ser ring notations may be most suitable. Again, since there is undoubted advantage in having ring systems uniquely identified in the notation for search purposes, the Hyde ring descriptions could possibly be developed from canonical connection table descriptions of the ring systems alone.

Further alternative approaches to ring systems exist, of course, including that described in this Symposium by Bowman and his colleagues,[1] or again, table-lookup of canonical Wiswesser ring notations from canonical CT ring descriptions, which might be useful in specialized collections.

To conclude, it would appear that the future provides scope for many alternative schemes for using notations. Hopefully, adequate means of interconverting them can also be provided.

## ACKNOWLEDGMENT

Financial support from the office for Scientific and Technical Information, London, and assistance provided by Miss Janet Armitage in the preparation of the program are gratefully acknowledged. Thanks are also due to the National Science Foundation for providing funds for attendance at the Symposium on Notation Systems.

## LITERATURE CITED

(1) Bowman, C. M., F. A. Landee, N. W. Lee, and M. H. Reslock, J. Chem. Doc. **8**, 000 (1968).

(2) Bragg, J., M. F. Lynch, J. Orton, and W. G. Town, (in preparation).

(3) Conrow, K., J. Chem. Doc. **6**, 206 (1966).

(4) Cossum, W. E., M. E. Hardenbrook, and R. N. Wolfe, *Proc. Amer. Doc. Inst.* **1**, 270 (1964).

(5) Dyson, G. M., W. E. Cossum, M. F. Lynch, and H. L. Morgan, *Information Storage Retrieval* **1**, 69 (1963).

(6) Dyson, G. M., *Information Storage Retrieval* **2**, 59 (1964).

(7) Dyson, G. M., W. E. Cossum, M. F. Lynch, and H. L. Morgan, "Mechanical Encipherment of Chemical Ring Structures from the Random Matrix," in H. P. Luhn, Ed., *Automation and Scientific Communication* American Documentation Institute, Washington, D. C., 1963.

(8) Feldman, A., D. B. Holland, and D. P. Jacobus, *ibid.* **3**, 187 (1963).

(9) Garfield, E., *Nature* **192**, 192 (1961).

(10) Hayward, H. W., *Pat. Off. Res. Dev. Rept.* No. 21, U.S. Patent Office, Washington, D. C., 1961.

(11) Kulpinski, S., *et al.*, 4 "A Study and Implementation of Mechanical Translation from WLN to CT," Vol. 1, Ann. Rept. on Contract NSF C-467, University of Pennsylvania, 1967.

(12) Lefkovitz, D., J. Chem. Doc. **7**, 186 (1967).

(13) Lefkovitz, D., *ibid.* **7**, 192 (1967).

(14) Mullen, J. M., *ibid.* **7**, 88 (1967).

(15) Opler, A., *Am. Doc.* **10**, 59 (1958).

(16) "Rules for IUPAC Notations for Organic Compounds," Longmans, Green and Co., London, 1961.

(17) Seifer, A. L., *Inform. Storage Retrieval.* **1**, 29 (1963).

(18) Tauber, S. J., *et al.*, "Chemical Structures as Information," in *Technical Preconditions for Retrieval Center Operations*, B. F. Cheydleur, Ed., Spartan Books Inc., Washington, D. C., 1965.

(19) Tauber, S. J., *et al.*, *Natl. Bur. Stds. Rept.* No. 9587, N.B.S., Washington, D. C., 1967.

(20) Tauber, S. J., *loc. cit.*

(21) Thomson, L. H., E. Hyde, and F. W. Matthews, *ibid.* **7**, 204 (1967).

(22) Tsukerman, A. M., and A. P. Terentiev, Proc. Intern. Conf. on Standards for a Common Language for Machine Searching and Translation **1**, 493, Interscience Press, 1960.

(23) Vander Stouw, G. G., I. Naznitsky, and J. E. Rush, J. Chem. Doc. **7**, 165 (1967).

