRITY\_NUMCUVETTES\_MAX | 8 |
| THRESHOLDS\_COUNT | 1 |
| THRESHOLD\_DESCRIPTION | Alinity CC Cuvette Integrity Segment X |

**APPENDIX 1:** B9XX PHN\_CC\_Cuvette\_Integrity Segment X

**Algorithm Code**

SELECT

final2.MODULESN

FROM

(SELECT

middle2.\*,

(middle2.NUM\_SAMPEVENTS\_GT20000\_PERCUV / middle2.NUM\_SAMPEVENTS\_PERCUV) AS PERC\_SAMPEVENTS\_GT20000\_PERCUV,

CASE WHEN (middle2.NUM\_SAMPEVENTS\_GT20000\_PERCUV / middle2.NUM\_SAMPEVENTS\_PERCUV) > 0.2

THEN 1

ELSE 0

END AS GT20000\_GT20PERC\_SAMPEVENTS

FROM

(SELECT

inner2.MODULESN,

inner2.CUVETTENUMBER,

COUNT(inner2.CUVETTENUMBER) AS NUM\_SAMPEVENTS\_PERCUV,

SUM(inner2.CHECK\_GT20000) AS NUM\_SAMPEVENTS\_GT20000\_PERCUV

FROM

(SELECT

SDP.SYSTEMSN,

SDP.LOGDATE\_LOCAL,

SDP.DISPENSEBEGINAVERAGE,

SDP.SAMPLEKEY,

SDP.TESTNUMBER,

SDP.REPLICATESTART,

SDP.REPLICATENUMBER,

DPM.MODULESN,

DPM.SYSTEMSN,

DPM.LOGDATE\_LOCAL,

DPM.SAMPLEKEY,

DPM.TOSHIBATESTNUMBER,

DPM.STARTINGREPLICATENUMBER,

DPM.REPLICATENUMBER,

R.SYSTEMSN,

R.TESTID AS RESULTS\_TESTID,

R.CUVETTENUMBER,

CASE WHEN SDP.DISPENSEBEGINAVERAGE > 20000

THEN 1

ELSE 0

END AS CHECK\_GT20000

FROM

IDAQOWNER.ICQ\_CCSAMPLEDISPCI SDP

LEFT JOIN IDAQOWNER.ICQ\_CCDISPENSEPM DPM

ON SDP.SYSTEMSN = DPM.SYSTEMSN

AND DPM.LOGDATE\_LOCAL

BETWEEN SDP.LOGDATE\_LOCAL - INTERVAL '0.1' SECOND AND SDP.LOGDATE\_LOCAL + INTERVAL '0.1' SECOND

AND SDP.SAMPLEKEY = DPM.SAMPLEKEY

AND SDP.TESTNUMBER = DPM.TOSHIBATESTNUMBER

AND SDP.REPLICATESTART = DPM.STARTINGREPLICATENUMBER

AND SDP.REPLICATENUMBER = DPM.REPLICATENUMBER

LEFT JOIN IDAQOWNER.ICQ\_RESULTS R

ON DPM.SYSTEMSN = R.SYSTEMSN

AND DPM.TESTID = R.TESTID

WHERE

SDP.LOGDATE\_LOCAL >= TRUNC(SYSDATE) - 7

AND SDP.LOGDATE\_LOCAL < TRUNC(SYSDATE)

AND R.CUVETTENUMBER IS NOT NULL) inner2

GROUP BY

inner2.MODULESN,

inner2.CUVETTENUMBER

ORDER BY

inner2.MODULESN,

inner2.CUVETTENUMBER) middle2

WHERE

middle2.NUM\_SAMPEVENTS\_PERCUV > 20

AND middle2.CUVETTENUMBER BETWEEN 1 AND 11) final2

WHERE

final2.GT20000\_GT20PERC\_SAMPEVENTS = 1

GROUP BY

final2.MODULESN

HAVING

COUNT(final2.MODULESN) <= 8

ORDER BY

final2.MODULESN

**Note:** Replace the yellow highlighted code with the value for the CUVETTEINTEGRITY\_SEGMENT parameter in the table below.

