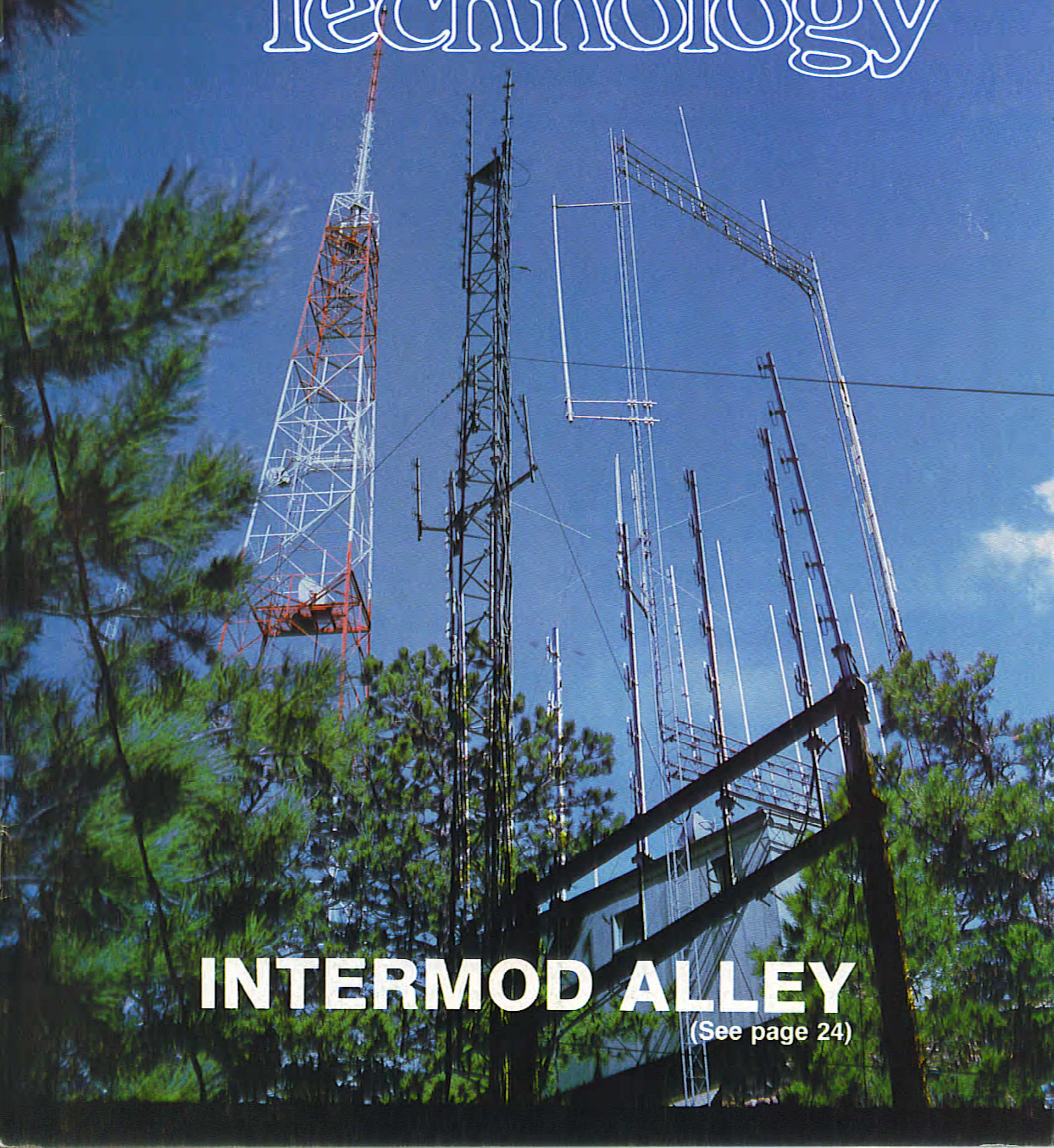
June 1983

Volume 1, Issue 4

Mobile Radio Technology



INTERMOD ALLEY

(See page 24)

Computers Track Down Intermodulation Interference

By Peter DeHaan
Custom Communications

A squawk breaks the squelch sounding like a cross between a Canadian goose and a scrap metal crusher, interrupting communications. The culprit? Intermodulation interference. Computers can help avoid the use of frequencies likely to receive intermod, and track down the cause of interference.