(24) Weizenbaum, J., *Comm. ACM.* **6**, 524 (1963).

(25) Wiswesser, W. J., "Line Formula Chemical Notation," E. G. Smith, Ed., mimeographed, 1965.

# A Chemically Oriented Information Storage and Retrieval System. II. Computer Generation of the Wiswesser Notations of Complex Polycyclic Structures*

CARLOS M. BOWMAN, FRANC A. LANDEE, NANCY W. LEE, and MARY H. RESLOCK
Computation Research Laboratory, The Dow Chemical
Company, Midland, Michigan 48640

Received March 26, 1968

**A computer program has been written to generate the canonical Wiswesser notation for complex polycyclic structures. The program accepts as input the connection between all the ring atoms and then selects the path which conforms to the notation rules. The operation of the program is described.**

A computer-based system has been devised to handle chemically oriented data and information.[2] The system is based on the revised Wiswesser Line Formula Notation.[4] The file organization and methods used to detect notation errors have been described earlier.[1] This paper discusses the difficulties of correctly encoding complex polycyclic structures and a computer program which was written to generate automatically and consistently the canonical notation for these structures.

The notation for a polycyclic structure is obtained by choosing a path through the network which will satisfy

*Presented before the Division of Chemical Literature, Symposium on Chemical Notations, 155th National Meeting of the American Chemical Society, San Francisco, Calif., April 4, 1968.

a series of hierarchical rules. For a given structure, there is one and only one correct path, and this unique path is not always readily determined. A discussion of the rules of precedence is presented first, and this is followed by a description of the computer program which selects the canonical path.

## NOTATION RULES

To understand the applicable notation rules, it will be necessary to define the meaning of certain terms used in the rules. Each atom or point in a polycyclic structure can be classified according to (1) the number of ring atoms to which it is attached, and (2) the number of rings in which it occurs. No special consideration is given here to an atom which is found in only one ring. The different types of atoms are listed in Table 1.

The basic hierarchical steps to be followed in denoting polycyclic structures are found in rules 30, 31, 32, and 43 of the revised Wiswesser Notation.[4] The first step involves finding a continuous path through the network which meets the requirements of rule 30. A continuous path is attained by assigning consecutive letter locants to the atoms in the ring system in such a way as to achieve the longest possible chain of consecutive ring locants. In some structures several such paths may be found, whereas in others it is impossible to find a continuous path which passes through each one of the atoms only once. When such a situation occurs, it is necessary to show one or more atoms as branches of the main chain.

There are, of course, restrictions and rules of precedence which permit the selection of only one path for a given ring system. It is the application of these restrictions which becomes difficult when dealing with complicated structures.

Before discussing these restrictions it is necessary to define more terms. A fused ring junction has been defined as the connection between two atoms which are jointly shared by only two rings. Thus in Figure 1, the connection between atoms *a* and *f* is defined as a fused ring junction. The fusion locant is the earliest (alphabetic) locant in each ring of a multicyclic structure. In this case, the fusion locant is *a*.

A multicyclic point is defined as a ring atom which is attached to three or more ring atoms and which is shared by three or more rings. In Figure 2, the locant *a* is a multicyclic point.

A perifused junction is defined as a bond which radiates from a multicyclic point and is part of two rings. In Figure 2, the junctions *ab*, *ae*, and *ai* are perifused junctions.

Table I. Types of Atoms or Points in a Polycyclic Structure

| Type of Atom or Point | Number of Rings | Number of Attachments |
|---|---|---|
| Bridged | 2 | 2 |
| Fused | 2 | 3 |
| Spiro | 2 | 4 |
| Bridged | 3 | 2 |
| Multicyclic | 3 | 3 |
| Spiro | 3 | 4 |
| Multicyclic | ≧4 | ≧4 |
| Spiro | ≧4 | ≧4 |

