

These were generated by a test program rather than the licor instrument; the measurement data in them is invalid.

Structurally, they are valid records; fields are delimited with commas, records are delimited by new line (aka linefeed/LF, ASCII 10d/0x0A).

Records are variable length, consisting of a header and one or more measurements. The header indicates the record structure, record type and how many measurements the record contains.

After taking a look at some records of different sizes, one will be broken down down to see what's inside.

Log record with one measurement:

1585958597000,0001,0002,282,256,1,1585958577000,1585958597000,0003,0.50,7,+1.
000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.
165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.
000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.
165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+8.165e-01,+1.000e+00,+2.000e+00,+3.000e+00,+8.165e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+8.165e-01,+0.000e+00,+1.000e+01,+2.000e+01,+3.000e+01,+4.
000e+01,+5.000e+01,+1.250e+00,-3.450e+01,+1.562e+01

Log record with two measurements:

1585958897000,0001,0003,283,512,2,1585958877000,1585958897000,0004,0.50,8,+1.
000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.
559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.
000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.
559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+7.559e-01,+0.000e+00,+1.000e+01,+2.000e+01,+3.000e+01,+4.
000e+01,+5.000e+01,+1.250e+00,-3.450e+01,+1.562e+01,1585958877000,15859588970
00,0004,0.50,8,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e
+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e
+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e
-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e
+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e
+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e
+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e
-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+0.000e+00,+1.000e+01,+2.000e
+01,+3.000e+01,+4.000e+01,+5.000e+01,+1.250e+00,-3.450e+01,+1.562e+01

Log record with six measurements:

1585960097000,0001,0003,287,1536,6,1585960077000,1585960097000,0008,0.50,12,+

[illegible]

header data is shown in red
measurement 1 is shown in blue
measurement 2 is shown in green

1585958897000,0001,0003,283,512,2,1585958877000,1585958897000,0004,0.50,8,+1.
000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.
559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.
000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.
559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.
000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.
000e+00,+3.000e+00,+7.559e-01,+0.000e+00,+1.000e+01,+2.000e+01,+3.000e+01,+4.
000e+01,+5.000e+01,+1.250e+00,-3.450e+01,+1.562e+01,1585958877000,15859588970
00,0004,0.50,8,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e
+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e
+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e
-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e
+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e
+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e
+00,+2.000e+00,+3.000e+00,+7.559e-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e
-01,+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01,+0.000e+00,+1.000e+01,+2.000e
+01,+3.000e+01,+4.000e+01,+5.000e+01,+1.250e+00,-3.450e+01,+1.562e+01

```
6 header fields
66 fields for each of 2 measurements
```

Record headers always have 6 fields.

Timestamp values represent time in epoch milliseconds, i.e. milliseconds since 00:00

Jan 01, 1970 UTC.

Record ID indicates the physical structure (format version) of the record; it is intended to allow future record types that are organized differently. There is currently only one record type.

Measurement ID indicates the logical type of record, i.e. sample, zcal, scal, etc. There are several valid record types, including user-defined.

The sequence number is intended to indicate the order in which records were put entered into the log and provide a means of checking for certain types of errors or system problems. Sequence numbers span multiple log files; they should increase monotonically.

Measurement Fields

Though the number of measurements in a record may vary, each measurement has 66 fields.

- 5 metadata fields
- 52 data fields min, max, mean and standard deviation for 13 licor parameters
- 6 extra fields for CO2 peak, area, trend line calculations
- 3 CO2 trend line parameters

The example record has two measurements; the autogenerated data field values are designed to be able to easily distinguish them, but they are the same for both measurements

Field	Name	Description.	Type	Value
6	timestamp0	measurement start time	uint64	
1585958877000				
7	timestamp1	measurement end time	uint64	
1585958897000				
8	meas ID	measurement type	uint16	0004
9	period	Licor output rate (Hz)	float	0.50
10	stat_n	Licor samples	uint16	8
11:14	co2	min,max,mean,sdev	float	
+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01				
15:18	co2abs	min,max,mean,sdev	float	
19:22	h2odewpoint	min,max,mean,sdev	float	
23:26	h2oabs	min,max,mean,sdev	float	
27:30	celltemp	min,max,mean,sdev	float	
31:34	cellpres	min,max,mean,sdev	float	

35:38	ivolt	min,max,mean,sdev	float
39:42	flowrate	min,max,mean,sdev	float
43:46	rawco2	min,max,mean,sdev	float
47:50	rawco2ref	min,max,mean,sdev	float
51:54	rawh2o	min,max,mean,sdev	float
55:58	rawh2oref	min,max,mean,sdev	float
59:62	co2	min,max,mean,sdev	float
63	X[0]	co2_area	float +0.000e+00
64	X[1]	co2_ep0	float +1.000e+01
65	X[2]	co2_ep1	float +2.000e+01
66	X[3]	MX	float +3.000e+01
67	X[4]	SX	float +4.000e+01
68	X[5]	SXY	float +5.000e+01
69	m.	CO2 trendline slope	float +1.250e+00
70	b	CO2 trendline offset	float -3.450e+01
71	r^2	CO2 trendline R^2	float +1.562e+01
72	timestamp0	measurement start time	uint64
1585958877000			
73	timestamp1	measurement end time	uint64
1585958897000			
74	meas ID	measurement type	uint16 0004
75	period	Licor output rate (Hz)	float 0.50
76	stat_n	Licor samples	uint16 8
77:80	co2	min,max,mean,sdev	float
+1.000e+00,+2.000e+00,+3.000e+00,+7.559e-01			
81:84	co2abs	min,max,mean,sdev	float for this
example, values are match measurement 1...			
85:88	h2odewpoint	min,max,mean,sdev	float
89:92	h2oabs	min,max,mean,sdev	float
93:96	celltemp	min,max,mean,sdev	float
97:100	cellpres	min,max,mean,sdev	float
101:104	ivolt	min,max,mean,sdev	float
105:108	flowrate	min,max,mean,sdev	float
109:112	rawco2	min,max,mean,sdev	float
113:116	rawco2ref	min,max,mean,sdev	float
117:120	rawh2o	min,max,mean,sdev	float
121:124	rawh2oref	min,max,mean,sdev	float
125:128	co2	min,max,mean,sdev	float
129	X[0]	co2_area	float
130	X[1]	co2_ep0	float
131	X[2]	co2_ep1	float
132	X[3]	MX	float
133	X[4]	SX	float
134	X[5]	SXY	float
135	m.	CO2 trendline slope	float
136	b	CO2 trendline offset	float

137

r^2

CO2 trendline R^2

float

Reading strategy

I'm guessing that an array of record structures would be used to manipulate the data in Matlab.

Each record structure would include a header and an (variable length) array of measurements.

A Matlab CSV record loader could use a pattern like this:

```
initialize data array
```

```
open file
```

```
while not end-of-file
```

```
    initialize a new record structure
```

```
    read line
```

```
    parse header fields into record structure
```

```
    for 0 < measurement < header.<meas_count>
```

```
        parse measurement into record measurement array
```

```
    add record structure to data array
```

```
end
```

```
close file
```