SOIL DRYNESS INDEX SYSTEM

VER 1

USER GUIDE

F H Yung & M Williams

2-6-1992
22-7-1992 updated

[Development status : phase 2, prototype.]

1. Introduction:

The system accepts historical data of daily total rainfall and maximum temperature from the input file, or from the keyboard, each line stands for a day's data.

The system has two modes of operation: the first start from an SDI value of the first day and calculates the SDI values for all the lines in the input file using the method of N D Burrows (1987). It writes the output file, which can in turn be used as the input in subsequent runs. It contains all the data in the input file plus the calculated data, one line per day.

The second mode updates an existing file of SDI data, with additional daily data entered from the keyboard.

This system was written in Lahey Fortran and is in compiled form. Customization requirements by each district can also be included at installation.

User must maintain his own periodic data backup systematically.

2. Method of SDI calculations :

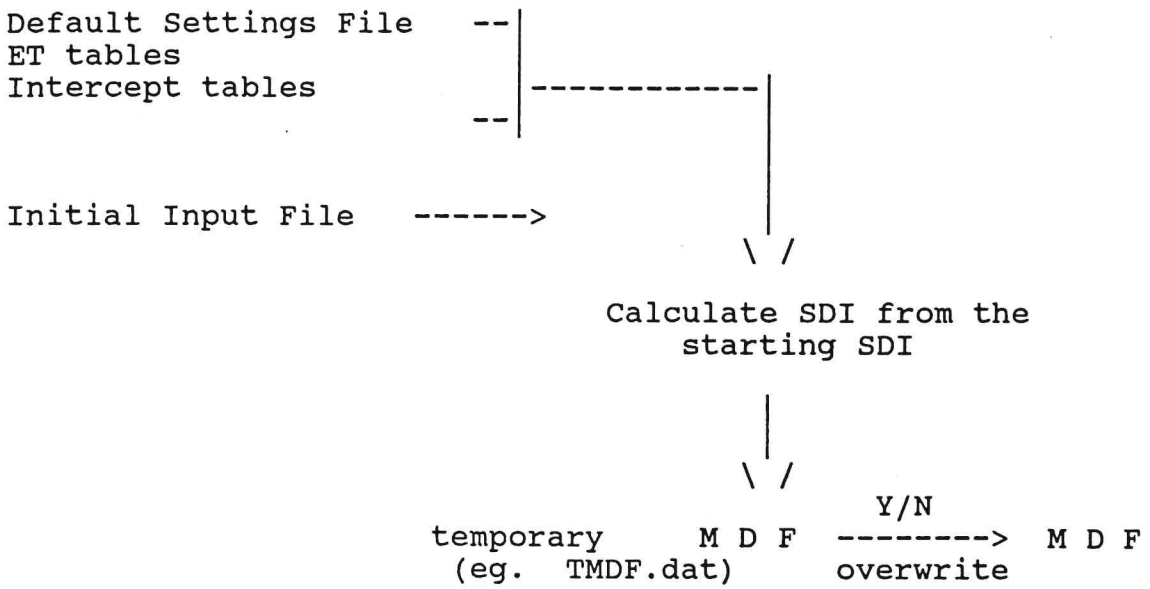
The method is identical to that in the publication by Burrows N D (CALM technical report, No 17, July 1987) and is not duplicated here.

In the program code, there are comments explaining what each variable and array is, in the terms used by the publication.

