Code Verification

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **ID** |
|  | WIDLogV2.c | **1** |
|  | Strings.c | **2** |
|  | MX\_SerComms.c | **3** |
|  | MX\_GM862.c | **4** |
|  | MX\_NRF24L01.c | **5** |
|  | MX\_SerFlash.c | **6** |
|  | MX\_USART.c | **7** |
|  | MX\_RTC.c | **8** |
|  | MList.c | **9** |
|  | TC\_driver.c | **10** |
|  | eeprom\_driver.c | **11** |
|  | Clksys\_driver.c | **12** |
|  |  |  |
|  | WIDLogV2.h | **1a** |
|  | Strings.h | **2a** |
|  | MX\_SerComms.h | **3a** |
|  | MX\_GM862.h | **4a** |
|  | MX\_NRF24L01.h | **5a** |
|  | MX\_SerFlash.h | **6a** |
|  | MX\_USART.h | **7a** |
|  | MX\_RTC.h | **8a** |
|  | MList.h | **9a** |
|  | TC\_driver.h | **10a** |
|  | eeprom\_driver.h | **11a** |
|  | Clksys\_driver.h | **12a** |
|  |  |  |

Contents

[1 General Functions 3](#_Toc304977268)

[1.1 Main Function 3](#_Toc304977269)

[1.2 Send Email Function 4](#_Toc304977270)

[1.3 ListUpdate 5](#_Toc304977271)

# 1 WIDLogV2.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | WIDLogV2.c | **int main(void);** |
|  | WIDLogV2.c | ISR(PORTB\_INT0\_vect) |
|  | WIDLogV2.c | ISR(TCC0\_OVF\_vect) |
|  | WIDLogV2.c | void MainLoop\_BatBackup\_NoBat\_Mode() |
|  | WIDLogV2.c | void MainLoop\_BatBackup\_LoBat\_Mode() |
|  | WIDLogV2.c | void WaitForMainBattery() |
|  | WIDLogV2.c | void SetupHardware() |
|  | WIDLogV2.c | uint8\_t EnterBatBakMode(uint8\_t Reason) |
|  | WIDLogV2.c | uint8\_t WakeFromBatBakMode() |
|  | WIDLogV2.c | float CheckBatteryVoltage() |
|  | WIDLogV2.c | void CalibrateADCOffset() |
|  | WIDLogV2.c | int ADCRead(uint8\_t SAMP) |
|  | WIDLogV2.c | void RTCWakeChecks() |
|  | WIDLogV2.c | void ReelInTheYears() |
|  | WIDLogV2.c | void ShowMenu() |
|  | WIDLogV2.c | void doRFWork(uint8\_t dispLvl) |
|  | WIDLogV2.c | void UpdateTagList() |
|  | WIDLogV2.c | uint8\_t saveRecord(RTC\_Time\_t\* mTime, uint32\_t\* ID) |
|  | WIDLogV2.c | uint8\_t loadRecord(uint32\_t RecordNo, RTC\_Time\_t\* mTime, uint32\_t\* ID) |
|  | WIDLogV2.c | uint8\_t RecordToFlash(RTC\_Time\_t\* mTime, uint32\_t ID, uint8\_t Flags) |
|  | WIDLogV2.c | void StartTim10ms(uint16\_t Time) |
|  | WIDLogV2.c | void StopTim10ms() |
|  | WIDLogV2.c | void save\_PC(void) |
|  | WIDLogV2.c | uint8\_t ReadCalibrationByte( uint8\_t index ) |
|  | WIDLogV2.c | void FilterStr2Mask(char\* filtStr) |
|  | WIDLogV2.c | uint32\_t FindRecordCount() |
|  | WIDLogV2.c | void SetClkSpeed(uint8\_t speed) |

## 1.1 Main Function

**Synopsis**

**int main(void);**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description/Reasoning | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection |  |  |  |  |
| Class 2 –Practical Use Test | Code has been used | No Problems Caused |  |  |

## 1.2 Main Function

**Synopsis**

ISR(PORTB\_INT0\_vect)

This function is called when the power good pin on the DC-DC switching converter goes low, indicating that the main battery has been removed. It’s job is to put the logger into a low-power ‘battery backup’ mode if appropriate and to return to a different

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description/Reasoning | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection |  |  |  |  |
| Class 2 –Practical Use Test | Code has been used | No Problems Caused |  |  |

# 2 Strings.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | WIDLogV2.c | **int main(void);** |
|  | WIDLogV2.c | ISR(PORTB\_INT0\_vect) |

## 2.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 3 MX\_SerComms.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_SerComms.c | **int main(void);** |
|  | MX\_SerComms.c | ISR(PORTB\_INT0\_vect) |

## 3.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 4 MX\_GM862.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_GM862.c | **int main(void);** |
|  | MX\_GM862.c | ISR(PORTB\_INT0\_vect) |