**Apollo Algorithm Details**

(\* is Mandatory)

|  |  |
| --- | --- |
| **Apollo Details** |  |
| Algorithm ID \* | Alinity CC Cuvette Integrity Segment X where x=1-17 |
| Algorithm Name \* | Alinity CC Cuvette Integrity Segment X where x=1-17 |
| Algorithm Description \* | To detect broken or cracked cuvettes. |
| Product Family \* | Alinity CC |
| Algorithm Group \* | Alinity Cuvette |
| Functional Area | N/A |
| Algorithm Category 1 | N/A |
| Algorithm Category 2 | N/A |
| Algorithm Category 3 | N/A |
| Remaining Useful Life Value | 7 |
| Remaining Useful Life Unit | Day |
| Keep Results Num Days | 14 |
| **Routine Details** |  |
| Routine Source | Use Reusable Group Routine |
| Reusable Routines | Alinity CC Cuvette Integrity Group Algorithm |
| Run Mode | Batch |
| Status | Enable |
| **Predictive Health Notification Details** |  |
| PHN Code | PHN\_Alinity\_CC\_B9XX  Note:Replace the exp code based on segment# |
| Issue Description (Use Algorithm Name) | N/A |
| Experience Code | B972(1), B973(2), B974(3), B975(4), B976(5), B978(6), B979(7), B97A(8), B97B(9), B97C(10), B97D(11), B97E(12), B97F(13), B97G(14), B97H(15), B97I(16), B97J(17) |
| **Knowledge Management DB Articles** |  |
| KM Article ID | K31460679 |
| KM Article | PHN\_Alinity\_CC: Cuvette Integrity |
| **Parameters** |  |
| Parameter Group Name | ICQ\_Cuvette\_Integrity |
| **Parameter Name** | See below |
| **Parameter Values** | See below |
| **Chart Details** |  |
| Chart Title | Alinity CC Cuvette Integrity Segment X  Note: Replace X based on Segment Number |
| Chart Type | Line |
| Chart Threshold Parameter | ICQ\_Cuvette\_Integrity-THRESHOLDS\_COUNT |
| Group ID | Group 7 |
| Chart X Axis Name | Date |
| Chart Y Axis Name | Threshold Count |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Parameter Values** | | | | | |
| **Parameter Name** | Segment 1 | Segment 2 | Segment 3 | Segment 4 | Segment 5 | Segment 6 |
| CUVETTEINTEGRITY\_SEGMENT | 1 AND 11 | 12 AND 22 | 23 AND 33 | 34 AND 44 | 45 AND 55 | 56 AND 66 |
| CUVETTEINTEGRITY\_DISBEGAVG\_MIN | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 |
| CUVETTEINTEGRITY\_SAMPEVENTS\_MIN | 20 | 20 | 20 | 20 | 20 | 20 |
| CUVETTEINTEGRITY\_PERCSAMPEVENTS\_MIN | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |
| CUVETTEINTEGRITY\_NUMCUVETTES\_MAX | 8 | 8 | 8 | 8 | 8 | 8 |
| THRESHOLD\_COUNT | 1 | 1 | 1 | 1 | 1 | 1 |
| THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION |
|  |  |  |  |  |  |  |
|  | Segment 7 | Segment 8 | Segment 9 | Segment 10 | Segment 11 | Segment 12 |
| CUVETTEINTEGRITY\_SEGMENT | 67 AND 77 | 78 AND 88 | 89 AND 99 | 100 AND 110 | 111 AND 121 | 122 AND 132 |
| CUVETTEINTEGRITY\_DISBEGAVG\_MIN | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 |
| CUVETTEINTEGRITY\_SAMPEVENTS\_MIN | 20 | 20 | 20 | 20 | 20 | 20 |
| CUVETTEINTEGRITY\_PERCSAMPEVENTS\_MIN | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |
| CUVETTEINTEGRITY\_NUMCUVETTES\_MAX | 8 | 8 | 8 | 8 | 8 | 8 |
| THRESHOLD\_COUNT | 1 | 1 | 1 | 1 | 1 | 1 |
| THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION |
|  |  |  |  |  |  |  |
|  | Segment 13 | Segment 14 | Segment 15 | Segment 16 | Segment 17 |  |
| CUVETTEINTEGRITY\_SEGMENT | 133 AND 143 | 144 AND 154 | 155 AND 165 | 166 AND 176 | 177 AND 187 |  |
| CUVETTEINTEGRITY\_DISBEGAVG\_MIN | 20,000 | 20,000 | 20,000 | 20,000 | 20,000 |  |
| CUVETTEINTEGRITY\_SAMPEVENTS\_MIN | 20 | 20 | 20 | 20 | 20 |  |
| CUVETTEINTEGRITY\_PERCSAMPEVENTS\_MIN | 0.2 | 0.2 | 0.2 | 0.2 | 0.2 |  |
| CUVETTEINTEGRITY\_NUMCUVETTES\_MAX | 8 | 8 | 8 | 8 | 8 |  |
| THRESHOLD\_COUNT | 1 | 1 | 1 | 1 | 1 |  |
| THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION | THRESHOLD\_DESCRIPTION |  |

