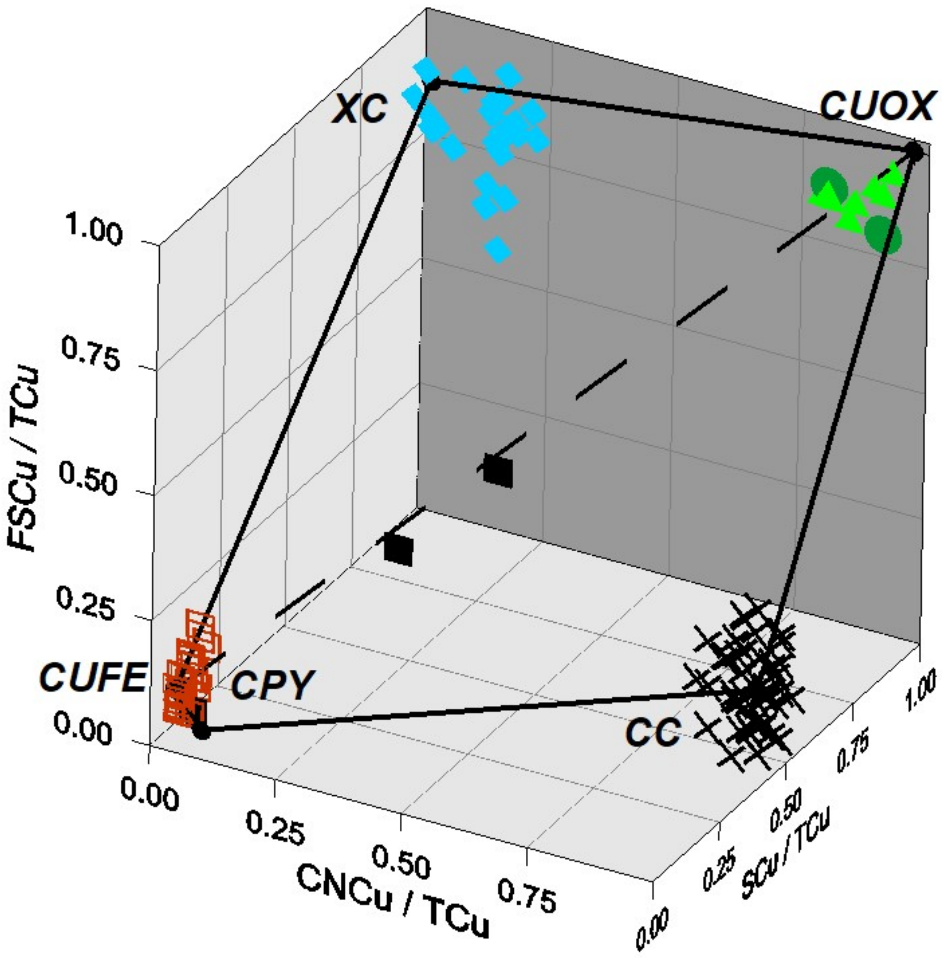


23-Aug-2022

NorMin Programming and User's Guide

NorMin V 2.3

Richard Preece
Principal Consultant

Richard Preece
Services LLC

Richard Preece
Services LLC

NorMin Programming and User’s Guide

NorMin V 2.3

Richard Preece

Disclaimer

This report and referenced computer program were prepared by Richard Preece Services LLC for the use of Minera Escondida Ltda. The computer program NORMINESC.F90 was written in gfortran, the GNU Fortran compiler and is released under the GNU General Public License, which accompanies this document.

Table of Contents

List of Appendices

List of Figures

List of Tables

NorMin Programming and User's Guide

NorMin V 2.3

Richard Preece Services LLC (RPS) was approached in mid-2019 by Minera Escondida, Ltda. (MEL) to recommend modifications to MEL methods and processes in the estimation of copper-iron sulfide mineral abundance. Significant levels of bornite had been encountered in recent drilling of selected areas of the Escondida mineral deposit. Partial extraction analyses of these domains are consistent with logged mineralogy, as most samples fall along a tie-line between the chalcopyrite and bornite end-member extraction rates. However, normative mineral estimates continue to assume the supergene association of chalcopyrite-chalcocite-covellite. MEL required that bornite be incorporated into the standard normative mineral calculation methods, including an automated method for calculating bornite-chalcopyrite±(chalcocite-covellite) to replace the use of Excel spreadsheets.

As part of that study, *NorMinEsc.f90* was written (Preece, 2019a) and distributed to MEL. Source code was written in GFORTRAN, a FORTRAN compiler licensed under the GNU Project, compatible with FORTRAN 90. The program provides the computational framework for incorporating bornite-bearing assemblages into the Escondida process, but required that the input and output data be exported and imported to the Acquire data base via ASCII files. Additional work on the part of MEL was expected to fully integrate the program into the Acquire data base or provide a user-friendly interface to the program.

In November 2019, RPS was engaged by MEL to further extend normative mineral calculations to economic mineralization occurring in the oxide cap that overlies the supergene sulfide deposit. MEL was interested in recommendations for improving estimation of normative minerals in the oxide and mixed oxide/sulfide resources. These resource represent potential ore feed to a high-chloride crush leach plant currently under economic and engineering studies. The extended calculations were added to the NorMinEsc.f90 computer program, resulting in the NorMin_v2.f90 program.

Version 2 has now underwent three minor revisions to:

- correct a programming error in the subroutine that reads the input data file (Version 2.1)
- correctly execute the program structure when the data is coded for Mineral Zone using the block model coding convention (Version 2.2)
- add TCu, Fe, and sulfide sulfur to the output recored (Version 2.3)

This report serves as a user guide to the program, with descriptions and examples to set up and execute the program. It also provides a guide to programming details to support any future modifications to the program. This report assumes that user understands the theoretical and practical basis for the normative mineral calculations. Theoretical background for the methods are described by Preece, et al. (2018) and the program code closely follows the algorithms coded in normative mineral Excel spreadsheets, such as *May18_01PtXt_cpycccv.xlsx*.

Executive Summary

NORmativeMINeral_v2, (NorMin_v2), a FORTRAN computer program, was developed to reproduce the calculation of normative sulfide mineralogy from partial extraction analytical data. The program provides the option of calculating normative minerals for a number of assumption of the mineral association:

- chalcopyrite-chalcocite-covellite (cp-cc-cv)
- chalcopyrite-bornite±chalcocite-covellite (cp-bn-cc and cp-bn-cv)
- chalcopyrite-chalcocite-covellite-brochantite (cp-cc-cv-br)
- chalcopyrite-chalcocite-brochantite-chrysocolla (cp-cc-br-xc)
- chalcocite- brochantite-chrysocolla-Cu-goethite (cc-br-xc-go)

The program uses mineral zone information supplied with the input data for initial normative mineral calculations. If the modal mineral association is incompatible with initial assumption, errors are generated in the normative mineral abundance (e.g., relative abundance greater than 1.0 or less than 0.0) and caught by the program. The program will then test the sample partial extraction geochemistry for consistency with other associations. The order for testing mineral associations was developed from testing a sub-set of samples collected from the oxide, mixed oxide-sulfide, and partially leached sulfide mineral zones in Escondida and Escondida Norte.

The application requires two input files supplied by the user:

- a parameter file that provides compositional information of the copper and iron minerals, and structure of the input data file
- a data file in CSV format that contains sample information that includes mineralogy zone code, indicator of bornite, and partial extraction chemical analyses

Data fields can be in any order, but need to be described in the parameter import file. The program reads partial extraction analytical values from the input data file and calculate CSR, CSP, weight percent mineralogy, and volume percent mineralogy. Pyrite content will be calculated for samples with sulfide sulfur and iron, and an arsenic-bearing mineral will be calculated for sulfide samples, if arsenic is included. Finally, an error flag is set, indicating the consistency of the partial extraction analyses with the final normative mineral association. The calculations are exported in a comma-separated variable (CSV) file to facilitate transfer of the normative mineralogy into a data base or Excel spreadsheet application.

The NorMin_v2 program was written in gfortran a public domain FORTRAN compiler issued under the GNU General Public License. As such, the program is provided to MEL without warranty nor restrictions for distribution or modifications. The program consists of three major sections: data input and checking, normative mineral calculations, and data output.

NorMinEsc Program Components

Program components delivered to MEL include the following items:

- *normin_v2_3.f90*: Source code for the computer program, written in Fortran 90. This file can be read and modified using standard text editors
- *normin_v2_3.exe*: Executable application
- *testparam.dat*: Parameter file template provided by MEL to test the Sept. 2022 test data file
- *input_Sep22.csv*: Example of the data input file provided by MEL
- *test_Sep22.csv*: Example of program output from this version of the program
- *NorMinv2.3_UserGuide.pdf*: This report
- *GNU_GPL.pdf***:** GNU general public license for programs written in gfortran

NorMinEsc User's Guide

Program NorMinEsc was written in gfortran (release GCC-8.2.0-3), an open source FORTRAN compiler developed by the GNU Compiler Collection (https://gcc.gnu.org). While the compiler is compatible with Fortran 95, 2003, and 2008, the program was written to be consistent with Fortran 90 to maximize portability of the source code.

Program Set-up

The file normin_v2.exe is an executable version of the program that will run in a DOS shell. The file is copied to the hard drive of a computer and run in one of two ways:

- double clicking on norminesc.exe in Windows Explorer, which automatically opens a DOS command window and executes the program, or
- adding the program directory to the user's PATH environmental variable and running the program by the normin_v2 command

In both cases, the command must be issued from the directory holding the parameter and input data files. The first option requires that the program file also be in the same directory, while the program and data files can be stored in different directories under the second option.

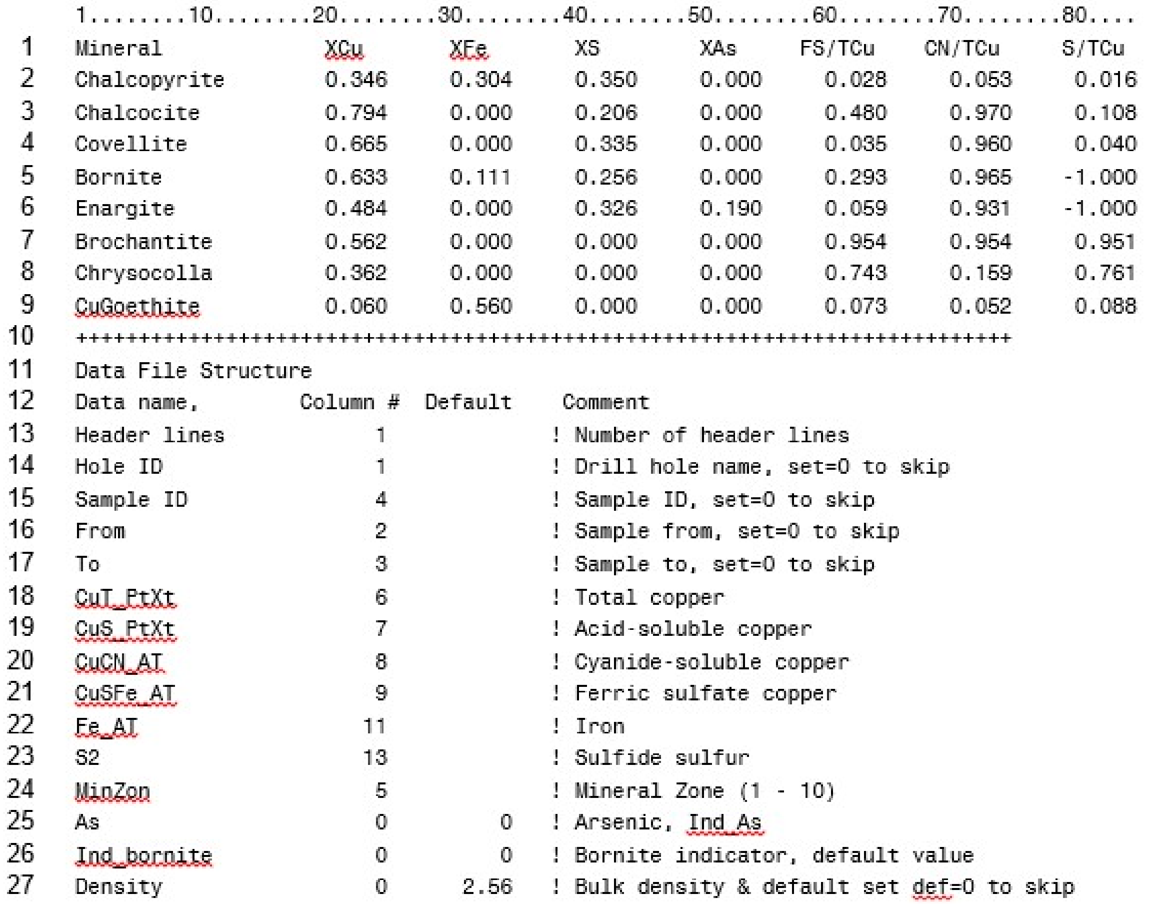
Parameter File Set-up

The parameter file consists of two sections, each containing critical information for proper execution of NorMInEsc. The parameter file template supplied with the program package (*oxdparam.dat*) is populated with values consistent with findings of SRK (2019) for copper sulfide minerals, with findings of Preece (2020) for the oxide minerals and the physical composite data provided for this study.

