

■ 셀 추적을 위한 데이터 저장 폴더 이름, 파일 이름, 파일 형식 규정

목적 : 생성된 데이터 파일을 별도의 프로그램에서 읽고 ECS 로 Ethernet 기반 Socket 통신으로 전달하기 위한 데이터 파일 저장 규칙 서술

작성일 : 2019-10-24

작성자 : 에이아이테크 손호영

Version : 1.0(2019-10-24)

Version : 1.1(2019-10-25)

Version : 1.2(2019-11-04)

Version : 1.3(2019-11-06)

Version : 1.4(2019-11-11)

Version : 1.5(2019-11-19)

Version : 1.6(2020-07-27)

Version : 1.7(2020-08-04)

내용

- 셀 추적을 위한 데이터 저장 폴더 이름, 파일 이름, 파일 형식 규정..... 1
- Directory Rule..... 3
 - ✓ 저장 파일 경로 규칙 : "저장 Root 폴더 이름" + "" + "날짜" + "" 3
- File Name Rule..... 4
 - ✓ 저장 파일 규격 4
 - ✓ 저장 파일 이름 규칙 : "A Name" + "_" + "B Name" + "_" + "날짜" + "_" + "시간".csv 4
- File Save In Fold Rule..... 6
 - ✓ 저장 폴더내부 규정 6
- File Format Rule..... 7
 - ✓ 저장파일 내부 형식 규정 7
- 파일 저장 옵션 방법 예제 코드 10
 - ✓ CSV 파일 저장 함수에서 다른 프로그램에서 읽기 권한을 부여하여야 한다..... 10

■ Directory Rule

✓ 저장 파일 경로 규칙 : “저장 Root 폴더 이름” + “W” + “날짜” + “W”

- a. 저장 Root 폴더 이름(아래 예제에서 'Data')
 - 특수 문자가 포함되지 않은 문자로 구성된 문자열
- b. 날짜(아래 예제에서 '20190905')
 - 문자 개수 : 8자(년도 4자리, 월 2자리, 일 2자리)
 - 형식 : YYYYMMDD
- c. 예: "D:\LGC_Vision\Files\WDataW20190905W"

■ File Name Rule

✓ 저장 파일 규격

a. ASCII기반 text 형식으로 파일 저장

✓ 저장 파일 이름 규칙 : "A Name" + "_" + "B Name" + "_" + "날짜" + "_" + "시간".csv

a. "A Name", "B Name" : 임의의 명칭, 구분자 '_' 를 제외한 문자로 구성(구분자('_')는 총 3개)

b. A Name

- 붙이는 순서

i. 생산공장

1. (오창/중국/폴란드/미국) : OC / CNJ, CNA, CNB / CWA / CMI

ii. 생산라인

1. 노칭 - NTC / 라미 - LAM / 스테킹, 폴딩 - STK, FOL / 패키지 - PKG

2. ESS라인이 아니면

A. 예 : LAM

3. ESS라인이면 : 생산라인명+ESS

A. 예 : LAM**ESS**

iii. 생산라인번호(예 : "1-3"), 구분은 '-'

iv. 예제

1. 오창 라미 14-1호

A. "OCLAM14-1"

2. CNB 스테킹 1-2호

A. "CNBSTK1-2"

3. CNB 라미 **ESS** 1-1호

A. "CNBLAM**ESS**1-1"

c. B Name

- 비전 Data

i. "VISION"

- 쇼트 Data

- i. "SHORT"
- d. 날짜
 - 문자 개수 : **8자**(년도 4자리, 월2자리, 일 2자리)
 - 형식 : YYYYMMDD
- e. 시간
 - 문자 개수 : **2자**(24시간 기준 00~23)
 - 형식 : HH
- f. 파일 확장자
 - **csv**
- g. 예제 파일
 - 예 1: 오창 라미 14-1호에서 2019년 11월 11일 오후 2시에 생성된 비전 Data
 - i. OCLAM14-1_VISION_20191111_14.csv
 - 예 2: CNB 스테킹 1-2호에서 2019년 11월 11일 오후 2시에 생성된 쇼트 Data
 - i. CNBSTK1-2_SHORT_20191111_14.csv
 - 예 3: CNB 라미 **ESS** 1-1호에서 2019년 11월 11일 오후 2시에 생성된 비전 Data
 - i. CNBLAM**ESS**1-1_VISION_20191111_14.csv
- h. **주의 사항**
 - **ESS** 라인의 경우 자동차전지 라인과의 구별을 위해 공정명 다음에 반드시 표기해주시길 바랍니다.

