

Appendix A: 1 kt Nuclear Blast Standard

This appendix contains a FORTRAN compilable listing of the 1 kt nuclear blast standard. This standard has been adopted as the blast standard by both the American National Standards Institute (ANSI) and the International Standards Committee. The input and output units are MKS, with the yield in kilotons. Internally the program uses cgs units; all scaling is handled internally. Program main is included to provide a simple example of how the standard can be used. The reader is encouraged to write their own driver.

The program includes a subroutine for the 1962 standard atmosphere which is used for scaling. This is a stand alone routine which can be used for many purposes.

The program also contains a subroutine for the equation of state for air. This is the equation of state referenced in Chap. 3. It includes the effects of vibrational and rotational excitation of nitrogen and oxygen, dissociation and first ionization levels of nitrogen and oxygen. Heed the warnings on limitations given in Chap. 3.

The subroutine “well” represents the fireball. At late times the fireball will rise out of the center of the nearly spherical blast wave bubble. This routine attempts to account for the fireball motion.

The function “energy1” takes as input the density and the pressure and iteratively finds the specific internal energy for air.

```
Program MAIN
  yield=1.
  height=100.
  tim=.25
  do 1 I=1,250
    ra=float(I)
    call shock(yield,height,tim,ra,opr,odr,vr)
    Print 950, tim,ra,opr,odr,vr
1   continue
950  format(1P8e12.4)
end
```

```
      subroutine shock (yield,height,tim,ra,opr,odr,vr)
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      alt=height
c
      call scalkt (alt,yield,vsl,dsl,tsl,csl,psl)
c
      t=tim*tsl
      r=ra/csl
c
      call peak (t,r,prado,oppko,odpko,opro,odro,vpko,vro)
c
      opr=opro*psl
      odr=odro*dsl
      vr=vro*vsl
c
      return
      end
```

```

      subroutine peak (t,ra,prado,oppko,odpko,opro,odro,vpko,vro)
c
      save
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfrt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c              ,vmn
c
      data told /0./
      data psca /0.1/, vsca /0.01/, rhosca /1000./
c
c   Convert distance to cm
c
      r=ra*100.
c
c   Calculate waveform radius at peak overpressure.
c
      if (t .ne. told) then
c
          rzp=wfzr(t)
          rzd=wfdzr(t)
          rzv=wfvzr(t)
          prad=wfpr(t)
c
          prado=prad*vsca
c
c   Calculate waveform peak overpressure.
c
          oppk=wfpkop(prad)
          oppko=oppk*psca
c
c   Calculate waveform peak overdensity.
c
          odpk=wfpkod(prad)
          odpko=odpk*rhosca
c
c   Calculate waveform peak velocity.
c
          vpk=wfpkv(prad)
          vpko=vpk*vsca
          told=t
      endif
c
      opr=0.
      odr=0.
      vr=0.
      opro=0.
      odro=0.
      vro=0.
c
      if (r .le. prad) then
c
c   Calculate overpressure at r.
c
          call wfprmt (t,r)

```

```
      opro=opr*psca
c
c   Calculate overdensity at r.
c
      call wfdrrmt (t,r)
      call well (t,r,odw)
      odro=odw*rhosca
c
c   Calculate velocity at r.
c
      call wfvrrmt (t,r)
      vro=vr*vsca
    endif
c
    return
end
```

```

      subroutine scalkt (hfpt,wb,vscale,dscale,tscale,cscale,pscale)
c
c      save
c
c      This routine sets up the scale factors for each burst so the 1 KT
c      sea level data may be used
c
c      Inputs
c
c      hfpt - Height of field point above sea level in meters
c      wb   - Yield of the burst in KT
c
c      Output
c
c      tscale - scales the actual time to the 1 KT sea level time
c      vscale - scales the 1 KT sea level velocities to actual
c      pscale - scales the 1 KT sea level pressures to actual
c      cscale - scales the 1 KT sea level dimensions to actual
c      dscale - scales the 1 KT sea level densities to actual
c
c      This routine is part of the AFWL 1KT Standard by Needham, et al.
c
c      dimension hn(4)
c      data p1 /1.01325e5/, c1 /3.4029399e2/, r1 /1.225/, t1 /288.15/
c
c      cubrt(x)=sign(exp(alog(abs(x)))/3. ),x)
c
c      call matm62 (hfpt,p3,c3,r3,t3)
c      rp3=cubrt(p1/p3)
c      wsw=cubrt(wb)
c      tscale=c3/(wsw*rp3*c1)
c      vscale=c3/c1
c      pscale=p3/p1
c      dscale=r3/r1
c      cscale=rp3*wsw
c
c      return
c      end

