

MCP79410 Alarm issue



Amond Lin
2024 SEP

Application

- 電池使用時間紀錄
- 每23小時紀錄一次，每一次增加一個計數
- 當計數達到上限，預估五年，則判定電池已經老化

初始化

- RTCSEC=0x00
- RTCMIN=0x00
- RTCHOUR=0x13
- RTCWKDAY=0x0B
- RTCDATE=0x22
- RTCMTH=0x06
- RTCYEAR=0x22
- CONTROL=0x10
- RTCSEC=0x80

- ALMOSEC=0x00
- ALMOMIN=0x00
- ALMOHOUR=0x00
- ALMOWKDAY=0x23

Hour match

正常情況

- 初始化的時間為2022, JUN, 22, Wednesday, 13:00:00，當ST(RTCSEC bit <7>)=1時，RTCC開始工作
- 當計數到00:00:00，ALM0IF=1
- 軟體清除ALM0IF，並設定ALM0HOUR=0x23
- 每次ALM0IF=1時，軟體將會清除ALM0IF以及將ALM0HOUR讀回後減1寫入

異常情況

- 初始化的時間為2022, JUN, 22, Wednesday, 13:00:00，當ST(RTCSEC bit <7>)=1時，RTCC開始工作
- 當計數到00:00:00，ALM0IF=1
- 軟體清除ALM0IF，並設定ALM0HOUR=0x23
- 此時RTCHOUR=0x01就會發生ALM0IF=1，軟體執行ALM0IF=0且ALM0HOUR讀回減1寫入(ALM0HOUR=0x22)，當RTCHOUR=0x22，又可以正常ALM0IF=1
- 每次ALM0IF=1時，軟體將會清除ALM0IF以及將ALM0HOUR讀回後減1寫入，經過此設定隔一小時就會發生觸發

