Input Parameters for WPHACT 1.0

The way the input parameters must be given in the command file is easily understood just looking at the program lines containing the read statements. We therefore reproduce them and explain their meaning in the following.

Every parameter whose initial is `i` or `n` is `integer*4`. All others are `real*8`. When a variable has a yes/no option the value `1` corresponds to `YES`, `0` to `NO`. All energies must be expressed in GeV.

```
READ*,e_cm
```
`e_cm` is the centre of mass energy.

```
READ*,iproc
READ*,ich
```
The value for these two parameters must be read directly from Table 2 and 3. `iproc` selects the group of processes with the same cross section at parton level with symmetric cuts, while `ich` selects among them the specific final state. The choice of `ich` is relevant only if hadronization is performed via Jetset or if in CC processes cuts are considered for which a final configuration and its charge conjugate behave differently. In all other cases `ich=1` gives the same results as any other value indicated in the tables.

```
IF(iproc.GE.33)THEN
  READ*,rmb
  READ*,icch
  IF(icch.NE.2)THEN
    READ*,isusy
    IF(isusy.EQ.0)THEN
      READ*,rmh
    ELSE
      READ*,rma
      READ*,tgb
    ENDIF
  ENDIF
ENDIF
```
The processes for which `iproc` $\geq$ `33` are those with massive b's in the final state. `rmb` fixes the b mass for the phase space and the matrix elements. The b mass in the Higgs coupling may be different from `rmb` and it has to be set in the DATA. In these processes one can have diagrams with SM or Susy MSSM neutral (A or h) Higgs. One has the possibility of choosing to compute only these diagrams (Higgs signal: `icch=1`), only those without Higgs (to which we refer as Higgs background: `icch=2`) or the whole set of diagrams (Higgs+Background+interference: `icch=3`).
`isusy=0` corresponds to SM Higgs, `isusy=1` to Susy Higgs. In the first case one has to specify the Higgs mass (`rmh`), in the second the mass of the pseudoscalar Higgs A (`rma`), and the ratio between the two vacuum expectation values $tg(\beta)$ (`tgb`).

```
IF (iproc.GE.6.AND.iproc.LE.8) THEN
  READ*,iccnc
ENDIF
```

The processes for which 6 $\leq$ `iproc` $\leq$ 8 are those which have both CC and NC diagrams (Mixed). For these one may choose to compute only the CC contribution (`iccnc=1`), the NC one (`iccnc=2`) or the total process including CC+NC+interference (`iccnc=3`).

```
READ*,ips_cc
READ*,ips_nc
```

`ips_cc` and `ips_nc` allow to choose among the various phase space mappings for the integration. `ips_cc` refers to the phase space of CC or Higgs signal contributions. `ips_nc` to NC contributions. Of course when only CC processes or only Higgs signal are considered `ips_nc` is irrelevant. The same happens to `ips_cc` when only neutral processes without Higgs are computed. Both parameters can assume 3 values: `1` for double resonant mapping, `2` for single resonant, `3` for non resonant. When `2` is chosen the invariant mass over which a transformation is performed to take care of the resonant peak is that formed by particles 3 and 4.

```
READ*,icc3
```

yes/no CC3 contribution only. When `icc3` is set to `1` only the three double resonant diagrams (CC3) corresponding to WW production and decay are computed. If `icc3=0` all CC contributions are computed.

```
READ*,isr
```

yes/no ISR: initial state radiation (ISR) is included when `isr=1`, not computed if `isr=0`.

```
READ*,ipr
```

yes/no running widths: this flag selects among running or constant Z, W, Higgs widths in s-channel propagators:

`ipr=0` Z, W, Higgs boson constant width

`ipr=1` Z, W, Higgs boson s-dependent width

```
READ*,iswgcomp
```

yes/no $sin^2\theta_W$ and g computed. If this flag is set to `1`, it is used the renormalization scheme in which $sin^2\theta_W$ and g are computed from Z mass,W mass, G_f. If it is set to `0`, the values for $sin^2\theta_W$ and α_{em} are taken from the DATA. The relation between g and α_{em} is always $g^2 = 4\pi\alpha_{em}/sin^2\theta_W$.

```
READ*,igwcomp,igzcomp,ighcomp
```

yes/no W,Z,H width computed. When one of these flags is `=1`, the corresponding W, Z or Higgs width is computed by standard formulas. If it is `=0`, the value for the corresponding width is the one given in the DATA.

```
READ*,icoul
```