## 4.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 5 MX\_NRF24L01.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_NRF24L01.c | **int main(void);** |
|  | MX\_NRF24L01.c | ISR(PORTB\_INT0\_vect) |

## 5.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 6 MX\_SerFlash.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_SerFlash.c | **int main(void);** |
|  | MX\_SerFlash.c | ISR(PORTB\_INT0\_vect) |

## 6.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 7 MX\_USART.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_USART.c | **int main(void);** |
|  | MX\_USART.c | ISR(PORTB\_INT0\_vect) |

## 7.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 8 MX\_RTC.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_RTC.c | **int main(void);** |
|  | MX\_RTC.c | ISR(PORTB\_INT0\_vect) |

## 8.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 9 MList.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MList.c | **int main(void);** |
|  | MList.c | ISR(PORTB\_INT0\_vect) |

## 9.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 10 TC\_driver.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | TC\_driver.c | **int main(void);** |
|  | TC\_driver.c | ISR(PORTB\_INT0\_vect) |

## 10.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 11 MX\_NRF24L01.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_NRF24L01.c | **int main(void);** |
|  | MX\_NRF24L01.c | ISR(PORTB\_INT0\_vect) |

## 11.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 12 MX\_NRF24L01.c Functions

|  |  |  |
| --- | --- | --- |
| **Verified** | **Source File** | **Function** |
|  | MX\_NRF24L01.c | **int main(void);** |
|  | MX\_NRF24L01.c | ISR(PORTB\_INT0\_vect) |

## 12.1 Main Function

**Synopsis**

mx\_err\_t mList\_Init(void);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Visual Code Inspection | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Use Test | |  |  |  |  |
| 2.1 |  |  |  |  |  |

# 13 Other tests

## 13.1 Send Email Function

**Synopsis**

**GM862\_err\_t** **GM862\_SendEmail**(**void**);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – | |  |  |  |  |
| 1.1 |  |  |  |  |  |
| Class 2 –Practical Test | | Simulated use in the field. |  |  |  |
| 2.1 | First Send | Device with 393 logs on it, scheduled to send email at 10:53AM on 28-09-2011. Send Period set to 10 minutes, all other properties set up correctly (email address, GPRS settings) (set to split email every 100 logs) | 4 emails received with all data. | 4 emails received with all data | Pass |
| 2.2 | Second Send | Simulated some tag pickups. 0x1DC logs total now. | One email with log 0x1C9 through to log 0x1DC | One email with log 0x1C9 through to log 0x1DC | Pass |
| 2.3 | Third Send | Simulated some tag pickups. 0x1F3 logs total now. | One email with log 0x1DC through to log 0x1F3 | One email with log 0x1DC through to log 0x1F3 | Pass |
| 2.4 | Fourth Send | No more tags pickups simulated. 0x1F5 logs total now. | One email with log 0x1F3 through to log 0x1F5 | One email with log 0x1F3 through to log 0x1F5 | Pass |
| 2.5 | Repeated entries |  |  |  |  |

## 13.2 ListUpdate

**Synopsis**

**Not a specific function but the whole List-Update system.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Random Entries | |  |  |  |  |
| 1.1 | Random Tags 1 | Randomly sending 8 different tags. | Exported to csv, sorted by tag number. Separate into groups of logs from single tags. Each start log has an accompanying stop log and vice versa. There are never two start or two stop logs from one tag in a row (ordered by time). | Exported to csv, sorted by tag number. Separate into groups of logs from single tags. Each start log has an accompanying stop log and vice versa. There are never two start or two stop logs from one tag in a row (ordered by time). | Pass |
| Class 2 – Full List | |  |  |  |  |
| 2.1 | Overflow | Set List max size to 5. Sent 8 different tags. | The first 5 go into list. Other sent tags after that just get a ‘start’ and no accompanying ‘stop’. | The first 5 go into list. Other sent tags after that just get a ‘start’ and no accompanying ‘stop’. | Pass |

## 13.3 Wait For Main Battery on Startup Function

NOT IMPLEMENTED!!!! – Main battery MUST be plugged in before backup battery!

**Synopsis**

No function, just the way it works.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Description/Reasoning | | Input | Expected Output | Actual Output | Pass/Fail |
| Class 1 – Startup Tests | |  |  |  |  |
| 1.1 | Power Good | Main battery is already connected on start-up. Tested using Debugger. | Startup and operation as normal. | Startup and operation as normal. | Pass |
| 1.2 | Power Bad | Main battery is not connected on start-up. Tested using Debugger. | Does necessary startup tasks then enters battery backup mode. Executes ‘Startup’ instructions in main ‘awake’ loop when getting power. Record count is retrieved in wakefrombatbak() function. | Does necessary startup tasks then enters battery backup mode. Executes ‘Startup’ instructions in main ‘awake’ loop when getting power. Record count is retrieved in wakefrombatbak() function. | Pass |
| Class 2 – | |  |  |  |  |
|  |  |  |  |  |  |