IT IS IMPORTANT THAT ANY MODIFICATION TO THE TEMPLATE ONLY CHANGES THE VALUES AND DOES NOT CHANGE THE FILE FORMAT AND STRUCTURE

An example of the parameter file is shown in Figure 1, with column spacing shown along the top of the figure and record numbers along the left side. The file consists of two sections:

- the upper section (records 2 – 9, Figure 1) provides compositional information for the copper minerals,
- the lower section (records 13 – 27, Figure 1) provides the field location for variables from the input data file



```
    1........10........20........30........40........50........60........70........80....
1   Mineral             XCu      XFe      XS       XAs      FS/TCu   CN/TCu    S/TCu
2   Chalcopyrite        0.346    0.304    0.350    0.000    0.028    0.053    0.016
3   Chalcocite          0.794    0.000    0.206    0.000    0.480    0.970    0.108
4   Covellite           0.665    0.000    0.335    0.000    0.035    0.960    0.040
5   Bornite             0.633    0.111    0.256    0.000    0.293    0.965   -1.000
6   Enargite            0.484    0.000    0.326    0.190    0.059    0.931   -1.000
7   Brochantite         0.562    0.000    0.000    0.000    0.954    0.954    0.951
8   Chrysocolla         0.362    0.000    0.000    0.000    0.743    0.159    0.761
9   CuGoethite          0.060    0.560    0.000    0.000    0.073    0.052    0.088
10  +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
11  Data File Structure
12  Data name,        Column #  Default    Comment
13  Header lines            1             ! Number of header lines
14  Hole ID                 1             ! Drill hole name, set=0 to skip
15  Sample ID               4             ! Sample ID, set=0 to skip
16  From                    2             ! Sample from, set=0 to skip
17  To                      3             ! Sample to, set=0 to skip
18  CuT_PtXt                6             ! Total copper
19  CuS_PtXt                7             ! Acid-soluble copper
20  CuCN_AT                 8             ! Cyanide-soluble copper
21  CuSFe_AT                9             ! Ferric sulfate copper
22  Fe_AT                  11             ! Iron
23  S2                     13             ! Sulfide sulfur
24  MinZon                  5             ! Mineral Zone (1 - 10)
25  As                      0        0    ! Arsenic, Ind_As
26  Ind_bornite             0        0    ! Bornite indicator, default value
27  Density                 0     2.56    ! Bulk density & default set def=0 to skip
```

Figure 1. Structure of parameter file for the NorMin_v2 program, showing parameters consistent with the May 2018 and December 2019 physical composite data

Mineral Parameters

The mineral parameters required by NorMin_v2 include the compositions of nine copper minerals used by the normative mineral associations. The program expects that the compositional data will begin on record 2 of the file and that the minerals remain in the order listed in the file. The first field of each record lists the copper sulfide mineral, and is skipped by the program as currently written. The next four fields list the mineral composition in terms of weight fraction for copper (XCu), iron (XFe), sulfide sulfur (XS), and arsenic (XAs). The final three fields provide the end-member extraction rates of the mineral in the FSCu, CNCu, and SCu partial extraction analyses.

Mineral composition for chalcopyrite and bornite are relative fixed, but chalcocite and covellite compositions can vary (Preece, et al., 2018), and the user may wish to modify the values from those listed in Figure 1. In addition, while enargite is generally assumed to be the more common As-bearing mineral in Escondida and Escondida Norte, this assumption can also be modified, depending on available data. More importantly, the end-member extraction rate can also vary, depending on analytical laboratory and changing mineral compositions The values listed in the parameter file can therefore be modified to be consistent with future changes in analytical laboratory or specific mineral deposit.

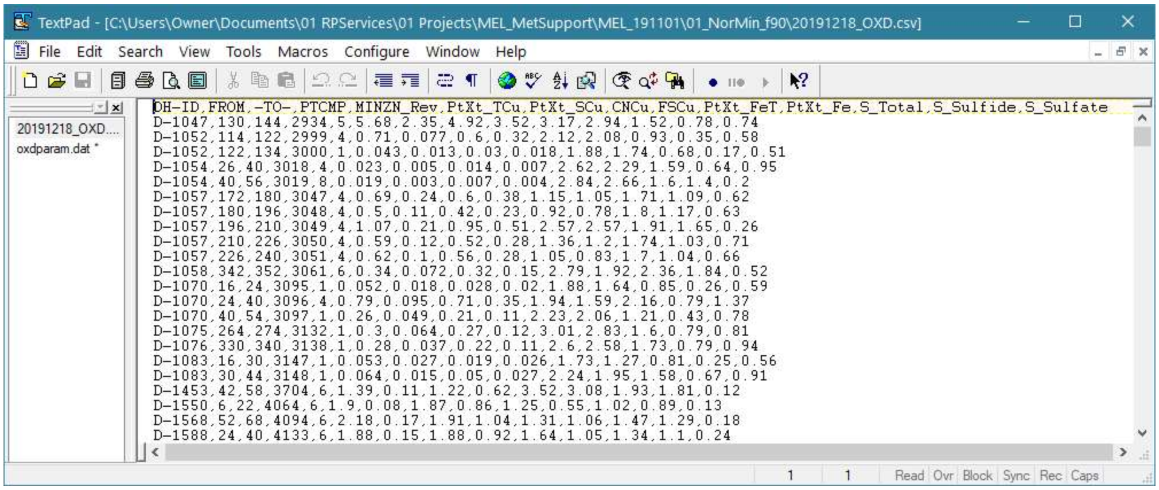
The chalcocite and covellite compositional data and their respective end-member extraction rates are listed in a table located at **Standard!G17:O25** of any normative mineral calculation file. Arsenic-bearing minerals encountered in Escondida Norte are located at **Standard!G32:Q37**. The bornite compositional data replace one of the copper sulfide minerals in the compositional table of **Standard!G17:O25** in any bornite spreadsheet, such as *May18_02PtXt_cpybncc.xlsx.*

End-member extraction rates for copper oxide minerals are not as well studied or standardized. It is important to note that oxide cap modal mineralogy is much more complex and variable than the supergene sulfide associations. Most cupric oxide minerals, including cupric sulfate, cupric carbonate are highly soluble in acid and cyanide and can all be included in a single component. Due to the abundance of brochantite in the Escondida and Norte deposits compared to anterlite or atacamite, that is the component defined in this study. End-member extraction rates for chrysocolla and copper-bearing goethite are highly variable, particularly for the CNCu/TCu and SCu/TCu analyses. This is in part due to the variability of SCu digestion protocols and partly due to variability in mineral composition and crystal structure. The composition used in this study was obtained from the QEMScan SIP (composition file), while end-member extraction rates are consistent with the December 2019 data set. As discussed by Preece (2020), the chrysocolla and Cu-goethite components also include the presence of CNCu-insoluble phases, such as Cu-clays, Cu-Mn oxides, and copper phosphates, such as pseudomalachite.

When replacing values for the sulfide minerals in records 2 – 6 of the parameter file, the user MUST maintain their location in the respective record. This is not required for the three oxide components listed in lines 7 – 9. The phase name, composition, and end-member extraction rates can be modified as needed for consistency with the modal mineralogy. Modification to the program read and write capabilities must be modified in order to change the header label of any new copper oxide phase, however. Contents of the header record in line 1 can be modified to document the parameter file. However, a header line must be present as the program will skip the first record of the file and expect chalcopyrite mineral data to start in record 2.

Input Data Structure

The lower section of the parameter file (Lines 10 through 27, Figure 1) lists the file structure for an example input file, a portion of which is presented in Figure 2. The cvs file contains a single header, which lists the variables of the input file. Note that each input record starts with the drill hole name, followed by the sample interval meterage (FROM and TO), and sample identification. Interval parameters are subsequently listed for each record, as discussed below.


TextPad - [C:\Users\Owner\Documents\01 RPServices\01 Projects\MEL_MetSupport\MEL_191101\01_NorMin_f90\20191218_OXD.csv]
File Edit Search View Tools Macros Configure Window Help
20191218_OXD....
oxdparam.dat *
1 1 Read Ovr Block Sync Rec Caps

```
DH-ID,FROM,-TO-,PTCMP,MINZN_Rev,PtXt_TCu,PtXt_SCu,CNCu,FSCu,PtXt_FeT,PtXt_Fe,S_Total,S_Sulfide,S_Sulfate
D-1047,130,144,2934,5,5.68,2.35,4.92,3.52,3.17,2.94,1.52,0.78,0.74
D-1052,114,122,2999,4,0.71,0.077,0.6,0.32,2.12,2.08,0.93,0.35,0.58
D-1052,122,134,3000,1,0.043,0.013,0.03,0.018,1.88,1.74,0.68,0.17,0.51
D-1054,26,40,3018,4,0.023,0.005,0.014,0.007,2.62,2.29,1.59,0.64,0.95
D-1054,40,56,3019,8,0.019,0.003,0.007,0.004,2.84,2.66,1.6,1.4,0.2
D-1057,172,180,3047,4,0.69,0.24,0.6,0.38,1.15,1.05,1.71,1.09,0.62
D-1057,180,196,3048,4,0.5,0.11,0.42,0.23,0.92,0.78,1.8,1.17,0.63
D-1057,196,210,3049,4,1.07,0.21,0.95,0.51,2.57,2.57,1.91,1.65,0.26
D-1057,210,226,3050,4,0.59,0.12,0.52,0.28,1.36,1.2,1.74,1.03,0.71
D-1057,226,240,3051,4,0.62,0.1,0.56,0.28,1.05,0.83,1.7,1.04,0.66
D-1058,342,352,3061,6,0.34,0.072,0.32,0.15,2.79,1.92,2.36,1.84,0.52
D-1070,16,24,3095,1,0.052,0.018,0.028,0.02,1.88,1.64,0.85,0.26,0.59
D-1070,24,40,3096,4,0.79,0.095,0.71,0.35,1.94,1.59,2.16,0.79,1.37
D-1070,40,54,3097,1,0.26,0.049,0.21,0.11,2.23,2.06,1.21,0.43,0.78
D-1075,264,274,3132,1,0.3,0.064,0.27,0.12,3.01,2.83,1.6,0.79,0.81
D-1076,330,340,3138,1,0.28,0.037,0.22,0.11,2.6,2.58,1.73,0.79,0.94
D-1083,16,30,3147,1,0.053,0.027,0.019,0.026,1.73,1.27,0.81,0.25,0.56
D-1083,30,44,3148,1,0.064,0.015,0.05,0.027,2.24,1.95,1.58,0.67,0.91
D-1453,42,58,3704,6,1.39,0.11,1.22,0.62,3.52,3.08,1.93,1.81,0.12
D-1550,6,22,4064,6,1.9,0.08,1.87,0.86,1.25,0.55,1.02,0.89,0.13
D-1568,52,68,4094,6,2.18,0.17,1.91,1.04,1.31,1.06,1.47,1.29,0.18
D-1588,24,40,4133,6,1.88,0.15,1.88,0.92,1.64,1.05,1.34,1.1,0.24
```

Figure 2. Portion of and example input data file.