**APPENDIX 2:** Algorithm Understanding Check – Algorithm Developer to Predictive Health Monitoring (PHM) Specialist Transition

This step is not necessary. Originally designed for Architect, this algorithm was directly converted/re-coded to work on Alinity. As such, there was no “hand-off” of the algorithm from Algorithm Developer to PHM Specialist, and no Understanding Check was necessary. Proper algorithm functionality was checked by the PHM Specialist.

**APPENDIX 3:** Algorithm Verification in Apollo

**Data Set Description**

The data set for this transition was retrieved from the IDAQOWNER.ICQ\_CCSAMPLEDISPCI, IDAQOWNER.ICQ\_CCDISPENSEPM and IDAQOWNER.ICQ\_RESULTS tables within the DABBTO database. Data was collected for all available instruments on November 5 and November 6 of 2018. Note that in order to generate algorithm violations, for verification purposes only, the DISPENSEBEGINAVERAGE threshold was lowered from 20,000 to 8,500. The threshold will be returned to 20,000 once verification is complete. Verification was conducted on three of the 17 Alinity c cuvette segments: segments 1, 9, and 17. Verification was conducted on a subset of cuvette segments as the Cuvette Integrity algorithm utilizes a reusable routine, run on each of the 17 segments, that does not differ between segments.

**PHM Specialist Analysis Output**

The following instruments (MODULESN) were identified as violating the algorithm by the PHM Specialist:

**11/5/2018:**

|  |  |  |
| --- | --- | --- |
| **Segment 1** | **Segment 9** | **Segment 17** |
|  |  |  |

**11/6/2018:**

|  |  |  |
| --- | --- | --- |
| **Segment 1** | **Segment 9** | **Segment 17** |
|  |  |  |

**Apollo Developer Analysis Output**

The following instruments (SN) were identified as violating the algorithm by the Apollo Developer:

**11/5/2018:**

|  |  |  |
| --- | --- | --- |
| **Segment 1** | **Segment 9** | **Segment 17** |
|  | cid:image006.png@01D47672.CE277A30 | cid:image007.png@01D47672.CE277A30 |

**11/6/2018:**

|  |  |  |
| --- | --- | --- |
| **Segment 1** | **Segment 9** | **Segment 17** |
| cid:image002.png@01D47672.CE277A30 | cid:image003.png@01D47672.CE277A30 | cid:image004.png@01D47672.CE277A30 |

**Algorithm Transition Summary**

Based on the outputs from both the Apollo Developer and PHM Specialist, the Apollo Developer’s implementation of the delivered algorithm is confirmed. Both the Apollo Developer and PHM Specialist flagged the same instruments for each cuvette segment on November 5, 2018 and November 6, 2018.