In the following, a general format for the input file of MITHRA is presented. The red icons or groups can be repeated in the text.

```
MESH
{
  length-scale                  = < real value | METER | DECIMETER | CENTIMETER |
                                    MILLIMETER | MICROMETER | NANOMETER | ANGSTROM >
  time-scale                    = < real value | SECOND | MILLISECOND | MICROSECOND |
                                    NANOSECOND | PICOSECOND | FEMTOSECOND | ATTOSECOND >
  mesh-lengths                  = < ( real value,  real value, real value) >
  mesh-resolution               = < ( real value,  real value, real value) >
  mesh-center                   = < ( real value,  real value, real value) >
  total-time                    = < real value >
  bunch-time-step               = < real value >
  bunch-time-start              = < real value >
  mesh-truncation-order         = < 1 | 2 >
  space-charge                  = < true | false >
}

BUNCH
{
  bunch-initialization
  {
    type                        = < manual | ellipsoid | 3D-crystal | file >
    distribution                = < uniform | gaussian >
    charge                      = < real value >
    number-of-particles         = < integer value >
    gamma                       = < real value >
    beta                        = < real value >
    direction                   = < ( real value, real value, real value ) >
    position                    = < ( real value, real value, real value ) >
    sigma-position              = < ( real value, real value, real value ) >
    sigma-momentum              = < ( real value, real value, real value ) >
    transverse-truncation       = < real value >
    longitudinal-truncation     = < real value >
    bunching-factor             = < real value between zero and one >
  }

  bunch-sampling
  {
    sample                      = < true | false >
    directory                   = < address according to UNIX convention >
    base-name                   = < name of the file >
    rhythm                      = < real value >
  }

  bunch-visualization
  {
    sample                      = < true | false >
    directory                   = < address according to UNIX convention >
    base-name                   = < name of the file >
    rhythm                      = < real value >
  }

  bunch-profile
  {
    sample                      = < true | false >
    directory                   = < address according to UNIX convention >
    base-name                   = < name of the file >
    time                        = < real value >
    rhythm                      = < real value >
```

```
        }
}

FIELD
{
  field-initialization
  {
    type                    = < plane-wave | confined-plane-wave | gaussian-beam >
    position                = < ( real value , real value , real value ) >
    direction               = < ( real value , real value , real value ) >
    polarization            = < ( real value , real value , real value ) >
    radius-parallel         = < real value >
    radius-perpendicular    = < real value >
    signal-type             = < neumann | gaussian | secant-hyperbolic | flat-top >
    strength-parameter      = < real value >
    offset                  = < real value >
    variance                = < real value >
    wavelength              = < real value >
    CEP                     = < real value >
  }

  field-sampling
  {
    sample                  = < true | false >
    type                    = < over-line | at-point >
    field                   = < Ex | Ey | Ez | Bx | By | Bz | Ax | Ay | Az | Jx | Jy
                              | Jz | F | Q >
    directory               = < address according to UNIX convention >
    base-name               = < name of the file >
    rhythm                  = < real value >
    position                = < ( real value , real value , real value ) >
    line-begin              = < ( real value , real value , real value ) >
    line-end                = < ( real value , real value , real value ) >
    resolution              = < real value >
  }

  field-visualization
  {
    sample                  = < true | false >
    field                   = < Ex | Ey | Ez | Bx | By | Bz | Ax | Ay | Az | Jx | Jy
                              | Jz | F | Q >
    directory               = < address according to UNIX convention >
    base-name               = < name of the file >
    rhythm                  = < real value >
  }

  field-profile
  {
    sample                  = < true | false >
    field                   = < Ex | Ey | Ez | Bx | By | Bz | Ax | Ay | Az | Jx | Jy
                              | Jz | F | Q >
    directory               = < address according to UNIX convention >
    base-name               = < name of the file >
    rhythm                  = < real value >
    time                    = < real value >
  }
}

UNDULATOR
{
  static-undulator
```

```
   {
     undulator-parameter       = < real value >
     period                    = < real value >
     length                    = < integer value >
     polarization-angle        = < real value >
     offset                    = < real value >
   }

   static-undulator-array
   {
     undulator-parameter       = < real value >
     period                    = < real value >
     length                    = < integer value >
     polarization-angle        = < real value >
     gap                       = < real value >
     number                    = < integer value >
     tapering-parameter        = < real value >
   }

   optical-undulator
   {
     beam-type                 = < plane-wave | confined-plane-wave | gaussian-beam >
     position                  = < ( real value , real value , real value ) >
     direction                 = < ( real value , real value , real value ) >
     polarization              = < ( real value , real value , real value ) >
     radius-parallel           = < real value >
     radius-perpendicular      = < real value >
     signal-type               = < neumann | gaussian | secant-hyperbolic | flat-top >
     strength-parameter        = < real value >
     offset                    = < real value >
     variance                  = < real value >
     wavelength                = < real value >
     CEP                       = < real value >
   }
}

EXTERNAL-FIELD
  {
  electromagnetic-wave
  {
     type                      = < plane-wave | confined-plane-wave | gaussian-beam >
     position                  = < ( real value , real value , real value ) >
     direction                 = < ( real value , real value , real value ) >
     polarization              = < ( real value , real value , real value ) >
     radius-parallel           = < real value >
     radius-perpendicular      = < real value >
     signal-type               = < neumann | gaussian | secant-hyperbolic | flat-top >
     strength-parameter        = < real value >
     offset                    = < real value >
     variance                  = < real value >
     wavelength                = < real value >
     CEP                       = < real value >
  }
}

FEL-OUTPUT
{
  radiation-power
  {
     sample                    = < false | true >
     type                      = < at-point | over-line >
```

```
        directory                = < address according to UNIX convention >
        base-name                = < name of the file >
        plane-position           = < real value >
        line-begin               = < real value >
        line-end                 = < real value >
        resolution               = < real value >
        normalized-frequency     = < real value >
        minimum-normalized-frequency      = < real value >
        maximum-normalized-frequency      = < real value >
        normalized-frequency-resolution   = < real value >
    }
}
```