```

```
      subroutine terror (t, name)
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
c      character*6 name
c
c      write (6,2) t, name
c      stop
c
c 2 format (1x/' Time ERROR.'/' t = ',1pe13.5,' seconds in routine '
c          ,a6/)
c
c      end
```

```

      subroutine well (t,r,depth)
c
      save
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfprt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c              ,vmn
c
      depth=odr
      rmsw=amin1(0.7*rzd,1.55e4)
c
      if (t .le. 1.2 .and. r .le. rmsw) then
c
         depth=amax1(-1.21e-3,-1.5e-3*exp(-8.e-13*r**3))
         if (t .gt. 1.) depth=(1.2-t)*depth*5.
         if (r .ge. 0.9*rmsw) depth=10.*(depth*(rmsw-r)
c                               +odr*(r-0.9*rmsw))/rmsw
      endif
c
      return
      end

```

```

function wfzr (t)
c
c   save
c
c   Calculates the waveform radius at zero overpressure, (wfzr).
c   (i.e., the radius separating the negative and positive phase
c   portions of the waveform, at the specified time (t)).
c
c   wfzr is not defined for times less than approximately 0.1 sec
c   because for these times the waveform does not exhibit a
c   negative phase.
c
c   t      - Time (sec)
c   wfzr   - Radius (cm)
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
c   data b /0.03291/, c /-1.086/, cz /33897./, bz /8490./
c
c   wfzr=0.
c
c   if (t .lt. 0.) then
c       call terror (t, 'wfzr  ')
c   else
c       wfzr=(1.-b*t**c)*(cz*t+bz)
c   endif
c
c   return
c   end

```



```

        function wfpr(t)
c
        save
c
c  This function also calculates the shock front radius at the
c  specified time.
c
        r1=24210.*t**0.371*(1.+(1.23*t+0.123)*(1.-exp(-26.25*t**0.79)))
        r2=(1.-0.03291*t**(-1.086))*(33897.*t+8490.)+8.36e3+2.5e3
c    *alog(t)+800.*t**(-0.21)
c
        if (t .lt. 0.21) then
            wfpr=r1
        else if (t .gt. 0.28) then
            wfpr=r2
        else
            wfpr=(r2*(t-0.21)+r1*(0.28-t))/0.07
        endif
c
        return
        end

```

```

      function wfpkod (dummy)
c
      save
c
c   This routine computes the peak overdensity given the peak
c   overpressure using Rankine-Hugoniot relations.
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfrt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c              ,vmn
c
      data rhoz /1.225e-3/, opz /1.01325e6/, gamm1 /0.404574/
c
      op=oppk
      rtio=op/opz
      p=op+opz
      gmone=gamm1
      gamma=gamm1+1.
      gamra=gamma
c
      do 2 n=1,20
      rho1=rhoz*((2.*gamra+(gamra+1.)*rtio)/(2.*gamra+(gamra-1.)*rtio))
      ee=p/(gmone*rho1)
      call air (ee,rho1,gmone,dum1,dum2)
      gamrao=gamra
      gamra=2.*gmone/(2.5*gmone+1.)+1.
      if (abs(gamra-gamrao) .lt. 1.e-5) go to 3
2 continue
c
3 wfpkod=rho1-rhoz
c
      return
      end

```

```
      function wfpkv (dummy)
c
      save
c
c   This routine computes the particle velocity at peak overpressure
c   given the overdensity and overpressure using Rankine-Hugoniot
c   relations.
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfprt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c              ,vmn
c
      data rhoz /1.225e-3/
c
      wfpkv=sqrt(oppk*odpk/(rhoz*(rhoz+odpk)))
c
      return
      end
```

```

      subroutine wfdrmt (t,r)
c
      save
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfprt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c              ,vmn
c
      data ttold /0./
c
      rpk=prad
c
      if (t .ge. 0.2) then
c
        rpk=rpk*1.e-5
        r=r*1.e-5
c
        if (t .ne. ttold) then
          rz=rzd
          rmn=rz-9.7163e3*t**0.12115
          rz=rz*1.e-5
          rmn=rmn*1.e-5
          odmn=-0.5*odpk+2.2e-5*t**(-1.6026)
          rneg=rz-rmn
          rpls=rpk-rz
          rnp=rpk-rmn
          aln=odpk/rpls
          bln=odpk-aln*rpk
          odmln=aln*rmn+bln
          fmlt=abs(odmn/odpk)
          odmhy=odmn+fmlt*(odmln-odmn)
          alpha=(rnp/(odmhy-odpk)+rpls/odpk)/rneg
          beta=-rpls*(alpha+1./odpk)
          fngz=rnp/rpls
          dnom=alpha*rnp+beta
          bcrmn=1.-odmn/(rnp/dnom+odpk)
          crmlb=log(bcrmn)
          cgz1=(beta*(1./bcrmn-1.)/dnom)/((fngz*crmlb
c              *rnp** (fngz-1.))*(rnp+odpk*dnom))
          cgz=exp(cgz1)
          bgz1=crmlb/cgz** (rnp**fngz)
          bgz=exp(bgz1)
        endif
c
        rbr=rpk-r
        gr=1.-bgz** (cgz** (rbr**fngz))
        if (r .gt. rz) gr=(rpk-r)/rpls*gr+(r-rz)/rpls
        hr=rbr/(alpha*rbr+beta)+odpk
        odr=gr*hr
        rpk=rpk*1.e5
        r=r*1.e5
      endif
c
      if (t .le. 1.) then
c