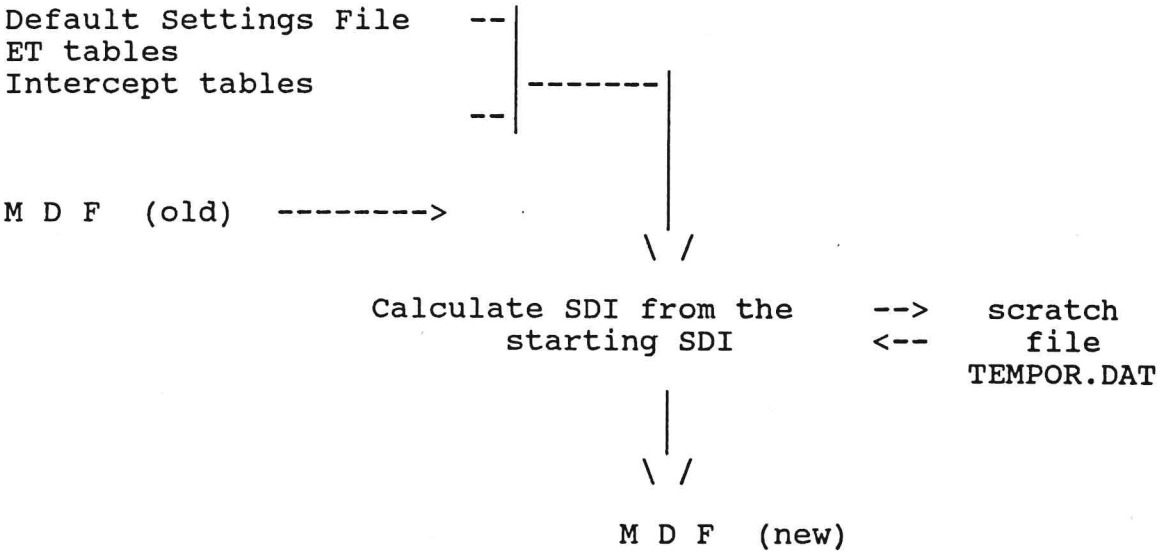
3. Program modules and data flow diagram

Options 1 & 2 correspond to two separate but similar program modules.

The data flow diagram of option 1 :



The data flow diagram of option 2:



In option 2, the program does not update the MDF directly, but copies it to a scratch file, unit 9, first. Then it prompts the user to enter one or more lines of data and appends them to this scratch file, which is then treated as the Initial Input File, from which data is read-in. During calculations it writes out a new MDF. The former MDF is then deleted.

At a program crash or power failure, you can recover the data from the scratch file tempor.dat, without reloading the MDF from backup.

4. Start run :

Get into the subdirectory that contains this system \SDI say.

At the DOS prompt, key-in

\SDI:> SDIMENU <CR>

This version starts with a User Menu showing the options, as follows.

User can also choose to use his own Evaportranspiration table.

5. Menu Options :

Option 1 :

It reads-in the Initial Input File, eg. test.dat, and for each line it calculates the SDI and a few other values listed in A2 below.

It prompts you to give a DOS filename, eg. TMDF.dat, which we shall call the Temporary Master Data File (TMDF).

* When the calculation is completed, you will be prompted to overwrite the current MDF if you like.

(**Caution:** If you overwrite the MDF with the new TMDF, you will end up losing all the extra days you might have entered in the current MDF.)

If you chose not to overwrite, the TMDF will be saved.

Option 1 must be the first run immediately after the installation of the system to generate the MDF for subsequent runs.

This option should also be run after a correction of errors on the MDF as all the SDI values must be recalculated.

All date validations in Option 4 are done here. Warning error messages appear on your screen.

Option 2 :

This option uses the MDF instead of the Initial Input File.

It reads-in the MDF, for instance which may contain data up to yesterday or a few days before, then it prompts you to enter the line of data for the next day until to-day's date. Then it executes the program. As you enter the date, this program checks that the Day value is sequentially increasing. If this is not so, you will see a warning message.

Only minor date validation is provided here. If you want to validate your dates more extensively, please use Option 4.

This is the most commonly used option for the daily SDI calculation.

Please also note that you can key-in more than one line of data. For example, after a weekend or a long weekend, you may have available only the total rainfall in this period. You can key-in a line of data for each day so to distribute the total rainfall in the manner you see most fit.

Option 3 :

You can modify the Default Settings File only here, by using any ASCII editor and then save this file. See section A1.3 of appendix 1 below for the fields and their formats.

Option 4 :

This option is for validating the dates in the Initial Input File, the MDF or any file in the same format.

It is not yet an exhaustive algorithm, but it is already quite extensive and is described in section A1.5 of appendix 1 below, for the programmer.

You are prompted to enter the filename you wish to validate and the filename you like the error messages to be written. The latter will be called the Date Error File.

After the run, you must use an ASCII editor to correct the errors.