Electronics is called a science. Yet, things occur at times without apparent reason or explanation. Communication systems can suffer seemingly mysterious interference whose cause defies discovery. So many variables enter into the interference equation that the problem may appear incomprehensible. Intermodulation is that kind of problem. Technician and engineer alike can be frustrated by its illusive nature.

Intermodulation is one of many unwanted responses which can occur in RF communications. Don't confuse it with its cousin cross modulation. Intermodulation can occur at either transmitter or receiver. In its basic form, intermodulation is the mixing of two frequencies. "Intentional intermodulation" (heterodyning) enables modern transmitters and receivers to work.

What is Intermodulation?

By definition, intermodulation occurs in a non-linear amplifier. Frequencies are produced corresponding to the sums and differences of the fundamentals and harmonics of the original two frequencies. A test applied to audio amplifiers for distortion caused by non-linear amplification gives an example. Two frequencies, 400 and 1,000 hertz are applied to the amplifier input. The output is then analyzed. If 600 hertz (the difference frequency) or 1,400 hertz (the sum frequency) is present, then the amplifier contains a non-linear element.

This same type of mixing (intermodulation) can occur in RF communications. In some instances intermodulation takes place in a receiver. With multiple signals present at the receiver's antenna, mixing can occur. A new frequency may be produced, passed through the conversion stages and demodulated.

"In its basic form, intermodulation is the mixing of two frequencies."

This demodulated signal would possess the modulated audio signals of both the carriers that created it. Filters placed on the antenna line are often the simplest method of eliminating this type of intermodulation interference. Before a filter is installed, the frequency causing the problem must be determined. The filter is puchased with that specific frequency in mind.

In other instances, intermodulation occurs at the transmitter site. This is often the case when two transmitters are in close proximity. When both transmitters are on the air, one may induce energy into the antenna of the other. The antenna line couples the signal into the transmitter's final amplifier.

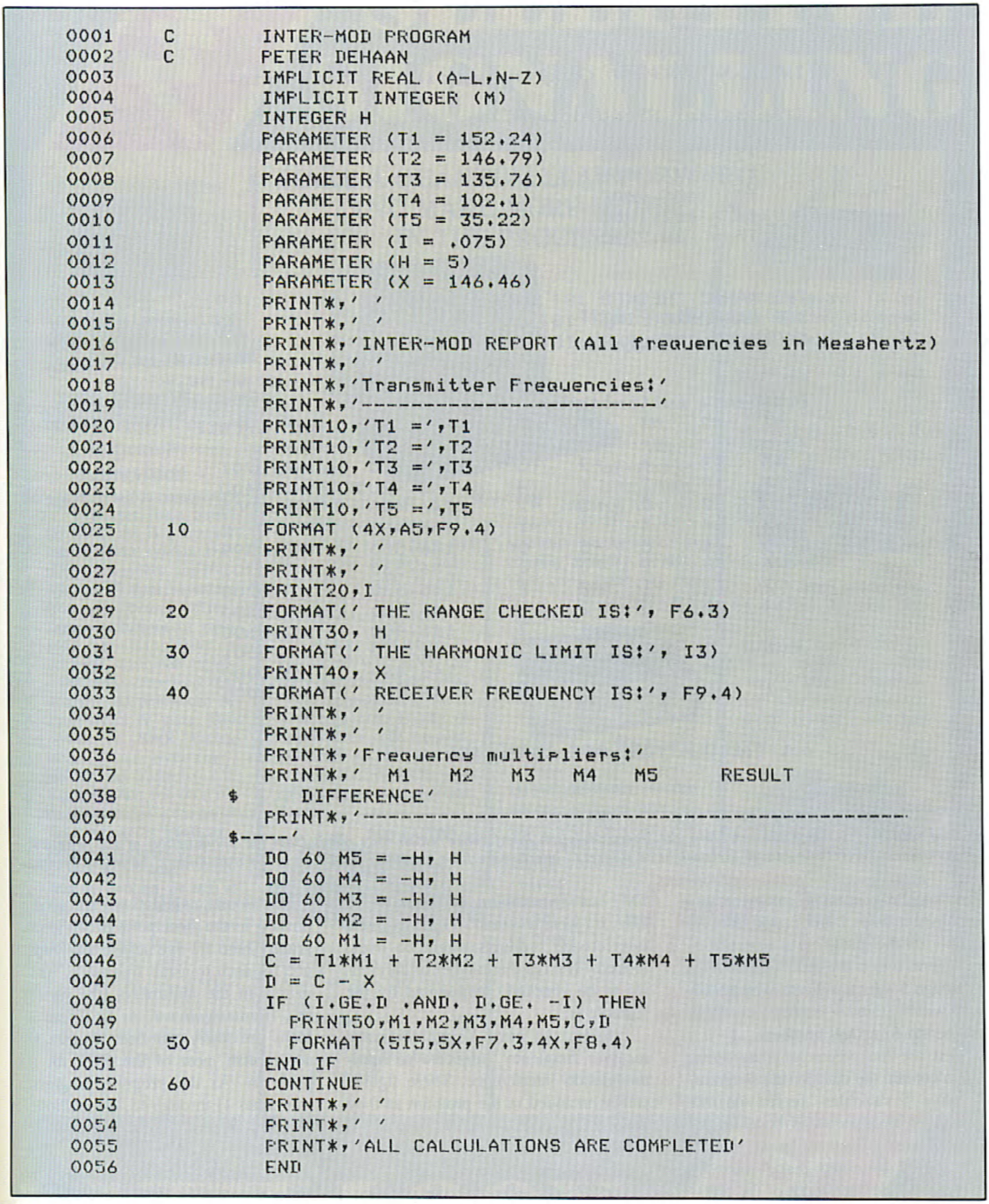
The final amplifier, containing energy from both transmitters, mixes or intermodulates the two signals. The result is not only both original frequencies being transmitted, but also their sum and difference, *and* the sums and differences of their *harmonics*.