The lower section describes structure of each data record of the input CSV file. Lines 11 – 13 are expected to be informational records and can be modified by the user to document the parameter file as needed.

Record #14 of the parameter file provides the number of header records of the input CSV file. The user should enter 0 (zero) for the case where a header record is not present in the input file.

Line numbers 15 – 28 provide the locations of the input variables within the CSV record. For this example, the drill hole name is in the first field of each record, which is entered on line 15 of the parameter file (Figure 1). The sample identification, which is treated as a string variable, is in the 4th record field, while the FROM and TO meterage of the sample are in the 2nd and 3rd fields, respectively (compare lines 15 – 18, Figure 1 with the cvs file records in Figure 2).

The copper assay data are in fields 6 through 9 and the iron and sulfur assays are in fields 11 and 13, respectively. The data fields are not required to be listed in the order that they occur in the input file. Instead, each variable should be maintained in the same order as they occur in the template file. Variables that are missing from a given data input data file is indicated by entering a value of 0 (zero) in the field location, as seen for the arsenic, bornite indicator flag, and rock density variables (lines 26 – 28, Figure 1)

In addition to the record field location for the arsenic assay (columns 24 – 25, line 26; see Figure 1), the unit of concentration is provided by the arsenic indicator (IND_AS, 1 = ppm, 2 = weight percent, see Table 1). The input data file did not include the arsenic assay, which is indicated by a zero value in the field specification and setting the IND_AS flag equal to zero.

Table 1. Indicator flags for arsenic assay units and presence of bornite

IND_AS	Meaning
0	Ignore As assay
1	As assay in ppm
2	As assay in wt. %
IND_BN	Meaning
0	No bornite in sample
1	Bornite in sample

Line 25 of the parameter defines the field location of the sample mineral zone (MinZon, see Table 2). As discussed below, the code is used for choosing the initial normative mineral zone and should reflect the presence and absence of copper oxide minerals as accurately as possible. While not critically important to be a highly accurate reflection of logged mineralogy, the code should be based on original logging and analytical data, rather than geological domain assignment. Two different conventions have been used by Escondida for coding the leached capping and oxide mineral zones, as shown in Table 2. Version v2.3 of the program will function correctly with either convention.

Table 2. List of Mineral Zone codes, as implemented in NorMin_v2.2

DH MinZon	Model MinZon	Meaning
1	0	Leach Cap
2	1	Oxide
4	4	Part Leach
5	5	Mixed Oxide-Sulfide
6	6	High Enrichment
7	7	Low Enrichment
8	8	Primary (cp-py)
10	10	Primary (cp-bn)

A bornite indicator flag is on line 27 of the parameter file (IND_BN, 0 = no bornite, 1 = bornite; see Table 1). The default value for the bornite indication flag is entered in column 30, and will be applied to any input data records that are missing the bornite flag. If bornite is not a constituent of any sulfide samples in the data file, the field location and default bornite flag can be set to 0 and all sulfide samples will be calculated with the cp-cc-cv association (see Figure 1).

Bulk rock density is required to convert weight percent mineral to volume percent mineral, which a useful transformation in the context of supergene enrichment potential. Realizing that this property may not be easily available, the default value can be provided, which will be used for all input file records that are missing a density value. If density is not included in the data file and the default density is set to zero, volume percent will not be calculated.

There are no restrictions on naming the parameter file for a given project, except that it must be less than 25 characters. NorMin_v2 must be invoked from the directory that contains the parameter and data entry files. The output file will be written to the same directory.

Data Entry File Set-up

The NorMin_v2 program is designed to read drill hole data, exported from the Acquire data base in CSV format. The program will accept and use the complete set of variables listed in the parameter file, but the program will run with a minimum number of variables that include TCu_AT, FSCu_AT, CNCu_AT, and S2. Variables other than those listed in Figure 1 will not affect execution of the program, but will not be read nor transferred to the output file.

Two additional variables are required than is normally provided for the physical composite data base: a bornite indicator flag and bulk density of the sample material. The bornite indicator flag selects between cp-cc-cv and cp-bn±(cc-cv) associations in the normative mineral calculations. As per recommendations by Preece (2019b), use of this flag should be phased out. Mineralogy-based codes should be phased in to eventually excluding all supergene-affected samples from the Primary MinZone. At that point, the bornite flag will be redundant to the Primary MinZone Code.

There are no restrictions on naming the data entry file for a given project, except that it must be less than 25 characters. NorMinEsc must be invoked from the directory that contains the parameter and data entry files. The output file will be written to the same directory. The program reads and processes the input file one line at a time, which means there is no limit on the size of the input file.

Output File Set-up

The NorMinEsc program prompts for the name of the output file during program execution. There are no restrictions on the file name, except for a 25-character limit on the length of the name. The program writes the data to the file in CSV format, without apostrophes for the string variables. The program will write the calculation results in a fixed order (Table 3).

If the program encounters missing variables for a particular record or for entire run, it will set the variable as missing (-99 for numeric data types, a single space (‘ ‘) for string data type) and print the variables to output file. All projects will therefore have the same number and order of variables in the CSV-formatted output file.

Table 3. List of calculation results written to the NorMin_v2.3 output file

	Variable	Meaning		Variable	Meaning
1	dhname	Drill Hole ID	23	cspbr	Copper Source Percent, brochantite
2	f_int	From	24	cspxc	Copper Source Percent, chrysocolla
3	t_int	To	25	cspen	Copper Source Percent, As mineral
4	sampid	Sample ID	26	cp_wtpct	Chalcopyrite, wt. %
5	err_flag	Error Flag*	27	cc_wtpct	Chalcocite, wt %
6	minzon	Mineral Zone	28	cv_wtpct	Covellite, wt %
7	ind_bn	Bornite Indicator	29	bn_wtpct	Bornite, wt %
8	tcu	Total copper from input file	30	go_wtpct	Cu-goethite, wt %
9	fe	Iron from input file	31	br_wtpct	Brochantite, wt %
10	S2	Sulfide sulfur from input file	32	xc_wtpct	Chrysocolla, wt %
11	csrcp	Copper Source Ratio, chalcopyrite	33	en_wtpct	As mineral, wt %
12	csrcc	Copper Source Ratio, chalcocite	34	py_best	Pyrite, wt %
13	csrcv	Copper Source Ratio, covellite	35	xfe	Non-sulfide iron, wt. %
14	csrbn	Copper Source Ratio, bornite	36	cp_volpct	Chalcopyrite, vol. %
15	csrgo	Copper Source Ratio, Cu-goethite	37	cc_volpct	Chalcocite, vol %
16	csrbr	Copper Source Ratio, brochantite	38	cv_volpct	Covellite, vol %
17	csrxc	Copper Source Ratio, chrysocolla	39	bn_volpct	Bornite, vol %
18	cspcp	Copper Source Percent, chalcopyrite	40	go_volpct	Cu-goethite, vol %
19	cspcc	Copper Source Percent, chalcocite	41	br_volpct	Brochantite, vol %
20	cspcv	Copper Source Percent, covellite	42	xc_volpct	Chrysocolla, vol %
21	cspbn	Copper Source Percent, bornite	43	en_volpct	As mineral, vol %
22	cspgo	Copper Source Percent, Cu-goethite	44	py_volpct	Pyrite, vol %

** Note: See discussion in the Program Execution section for details*

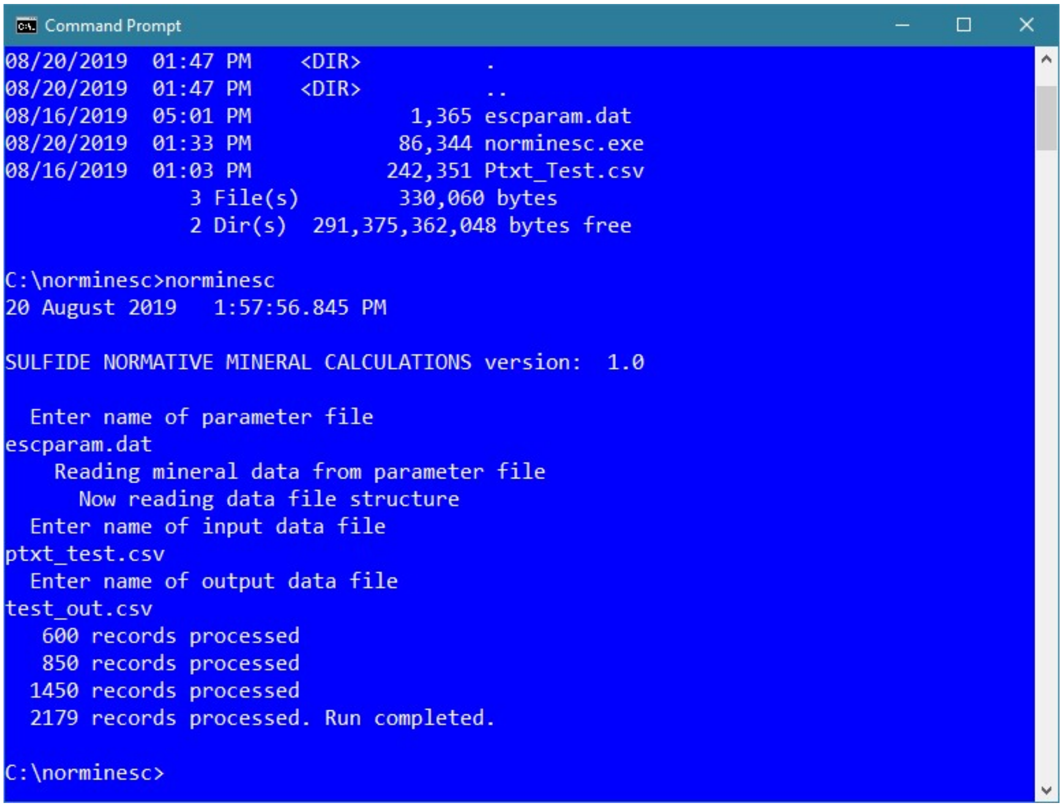
Program Execution

NorMin_v2 is executed in a DOS command shell that can be opened by:

- double-clicking on norminesc.exe in MS File Explorer, or
- right-clicking on the Windows Start button and selecting Command Prompt

Once in the command prompt, the user needs to relocate the prompt to the correct directory and issue the command **normin_v2** to start execution of the application. The program issues prompts for the user to enter the parameter and input data files. At both of these steps, the program will stop with an error message if the requested file is not in the directory.