6. Future version to include :

- plotting of the SDI
- interpolation of the evapotranspiration data
- double validate any change of data by the user - for data integrity reasons.
- default procecure to handle missing data.

*

7. Notes to the programmer:

7.1 Program design

Options 1 & 2 correspond to two separate but similar programs.

In the corresponding programs for options 1 & 2, the codes in the dotted bracket labeled as for the preparation of the Default Settings File must be kept identical.

This also applies to the dotted bracket labeled as "to read & display Default Settings".

Option 4 is specially written for doing date validations on a file of the same format as the MDF. It consists of a main program CHKDATEn that calls the subroutine VDATE, which does the validations and issue warning messages. Thus, when you generate the executable file with extension .exe, you must use eg.

```
C:\> OPTLINK CHKDATE4 + VDATE1;
```

Only minor date validations is done in Option 1.

Option 2 also calls the subroutine VDATE to do the date validations. Thus to generate the executable file, you also must use eg.

```
C:\> OPTLINK SDI20 + VDATE1;
```

7.2 Program compilation

For PC's without a Maths Coprocessor, you must generate the executable file .exe by compiling with the /E option.

7.3 System installation

You only have to copy all the relevant files into a subdirectory and revise the Defaults Settings File.

Appendix 1

A1.1 Initial Input File format :

This is in free format, in which a space is used as the delimiter between each data value, one line of data per day as follows :

Year	month	day	total daily rainfall	max temperature	
(I4)	(I2)	(I2)	(F6.2)	(F4.1)	
1980	12	2	0	22.3	(eg)

A1.2 Output data file format :

Its format is the same as the input file above except that its records are longer containing the following additional calculated values :

I	effective rainfall	evapo-transpiration	amSDI	pmSDI
(I2)	(F6.1)	(I2)	(F6.1)	(F6.1)

A1.3 Default Settings File format :

After column 15, it is just comments for your editing convenience. eg

```
1500            (starting value for  amSDI,  <  5 digits)
   0            (rain / no rain, on the day before start of
                                        data= 1 / 0 )
collie.dat      (the Initial Input Filename, < 16 characters )
collie.out      (the Master Data Filename,   < 16 characters )
   1            (forest type : 1 - 4 )
   1            (place : 1 - 6 )
```

A1.4 Test data sets.

Initial Input File

```
1980 1 1  0 23.4  0  0.0  0  0.0  0.0
1980 1 1  0 22.5  0  0.0  0  0.0  0.0
1980 1 1 12 21.0  0  0.0  0  0.0  0.0
1980 1 1  0 0     0  0.0  0  0.0  0.0
1981 1 1  0 46.9  0  0.0  0  0.0  0.0
```

Master Data File

```
1980  1  1   0.00 23.4  2    0.0  8 1500.0 1508.0
1980  1  1   0.00 22.5  2    0.0  8 1508.0 1516.0
1980  1  1  12.00 21.0 13  107.0  8 1409.0 1417.0
1980  1  1   0.00  0.0 13  107.0  8 1409.0 1417.0
1981  1  1   0.00 46.9  2    0.0 17 1417.0 1434.0
```

A1.5 Validation of the Date values:

The date format is required to be "YYYY MM DD".
The Date value in the first line is validated only for its Month and Day values; its Year value is assumed correct.

Then it validates the subsequent dates with respect to the starting date, for consistency only, as the program has not got an inbuilt calendar as in packages like SAS. It igmores leap years.

For each Date, it first tests for the consistency between the Year and the Month value with respect to the previous Date, then it tests the Day value.

All modules of the F77 codes used for these tests are coded sequentially one after the other in the program to minimize confusion when we add further codes in future.

The conditions that will trigger an error or warning message written to your Date Error File are :

For Year-Month value validation:

- year decreased.
- month not within [1, 12].
- month not sequentially increasing.
- year changed, but month not changed.
- year changed, but not from December to January.

For Day value validation:

- day out of the valid range of that month.
- date identical to the previous date.
- day repeated.

- not: day increases sequentially until end of month
  & not like : 1980 1 31
  1980 2 1

  & not like : 1980 12 31
  1980 1 1

REFERENCES

Burrows N D, CALM technical report, **No 17**, July 1987.

(end)