If two frequencies were mixed together in this manner and each frequency had five harmonics, a total of 121 separate carriers would result. Each one of these carriers would possess the modulation of the orignals. (If this figure seems unbelievably high, run the intermodulation program and have it print all combinations!)

A Fortran Program

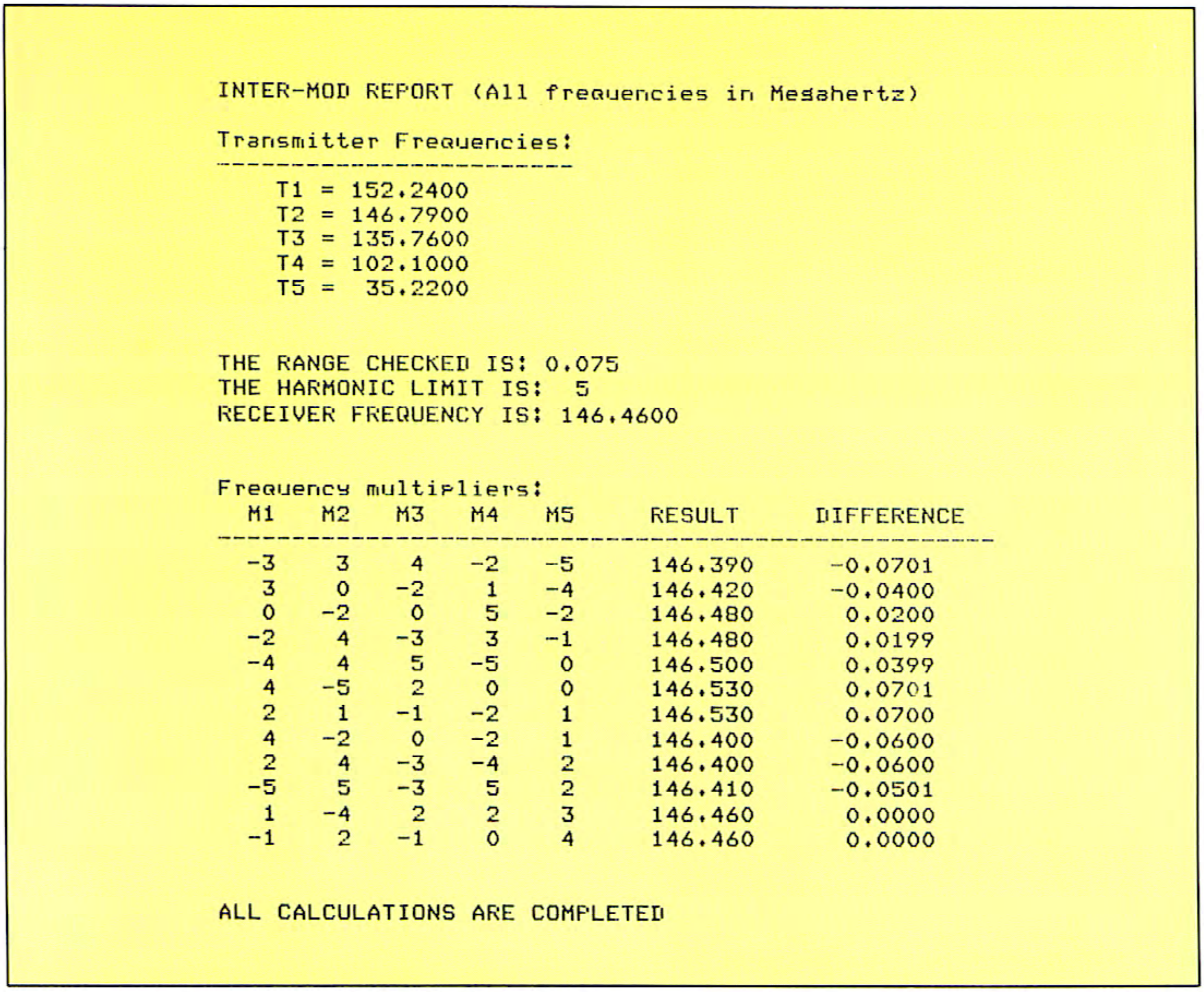
The task of calculating all 121 combinations in the example would be an ominous task, even using a calculator. Fortunately, this is just the kind of assignment that the computer was designed to do. Because the task is mostly mathematical, Fortran, the engineering language, is used to write the program.