Coulomb corrections may (`icoul=1`) or not (`icoul=0`) be computed.

```
READ*,istrcor
```
yes/no 'naive' QCD corrections to the cross sections. It has to be noticed that, if `istrcor=0` and quarks are present in the final state, the eventual width computations (performed when `igwcomp, igzcomp or ighcomp` is `=1`) do not include QCD corrections.

```
READ*,iqu
```
yes/no QCD diagrams for 4-quarks NC. The order at which one computes four fermion diagram is α^4. There are some diagrams in NC processes $O(\alpha^2\alpha_s^2)$ whose contribution can be important and can be included choosing `iqu=1`.

```
READ*,icut
IF(icut.EQ.1)THEN
  READ*,e_min
  READ*,e_max
  READ*,rm_min
  READ*,rm_max
  READ*,pt_min
  READ*,pt_max
  READ*,icos
  READ*,thbeam_min
  READ*,thbeam_max
  READ*,thsep_min
  READ*,thsep_max
ENDIF
```
Cuts may (`icut=1`) or not (`icut=0`) be implemented. If `icut=1` all default cuts of the above list must be specified.
`e_min` and `e_max` correspond to the 4 lower and upper energies for particle 3, 4, 5, 6 respectively.
`rm_min` and `rm_max` are the 6 invariant mass lower and upper limits respectively. They must be given in the following order: m(34), m(35), m(36), m(45), m(46), m(56).
`pt_min` and `pt_max` are the 4 lower and upper values of the transverse momenta. The order is as before 3, 4, 5, 6.
`icos = 1` implies that the following angular cuts must be expressed in terms of the cosines of the angles. With `icos = 0` one must instead specify the angles in degrees.
`thbeam_min` and `thbeam_max` are the 4 lower and the 4 upper limits for the angle that particles 3,4,5,6 produce with the beam (e^+).
`thsep_min` and `thsep_max` are the 6 lower and the 6 upper limits for particle-particle angles: the order is again (3 4), (3 5), (3 6), (4 5), (4 6), (5 6).

```
READ*,ianc
IF(ianc.EQ.1)THEN
  READ*, delz,xf,xz,yf,yz,zz
ENDIF
```

Anomalous couplings contributions may (`ianc=1`) or not (`ianc=0`) be computed. If one wants to compute them, he must also specify the values of the parameters $\delta_z, x_\gamma, x_Z, y_\gamma, y_Z, z_Z$ defined in M. Bilenky, J.L. Kneur, F. M. Renard, D. Schildknecht *Nucl. Phys.* **B409** (1993) 22.

```
READ*,idistr
IF(idistr.EQ.1)THEN
  READ*,ndistr
  DO i=1,ndistr
    READ*,nsubint(i)
    READ*,(distr_estrinf(i,j),j=1,nsubint(i)+1)
    READ*,(nbin_number(i,j),j=1,nsubint(i))
  ENDDO
ENDIF
```

Distributions at the parton level can be implemented very easily. If one wants to use this possibility the flag `idistr` must be `=1`, `=0` otherwise. In the program there is the line:

```
* include 'abdis.dis'
```

which must be uncommented before compiling if one choose `idistr=1`. In this case the file `abdis.dis` must be written for implementing distributions, as in the following example:

```
      string(1)='Distribution: bb~ invariant mass'
      distr_var(1)=sqrt((p3(0)+p4(0))**2-(p3(1)+p4(1))**2-
     &  (p3(2)+p4(2))**2-(p3(3)+p4(3))**2)
      string(2)='Distribution: Charged lepton energy '
      distr_var(2)=p5(0)
```

In it, the title of the ith distribution is given in the character string(i) and the ith quantity to be distributed in bins, computed from the particle momenta, is assigned to `distr_var(i)`. The resulting cross sections, corresponding to every single bin will be stored in the file abdis.dat, where each line will contain 3 numbers: the value of the central point of the bin, the distribution for the bin (cross section divided by the width of the bin, in order to reproduce $d\sigma/dx$ for a distribution of the variable x) and the estimated statistical error.