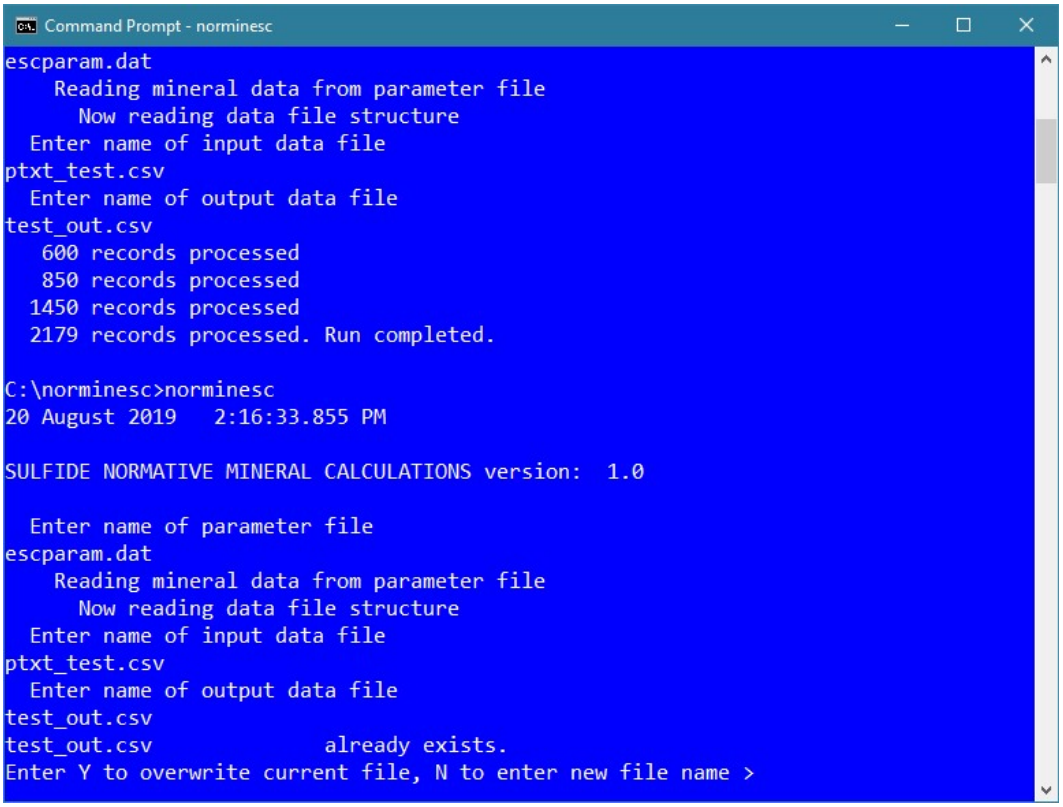
The program will read the parameter file, initialize the program variables and ask for the name of the output file. If the output file name does not exist in the project directory, the program will begin execution and issue notice upon completion of the program. A view of the successive prompts and successful completion is presented in Figure 3.



```
08/20/2019  01:47 PM    <DIR>          .
08/20/2019  01:47 PM    <DIR>          ..
08/16/2019  05:01 PM             1,365 escparam.dat
08/20/2019  01:33 PM            86,344 norminesc.exe
08/16/2019  01:03 PM           242,351 Ptxt_Test.csv
               3 File(s)        330,060 bytes
               2 Dir(s)  291,375,362,048 bytes free

C:\norminesc>norminesc
20 August 2019   1:57:56.845 PM

SULFIDE NORMATIVE MINERAL CALCULATIONS version:  1.0

  Enter name of parameter file
escparam.dat
    Reading mineral data from parameter file
      Now reading data file structure
  Enter name of input data file
ptxt_test.csv
  Enter name of output data file
test_out.csv
   600 records processed
   850 records processed
  1450 records processed
  2179 records processed. Run completed.

C:\norminesc>
```

Figure 3. Program execution in the command shell, showing the program execution command, program prompts, and successful completion of the program.

If the output file name already exists on disk, the program will issue a warning and ask the user if they wish to continue and overwrite the existing file, or create a new file with a different name. A **Y** response will tell the program to continue with setting up the output file, thus overwriting the file on disk. A response of **N** will result in the program re-issuing the output file name prompt to open a new file (Figure 4).



```
escparam.dat
    Reading mineral data from parameter file
      Now reading data file structure
  Enter name of input data file
ptxt_test.csv
  Enter name of output data file
test_out.csv
   600 records processed
   850 records processed
  1450 records processed
  2179 records processed. Run completed.

C:\norminesc>norminesc
20 August 2019   2:16:33.855 PM

SULFIDE NORMATIVE MINERAL CALCULATIONS version:  1.0

  Enter name of parameter file
escparam.dat
    Reading mineral data from parameter file
      Now reading data file structure
  Enter name of input data file
ptxt_test.csv
  Enter name of output data file
test_out.csv
test_out.csv              already exists.
Enter Y to overwrite current file, N to enter new file name >
```

Figure 4. Program execution in the command shell, showing status of the program prompts after entering an existing output file name.

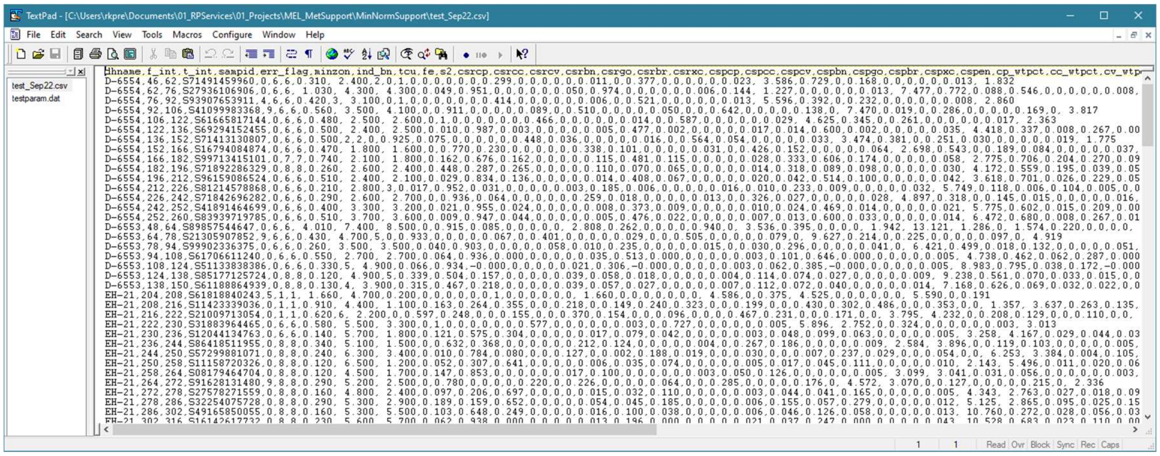
Program Flow and Results

NorMin_v2 first reads the parameter file, initializes program variables, and finds the determinant for the algebraic matrices for end-member extraction rates. The program then reads the data input file one record at a time, and extracts the required variables listed in the parameter file (Figure 1). The program traps any read errors and writes a warning message to the command shell. Generation and source of errors that may be encountered during the read process are discussed further in Error Codes, below. While read errors may result in the program skipping individual records, the program will continue to run.

After a data record has been read, it checks that all four copper assays (TCu, SCu, CNCu and FSCu) are present for calculation of the normative mineralogy. If one or all assays are missing (value < 0.0), the program cycles to the next record and the output record is not written to file. Otherwise, the program begins processing the partial extraction data, as discussed in the following Programming Guide section. The Copper Source Ratio of the normative mineralogy is calculated with the most appropriate normative components, as determined from the MinZone and partial extraction analyses. Where necessary, the calculated CSR values are adjusted within the limits of 0.0 and 1.0. The copper mineralogy content is then converted to Copper Source Percent and wt. % mineral. Pyrite content is calculated from both iron and sulfide sulfur content and the best value selected using the same criteria as the spreadsheets. The program subsequently calculates the non-sulfide iron content (XFe) and volume percent mineral content.

The program writes the calculation results to the user-designated output file in CSV format and in a consistent order, as listed in Table 3. Variables that occur in both the input and output files are the sample identifiers (DH Name, From, To, and Sample ID), Mineral Zone, bornite indicator flag, and the copper, iron, and sulfur assay values. Those variables that are not calculated due to user instructions or incomplete input data will be assigned as missing (-99) and written to the file. It should be noted that the CSR values printed in the output file do not include the enargite component, if calculated. This is to help with debugging any potential analytical error flags discussed in the following section.

Part of an example output file is shown in Figure 5.


