```
%macro converg;
        call symput('f18ap', PRM_EVM);
        call symput('f18bp', PRM_IM);
        call symput('cyc'  , '1');
        call symput('it'   , '1');
        call symput('m2'   , PRM_EVM);
        call symput('den'  , '1');
        call symput('target', TARGET);

data results;
      nit=.;
      output;
      %let flag= 0;
      %do %until
          (%eval(&flag)=1);

data eqa_form;
      set eqa_form;
      %if (&cyc)=1 %then
          %do;
            PRM_EVM= ROUND((&m2), .000000001);
            PRM_IM= 0;
          %end;
      %else
          %if (&cyc)=2 %then
            %do;
                PRM_EVM= 0;
                PRM_IM= ROUND((&m2), .000000001);
            %end;
          %else
            %if (&cyc)=3 %then
                %do;
                  PRM_EVM= round((&m2/&f18bp)*&f18ap, .000000001);
                        PRM_IM= ROUND((&m2), .000000001);
                  call symput('den', '2');
                %end;
            %else
                %do;
                  endsas;
                %end;
 run;

 DATA EQA_FORM;
      SET EQA_FORM;
      EQA_EVAP=ROUND(((WLT_EQVL*PRM_EVM)/&DEN), 1);
      EQA_INAP=ROUND(((WLT_INCM*PRM_IM)/&DEN), 1);
      EQA_ABPY=ROUND(SUM(EQA_INAP, EQA_EVAP), 1);
            IF WENR_ENR>0 THEN
          DO;
            EQA_DLSR=1;
              EQA_DSLS=ROUND((EQA_ABPY*EQA_DLSR), 1);
          END;
        ELSE
          DO;
            EQA_DLSR=0;
            EQA_DSLS=0;
          END;
```

```
PROC SORT DATA=EQA_FORM;
      BY CO;

PROC MEANS SUM NOPRINT DATA=EQA_FORM;
      BY CO;
      VAR EQA_DSLS ADQ_BUD;
      OUTPUT OUT=COLSHR SUM=EQA_COLS EQA_COTE;

PROC SORT DATA=COLSHR;
      BY CO;

DATA COLSHR;
      SET COLSHR;
      IF EQA_COTE>0 THEN
          EQA_CLSR=ROUND((EQA_COLS/EQA_COTE), .0001);

DATA EQA_FORM(DROP=EQA_CLSR EQA_COLS EQA_COTE);
      MERGE EQA_FORM COLSHR;
      BY CO;
      EQA_CLS2=EQA_CLSR;
      EQA_COL2=EQA_COLS;
      EQA_COT2=EQA_COTE;
      IF OPTYPE='8' THEN
          DO;
                  EQA_VOLS=ROUND((EQA_CLSR*ADQ_BUD), 1);
            EQA_LSHR=EQA_VOLS;
            END;
      ELSE
          EQA_LSHR=EQA_DSLS;
      EQA_FEQA=MAX(0,SUM(ADQ_BUD, -EQA_LSHR));

run;

PROC SORT;
      BY CO DIST;

proc summary data=eqa_form;
      var EQA_FEQA;
      output out=total sum=;

data _null_;
      set total;
      dtotal= EQA_FEQA;
      datd2= (&target - dtotal);
      datcv= 0;
      if abs(sum(&target., -EQA_FEQA)) < 10000 then
          datcv=1;
      call symput('d2' , datd2);
      call symput('cv', datcv);
      call symput('total', dtotal);
run;

 %if (&cv)=1 %then
      %DO;
          data line;
          keep cycl nit targ tot datd2 f18amult f18bmult;
```

```
          label   cycl= 'Cycle'
                  nit=  'Iteration'
                  targ= 'Target'
                  tot=  'EQA'
                  datd2= 'Error'
                  f18amult= 'W. mult.'
                  f18bmult= 'I. mult.';
          FORmat  targ tot comma16. datd2 comma14. cycl nit 2.
                  f18amult f18bmult 12.10;
          cycl= &cyc;
            nit= &it;
          tot= &total;
          targ= &target;
          %let tm2= ROUND((&m2), .000000001);
          %let td2= (&d2);
          datd2= &td2;
          %if &cyc=1 %then
            %do;
                f18amult= ROUND((&tm2), .000000001);
            %end;
          %else
            %if &cyc=2 %then
                %do;
                  f18bmult= ROUND((&tm2), .000000001);
                %end;
            %else
                %if &cyc=3 %then
                  %do;
                   *  f18amult= (&tm2/&f18bp)*&f18ap;
                   *  f18bmult= &tm2;
                         f18amult= round((&tm2/&f18bp)*&f18ap,
.000000001);
                         f18bmult= ROUND((&tm2), .000000001);
                  %end;
          run;
          data results;
            set results line;
                %if (&cyc)=1 %then
                  %do;
                         %let f18ap  = ROUND((&m2), .000000001);
                         %let m2     = ROUND((&f18bp), .000000001);
                         %let it     = 1;
                         %let cyc    = 2;
                      %end;
                %else
                  %if &cyc=2 %then
                      %do;
                           %let f18bp  = ROUND((&m2), .000000001);
                           %let it     = 1;
                           %let cyc    = 3;
                         %end;
                  %else
                      %if &cyc=3 %then
                        %let flag= 1;
                            %else
                        endsas;
      %end;
```

```
%else
     %do;
         %let tm2= ROUND((&m2), .000000001);
         %let td2= (&d2);
           data _null_;
           %if &it=1 %then
               %do;
                 datm2= (&target - &d2) * (&m2) /&target;
               %end;
           %if &it ne 1 %then
               %do;
                       datm2= &m1 + (&m2-&m1)*(&d1/(&d1-&d2));
               %end;
           call symput('m2', datm2);
           run;
         %let m1= ROUND((&tm2), .000000001);
         %let d1= (&td2);
         data line;
           keep cycl nit targ tot datd2 f18amult f18bmult;
           label       cycl= 'Cycle'
                 nit=  'Iteration'
                 targ= 'Target'
                 tot=  'EQA_FEQA'
                 datd2= 'Error'
                 f18amult= 'W. mult.'
                 f18bmult= 'I. mult.';
           format      targ tot comma16. datd2 comma14. cycl nit 2.
                 f18amult f18bmult 12.10;
         cycl= &cyc;
         nit= &it;
         tot= &total;
         targ= &target;
         datd2= &td2;
         %if &cyc=1 %then
           %do;
               f18amult= ROUND((&tm2), .000000001);
           %end;
         %else
           %if &cyc=2 %then
               %do;
                 f18bmult= ROUND((&tm2), .000000001);
                 %end;
           %else
               %if &cyc=3 %then
                 %do;
                     f18amult= round((&tm2/&f18bp)*&f18ap, .000000001);
                     f18bmult= ROUND((&tm2), .000000001);
                  %end;
               nextit= &it + 1;
           call symput('it',nextit);
           run;
         data results;
           set results line;
      ******check error***;
        proc print;

     %end;
```

```
%end;

data results;
      set results;
      if _n_ ne 1;

proc print label data=results;
      id cycl nit;
      var targ tot datd2 f18amult f18bmult;
      title3 "Iteration results";
%mend;
```