`ndistr` is the number of distributions defined in `abdis.dis`.

For each distribution i, one must specify:

`nsubint(i)`, the number of sub-intervals with different binning in the ith distribution (=1 when all bins are of the same length). The subintervals must be contiguous.

`(distr_estrinf(i,j),j=1,nsubint(i)+1)`, the lower limits of each subinterval (which coincide with the upper limit of the previous one as they must be contiguous) and, as last entry, the upper limit of the last subinterval. In case all bins are of the same length, this corresponds only to the lower and upper limit of the interval for the distribution.

`(nbin_number(i,j),j=1,nsubint(i))`, the number of bins in each subinterval.

```
READ*,iflat
IF(iflat.EQ.1)THEN
  READ*,scalemax
  READ*,istorvegas
  READ*,irepeat
  IF(irepeat.eq.2)THEN
    READ*, nflevts
  END IF
  READ*,istormom
  READ*,ijetset
  IF(iproc.GE.6.AND.iproc.LE.8)THEN
    READ*,interf
  ENDIF
ENDIF
```

One may choose (`iflat=1`) or not (`iflat =0`) to generate unweighted events. In the first case, the number of iterations (`itmx`) which must be specified in the following must be 2. The integration routine will perform the requested number of iterations for thermalization (see `iterm, ncall_term,itmx_term` below) and then the two iterations in which the integral is evaluated. In the first iteration the maximum for the hit-or-miss procedure will be determined and used in the second iteration where the unweighted generation will take place. After the run the .log file will report as usual the result of the integration and its error. It will also report the maximum used for the hit-or-miss procedure, the maximum found in the second iteration, and the number of events which were greater than the maximum used.

There is also the possibility to repeat the generation just starting directly from the second iteration. This might be useful if too many events exceeded the maximum chosen.

Another interesting possibility is to rerun the program asking for a predetermined number of unweighted events to generate. So one may for instance run the first time just to find an appropriate maximum and a correct cross section (and eventually to produce partonic distributions). One controls then if it is better to multiply the maximum for an appropriate factor and with a second run one generates the desired number of events. To save time, the hadronization procedure with the link to JETSET may be activated only in this second run.

`scalemax` is the coefficient by which the maximum of the first iteration can be multiplied, in order to vary the efficiency of the hit-or-miss procedure or in order to avoid values exceeding the maximum.

VEGAS data are (if `istorvegas=1`) or not (if `istorvegas=0`) stored in ABVEGAS.DAT (in ABVEGAS_CC.DAT and ABVEGAS_NC.DAT for mixed and Higgs+background processes) after the first iteration. Stored Vegas data are necessary if one wants to rerun the programs to generate again unweighted events. When the program is rerun using Vegas data stored, the maximum of the second iteration will be automatically used as the new maximum.

`irepeat` has to be set to `0` for the first run It has to be set to `1` if one wants to rerun exactly with the same input starting from the second iteration. In this case the same weighted points will be riproduced. `irepeat=2` has to be chosen if one wants to rerun with the same imput and grid as before, but letting the program run until a requested number of events `nfltevts` is reached. For both cases `irepeat=1 and 2` one might of course vary `scalemax`, `ijetset` or `istormom`.
The momenta of the unweighted events are written in ABMOM.DAT (or ABMOM_SIGN.DAT and ABMOM_BACK.DAT) file if `istormom=1`, they are not written if `istormom=0`. ABMOM_SIGN.DAT is used for CC or Higgs events, ABMOM_BACK.DAT for NC events when mixed or Higgs+background processes are computed.
Every unweighted event is passed to the standard COMMON HEPTV. If `ijetset=1` the subroutine AB_LU4FRM is called. This routine is an interface to Jetset that we have produced modifying the subroutine LU4FRM by Sjostrand to adapt it to WPHACT. The comment lines of the routine AB_LU4FRM may be useful. If `ijetset=0`, HEPTV is still filled but the interface to Jetset not called.
One may choose to produce events for mixed processes in which the interference is added to CC (`interf= 1`), or to NC (`interf=0`). This implies that also the interference will be considered to have the same color structure of CC or NC respectively (see section 5.3 of 'QCD event generators' vol. 2 in Report of the Workshop on Physics at LEP2, G. Altarelli, T. Sjostrand and F.Zwirner (eds), CERN 96-01).

```
READ*,acc
READ*,iterm
READ*,ncall_term
READ*,itmx_term
READ*,ncall
READ*,itmx
```