TextPad - [C:\Users\rkpre\Documents\01_RPServices\01_Projects\MEL_MetSupport\MinNormSupport\test_Sep22.csv]
File
Edit
Search
View
Tools
Macros
Configure
Window
Help
test_Sep22.csv
testparam.dat
1
1
Read
Ovr
Block
Sync
Rec
Caps

```
dhname,f_int,t_int,sampid,err_flag,minzon,ind_bn,tcu,fe,s2,csrcp,csrcc,csrcv,csrbn,csrgo,csrbr,csrxc,cspcp,cspcc,cspcv,cspbn,cspgo,cspbr,cspxc,cspen,cp_wtpct,cc_wtpct,cv_wtp
D-6554,46,62,S71491459960,0,6,6,0.310, 2.400,2,0,1,0,0,0,0,0,0,0.299,0,0,0,0,0,0.011,0,0.377,0,0,0,0,0,0.023, 3.586,0.729,0,0.168,0,0,0,0,0,0.013, 1.832
D-6554,62,76,S27936106906,0,6,6, 1.030, 4.300, 4.300,0.049,0.951,0,0,0,0,0,0.050,0.974,0,0,0,0,0,0.006,0.144, 1.227,0,0,0,0,0,0.013, 7.477,0.772,0.088,0.546,0,0,0,0,0,0.008,
D-6554,76,92,S93907653911,4,6,6,0.420,3, 3.100,0,1,0,0,0,0,0,0,0.414,0,0,0,0,0,0.006,0,0.521,0,0,0,0,0,0.013, 5.596,0.392,0,0.232,0,0,0,0,0,0.008, 2.860
D-6554,92,106,S41099983368,9,6,6,0.560, 3.500, 4.100,0,0.911,0,0,0,0,0.089,0,0.510,0,0,0,0,0.050,0,0,0.642,0,0,0,0,0.138,0, 7.470,0.019,0,0.286,0,0,0,0,0.169,0, 3.817
D-6554,106,122,S61665817144,0,6,6,0.480, 2.500, 2.600,0,1,0,0,0,0,0,0,0.466,0,0,0,0,0,0.014,0,0.587,0,0,0,0,0,0.029, 4.625,0.345,0,0.261,0,0,0,0,0,0.017, 2.363
D-6554,122,136,S69294152455,0,6,6,0.500, 2.400, 2.500,0.010,0.987,0.003,0,0,0,0,0.005,0.477,0.002,0,0,0,0,0.017,0.014,0.600,0.002,0,0,0,0,0.035, 4.418,0.337,0.008,0.267,0.00
D-6554,136,152,S71413130807,0,6,6,0.500,2,2,0,0.925,0.075,0,0,0,0,0,0.448,0.036,0,0,0,0,0,0.016,0,0.564,0.054,0,0,0,0,0.033, 3.474,0.381,0,0.251,0.030,0,0,0,0,0.019, 1.775
D-6554,152,166,S16794084874,0,6,6,0.470, 1.800, 1.600,0,0.770,0.230,0,0,0,0,0,0.338,0.101,0,0,0,0,0,0.031,0,0.426,0.152,0,0,0,0,0.064, 2.698,0.543,0,0.189,0.084,0,0,0,0,0.037,
D-6554,166,182,S99713415101,0,7,7,0.740, 2.100, 1.800,0.162,0.676,0.162,0,0,0,0,0.115,0.481,0.115,0,0,0,0,0.028,0.333,0.606,0.174,0,0,0,0,0.058, 2.775,0.706,0.204,0.270,0.09
D-6554,182,196,S71892286329,0,8,8,0.260, 2.600, 2.400,0.448,0.287,0.265,0,0,0,0,0.110,0.070,0.065,0,0,0,0,0.014,0.318,0.089,0.098,0,0,0,0,0.030, 4.172,0.559,0.195,0.039,0.05
D-6554,196,212,S96159086524,0,6,6,0.510, 2.400, 2.100,0.029,0.834,0.136,0,0,0,0,0.014,0.408,0.067,0,0,0,0,0.020,0.042,0.514,0.100,0,0,0,0,0.042, 3.618,0.701,0.026,0.229,0.05
D-6554,212,226,S81214578868,0,6,6,0.210, 2.800,3,0.017,0.952,0.031,0,0,0,0,0.003,0.185,0.006,0,0,0,0,0.016,0.010,0.233,0.009,0,0,0,0,0.032, 5.749,0.118,0.006,0.104,0.005,0,0
D-6554,226,242,S71842696282,0,6,6,0.290, 2.600, 2.700,0,0.936,0.064,0,0,0,0,0,0.259,0.018,0,0,0,0,0.013,0,0.326,0.027,0,0,0,0,0.028, 4.897,0.318,0,0.145,0.015,0,0,0,0,0.016,
D-6554,242,252,S41891464699,0,6,6,0.400, 3.300, 3.200,0.021,0.955,0.024,0,0,0,0,0.008,0.373,0.009,0,0,0,0,0.010,0.024,0.469,0.014,0,0,0,0,0.021, 5.775,0.602,0.015,0.209,0.00
D-6554,252,260,S83939719785,0,6,6,0.510, 3.700, 3.600,0.009,0.947,0.044,0,0,0,0,0.005,0.476,0.022,0,0,0,0,0.007,0.013,0.600,0.033,0,0,0,0,0.014, 6.472,0.680,0.008,0.267,0.01
D-6553,48,64,S89857544647,0,6,6, 4.010, 7.400, 8.500,0,0.915,0.085,0,0,0,0,0, 2.808,0.262,0,0,0,0,0.940,0, 3.536,0.395,0,0,0,0, 1.942, 13.121, 1.286,0, 1.574,0.220,0,0,0,0,
D-6553,64,78,S21305907852,9,6,6,0.430, 4.700,5,0,0.933,0,0,0,0,0.067,0,0.401,0,0,0,0,0.029,0,0,0.505,0,0,0,0,0.079,0, 9.627,0.214,0,0.225,0,0,0,0,0.097,0, 4.919
D-6553,78,94,S99902336375,0,6,6,0.260, 3.500, 3.500,0.040,0.903,0,0,0,0,0.058,0.010,0.235,0,0,0,0,0,0.015,0,0.030,0.296,0,0,0,0,0.041,0, 6.421,0.499,0.018,0.132,0,0,0,0,0.051,
D-6553,94,108,S61706611240,0,6,6,0.550, 2.700, 2.700,0.064,0.936,0.000,0,0,0,0,0.035,0.513,0.000,0,0,0,0,0.003,0.101,0.646,0.000,0,0,0,0,0.005, 4.738,0.462,0.062,0.287,0.000
D-6553,108,124,S51133838386,0,6,6,0.330,5, 4.900,0.066,0.934,-0.000,0,0,0,0,0.021,0.306,-0.000,0,0,0,0,0.003,0.062,0.385,-0.000,0,0,0,0,0.005, 8.983,0.795,0.038,0.172,-0.000
D-6553,124,138,S85177125724,0,8,8,0.120, 4.900,5,0.339,0.504,0.157,0,0,0,0,0.039,0.058,0.018,0,0,0,0,0.004,0.114,0.074,0.027,0,0,0,0,0.009, 9.238,0.561,0.070,0.033,0.015,0,0
D-6553,138,150,S61188864939,0,8,8,0.130,4, 3.900,0.315,0.467,0.218,0,0,0,0,0.039,0.057,0.027,0,0,0,0,0.007,0.112,0.072,0.040,0,0,0,0,0.014, 7.168,0.626,0.069,0.032,0.022,0,0
EH-21,204,208,S61818840243,5,1,1, 1.660, 4.700,0.200,0,0,0,0,0,0,1,0,0,0,0,0,0, 1.660,0,0,0,0,0,0,0, 4.586,0,0.375, 4.525,0,0,0,0,0,0, 5.590,0,0.191
EH-21,208,216,S11423339036,0,1,1,0.910, 4.400, 1.100,0.163,0.264,0.355,0,0,0.218,0,0.149,0.240,0.323,0,0,0.199,0,0,0.430,0.302,0.486,0,0,0.353,0,0, 1.357, 3.637,0.263,0.135,
EH-21,216,222,S21009713054,0,1,1,0.620,6, 2.200,0,0.597,0.248,0,0,0.155,0,0,0.370,0.154,0,0,0.096,0,0,0,0.467,0.231,0,0,0.171,0,0, 3.795, 4.232,0,0.208,0.129,0,0,0.110,0,0,
EH-21,222,230,S31883964465,0,6,6,0.580, 5.500, 3.300,0,1,0,0,0,0,0,0,0.577,0,0,0,0,0,0.003,0,0.727,0,0,0,0,0,0,0.005, 5.896, 2.752,0,0.324,0,0,0,0,0,0.003, 3.013
EH-21,230,236,S12044134763,0,6,6,0.140, 5.700, 1.800,0.121,0.575,0.304,0,0,0,0,0.017,0.079,0.042,0,0,0,0,0.003,0.048,0.099,0.063,0,0,0,0,0.005, 3.258, 4.167,0.029,0.044,0.03
EH-21,236,244,S86418511955,0,8,8,0.340, 5.100, 1.500,0,0.632,0.368,0,0,0,0,0,0.212,0.124,0,0,0,0,0.004,0,0.267,0.186,0,0,0,0,0.009, 2.584, 3.896,0,0.119,0.103,0,0,0,0,0.005,
EH-21,244,250,S57299881071,0,8,8,0.240, 6.300, 3.400,0.010,0.784,0.080,0,0,0.127,0,0.002,0.188,0.019,0,0,0.030,0,0,0.007,0.237,0.029,0,0,0.054,0,0, 6.253, 3.384,0.004,0.105,
EH-21,250,258,S11158720326,0,8,8,0.120, 6.500, 1.200,0.052,0.307,0.641,0,0,0,0,0.006,0.035,0.074,0,0,0,0,0.005,0.017,0.045,0.111,0,0,0,0,0.010, 2.143, 5.496,0.011,0.020,0.06
EH-21,258,264,S08179464704,0,8,8,0.120, 4.500, 1.700,0.147,0.853,0,0,0,0,0,0.017,0.100,0,0,0,0,0,0.003,0.050,0.126,0,0,0,0,0,0.005, 3.099, 3.041,0.031,0.056,0,0,0,0,0,0.003,
EH-21,264,272,S91628131480,9,8,8,0.290, 5.200, 2.500,0,0.780,0,0,0,0,0.220,0,0.226,0,0,0,0,0.064,0,0,0.285,0,0,0,0,0.176,0, 4.572, 3.070,0,0.127,0,0,0,0,0.215,0, 2.336
EH-21,272,278,S27578271559,0,8,8,0.160, 4.800, 2.400,0.097,0.206,0.697,0,0,0,0,0.015,0.032,0.110,0,0,0,0,0.003,0.044,0.041,0.165,0,0,0,0,0.005, 4.343, 2.763,0.027,0.018,0.09
EH-21,278,286,S32254075728,0,8,8,0.290, 5.300, 2.900,0.189,0.159,0.652,0,0,0,0,0.054,0.045,0.185,0,0,0,0,0.006,0.155,0.057,0.279,0,0,0,0,0.012, 5.125, 2.865,0.095,0.025,0.15
EH-21,286,302,S49165850055,0,8,8,0.160, 5.300, 5.500,0.103,0.648,0.249,0,0,0,0,0.016,0.100,0.038,0,0,0,0,0.006,0.046,0.126,0.058,0,0,0,0,0.013, 10.760,0.272,0.028,0.056,0.03
EH-21,302,316,S16142617732,0,8,8,0.230, 5.600, 5.700,0.062,0.938,0.000,0,0,0,0,0.013,0.196,0.000,0,0,0,0,0.021,0.037,0.247,0.000,0,0,0,0,0.043, 10.528,0.683,0.023,0.110,0.00
```

Figure 5. Portion of the normative mineral output from an example data file

Error Codes

Two types of errors are reported by NorMinEsc:

- Errors reported during program execution, caused by unexpected conditions during the opening and reading of data files
- Errors reported with the calculation results, flagging the samples with partial extraction assays that are inconsistent with assumptions on sulfide mineralogy

The following section briefly describes the identification of the errors and the meaning of status codes that are reported with the error warning.

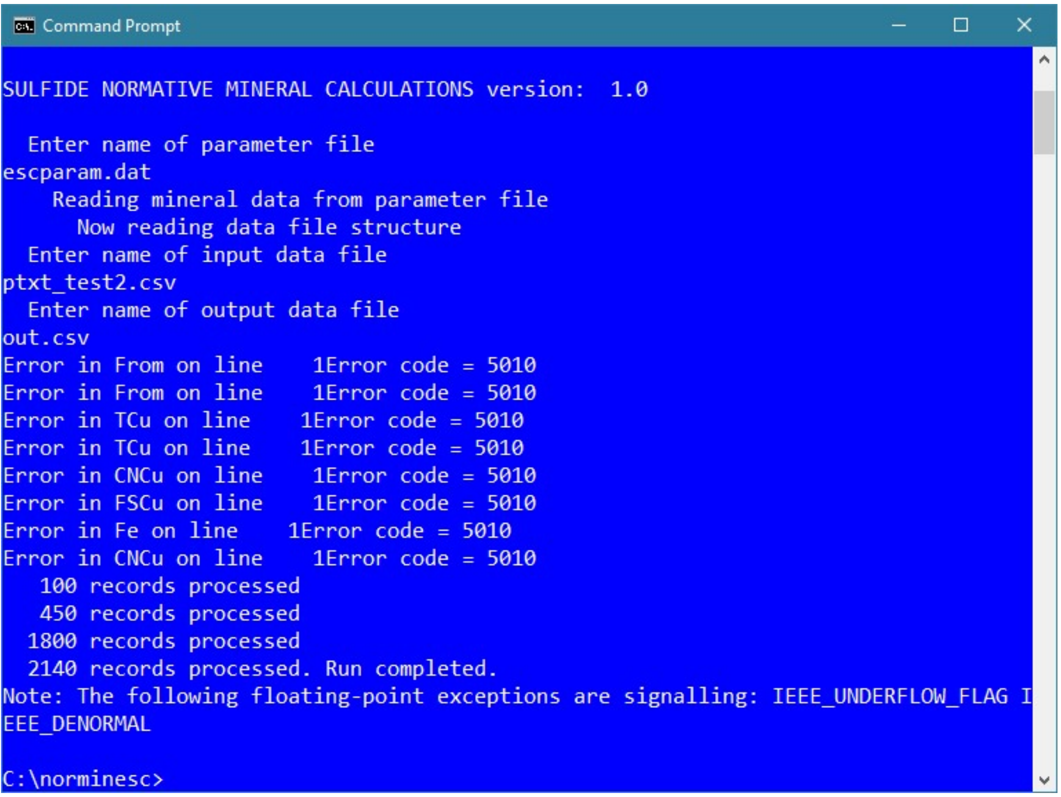
Program Status Codes. Fatal errors (errors that stop program execution) can happen during the opening of parameter and data files, as previously discussed. These errors are explained on screen and are caused by either the parameter or input data file not existing in the same directory as the command shell, or by an output file that already exists. Please note that if NorMin_v2 is executed by double-clicking in File Explorer, the command shell will close when the program closes, and the on-screen warnings may not be visible to the user.