■ File Save In Fold Rule

✓ 저장 폴더내부 규정

- 저장 폴더내의 파일형식은 확장자가 CSV 만 허용한다.
- 하루 24시간, 1시간당 1개의 파일만 생성한다.
 - 예:"A_B_20191104_13.csv", "A_C_20191104_13.csv" ←(X) 2019년11월4일 13시 파일은 2개
 - 예:A_B_20191104_13.csv ←(○) 2019년11월4일 13시 파일은 1개
- 저장 폴더내의 파일은 동일한 데이터를 저장하는 파일만을 허용한다.(아래 그림1 참조)
 - ProcessLog_20190926.txt ←(X)
 - Shortage_20190926.csv ←(X)
 - STATS_USIP11LMC2_20190926.csv, STATS_USIQ11LMC1_20190926.csv, STATS_USIQ11LMC2_20190926.csv ←(X)
 - E63F_CAM#00_20191020_00.csv~E63F_CAM#00_20191020_23.csv ←(○)

이름	수정된 날짜	유형	크기
E63F_CAM#00_20191020_00.csv	2019-10-20 오전 12:21	Microsoft Excel ...	599KB
E63F_CAM#00_20191020_01.csv	2019-10-20 오전 1:59	Microsoft Excel ...	322KB
E63F_CAM#00_20191020_02.csv	2019-10-20 오전 2:53	Microsoft Excel ...	509KB
E63F_CAM#00_20191020_03.csv	2019-10-20 오전 3:54	Microsoft Excel ...	343KB
E63F_CAM#00_20191020_04.csv	2019-10-20 오전 4:57	Microsoft Excel ...	783KB
E63F_CAM#00_20191020_05.csv	2019-10-20 오전 5:59	Microsoft Excel ...	923KB
E63F_CAM#00_20191020_06.csv	2019-10-20 오전 6:18	Microsoft Excel ...	724KB
E63F_CAM#00_20191020_10.csv	2019-10-20 오전 10:59	Microsoft Excel ...	470KB
E63F_CAM#00_20191020_11.csv	2019-10-20 오전 11:59	Microsoft Excel ...	810KB
E63F_CAM#00_20191020_12.csv	2019-10-20 오후 12:37	Microsoft Excel ...	144KB
E63F_CAM#00_20191020_13.csv	2019-10-20 오후 1:28	Microsoft Excel ...	23KB
E63F_CAM#00_20191020_14.csv	2019-10-20 오후 2:53	Microsoft Excel ...	93KB
E63F_CAM#00_20191020_15.csv	2019-10-20 오후 3:26	Microsoft Excel ...	731KB
E63F_CAM#00_20191020_16.csv	2019-10-20 오후 4:56	Microsoft Excel ...	710KB
E63F_CAM#00_20191020_18.csv	2019-10-20 오후 6:58	Microsoft Excel ...	107KB
E63F_CAM#00_20191020_19.csv	2019-10-20 오후 7:38	Microsoft Excel ...	683KB
E63F_CAM#00_20191020_20.csv	2019-10-20 오후 8:04	Microsoft Excel ...	9KB
E63F_CAM#00_20191020_21.csv	2019-10-20 오후 9:59	Microsoft Excel ...	788KB
E63F_CAM#00_20191020_22.csv	2019-10-20 오후 10:55	Microsoft Excel ...	2,345KB
E63F_CAM#00_20191020_23.csv	2019-10-21 오전 12:00	Microsoft Excel ...	1,685KB
ProcessLog_20190926.txt	2019-09-26 오후 11:53	텍스트 문서	1,209KB
Shortage_20190926.csv	2019-09-26 오후 11:53	Microsoft Excel ...	640KB
STATS_USIP11LMC2_20190926.csv	2019-09-26 오전 9:25	Microsoft Excel ...	2KB
STATS_USIQ11LMC1_20190926.csv	2019-09-26 오후 8:21	Microsoft Excel ...	2KB
STATS_USIQ11LMC2_20190926.csv	2019-09-26 오후 11:53	Microsoft Excel ...	2KB

그림 1. 폴더 내 파일 저장 예제

■ File Format Rule

✓ 저장파일 내부 형식 규정

a. 파일이 신규로 생성되면 1번째 줄에 항목 이름을 **구분자(Comma',')로 분리하여 모두 명기**한다.