時間軸：正確Alarm

RTCC

2022/6/22

WED

13:00:00

2022/6/23

TUE

00:00:00

2022/6/23

TUE

23:00:00

2022/6/24

FRI

22:00:00

2022/6/25

SAT

21:00:00

Set

00:00:00

Set

23:00:00

Set

22:00:00

Set

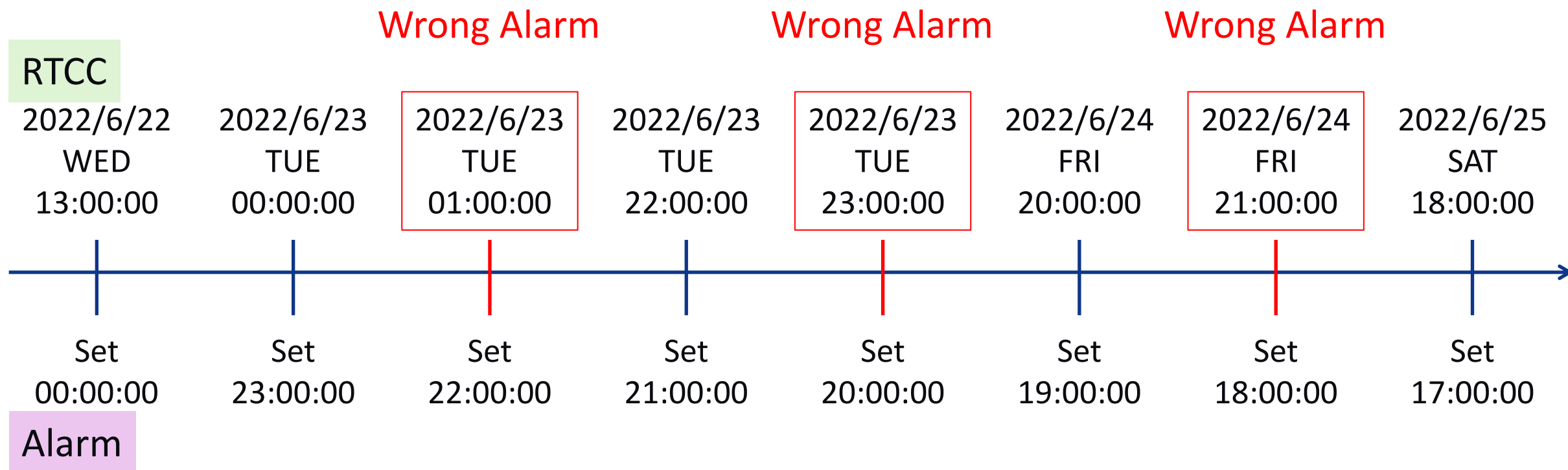
21:00:00

Set

20:00:00

Alarm

時間軸：錯誤Alarm



測試流程

- 初始化以下暫存器
- RTCSEC=0x00
- RTCMIN=0x00
- RTCHOUR=0x05
- RTCWKDAY=0x03
- RTCDATE=0x18
- RTCMTH=0x09
- RTCYEAR=0x24
- CONTROL=0x14
- ALMOSEC=0x00
- ALMOMIN=0x00
- ALMOHOUR=0x10
- ALMOWKDAY=0x20
- RTCSEC=0x80