Errors that may occur while the program reads the parameter file are not captured, some of which may be fatal. The program will close and the Fortran compiler will issue system status errors and minor debugging information. It is highly likely that these types of errors are caused by unexpected changes to the parameter file structure when variables are modified. Carefully check the location of required variables to assure they are on the correct line and column.

Errors that occur while the program reads the input data file are captured by the program code and the occurrence and status of the error will be issued to the command shell screen during program execution, as shown in Figure 6. The warning includes information on the input data file record number, the variable that was being read when error occurred, and a status code indicating the source of the error. One of the expected sources for read errors is the unexpected termination of the csv_io subroutines, which will generate one of the two following status codes:

- Code 1: flag issued when the input field number is greater than the number of fields in the record, generated by one of the csv_io subroutines **get_real**, **get_integer**, or **get_char**
- Code 9: flag issued when the input field cannot be located for some other reason, generated by of csv_io subroutines above

The other expected source is a Fortran error that occurs during a READ statement. The Fortran compiler will issue a status code greater than 0, and different from 1 or 9. The exact number will depend on the nature of the read error and the Fortran compiler. The example of the program warning statement in Figure 6 shows a Fortran-generated status code of 5010, which was caused by placing character data where numeric data was expected. Note that the program continues to execute even with these errors, but that the user can expect issues for that particular sample.



```
SULFIDE NORMATIVE MINERAL CALCULATIONS version:  1.0

  Enter name of parameter file
escparam.dat
    Reading mineral data from parameter file
      Now reading data file structure
  Enter name of input data file
ptxt_test2.csv
  Enter name of output data file
out.csv
Error in From on line    1Error code = 5010
Error in From on line    1Error code = 5010
Error in TCu on line    1Error code = 5010
Error in TCu on line    1Error code = 5010
Error in CNCu on line    1Error code = 5010
Error in FSCu on line    1Error code = 5010
Error in Fe on line    1Error code = 5010
Error in CNCu on line    1Error code = 5010
   100 records processed
   450 records processed
  1800 records processed
  2140 records processed. Run completed.
Note: The following floating-point exceptions are signalling: IEEE_UNDERFLOW_FLAG I
EEE_DENORMAL

C:\norminesc>
```

s

Figure 6. Program execution in the command shell, showing an example of read errors during program execution.

Assay Status Codes. NorMin_v2 makes a series of adjustments to the calculated Copper Source Ratios (CSR) to ensure each mineral CSR falls within the limits of 0.0 and 1.0. When one or more of the calculated (raw) CSR values exceeded these limits, the sample mineralogy was adjusted to bring the mineral to nearest physical limit, while holding the sum of CSR to 1.0 and maintaining the ratio of the other calculated minerals. When a raw CSR value of a sample is more than 1.10 or less than -0.10, the sample is flagged as exceeding these limits. The limit of ±10% is approximately equal to the pooled analytical uncertainty of the partial extraction analyses (Preece, et al., 2018).

The criteria for assigning status code are listed in Table 4, and reflect the relative values of the XCu/TCu, FSCu/TCu and/or CNCu/TCu compared with each other. As discussed below, exceeding CSR tolerances under one set of assumptions may also result in the program recalculating normative mineralogy under a different set of assumptions. In that case, the status code will only be reported for the final set of calculations

Table 4. List of Assay Status Codes and raw CSR criteria

Code	Matrix	Criteria	Source
0	All	min(CSR) ≥ -0.1 AND max(CSR) ≤ 1.1	Successful calculation
1	All	TCu < 0.15 and err_flag > 0	Analytical uncertainty (round-off) at low grades
2	cpcccv	CSRcc > 1.1 and CSRcv or CSRbn <-0.1	FSCu in excess of CNCu near CC-CP, indicator of oxidation
3	cpcccv	CSRcp <-0.1	CNCu in excess of FSCu near CC end-member
4	cpcccv	Fails criteria 0, passes criteria 1-3	Analytical uncertainty near CP end-member
5	goccbrxc	CSRbr < -0.1, off-set by elevated CSRcc	Analytical uncertainty near XC-CuGOE tie-line, FSCu > SCu
6	goccbrxc	CSRcc < -0.1 or CSRxc < -0.1	Analytical uncertainty near BR-GOE tie-line, SCu > FSCu
7	goccbrxc	CSRgoe <-0.1	Analytical uncertainty near XC end-member
8	goccbrxc	Fails criteria 0, passes criteria 5-7	Undefined analytical issues
9	cpccbrxc	CSRbr < -0.1, CSRcp < 0.1	Inconsistent SCu > FSCu along CC-CP tieline
10	cpccbrxc	Fails criteria 0, passes criteria 9	Analytical uncertainty near end-members & tie-lines
11	cpcccvbr	min(CSR) < -0.1 OR max(CSR) > 1.1	Analytical uncertainty near end-members & tie-lines

Programming Guide

The NorMin_v2 program was written to be consistent with Fortran 90, although it was compiled with the gfortran compiler, which is compatible with Fortran 95, 2003 and 2007. The source code was provided with the distribution package and can be viewed as a normal text file. Appendix A includes a listing of important variables used in the MAIN section of the NorMin_v2.

Program Description and Flowchart

Calculation of the normative mineralogy was based on the algorithms developed for Excel spreadsheets (see Preece, et al., 2018), that has been incorporated in the Acquire data base for the cp-cc-cv association. The first version of NorMin included calculation of the cp-cc-cv mineral association and extended the calculations for bornite-bearing associations. This second version further extends the capability of the program to calculate the 4-component mineral models that are required for samples collected within the oxidized leach cap.

FORTRAN does not have internal functions for reading string variables (e.g., drill hole name) from CSV-formatted files, nor for writing files in CSV formats. For that reason, external subroutines and functions were required to import and export the data. A set of subroutines were written by John Burkardt and distributed under the GNU public license at https://people.sc.fsu.edu/~jburkardt/f_src/csv_io/csv_io.html. These sub-programs were augmented by subprograms written by RPS to read variables from selected fields of a CSV-formatted record.

Fortran 90 functions named M33DET and M44DET were used to calculate the determinate of the 3x3 and 4x4 algebraic matrices for solving the partial extraction equations. The functions were written in gfortran by David Simpson, and located at https://caps.gsfc.nasa.gov/simpson/software.html.

The computational core of the program solve the simultaneous equations for the estimation of copper minerals on the basis of three partial extraction analyses and the total copper (TCu) analysis: acid-soluble copper (SCu), cyanide-soluble copper (CNCu), and ferric sulfate-soluble copper (FSCu). The mineral composition is first calculated in units of Copper Source Ratio (CSR), which are checked for compliance with mineral ratio tolerances (Table 4). After completing the CSR calculations, the program will recalculate mineral ratios to Copper Source Percent (CSP). These are subsequently converted to weight and volume percent mineral, using mineral composition and bulk rock density supplied by the user. If the data file contains As analyses, the program will allocate copper content to the defined sulfosalt mineral (e.g., enargite) and report the As-mineral abundance. Pyrite content is calculated on the basis of sulfide sulfur and iron, accounting for copper sulfide mineral abundances, reporting the value that best accounts for analytical uncertainty and Fe or S2 mass constraints.

A graphical flowchart of the normative mineral calculations is shown in Figure 7. Upon reading a data record, the program with either follow an initial sulfide-only branch or the oxide ± sulfide branch, as determined from the MinZone code. Oxidized samples (Leach cap, Oxide, Mixed; Table 2) follow the oxide copper branch. Decision points based on Sulfide Sulfur content and/or SCu / CNCu ratios will branch to Oxide-only (goe-cc-br-xc) or Oxide-Sulfide (cp-cc-br-xc) associations (Figure 7). Samples that generate assay status code greater than 0 with the cp-cc-br-xc association will be recalculated under the assumption of cp-cc-cv-br. Normative mineral content that generate exceptions to the Assay Status Code are adjusted for maintaining maximum CSR of 1.0 and minimum CSR of 0.0, while maintaining calculated mineral ratios.

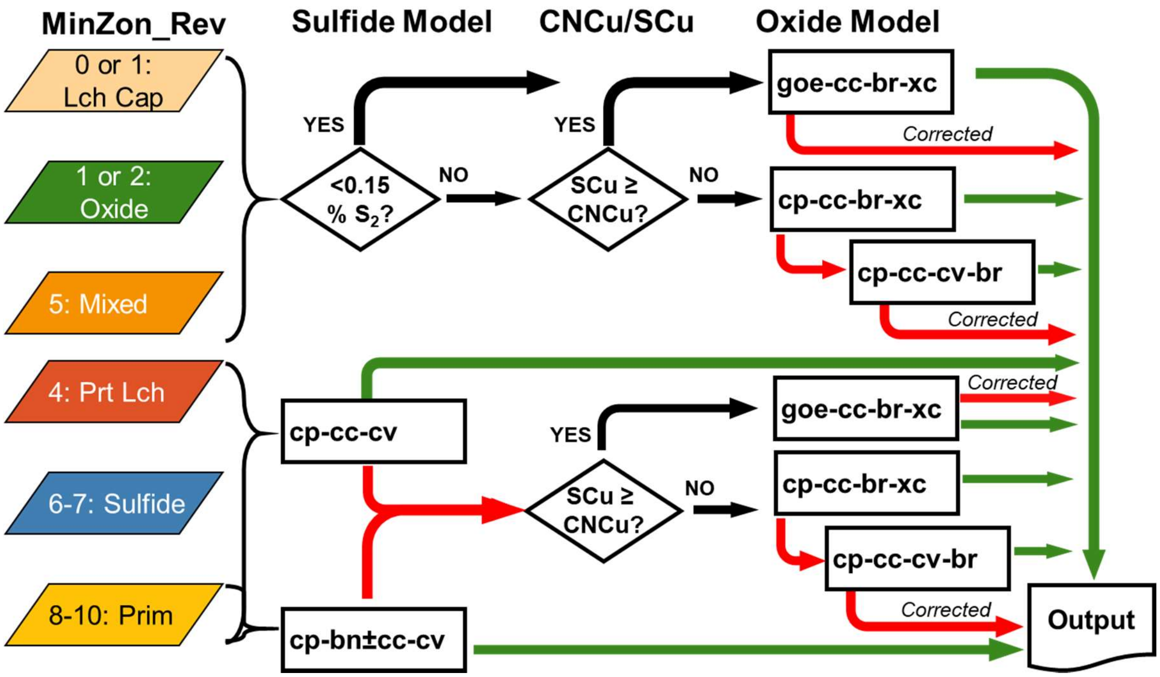
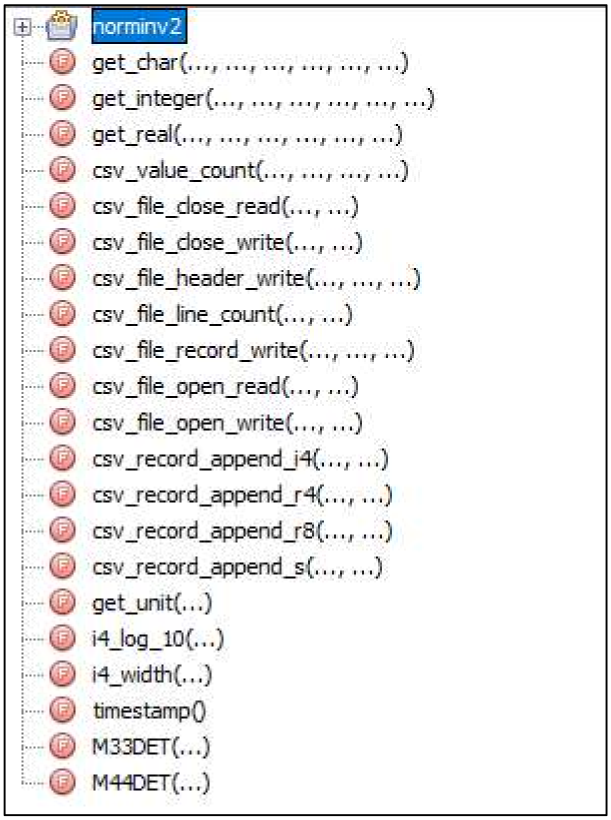


Figure 7. Graphical flowchart for calculation of normative mineralogy. Black arrows show decision branches, program follows green arrows for Assay Status Code 0 and red arrows for Assay Status > 0.