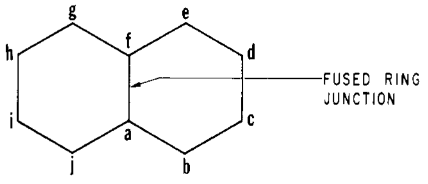


Figure 1. Example of a fused ring junction

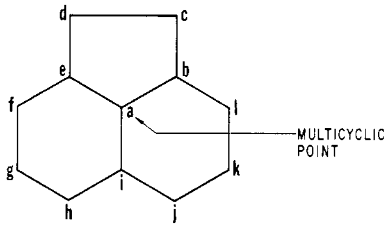


Figure 2. Example of a perifused ring

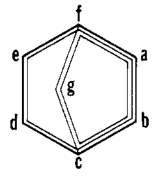
The chosen path must cross no fused ring junction and, according to hierarchical notation rules, it must:

a. cite the smallest rings present;
b. cite the fewest rings necessary to define the structure completely;
c. have the fewest branch ring locants, all of which must be citeable;
d. have the lowest sum of fusion locants cited;
e. have the earliest alphabetic set of fusion locants in the order of their appearance in the notation;
f. have the earliest set of notation symbols for denoting bridges and nonconsecutive locant pairs;
g. have the earliest sequence of ring numerals.

There are other considerations which must be made, but they are dependent on the state of saturation of the rings as well as the substitution of hetero atoms in the ring and will not be considered in this discussion.

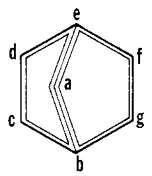
These rules of precedence can best be understood by examining the following examples:

Example 1



6,5

I*a*



5,5

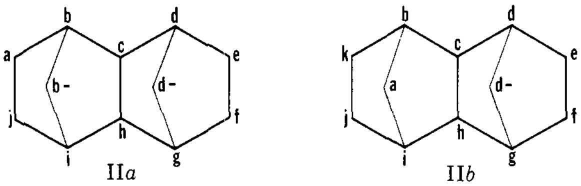
I*b*

*PREFERRED*

SMALLEST RINGS

This structure can be described as a combination of two five-membered rings or a combination of a six- and a five-membered ring. If the path shown in I*a* were followed, a six-membered ring (*abcdef*) would be defined first, followed by the five-membered ring (*abcgf*). The path shown in I*b* defines two five-membered rings (*abcde*) and (*abgfe*). This latter path is preferred because it cites the smallest rings present.

Example 2



IIa

IIb

*PREFERRED*

**MINIMUM NUMBER OF BRANCHED POINTS**

The path in IIa correctly defines the smallest rings. It has two branched points, *b-* and *d-*. The second path, IIb, also defines the smallest rings, but it has only one branched locant; therefore, it is the preferred path. Similar examples could be cited for the other rules, but the purpose of this paper is not to describe or teach the notation rules.

## INPUT

The first problem to be considered in the description of the program is the manner in which the structure will be presented to the machine. Several alternatives were considered, such as a modified notation or a connection table, but these were discarded as being too involved and error-prone. A polycyclic network can be unambiguously represented by finding the longest continuous path which passes through each point only once and then simply naming those points which are attached to points not directly before them or after them in the continuous lettered series. The example in Figure 3 illustrates this point. For this structure an arbitrary path is chosen. The first point (*a*) in the path has been assigned to the lower left-hand atom in the five-membered ring. The path moves clockwise from this point around the perimeter of the entire structure and back to the starting point. All the points in the network have been assigned a place in the path and no point has been mentioned more than once. The next step is to locate the so-called nonconsecutive locant pairs and these are: *ai*, *am*, *dm*, and *ej*. The nonconsecutive locant pairs simply indicate the connections between the atoms in the path which are not implied in the path. It is interesting to note that these four pairs give all the information needed for an unambiguous description of this ring structure. The structure can be reconstructed by building a chain which has 13 points and then attaching the first point to the ninth (*ai*) and last points (*am*) and by creating a link between the fourth and last (*dm*) and the fifth and tenth points (*ej*).