These parameters specify how the integration will be performed by VEGAS.
`acc` is the integration accuracy. When this accuracy is reached after a certain integration iteration, the remaining iterations are not performed.
If `iterm=1` a certain number of integration iterations (`=itmx_term`) are used only for adapting the integration grid. Their result is not used for the final integral. Each thermalizing iteration makes use of a maximum of `ncall_term` evaluations of the amplitude. If `iterm=0` these iterations are skipped. One or two `itmx_term` with few `ncall_term` are often useful.
`itmx` is the maximum number of iterations used to evaluate the integral. Each iteration makes use of a maximum of `ncall` evaluations of the amplitude. Normally it is better not to use more than about five iterations. If higher precision is requested it is convenient to increase `ncall` and not `itmx`
As a final remark about the choiche of these parameters, one must be aware of the fact that final results with a χ^2 much greater than the number of iterations are not to be trusted. When this happens, one has to increase `ncall`.

CC					
process type	iproc	ich	final state	ich	final state
CC9	1	1	$\mu^- \; \bar{\nu}_\mu \; \nu_\tau \; \tau^+$	2	$\mu^+ \; \nu_\mu \; \bar{\nu}_\tau \; \tau^-$
CC18	2	1	$e^- \; \bar{\nu}_e \; \nu_\mu \; \mu^+$	3	$e^+ \; \nu_e \; \bar{\nu}_\mu \; \mu^-$
		2	$e^- \; \bar{\nu}_e \; \nu_\tau \; \tau^+$	4	$e^+ \; \nu_e \; \bar{\nu}_\tau \; \tau^-$
CC10	3	1	$\mu^- \; \bar{\nu}_\mu \; u \; d$	5	$\tau^- \; \bar{\nu}_\tau \; u \; d$
		2	$\mu^- \; \bar{\nu}_\mu \; c \; \bar{s}$	6	$\tau^- \; \bar{\nu}_\tau \; c \; \bar{s}$
		3	$\mu^+ \; \nu_\mu \; \bar{u} \; d$	7	$\tau^+ \; \nu_\tau \; \bar{u} \; d$
		4	$\mu^+ \; \nu_\mu \; \bar{c} \; s$	8	$\tau^+ \; \nu_\tau \; \bar{c} \; s$
CC20	4	1	$e^- \; \bar{\nu}_e \; u \; d$	3	$e^+ \; \nu_e \; \bar{u} \; d$
		2	$e^- \; \bar{\nu}_e \; c \; \bar{s}$	4	$e^+ \; \nu_e \; \bar{c} \; s$
CC11	5	1	$s \; \bar{c} \; u \; d$	2	$\bar{s} \; c \; \bar{u} \; d$

MIX					
process type	iproc	ich	final state	ich	final state
MIX19	6	1	$\mu^- \; \mu^+ \; \nu_\mu \; \bar{\nu}_\mu$	2	$\tau^- \; \tau^+ \; \nu_\tau \; \bar{\nu}_\tau$
MIX56	7	1	$e^- \; e^+ \; \nu_e \; \bar{\nu}_e$		
MIX43	8	1	$d \; d \; u \; \bar{u}$	2	$s \; \bar{s} \; c \; \bar{c}$