The first part of the sulfide-only branch is essentially Version 1.0 of NorMinEsc, which calculates normative cp-cc-cv or cp-bn, depending on the status of the bornite indicator flag. Normative calculations that result in an Assay Status Code of 0 will exit the normative calculation section and written to the output file (Figure 7). Exceptions to the Status Code will branch to the Oxide program path, as previously outlined.

Program Modules

The program consists a MAIN section named **norminv2**, nineteen subroutines in the cvs_io package, and the functions M33DET and M44DET (Figure 8). The cvs_io package includes subroutines to extract variables from a csv-formatted input record, and to write variables to a csv-formatted output record. The two functions calculate the determinant for a 3x3 and 4x4 matrix, respectively.



```
normin v2
get_char(..., ..., ..., ..., ..., ...)
get_integer(..., ..., ..., ..., ..., ...)
get_real(..., ..., ..., ..., ..., ...)
csv_value_count(..., ..., ..., ...)
csv_file_close_read(..., ...)
csv_file_close_write(..., ...)
csv_file_header_write(..., ..., ...)
csv_file_line_count(..., ...)
csv_file_record_write(..., ..., ...)
csv_file_open_read(..., ...)
csv_file_open_write(..., ...)
csv_record_append_i4(..., ...)
csv_record_append_r4(..., ...)
csv_record_append_r8(..., ...)
csv_record_append_s(..., ...)
get_unit(...)
i4_log_10(...)
i4_width(...)
timestamp()
M33DET(...)
M44DET(...)
```

Figure 8. List of program modules, NorMinEsc Fortran program

The main program has a fairly linear path, and is briefly described here.

Line No.	Purpose	Comments
1	Program Name	
2 – 45	Header	Description and disclaimers
46 – 62	Type declaration	Variable types and properties declared
63 – 163	File declaration and set-up	Write screen prompts and open parameter, input, and output files, read and close parameter file, write header line for output file
164 – 227	Find determinant for 3x3 PtXt matrix	Load end-member extraction rates into 3x3 matrix and find determinant. Three sections for cp-cc-cv, cp-bn-cc and cp-bn-cv mineral equations
228 – 263	Find determinant for 4x4 PtXt matrix	Load end-member extraction rates into 4x4 matrix and find determinant. Three sections for go-cc-br-xc, cp-cc-br-xc and cp-cc-cv-br mineral equations

Line No.	Purpose	Comments
264 – 322	Read input record	Read next record as character string, on EOF stop program Parse character string to extract input variables
323 – 342	Check missing Cu and MinZon	Check for negative value (-99, no data) in the four copper assays. Check for MinZon out of limits. Cycle to next record if true
343 – 356	MinZone branch	Check for oxide mineral zones, else continue to sulfide mineral matrix
357 – 383	Adjust for As	Calculate As mineral content from As assay (wt % mineral) Calculate CSP As mineral Modify TCu, FSCu, and CNCu to remove contribution from arsenic mineral Calculate PtXt assay ratios for CSR calculation
384 – 458	cp-cc-cv CSR calculations	Check bornite indicator flag, if = 1, go to cp-bn-cc section Calculate raw CSRcp, CSRcc, CSRcv If CSR_raw exceeds tolerance of -0.1 to 1.1, assign assay status code then go to Oxide calculations Else, check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, fill non-calculated mineral CSR with zeros Go to CSP calculations
459 – 538	cp-bn-cc CSR calculations	Calculate raw CSRcp, CSRbn, CSRcc Check for raw CSRcc < 0, if yes go to cp-bn-cv section If CSRbn or CSRcp exceed tolerance of -0.1 to 1.1, assign assay status code and go to Oxide calculations Check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, fill non-calculated mineral CSR with zeros 0 Go to CSP calculations

Line No.	Purpose	Comments
539 – 609	cp-bn-cv CSR calculations	Calculate raw CSRcp, CSRbn, CSRcv If CSR_raw exceeds tolerance of -0.1 to 1.1, assign assay status code then go to Oxide calculations Check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, , fill non-calculated mineral CSR with zeros 0 Go to CSP calculations
610 - 629	Initialize Oxide calculations	Calculate PtXt ratios, set bornite and enargite to zero Checks on presence of sulfides: if Sulfide sulfur < 0.15 then go to goe-cc-br-xc calculations if SCu > CNCu then go to goe-cc-br-xc calculations else go to cp-cc-br-xc calculations Note that arsenic is not incorporated as a copper phase
630 – 725	go-cc-br-xc CSR calculations	Calculate raw CSRgo, CSRcc, CSRbr, CSRxc If CSR_raw exceeds tolerance of -0.1 to 1.1, assign assay status code Check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, , fill non-calculated mineral CSR with zeros 0 Go to CSP calculations
726 – 819	cp-cc-br-xc CSR calculations	Calculate raw CSRcp, CSRcc, CSRbr, CSRxc If CSRxc exceeds tolerance of -0.01, go to cp-cc-cv-br calculations Else if CSR_raw otherwise exceeds tolerance of -0.1 to 1.1, assign assay status code Check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, , fill non-calculated mineral CSR with zeros 0 Go to CSP calculations

Line No.	Purpose	Comments
820 – 891	cp-cc-cv-br CSR calculations	Calculate raw CSRcp, CSRcc, CSRcv, CSRbr If CSR_raw otherwise exceeds tolerance of -0.1 to 1.1, assign assay status code Check for CSR_raw tolerance of 0 to 1, adjust CSR for those samples outside of tolerance Assign adjusted CSR_raw values to CSR, , fill non-calculated mineral CSR with zeros 0 Go to CSP calculations
892 – 916	CSP and wt. % calculations	Calculate CSPcp, CSPcc, CSPcv, CSPbn, CSPgo, CSPxc, CSPbr Calculate weight percent mineral
917 – 961	Pyrite calculation	Calculate pyrite (py_fe) from Fe, skip if Fe < 0 Calculate pyrite (py_s) from S2, set pyrite =-99 if S2 < 0 Determine best pyrite Check that absolute relative difference < 0.11 No = minimum of py_fe, py_s Yes = average of py_fe, py_s Calculate excess iron (XFE)
962 – 1010	Volume percent calculations	If default density = 0, set to -99 and skip to next section Define mineral densities from public domain Calculate volume percent from weight percent mineral and bulk density
1011 – 1065	Output calculations	Write variables to character string in CSV format Write record string to output file
1066 – 1071	End of cycle	Return to read next input record
1072 – 1080	End of program	Close files and stop

Program Verification

A subset of the December 2019 PtXt data was tested to verify that the raw CSR normative mineral calculations are consistent with spreadsheet calculations under the same assumptions. The data set was also used to evaluate the type and importance of Assay Status results. These reviews found that the program reproduces the spreadsheet calculations for all program branches. Most of the exclusions to valid normative mineral calculation are due to analytical uncertainty near end-member nodes or tie-lines. A few samples exhibit high variances that are probably due to major analytical or sample labeling errors. See Preece (2020) for details.

List of References

Preece, R. K., 2019a, NorMinEsc Programming and User’s Guide: Unpublished report prepared for Minera Escondida, Ltda., 23 August 2019, 33 p.

Preece, R. K., 2019b, Proyecto Separación Primario de Escondida, Executive Summary: Unpublished report prepared for Minera Escondida, Ltda., 25 October 2019, 35 p.

Preece, RK, 2020, Calculation and Estimation of Normative Mineralogy in Oxide and Mixed Mineral Zones, Escondida: Unpublished summary and slide pack presentation prepared on behalf of Minera Escondida.

Preece, R. K., Williams, M. J., and Gilligan, J. M., 2018, Development of partial extraction methods to estimate abundance of copper-iron sulphide minerals in the Escondida Norte porphyry copper deposit, Chile: Geochemistry: Exploration, Environment, Analysis, DOI: 10.1144/geochem2017-002, v. 18, p. 13-30

SRK Consulting, 2019, Technical Audit of the Escondida May 2018 Geometallurgical Model: Unpublished report prepared for Minera Escondida, Ltda., 19 April 2019, 124 p.

Appendix A. Annotated List of Variables

Variable	Type	Meaning	Source
as	R4	Arsenic assay	Read from input data
ave_est	R4	Average of py_fe and py_s	Calculated
bn_volpct	R4	Bornite, vol %	Calculated
bn_wtpct	R4	Bornite, wt %	Calculated
bnras	R4	End-member extraction, SCu in bornite	Read from parameter file
bnrcn	R4	End-member extraction, CNCu in bornite	Read from parameter file
bnrfs	R4	End-member extraction, FSCu in bornite	Read from parameter file
bnxas	R4	Fraction of arsenic in bornite	Read from parameter file
bnxcu	R4	Fraction of copper in bornite	Read from parameter file
bnxfe	R4	Fraction of iron in bornite	Read from parameter file
bnxs	R4	Fraction of sulfur in bornite	Read from parameter file
br_volpct	R4	Brochantite, vol %	Calculated
br_wtpct	R4	Brochantite, wt %	Calculated
brras	R4	End-member extraction, SCu in brochantite	Read from parameter file
brrcn	R4	End-member extraction, CNCu in brochantite	Read from parameter file
brrfs	R4	End-member extraction, FSCu in brochantite	Read from parameter file
brxas	R4	Fraction of arsenic in brochantite	Read from parameter file
brxcu	R4	Fraction of copper in brochantite	Read from parameter file
brxfe	R4	Fraction of iron in brochantite	Read from parameter file
brxs	R4	Fraction of sulfur in brochantite	Read from parameter file
cc_volpct	R4	Chalcocite, vol %	Calculated
cc_wtpct	R4	Chalcocite, wt %	Calculated
ccras	R4	End-member extraction, SCu in chalcocite	Read from parameter file
ccrcn	R4	End-member extraction, CNCu in chalcocite	Read from parameter file
ccrfs	R4	End-member extraction, FSCu in chalcocite	Read from parameter file
ccxas	R4	Fraction of arsenic in chalcocite	Read from parameter file
ccxcu	R4	Fraction of copper in chalcocite	Read from parameter file
ccxfe	R4	Fraction of iron in chalcocite	Read from parameter file
ccxs	R4	Fraction of sulfur in chalcocite	Read from parameter file
cncu	R4	Cyanide-soluble copper assay	Read from input data
cncu_rat	R4	CNCu/TCu ratio, modified for As-mineral	Calculated
cncumod	R4	CNCu, modified for As-mineral	Calculated
cp_volpct	R4	Chalcopyrite, vol. %	Calculated
cp_wtpct	R4	Chalcopyrite, wt. %	Calculated
cpras	R4	End-member extraction, SCu in chalcopyrite	Read from parameter file
cprcn	R4	End-member extraction, CNCu in chalcopyrite	Read from parameter file
cprfs	R4	End-member extraction, FSCu in chalcopyrite	Read from parameter file
cpxas	R4	Fraction of arsenic in chalcopyrite	Read from parameter file
cpxcu	R4	Fraction of copper in chalcopyrite	Read from parameter file
cpxfe	R4	Fraction of iron in chalcopyrite	Read from parameter file
cpxs	R4	Fraction of sulfur in chalcopyrite	Read from parameter file
cspbn	R4	Copper Source Percent, bornite	Calculated
cspbr	R4	Copper Source Percent, brochantite	Calculated
cspcc	R4	Copper Source Percent, chalcocite	Calculated