The program can most easily be analyzed by breaking it down into three sections. The first section, lines 1 through 13, sets the initial conditions for the



```
C     INTER-MOD PROGRAM
C     PETER DEHAAN
      IMPLICIT REAL (A-L,N-Z)
      IMPLICIT INTEGER (M)
      INTEGER H
      PARAMETER (T1 = 152.24)
      PARAMETER (T2 = 146.79)
      PARAMETER (T3 = 135.76)
      PARAMETER (T4 = 102.1)
      PARAMETER (T5 = 35.22)
      PARAMETER (I = .075)
      PARAMETER (H = 5)
      PARAMETER (X = 146.46)
      PRINT*,' '
      PRINT*,' '
      PRINT*,'INTER-MOD REPORT (All frequencies in Megahertz)
      PRINT*,' '
      PRINT*,'Transmitter Frequencies:'
      PRINT*,'------------------------'
      PRINT10,'T1 =',T1
      PRINT10,'T2 =',T2
      PRINT10,'T3 =',T3
      PRINT10,'T4 =',T4
      PRINT10,'T5 =',T5
10    FORMAT (4X,A5,F9.4)
      PRINT*,' '
      PRINT*,' '
      PRINT20,I
20    FORMAT(' THE RANGE CHECKED IS:', F6.3)
      PRINT30, H
30    FORMAT(' THE HARMONIC LIMIT IS:', I3)
      PRINT40, X
40    FORMAT(' RECEIVER FREQUENCY IS:', F9.4)
      PRINT*,' '
      PRINT*,' '
      PRINT*,'Frequency multipliers:'
      PRINT*,'  M1    M2    M3    M4    M5      RESULT
     $   DIFFERENCE'
      PRINT*,'--------------------------------------------------------
     $----'
      DO 60 M5 = -H, H
      DO 60 M4 = -H, H
      DO 60 M3 = -H, H
      DO 60 M2 = -H, H
      DO 60 M1 = -H, H
      C = T1*M1 + T2*M2 + T3*M3 + T4*M4 + T5*M5
      D = C - X
      IF (I.GE.D .AND. D.GE. -I) THEN
        PRINT50,M1,M2,M3,M4,M5,C,D
50      FORMAT (5I5,5X,F7.3,4X,F8.4)
      END IF
60    CONTINUE
      PRINT*,' '
      PRINT*,' '
      PRINT*,'ALL CALCULATIONS ARE COMPLETED'
      END
```

Calculating possible intermodulation interference frequencies is a mostly mathematical task. Fortran, the engineering language, is used for the program. BASIC language can also be written which will work on other computers.



```
INTER-MOD REPORT (All frequencies in Megahertz)

Transmitter Frequencies:
------------------------
    T1 = 152.2400
    T2 = 146.7900
    T3 = 135.7600
    T4 = 102.1000
    T5 =  35.2200


THE RANGE CHECKED IS: 0.075
THE HARMONIC LIMIT IS:  5
RECEIVER FREQUENCY IS: 146.4600


Frequency multipliers:
  M1    M2    M3    M4    M5      RESULT      DIFFERENCE
-------------------------------------------------------------
  -3     3     4    -2    -5     146.390      -0.0701
   3     0    -2     1    -4     146.420      -0.0400
   0    -2     0     5    -2     146.480       0.0200
  -2     4    -3     3    -1     146.480       0.0199
  -4     4     5    -5     0     146.500       0.0399
   4    -5     2     0     0     146.530       0.0701
   2     1    -1    -2     1     146.530       0.0700
   4    -2     0    -2     1     146.400      -0.0600
   2     4    -3    -4     2     146.400      -0.0600
  -5     5    -3     5     2     146.410      -0.0501
   1    -4     2     2     3     146.460       0.0000
  -1     2    -1     0     4     146.460       0.0000


ALL CALCULATIONS ARE COMPLETED
```