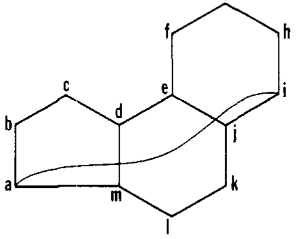


Figure 3. Path assignment for computer input

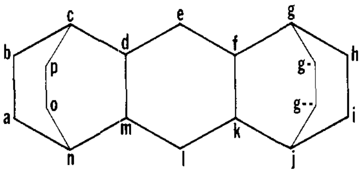


Figure 4. Example of branched chain

This representation is not unique, but it need not be so, because this information will be manipulated by the computer program to arrive at the unique path according to the notation rules of precedence. Any other point could have been chosen as a starting point, and a different path and set of nonconsecutive locant pairs would have resulted. Any one of these is equally acceptable to the program.

There are, of course, structures in which a continuous path which includes all the atoms in the network cannot be found. In such cases it is necessary to designate some points as branching points. The branching points are indicated by adding a dash to the earliest locant to which the branching atom is connected in the main chain. Figure 4 is an example of such a situation.

There is no way in which a path can be assigned to this structure so that all the points in the network are included in the main path and the points mentioned only once. In this case there are two atoms, the two in the bridge in the six-membered ring at the right, which must be considered as branching atoms. They are designated as *g-* and *g--* to show that the branch comes from the point *g* in the main chain. The single hyphen indicates that this atom is directly attached to the main chain and the double hyphen shows that atom is once removed from the continuous path. If this convention is followed, an adequate description of this network is given by citing the nonconsecutive locant pairs: *an, cp, dm, fk, g--j*. The fact that the branched point *g--* appears in one of the pairs is all that is needed to convey to the program the information pertaining to the branched nature of this structure.

It is also interesting to note that the number of nonconsecutive locant pairs is equal to the minimum number of rings which must be mentioned in order to describe the network completely. Thus, the input requirements of the program are such that a very complicated structure can easily be prepared for analysis. The only rules that need to be considered are the selection of the longest path and the method for designating branched points.

## COMPUTER PROGRAM

In this section the procedure used to find the canonical path for a structure is discussed in general. The specific steps in the computer program are not detailed. The program was written in Extended ALGOL-60 for the Burroughs B-5500 computer.

The program first reads in the input data which was discussed earlier and which consists of a series of nonconsecutive locant pairs. At this time those points which will not be considered as starting points for a path are read and stored in memory. Since computers are able to manipulate numbers much more easily and rapidly than alphabetical symbols, each locant is transformed to its numeric equivalent. The convention followed is the one specified in the revised rules (*4*). The *a* is equivalent to 1, the *b* to 2, up to *w* which is equivalent to 23. The 24th point is designated in the notation as *a*& and so on.

A limitation of a maximum of 512 points in a path was built into the program. The branched locants are made equivalent to the number + 512. Thus, *c*- becomes 515 and *g*- becomes 519. To obtain the numeric equivalent of the second branch atom, another 512 is added, thus *e*-- is transformed into 1029.

The next step is the construction of a complete connection table for the network from the information contained in the set of nonconsecutive locants. It is from this table that all the future manipulations will be made. A count is maintained of the number of points in the main path as well as the number of branched points.

At this point a procedure or subroutine, which shall be called CLOSER, is called. This procedure follows the path and examines the rings as they are formed by the path. The results from this procedure give the number of rings, the sizes of the rings, and a description of each atom relative to the number of rings it is in as well as the number of other atoms to which it is connected. This initial data is stored in memory for future reference.