Table 2

NC					
process type	iproc	ich	final state	ich	final state
NC6	9	1	$\nu_\mu\ \bar\nu_\mu\ \nu_\tau\ \bar\nu_\tau$		
NC12	10	1	$\nu_\mu\ \bar\nu_\mu\ \nu_e\ \bar\nu_e$	2	$\nu_\tau\ \bar\nu_\tau\ \nu_e\ \bar\nu_e$
NC12	11	1	$\nu_\mu\ \bar\nu_\mu\ \nu_\mu\ \bar\nu_\mu$	2	$\nu_\tau\ \bar\nu_\tau\ \nu_\tau\ \bar\nu_\tau$
NC36	12	1	$\nu_e\ \bar\nu_e\ \nu_e\ \bar\nu_e$		
NC10	13	1	$u\ \bar u\ \nu_\mu\ \bar\nu_\mu$	3	$c\ \bar c\ \nu_\mu\ \bar\nu_\mu$
		2	$u\ \bar u\ \nu_\tau\ \bar\nu_\tau$	4	$c\ \bar c\ \nu_\tau\ \bar\nu_\tau$
NC19	14	1	$u\ \bar u\ \nu_e\ \bar\nu_e$	2	$c\ \bar c\ \nu_e\ \bar\nu_e$
NC64	15	1	$u\ \bar u\ u\ \bar u$	2	$c\ \bar c\ c\ \bar c$
NC32	16	1	$u\ \bar u\ c\ \bar c$		
NC10	17	1	$\mu^-\ \mu^+\ \nu_\tau\ \bar\nu_\tau$	2	$\tau^-\ \tau^+\ \nu_\mu\ \bar\nu_\mu$
NC20	18	1	$e^-\ e^+\ \nu_\mu\ \bar\nu_\mu$	2	$e^-\ e^+\ \nu_\tau\ \bar\nu_\tau$
NC19	19	1	$\mu^-\ \mu^+\ \nu_e\ \bar\nu_e$	2	$\tau^-\ \tau^+\ \nu_e\ \bar\nu_e$
NC24	20	1	$\mu^-\ \mu^+\ u\ \bar u$	3	$\tau^-\ \tau^+\ u\ \bar u$
		2	$\mu^-\ \mu^+\ c\ \bar c$	4	$\tau^-\ \tau^+\ c\ \bar c$
NC48	21	1	$e^-\ e^+\ u\ \bar u$	2	$e^-\ e^+\ c\ \bar c$
NC19	22	1	$d\ \bar d\ \nu_e\ \bar\nu_e$	2	$s\ \bar s\ \nu_e\ \bar\nu_e$
NC10	23	1	$d\ d\ \nu_\mu\ \bar\nu_\mu$	3	$d\ d\ \nu_\tau\ \bar\nu_\tau$
		2	$s\ \bar s\ \nu_\mu\ \bar\nu_\mu$	4	$s\ \bar s\ \nu_\tau\ \bar\nu_\tau$
NC32	24	1	$s\ \bar s\ u\ \bar u$	2	$d\ \bar d\ c\ \bar c$
NC24	25	1	$\mu^-\ \mu^+\ \tau^-\ \tau^+$		
NC48	26	1	$e^-\ e^+\ \mu^-\ \mu^+$	2	$e^-\ e^+\ \tau^-\ \tau^+$
NC48	27	1	$\mu^-\ \mu^+\ \mu^-\ \mu^+$	2	$\tau^-\ \tau^+\ \tau^-\ \tau^+$
NC144	28	1	$e^-\ e^+\ e^-\ e^+$		
NC64	29	1	$d\ d\ d\ d$	2	$s\ \bar s\ s\ \bar s$
NC48	30	1	$e^-\ e^+\ d\ d$	2	$e^-\ e^+\ s\ \bar s$
NC24	31	1	$\mu^-\ \mu^+\ d\ \bar d$	3	$\tau^-\ \tau^+\ d\ \bar d$
		2	$\mu^-\ \mu^+\ s\ \bar s$	4	$\tau^-\ \tau^+\ s\ \bar s$
N32C	32	1	$d\ \bar d\ s\ \bar s$		

NC+HIGGS					
process type	iproc	ich	final state	ich	final state
NC21	33	1	$b\ b\ \nu_e\ \bar\nu_e$		
NC11	34	1	$b\ \bar b\ \nu_\mu\ \bar\nu_\mu$	2	$b\ \bar b\ \nu_\tau\ \bar\nu_\tau$
NC33	35	1	$b\ b\ u\ \bar u$	2	$b\ b\ c\ \bar c$
NC25	36	1	$b\ b\ \mu^-\ \mu^+$	2	$b\ b\ \tau^-\ \tau^+$
NC50	37	1	$b\ \bar b\ e^-\ e^+$		
NC33	38	1	$b\ \bar b\ d\ d$	2	$b\ \bar b\ s\ \bar s$
NC84	39	1	$b\ b\ \mathrm{b}\ b$		

Table 3