Variable	Type	Meaning	Source
cspcp	R4	Copper Source Percent, chalcopyrite	Calculated
cspcv	R4	Copper Source Percent, covellite	Calculated
cspen	R4	Copper Source Percent, As mineral	Calculated
cspgo	R4	Copper Source Percent, Cu-goethite	Calculated
cspxc	R4	Copper Source Percent, Chrysocolla	Calculated
csr_max	R4	Maximum value in set of CSR's	Calculated
csr_min	R4	Minimum value in set of CSR's	Calculated
csrbn	R4	Copper Source Ratio, bornite	Calculated
csrbn_raw	R4	Bornite Copper Source Ratio, calculated	Calculated
csrbn1	R4	Bornite Copper Source Ratio, adjusted	Calculated
csrbr	R4	Copper Source Ratio, brochantite	Calculated
csrbr_raw	R4	Brochantite Copper Source Ratio, calculated	Calculated
csrbr1	R4	Brochantite Copper Source Ratio, adjusted	Calculated
csrcc	R4	Copper Source Ratio, chalcocite	Calculated
csrcc_raw	R4	Chalcocite Copper Source Ratio, calculated	Calculated
csrcc1	R4	Chalcocite Copper Source Ratio, adjusted	Calculated
csrcp	R4	Copper Source Ratio, chalcopyrite	Calculated
csrcp_raw	R4	Chalcopyrite Copper Source Ratio, calculated	Calculated
csrcp1	R4	Chalcopyrite Copper Source Ratio, adjusted	Calculated
csrcv	R4	Copper Source Ratio, covellite	Calculated
csrcv_raw	R4	Covellite Copper Source Ratio, calculated	Calculated
csrcv1	R4	Covellite Copper Source Ratio, adjusted	Calculated
csrgo	R4	Copper Source Ratio, Cu-goethite	Calculated
csrgo_raw	R4	Cu-goethite Copper Source Ratio, calculated	Calculated
csrgo1	R4	Cu-goethite Copper Source Ratio, adjusted	Calculated
csrxc	R4	Copper Source Ratio, chrysocolla	Calculated
csrxc_raw	R4	Chrysocolla Copper Source Ratio, calculated	Calculated
csrxc1	R4	Chrysocolla Copper Source Ratio, adjusted	Calculated
cusul_fe	R4	Iron content of copper sulfides	Calculated
cusul_s	R4	Sulfide sulfur content of copper sulfides	Calculated
cv_volpct	R4	Covellite, vol %	Calculated
cv_wtpct	R4	Covellite, wt %	Calculated
cvras	R4	End-member extraction, SCu in covellite	Read from parameter file
cvrcn	R4	End-member extraction, CNCu in covellite	Read from parameter file
cvrfs	R4	End-member extraction, FSCu in covellite	Read from parameter file
cvxas	R4	Fraction of arsenic in covellite	Read from parameter file
cvxcu	R4	Fraction of copper in covellite	Read from parameter file
cvxfe	R4	Fraction of iron in covellite	Read from parameter file
cvxs	R4	Fraction of sulfur in covellite	Read from parameter file
d_cpccbn	R8	Matrix determinant, cp-cc-bn	Calculated
d_cpccbrxc	R4	Matrix determinant, cp-cc-br-xc	Calculated
d_cpcccv	R8	Matrix determinant, cp-cc-cv	Calculated
d_cpcccvbr	R4	Matrix determinant, cp-cc-cv-br	Calculated
d_cpcvbn	R8	Matrix determinant, cp-cv-bn	Calculated
d_goccbrxc	R4	Matrix determinant, go-cc-br-xc	Calculated

Variable	Type	Meaning	Source
d_min1	R4	Matrix determinant, Mineral 1 of association	Calculated, see in-program notes
d_min2	R4	Matrix determinant, Mineral 2 of association	Calculated, see in-program notes
d_min3	R4	Matrix determinant, Mineral 3 of association	Calculated, see in-program notes
d_min4	R4	Matrix determinant, Mineral 4 of association	Calculated, see in-program notes
dens_def	R4	Default density	Read from parameter file
density	R4	Bulk density	Read from input data
dhname	C25	Drill Hole ID	Read from input file
dumline	C80	Dummy line	Read from command screen
en_volpct	R4	As mineral, vol %	Calculated
en_wtpct	R4	As mineral, wt %	Calculated
enras	R4	End-member extraction, SCu in As-mineral	Read from parameter file
enrcn	R4	End-member extraction, CNCu in As-mineral	Read from parameter file
enrfs	R4	End-member extraction, FSCu in As-mineral	Read from parameter file
enxas	R4	Fraction of arsenic in As-mineral	Read from parameter file
enxcu	R4	Fraction of copper in As-mineral	Read from parameter file
enxfe	R4	Fraction of iron in As-mineral	Read from parameter file
enxs	R4	Fraction of sulfur in As-mineral	Read from parameter file
err_flag	I4	Status code for CSR calculations	Calculated
f_int	R4	From	Read from input file
fe	R4	Iron assay	Read from input data
fscu	R4	Ferric sulfate-soluble copper assay	Read from input data
fscu_rat	R4	FSCu/TCu ratio, modified for As-mineral	Calculated
fscumod	R4	FSCu, modified for As-mineral	Calculated
go_volpct	R4	Cu-goethite, vol %	Calculated
go_wtpct	R4	Cu-goethite, wt %	Calculated
goras	R4	End-member extraction, SCu in Cu-goethite	Read from parameter file
gorcn	R4	End-member extraction, CNCu in Cu-goethite	Read from parameter file
gorfs	R4	End-member extraction, FSCu in Cu-goethite	Read from parameter file
goxas	R4	Fraction of arsenic in Cu-goethite	Read from parameter file
goxcu	R4	Fraction of copper in Cu-goethite	Read from parameter file
goxfe	R4	Fraction of iron in Cu-goethite	Read from parameter file
goxs	R4	Fraction of sulfur in Cu-goethite	Read from parameter file
ias	I4	Field number, As	Read from parameter file
ibn	I4	Field number, bornite indicator flag	Read from parameter file
icncu	I4	Field number, CNCu	Read from parameter file
idebug	I4	Debug flag	Hard-coded in program
idef_bn	I4	Default bornite indicator flag	Read from parameter file
idens	I4	Field number, density	Read from parameter file
idh	I4	Field number, drill hole name	Read from parameter file
ife	I4	Field number, Fe	Read from parameter file
ifrom	I4	Field number, interval from	Read from parameter file
ifscu	I4	Field number, FSCu	Read from parameter file
imz	I4	Field number, MinZone	Read from parameter file
ind_as	I4	Field number, arsenic unit flag	Read from parameter file

Variable	Type	Meaning	Source
ind_bn	I4	Bornite Indicator	Read from input file
inputcsv	C25	File name, input data file	Read from command screen
ios	I4	Error status, OPEN and READ statements	Assigned
is2	I4	Field number, S2	Read from parameter file
iscu	I4	Field number, acid-soluble copper	Read from parameter file
ismp	I4	Field number, Sample name	Read from parameter file
itcu	I4	Field number, TCu	Read from parameter file
ito	I4	Field number, interval to	Read from parameter file
m_cpccbn	R8	End-member extraction matrix, cp-cc-bn	Assigned from input data
m_cpccbrxc	R8	End-member extraction matrix, cp-cc-br-xc	Assigned from input data
m_cpcccv	R8	End-member extraction matrix, cp-cc-cv	Assigned from input data
m_cpcccvbr	R8	End-member extraction matrix, cp-cc-cv-br	Assigned from input data
m_cpcvbn	R8	End-member extraction matrix, cp-cv-bn	Assigned from input data
m_goccbrxc	R8	End-member extraction matrix, go-cc-br-xc	Assigned from input data
m_min1	R8	End-member extraction matrix, Mineral 1	Assigned, see in-program notes
m_min2	R8	End-member extraction matrix, Mineral 2	Assigned, see in-program notes
m_min3	R8	End-member extraction matrix, Mineral 3	Assigned, see in-program notes
m_min4	R8	End-member extraction matrix, Mineral 4	Assigned, see in-program notes
m33det	R8	Calculated determinant, 3x3 matrix	Calculated
m44det	R8	Calculated determinant, 4x4 matrix	Calculated
ndata	I4	Count of data lines read	Calculated
nhdr	I4	No. header lines, input data file	Read from parameter file
outputcsv	C25	File name, output data file	Read from command screen
paramfile	C25	File name, parameter file	Read from command screen
py_best	R4	Pyrite wt.%, selected from Fe and S balance	Calculated
py_fe	R4	Pyrite calculated from iron	Calculated
py_s	R4	Pyrite calculated from sulfur	Calculated
py_volpct	R4	Pyrite, vol %	Calculated
rec_out	C256	Output data record	Written to output data file
record	C256	Input data record	Read from input data file
s2	R4	Sulfide sulfur assay	Read from input data
sampid	C25	Sample ID	Read from input file
scu_rat	R4	SCu/TCu ratio	Calculated from input data
sg_bn	R4	Density, bornite	Assigned
sg_br	R4	Density, brochantite	Assigned
sg_cc	R4	Density, chalcocite	Assigned
sg_cp	R4	Density, chalcopyrite	Assigned
sg_cv	R4	Density, covellite	Assigned
sg_en	R4	Density, As-mineral	Assigned
sg_go	R4	Density, Cu-goethite	Assigned
sg_py	R4	Density, pyrite	Assigned
sg_xc	R4	Density, chrysocolla	Assigned
sumcsr1	R4	Sum of CSR1 values	Calculated
t_int	R4	To	Read from input file
tcu	R4	Total copper assay	Read from input data

Variable	Type	Meaning	Source
tcumod	R4	TCu, modified for As-mineral	Calculated
xc_volpct	R4	Chyrsocolla, vol %	Calculated
xc_wtpct	R4	Chyrsocolla, wt %	Calculated
xcras	R4	End-member extraction, SCu in chyrsocolla	Read from parameter file
xcrcn	R4	End-member extraction, CNCu in chyrsocolla	Read from parameter file
xcrfs	R4	End-member extraction, FSCu in chyrsocolla	Read from parameter file
xcxas	R4	Fraction of arsenic in chyrsocolla	Read from parameter file
xcxcu	R4	Fraction of copper in chyrsocolla	Read from parameter file
xcxfe	R4	Fraction of iron in chyrsocolla	Read from parameter file
xcxs	R4	Fraction of sulfur in chyrsocolla	Read from parameter file
xfe	R4	Non-sulfide iron, wt. %	Calculated
yesno	C1	Decision on overwriting data file	Read from command screen