The program then turns to the procedure GETPATH which chooses the correct path. It is in this procedure that each locant or point is considered as a starting point for a path, unless specifically excluded at input time. Care is taken to make sure that fused ring junctions are not crossed. All possible paths are explored. Once a path has been chosen a procedure called FUSION determines the nonconsecutive locant pair set for this route. Detailed bookkeeping steps must be carried out to make sure that no paths are missed or studied more than once. Since a set of nonconsecutive locant pairs completely describes a given path, a table of those sets found by the program is maintained for future reference. Once a new path has been chosen, the new set of nonconsecutive locants is compared with those already stored in the table. If the set of pairs has been entered in the table previously, an equivalent path has been found, and no further analysis need be carried out on this set. The program then returns to choose another path. If the set of pairs is not in the table, it is entered, and further analysis is continued.

It is at this point that the notation rules are used to find the best path. The initial call on the procedure CLOSER resulted in information about the original path. This information has been saved and now that a new path has been chosen, the procedure CLOSER is called again. The new information is now compared with the original data to determine which is the better of the two. If the original path satisfies the rules of precedence better than the second path, the second path is discarded and the program is returned to look at the next path available. However, if the new path is preferred the original information is replaced by the data obtained from the more recent one, and this becomes the standard against which future paths will be tested. Thus, after all the paths have been explored, the best or preferred path will be the standard.

The selection procedures are carried out as follows:

1. The sizes of the rings are arranged in increasing numerical order within each set and compared. The earlier sequence is preferred (a 556 sequence would be chosen over 566). If there is a one to one correspondence then the comparison goes to the next step.
2. The number of nonconsecutive pairs are compared. The set containing fewer pairs will be the path which defines the fewer rings. If the number of rings are equal the next comparison is made.
3. The lengths of the main paths are compared. The longer path is preferred since it would indicate a smaller number of branched locants.
4. Comparison is now made between the sum of earliest ring locants for each set. The smaller sum is preferred. If they are equal the comparison proceeds further.
5. Comparison is made of the earliest ring locants as they are found by procedure CLOSER. The earlier sequence is chosen, otherwise the selection continues. (The sequence 1,1,2,5 would be preferred over 1,2,1,5.)
6. Comparison is now made of ring sizes as they are obtained by CLOSER rather than after rearranging them as in step 1. The earlier sequence is selected. (6557 would be preferred over 6575.)

If after going through these six selection steps the program is unable to differentiate between the two paths in question, the structure under consideration has equivalent paths which must be resolved on the basis of further considerations such as saturation and substitution. In such a case the equivalent path description is stored away, and the analysis is continued. The program now returns to select another path for examination. The data for the best path thus far encountered has been saved, and the data from the next path will be compared in a similar manner. This process is repeated until all the routes available have been exhausted. When this occurs, the path which has successfully passed all the comparisons is considered to be the path which will yield the canonical notation. From time to time equivalent paths are found to be the correct paths at the conclusion of the analysis. This program makes no attempt to differentiate any further, but produces an appropriate message and then treats each one as if it were the only one.

The program now takes the path thus selected and writes the notation. The notation is written according to the rules of the notation as follows:

1. The structure is assumed to be carbocyclic and therefore the notation is started with the symbol L.
2. The numerals showing the sizes of rings are cited next. These numerals are cited in the order in which each ring is completed by the locant path, but before each ring numeral is cited the earliest locant in that ring. The *a* locants are omitted.
3. If the structure is found to contain three cited rings which share the same pair of multicyclic points, the pairs of nonconsecutive locants required by Rule 43 are cited immediately after the last ring numeral. Each pair is preceded by a slash mark.
4. The locants for all bridge atoms are cited in alphabetical order, each preceded by a blank space. A bridge locant is repeated as many times as the atom is shared by more rings than two.
5. If multicyclic points are present a numeral preceded by a space indicating the number of such multicyclic points is written. The multicyclic point locants are cited immediately

after the numeral in ascending alphabetical order, without spaces preceding them. Any multicyclic point locant is repeated as many times as it is shared by more rings than three. Finally, after a space, the last locant in the path is cited.

6. Each spiro point is then listed by citing its locant followed by the letter X.

7. The notation is completed by writing a T to indicate that the molecule is completely saturated and a J to indicate the closing of the ring description.