- Title 줄이라고 명칭한다.
- Title 줄의 첫번째, 두번째 글자는 “//” 문자를 반드시 붙여야 한다.

i. 예 : No,Data,Time ◀(X) ⇔ //No,Data,Time ◀(○)

- Title 줄은 **항목 명칭 변경, 개수 변경, 항목 위치 변경**이 발생하면 Title 줄을 추가한다.

//No,Date,Time,Model,Refer-Result,Result,Lot-ID,Cell-ID,Position,Voltage

000001,20191028,230000,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000

000002,20191028,230000,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000

000003,20191028,230000,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000

//No,Date,Time,Refer-Result,Model,Result,Lot-ID,Cell-ID,Position,Voltage

000004,20191028,230000,OK,E63-C-TW-NEW,OK,NONE,NONE,Rear,0.0000

000005,20191028,230000,OK,E63-C-TW-NEW,OK,NONE,NONE,Front,0.0000

000006,20191028,230000,OK,E63-C-TW-NEW,OK,NONE,NONE,Rear,0.0000

000007,20191028,230002,OK,E63-C-TW-NEW,OK,NONE,NONE,Front,0.0000

//No,Date,Time,Model,Refer-Result,Result,Lot-ID,Cell-ID,Position,Voltage,Judge

000008,20191028,230002,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000,OK

000009,20191028,230002,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000,OK

000010,20191028,230002,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000,NG

000011,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000,NG

//No,Date,Time,Model,Refer-Result,Result,Lot-ID,Cell-ID,Position,Voltage

000012,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000

000013,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000

000014,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000

000015,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Front,0.0000

000016,20191028,230004,E63-C-TW-NEW,OK,OK,NONE,NONE,Rear,0.0000

i. 첫번째 Title 줄

1. 항목 개수는 10개

ii. 두번째 Title 줄(첫번째 Title 줄(i)과 비교)

1. 항목 개수(10)는 일치, 항목 순서가 변경
2. “Model,Refer-Result → Refer-Result,Model”

iii. 세번째 Title 줄(두번째 Title 줄(ii)과 비교)

1. **항목 개수 증가(10→11)**
2. “Judge” 항목이 추가

iv. 네번째 Title 줄(세번째 Title 줄(iii)과 비교)

1. **항목 개수 감소(11→10)**
2. “Judge” 항목이 제거

b. 데이터는 2번째 줄부터 **구분자(Comma ',')로 분리하여 모두 명기**한다. 단 Title 줄이 추가되면 Title 줄 다음부터 데이터 문자열을 추가한다.

- Value 줄이라고 명칭한다.

- Value 줄의 첫번째와 두번째 문자위치에 "//" 문자를 붙이지 않는다.

i. 예 : //1,20191103,140311 ←(X) ⇔ 1,20191103,140311←(○)

c. Value 항목 개수와 Title줄의 **항목 개수는 일치**한다.

- 예: //No,Data,Time : Title 항목 개수 3개

01,20191118,113710 : Value 항목 개수 3개 ← (○)

// No,Data,Time : Title 항목 개수 3개

01,20191118,113710,1, : Value 항목 개수 5개(구분자(',') 개수(4) + 1 = 5) ← (X)

d. 항목과 구분자(',') 간에 **공백문자는 제외**한다. 또한 **문자열의 마지막에 공백문자**는 없어야 한다.

- 단, []을 공백문자로 표현하였다.

- 예: //Name,[Date][Time][High Voltage,[Path] ←(X) ⇔ //Name,Date,Time,High[Voltage,Path ←(○)

- 예: Mono,[20190308,[230109[3.32,[d:₩Mono ←(X) ⇔ Mono,20190308,230109,3.32,d:₩Mono ←(○)

e. 항목 글자 내부의 공백문자([])는 허용한다.