Printed results of a computer run show the intermodulation interference which may result from the combination of five transmitter frequencies. A receiver on 146.46 MHz might be subject to interference from the intermodulation of the transmitters' harmonic products.

program. Programming comments are contained in lines 1 and 2, and have no effect on the program's execution. Statements follow the comments on lines 3 through 5 which indicate which variables will be stored as integers and which will be stored as real numbers.

Next are the parameter statements, which contain the data for the program. The five T variables are for the five transmitter frequencies to be checked. The variable *I* is for the inference range.

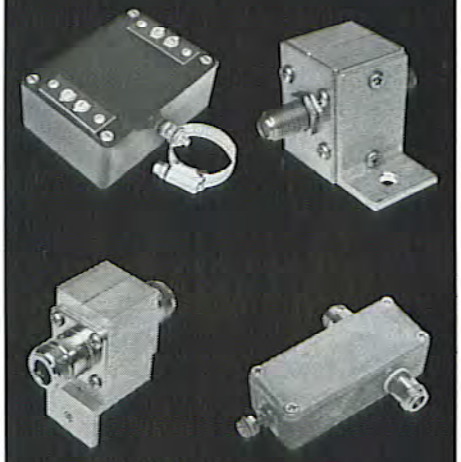
It should be noted that *I* must be greater than the bandwidth of the transmitters being checked. (For example, a wide band transmitter, like broadcast FM, has a maximum deviation of 75 kHz. If *I* were less than 75 kHz, possible interference products might be overlooked.) Next is the number of harmonics to be checked, represented by the variable *H*.

Finally, we have *X*, which is the receiver frequency subjected to intermodulation interference. These figures can be included in the program as they are here, or the program can be made to be user-interactive. The user would enter the data as the program runs.

The second section of the program, lines 14 through 40, prints the data that is to be used and the table headings.

The actual calculations are accomplished in the final portion of the program. There are five nested "Do loops," one for each transmit frequency, which increment the multipliers (the variable *M*), from *negative H* to *H*. These Do loops are really glorified counters and accomplish most of the work for the program. As the program progresses from lines 41 through 45, the multipliers (the *M* variables) are established for the first check.

The first time through all multipliers are a *negative* 5. This is equivalent to saying, "the difference of the fifth harmonic." Each multiplier is multiplied by



its respective transmit frequency to obtain C, the computed frequency, on line 46. The receive frequency is subtracted from the computed result (represented by the variable X), to yield D, the difference.

If D is within the interference limit, established by *I*, it is a potentially harmful combination and the multipliers (*M*), the resultant (C), and the difference (D) are printed. If the output is not significant then there is no printed result. In either case, line 52 of the program loops back to line 45 and the first multiplier is incremented from a *negative* 5 to a *negative* 4 and the process is repeated.

After eleven iterations, M1 has been incremented to positive five. On the next iteration, the program loops back to line 44 rather than 45. M2 is incremented from *negative* 5 to a 4, M1 reverts back to *negative* 5, and the sequence is repeated. The program will continue through the Do loops until all of the multipliers have been incremented to their *positive* limit, 5 (which equals the harmonic limit of H).

At this time the primary frequency calculations, lines 46 through 48, have been run over 160,000 times. The computer has systematically checked every possible combination of interference: 161,051 different combinations in all!

What about BASIC?

Although the sample program is written in Fortran, Fortran is not the only language that can be used. It is merely the easiest. BASIC can also be used, but the program would be longer and more difficult to write.

For someone familiar with programming in BASIC, it should be easy to make the conversion from Fortran to BASIC. The program explanation should be reviewed to ensure a full understanding of the logic behind the Fortran version. Next, the program is divided into three major programming steps.

The goal of the first step is to provide data for the program. This can easily be accomplished through the use of assignment statements or input statements. The choice here is one of personal preference.