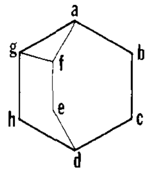
## OUTPUT

Once the program has selected the path which complies with all the restrictions and precedence rules of the Wiswesser notation, a rather detailed analysis of the structure is printed out. In addition to the correct notation the following information is listed:

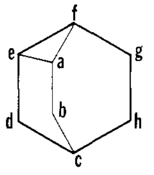
a. the nonconsecutive locant pairs as determined from the final path chosen by the program,
b. the earliest ring locant for each one of the rings cited in the notation,
c. the size of each ring cited,
d. the sum of earliest ring locants,
e. a detailed analysis of the network which indicates for each atom the number of rings it is in, the number of ring atoms to which it is attached, and the rings in which it appears.

Examples 3 through 6 are typical of some of the results obtained using this program. Examples 3 and 4 show on the left the paths which were used as input and on the right the path and notation produced by the computer. The times listed for the development of the canonical notation are small, but the molecules are not very complex. The number of paths mentioned are those paths which the program analyzed.

Example 3



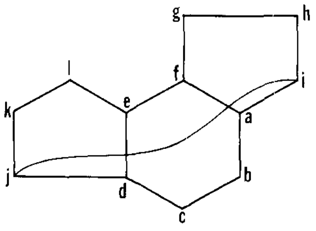
L636 B C 1A HTJ
AF, AG, DH



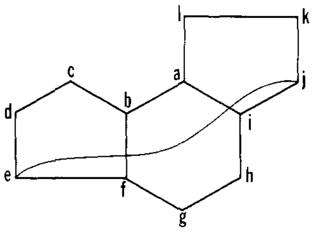
L536 B 1A HTJ
AE, AF, CH

Number of Paths = 10
Time Required = 8 seconds

Example 4



L656 D5 B C 2AD LTJ
AF, AI, DJ, EI

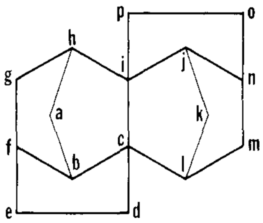


L B5665/EJ 4ABFI LTJ
BF, AI, EJ, AL

Number of Paths = 26
Time Required = 24 seconds

The next two examples, 5 and 6, are more complicated. The number of paths is greater, and a longer time was required for the solution. It is apparent that the time necessary to arrive at the canonical notation is related to the number of rings, atoms, and connections between atoms. It is not a linear relationship, and other factors such as symmetry and path equivalency also have a significant influence.

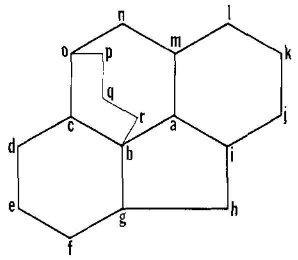
Example 5



L B555 C5 J5 I5 A K 4BCIJ P CX IXTJ
BF, AH, CI, CL, JN, IP

Number of Paths = 32
Time Required = 124 seconds

Example 6



L B6566 B6/CO 4ABBC R BXTJ
BG, AI, AM, CO, BR

Number of Paths = 62
Time Required = 90 seconds

## USES

The computer program which has been described is used for developing the canonical notation of fairly complicated polycyclic structures. The time required for the analysis of a complex structure is not trivial; therefore, it is not expected that the program will be utilized as a routine check of all the polycyclic structures encoded at this laboratory.

A modification to the program was made to permit the printing and punching on cards of the nonconsecutive locant pair sets for each path tried by the program. These cards have been assembled and an alphabetical index of all the structures thus processed has been prepared. This allows an encoder to find rapidly the canonical notation once he has chosen any one path through the structure and identified the nonconsecutive locant pairs. A number of the more complex structures reported in the *Ring Index*[3] has been processed by this program. It is anticipated that this index will grow as new structures which require computer analysis are encountered.