- 예: High[Voltage ←(○)

f. Title 줄에 **"Date"**와 **"Time"** 항목이 반드시 있어야 한다.

- 예: //Name,Date,High Voltage,Path ←(X) ⇔ //Name,**Date,Time**,High Voltage,Path ←(○)

- 예:// Name,Time,High Voltage,Path ←(X)

- 예:// Name,High Voltage,Path ←(X)

g. Value 줄에 데이터가 생성된 시간이 명기된 데이터가 반드시 있어야 한다.

- 문자수 : **6자**(시간 2자리, 분 2자리, 초 2자리), 단 시간은 24시간 기준

- 예 : 오후 5시 3분 20초

i. 17320 ←(X) ⇔ 170320 ←(○)

ii. 17:3:20, 17:03:20 ←(X)

h. Value 줄에 데이터가 생성된 날짜가 명기된 데이터가 반드시 있어야 한다.

- 자수 : **8자**(년도 4자리, 월 2자리, 일 2자리)

- 예 : 2019년 5월 9일

i. 190509 ←(X) ⇔ 20190509 ←(○)

ii. 19-5-9, 19-05-09, 190509, 201959, 2019-05-09 ←(X)

i. Value 줄에 데이터가 없는 경우에 **공백문자([])**나 **문자없음** 상태로 저장하지 않고 **디폴드 문자(상위 보고를 받는 담당자와 협의)**를 지정하여 저장한다.

- 예: //Mono, , 230109, 3.32, d:W Mono ←(X)
- 예: //Mono,, 230109, 3.32, d:W Mono ←(X)
- 예: //Mono, **Default**, 3.32, d:W Mono ←(○) 단: "Default"는 예제로 사용한 문자열이다.

j. Title 줄과 Value 줄의 끝에 별도의 문자를 붙이지 않는다.(아래 예제에서는 ' ' ' ')

- 예: //Name,Date,Time,High Voltage,Path; ⚡ (X) ↔ //Name,Date,Time,High Voltage,Path ⚡ (○)
- 예: //Name,Date,Time,High Voltage,Path,, ⚡ (X) ↔ //Name,Date,Time,High Voltage,Path ⚡ (○)

■ 파일 저장 옵션 방법 예제 코드

✓ CSV 파일 저장 함수에서 다른 프로그램에서 읽기 권한을 부여하여야 한다.

- 예제 코드의 CreateFile 함수내의 “[FILE_SHARE_READ](#)” 참조
- 아래 예제 코드에서 m_hFile 선언과 초기화
 - i. 선언 : HANDLE m_hFile;
 - ii. 초기화 : m_hFile = INVALID_HANDLE_VALUE;

```
bool CMFCApplicationIDlg::SaveFile(void)
{
    int nLength;
    DWORD nWritten;
    CStringA strA;
    CString strFoldFileName = _T("D:₩₩Data₩₩20191104₩₩A Name_B Name_20191104_11.csv");

    //파일 Open or Create 처리
    m_hFile = ::CreateFile(strFoldFileName, GENERIC_WRITE, FILE_SHARE_READ, NULL, OPEN_ALWAYS, FILE_ATTRIBUTE_NORMAL, NULL);

    //파일 Open 성공여부 확인
    if (INVALID_HANDLE_VALUE != m_hFile)
    {
        //파일의 끝으로 이동
        ::SetFilePointer(m_hFile, 0, NULL, FILE_END);

        //한줄 단위로 저장하기 위해서는 문자열의 끝은 "₩r₩n" 을 붙여야 한다.
        strA = "Data_1,Data_2,Data_3₩r₩n";

        //문자열의 길이값을 구한다.
        nLength = strA.GetLength();

        //파일에 한줄 Writing 한다.
        ::WriteFile(m_hFile, strA.GetString(), nLength, &nWritten, NULL);

        //파일 핸들을 닫는다.
        ::CloseHandle(m_hFile);
        m_hFile = INVALID_HANDLE_VALUE;
        return true;
    }
    else
    {
        AfxMessageBox(_T("Error Save"));
        return false;
    }
    return false;
}
```