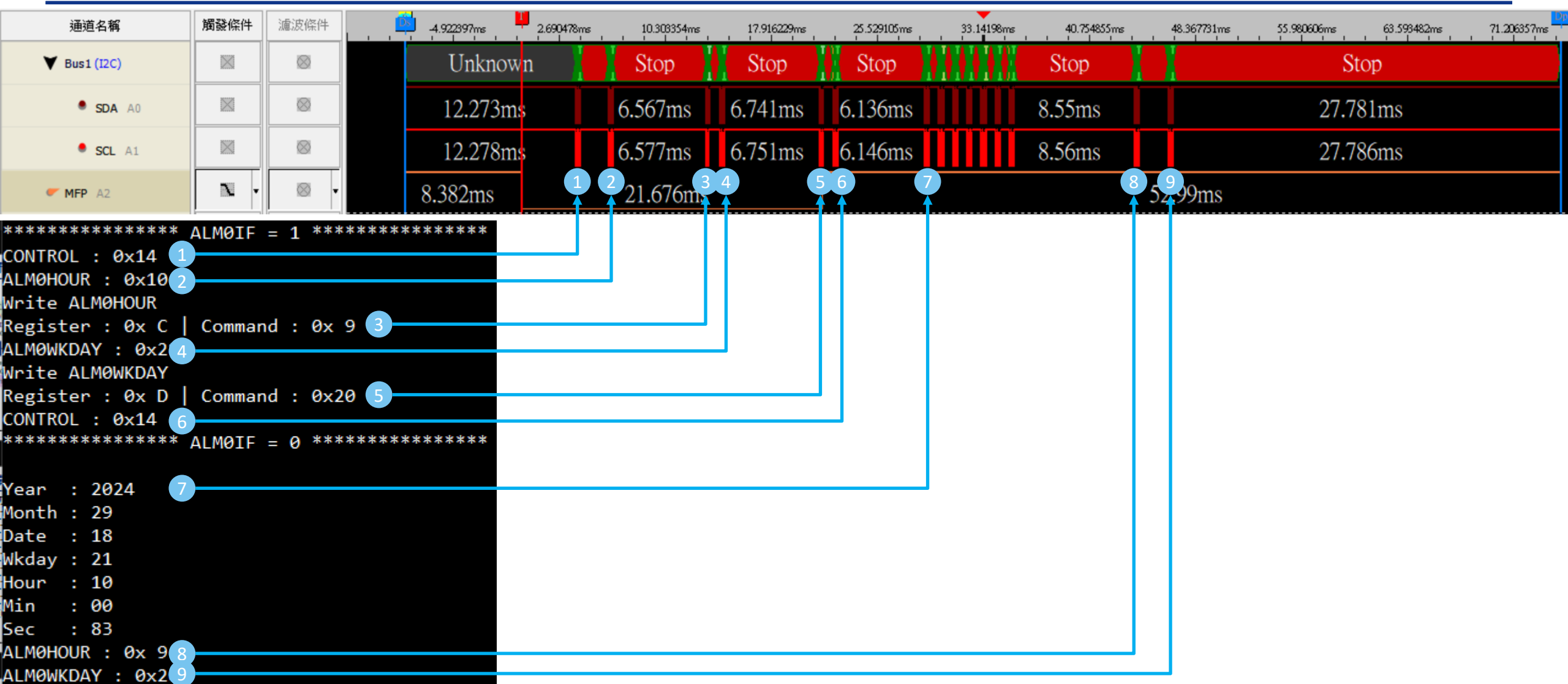
測試流程

- 使用dsPIC33CK256MP506，每秒讀取MCP79410一次，確認粗條模式啟用，等待MFP觸發MCU，並讀取暫存器，約10分鐘內可跑完23小時

封包紀錄

| 封包 | 名稱 | 起始點 | Start | Address | Write | A-ACK | Data | D-ACK | | | |
|----|-----------|------------|-------|---------|-------|-------|------|--------|------|-------------|------|
| 1 | Bus1(I2C) | 3.89079ms | Start | 6F | Write | A-ACK | 07 | D-ACK | | | |
| 2 | Bus1(I2C) | 4.09025ms | Start | 6F | Read | A-ACK | 14 | D-NACK | Stop | Master NACK | |
| 3 | Bus1(I2C) | 6.28412ms | Start | 6F | Write | A-ACK | 0C | D-ACK | | | |
| 4 | Bus1(I2C) | 6.48357ms | Start | 6F | Read | A-ACK | 10 | D-NACK | Stop | Master NACK | |
| 5 | Bus1(I2C) | 13.25091ms | Start | 6F | Write | A-ACK | 0C | D-ACK | 09 | D-ACK | Stop |
| 6 | Bus1(I2C) | 14.26477ms | Start | 6F | Write | A-ACK | 0D | D-ACK | | | |
| 7 | Bus1(I2C) | 14.46427ms | Start | 6F | Read | A-ACK | 28 | D-NACK | Stop | Master NACK | |
| 8 | Bus1(I2C) | 21.406ms | Start | 6F | Write | A-ACK | 0D | D-ACK | 20 | D-ACK | Stop |
| 9 | Bus1(I2C) | 22.41982ms | Start | 6F | Write | A-ACK | 07 | D-ACK | | | |
| 10 | Bus1(I2C) | 22.61933ms | Start | 6F | Read | A-ACK | 14 | D-NACK | Stop | Master NACK | |
| 11 | Bus1(I2C) | 28.95579ms | Start | 6F | Write | A-ACK | 00 | D-ACK | | | |
| 12 | Bus1(I2C) | 29.15526ms | Start | 6F | Read | A-ACK | 83 | D-NACK | Stop | Master NACK | |
| 13 | Bus1(I2C) | 29.97238ms | Start | 6F | Write | A-ACK | 01 | D-ACK | | | |
| 14 | Bus1(I2C) | 30.17176ms | Start | 6F | Read | A-ACK | 00 | D-NACK | Stop | Master NACK | |
| 15 | Bus1(I2C) | 30.98901ms | Start | 6F | Write | A-ACK | 02 | D-ACK | | | |
| 16 | Bus1(I2C) | 31.18843ms | Start | 6F | Read | A-ACK | 10 | D-NACK | Stop | Master NACK | |
| 17 | Bus1(I2C) | 32.0057ms | Start | 6F | Write | A-ACK | 03 | D-ACK | | | |
| 18 | Bus1(I2C) | 32.20512ms | Start | 6F | Read | A-ACK | 23 | D-NACK | Stop | Master NACK | |
| 19 | Bus1(I2C) | 33.02315ms | Start | 6F | Write | A-ACK | 04 | D-ACK | | | |
| 20 | Bus1(I2C) | 33.22271ms | Start | 6F | Read | A-ACK | 18 | D-NACK | Stop | Master NACK | |
| 21 | Bus1(I2C) | 34.03961ms | Start | 6F | Write | A-ACK | 05 | D-ACK | | | |
| 22 | Bus1(I2C) | 34.23898ms | Start | 6F | Read | A-ACK | 29 | D-NACK | Stop | Master NACK | |
| 23 | Bus1(I2C) | 35.05597ms | Start | 6F | Write | A-ACK | 06 | D-ACK | | | |
| 24 | Bus1(I2C) | 35.25542ms | Start | 6F | Read | A-ACK | 24 | D-NACK | Stop | Master NACK | |
| 25 | Bus1(I2C) | 44.00665ms | Start | 6F | Write | A-ACK | 0C | D-ACK | | | |
| 26 | Bus1(I2C) | 44.20609ms | Start | 6F | Read | A-ACK | 09 | D-NACK | Stop | Master NACK | |
| 27 | Bus1(I2C) | 46.48592ms | Start | 6F | Write | A-ACK | 0D | D-ACK | | | |
| 28 | Bus1(I2C) | 46.68535ms | Start | 6F | Read | A-ACK | 20 | D-NACK | Stop | Master NACK | |

封包波形：讀寫過程使用UART print info



封包timing