The next step involves programming the information output. Again, this step is left to personal preference. It can be simple or complex.

The final programming step will provide the greatest challenge. To simplify this section, it should be divided into three parts. The purpose of the first part is to generate the multipliers. If the user's version of BASIC has a 'For-to' statement, this part will be easy.

However, if a computer will not accept a 'For-to' statement, then a counter and a means to check it will be required. Whichever method is chosen, it must have one loop for each transmit frequency to be checked. Each loop is nested inside one another.

The second part consists of the actual calculations. The formulas shown in the Fortran program can be transferred to the BASIC program.

In part three, the result of the calculations is compared to the limit of the range to be checked. The ease of this portion is again determined by the chosen version of BASIC. The 'If-then do, else' statement is ideal for this situation.

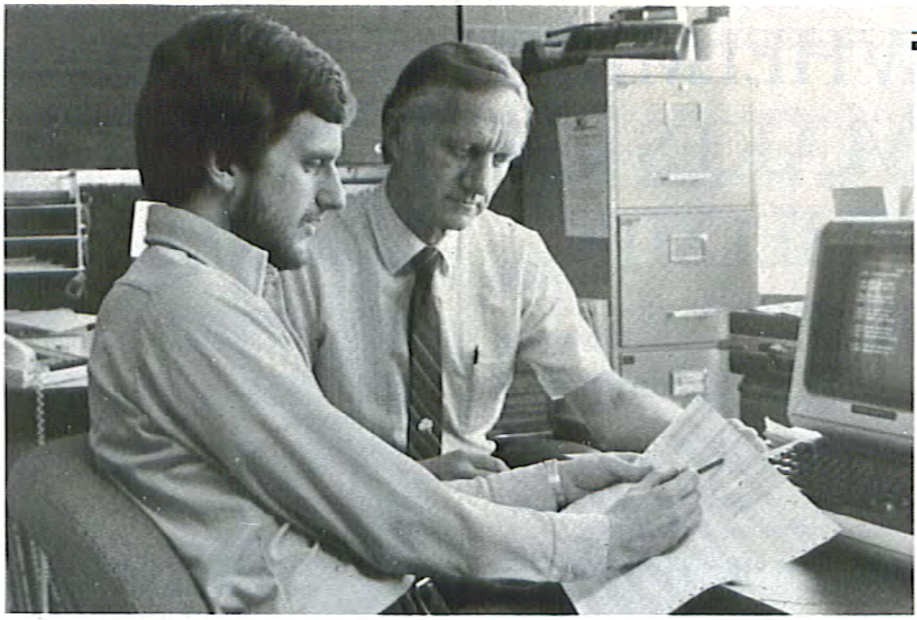
If however, a computer does not provide this command, several 'if-then' statements will replace it in conjunction with a 'go sub' routine. The relational operators available on the system should also be checked.

An outline of the program is as follows:

1. Input data
2. Print output and table headings
3. Perform calculations
    A. Generate multipliers
    B. Calculate new frequency
    C. Check result
    D. Print output if it is within limits
    E. Repeat until finished
4. End

Some Enhancements

Once the program is up and running, either the Fortran version or the BASIC version, some enhancements may be desired. Among these might be a method to check the results against additional receive frequencies. It is advisable to limit an expansion to no more than eight,



Peter DeHaan (l) reviews intermodulation report with technical supervisor Harold Timmer of Custom Communications in Kalamazoo, MI.

because of the time required to run the program.

A program with eight transmit frequencies and five harmonics will execute the main calculation loop almost 250 million times. The program run time would be measured in hours rather than seconds! Another modification would allow the program to accept a varying number of transmit (or receive) frequencies, rather than a fixed number. The program shown here is in its simplest form, but not its fastest. Several changes can be made to make it run faster.

Conclusion

The computer can be a very powerful tool applied to the problems caused by intermodulation interference. If communication on an existing system is hampered by the illusive squawks, tones, and bits of conversation from other systems caused by intermod, the computer can help. Knowing the frequencies involved in the mixture allows a selection of the right filters, traps or other devices likely to provide the cure.

Before a new system is constructed, an intermodulation analysis on the computer can allow the selection of a frequency for the system which may avoid creating a new intermodulation interference problem. **MRT**

Peter DeHaan is a technician with Custom Communications in Kalamazoo, MI, an RCC operator with a dozen systems including paging and mobile telephone service in Michigan and Florida. He is also studying engineering at Western Michigan University.