## SUMMARY

The rules of precedence of the Wiswesser Line Formula Notation for polycyclic structures are of such a nature that the canonical notation for complex structures is sometimes obtained only with time-consuming effort. A computer program has been written which accepts as input a set of nonconsecutive locant pairs which unambiguously describes a cyclic structure. The program operates on this information and produces a detailed analysis of the network according to the rules of the notation. The correct canonical notation is also prepared by the program. As a by-product of this analysis, the program punches on cards the nonconsecutive locant pair sets for each one of the paths tried. These data are used to accumulate an index of the structures analyzed and thus prevent unnecessary duplication of effort.

## LITERATURE CITED

(1) Bowman, C. M., Landee, F. A., and Reslock, M. H., "A Chemically Oriented Information Storage and Retrieval System. I. Storage and Verification of Structural Information," J. CHEM. DOC. **7**, 43 (1967).

(2) Landee, F. A., "Computer Programs for Handling Chemical Structures Expressed in the Wiswesser Notation," Presented before the Division of Chemical Literature, 147th National Meeting of the American Chemical Society, Philadelphia, Pa., April 8, 1964.

(3) Patterson, A. M., Capell, L. T., and Walker, D. F., "The Ring Index," 2nd ed., American Chemical Society, 1960; Suppl. **I**, 1963; Suppl. **II**, 1964; Suppl. **III**, 1965.

(4) Smith, E. G., "The Wiswesser Line-Formula Chemical Notation," McGraw-Hill Book Co., New York, N. Y., 1968.

# Structure Display*

ERNEST HYDE and LUCILLE THOMSON
Imperial Chemical Industries, Ltd., P.O. Box 25,
Alderley Park, Macclesfield, Cheshire, England

Received May 21, 1968

**Structure display as the end point of searching is essential if mechanization is to be acceptable to the organic chemist. The standard of display achieved must be as compatible as possible with current practice. Any system proposed for mechanization is incomplete without this facility. This paper gives details of a program suitable for the regeneration of a structure from a connectivity matrix derived from the Wiswesser notation.**

An investigation has been carried out to establish the suitability of the Wiswesser linear notation for computer systems.

A notation technique is based on describing the structural features of a chemical compound by a series of symbols whose order is governed by a set of precise rules. The resulting expression gives a unique representation of a compound in a form well suited to building up a compound registry. In its original form, the notation is not entirely suitable as a method of representing structural data for computer manipulation. However, the notation can be converted by computer into a connection table, a form which does give a precise chemically descriptive record suitable for all computer applications.

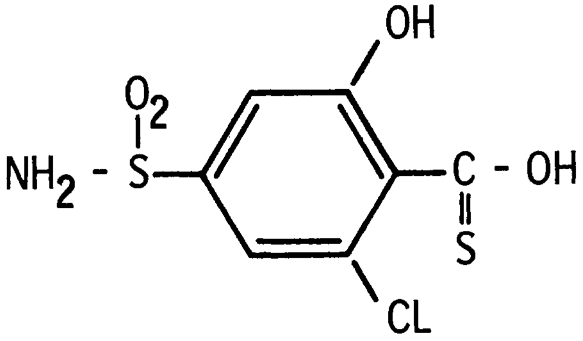
The connection table developed from the notation is written in a linear form and consists of three parts:

**Chemical Unit Section.** A linear string of symbols each representing an atom and its associated bond. The chemical units are assumed to be linked together in a linear string. This assumption is modified by the data contained in the second part of the table.

**Connection Transfers.** A set of numbers written in pairs which indicate: a) the unit position where a break in connectivity has occurred and b) the position of the branch from which the connectivity pattern is to be resumed.

**Ring Block.** This is the last section of the connection table and indicates those units which form a ring system. The data are expressed as a set of numbers that represent the pairs of atoms forming each ring closure.

For example,



[*] Presented before the Division of Chemical Literature, Symposium on Notation Systems, 155th Meeting, ACS, San Francisco, Calif., April 4, 1968.