```

```

        if (t .ne. ttold) then
            a=-1.2e-3
            c=log(max(1.e-20, -a/(odpk-a)))/(rzd-rpk)
            b=(odpk-a)*exp(-c*rpk)
        endif
c
        wflt=a+b*exp(c*r)
c
        if (t .le. 0.2) then
            odr=wflt
        else
            odr=(wflt*(1.-t)+odr*(t-0.2))*1.25
        endif
    endif
c
    ttold=t
c
    return
end

```

```
function wfdzr (t)
c
  save
c
c  This routine is part of the AFWL 1KT Standard by Needham, et al.
c
  data b /0.03499/, c /-1.068/, cz /33897./, bz /8490./
c
  wfdzr=0.
c
  if (t .lt. 0.) then
    call terror (t, 'wfdzr ')
  else if (t .gt. 0. .and. t .lt. 0.265) then
    wfdzr=2.568e4*t**0.395
  else
    wfdzr=(1.-b*t**c)*(cz*t+bz)+500.
  endif
c
  return
end
```

```

      subroutine wfprmt (t,r)
c
      save
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfprt/ prad,opk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c          ,vmn
c
      data ttold /0./
c
      rpk=prad
c
      if (t .ge. 0.1) then
c
        rpk=rpk*1.e-5
        r=r*1.e-5
c
        if (t .ne. ttold) then
          rz=rzp
          rmn=rz-9.7163e3*t**0.12115
          rz=rz*1.e-5
          rmn=rmn*1.e-5
          opmn=2.2e4/(t**sqrt(t))-0.5*opk
          rneg=rz-rmn
          rpls=rpk-rz
          rnp=rpk-rmn
          aln=opk/rpls
          bln=opk-aln*rpk
          opmnl=aln*rmn+bln
          fmlt=abs(opmn/opk)
          opmhy=opmn+fmlt*(opmnl-opmn)
          alpha=(rnp/(opmhy-opk)+rpls/opk)/rneg
          beta=-rpls*(alpha+1./opk)
          fngz=rnp/rpls
          dnom=alpha*rnp+beta
          bcrmn=1.-opmn/(rnp/dnom+opk)
          crmnlb=log(bcrmn)
          cgz1=(beta*(1./bcrmn-1.)/dnom)/((fngz*crmnlb
c              *rnp**(fngz-1.))*(rnp+opk*dnom))
          cgz=exp(cgz1)
          bgz1=crmnlb/cgz**(rnp**fngz)
          bgz=exp(bgz1)
        endif
c
        rbr=rpk-r
        gr=1.-bgz**(cgz**(rbr**fngz))
        if (r .gt. rz) gr=(rpk-r)/rpls*gr+(r-rz)/rpls
        hr=rbr/(alpha*rbr+beta)+opk
        opr=gr*hr
        rpk=rpk*1.e5
        r=r*1.e5
      endif
c
      if (t .lt. 0.95) then
c

```

```

      if (t .ne. ttold) then
        a=5.446e4*t**(-1.22)-7.135e5*(1.-t**2)
        if (t .lt. 0.2) c=7.41e-5*t**(-0.885)
        if (t .ge. 0.2) c=alog(max(1.e-20,-a/(oppk-a)))/(rzp-rpk)
        b=(oppk-a)*exp(-c*rpk)
      endif
c
      wflt=a+b*exp(c*r)
c
      if (t .lt. 0.1) then
        opr=wflt
      else
        opr=(wflt*(0.95-t)+opr*(t-0.1))/0.85
      endif
    endif
c
    ttold=t
c
    return
  end

```



```
      function wfvzr (t)
c
      save
c
c  This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      data b /0.08459/, c /-1.34/, cz /33897./, bz /8490./
c
      wfvzr=0.
c
      if (t .lt. 0.) then
        call terror (t, 'wfvzr ')
      else if (t .gt. 0. .and. t .lt. 0.263) then
        wfvzr=2.5e4*t**0.8
      else
        wfvzr=(1.-b*t**c)*(cz*t+bz)
      endif
c
      return
      end
```

```

      subroutine wfvrm (t,r)
c
      save
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
      common /wfrt/ prad,opk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c                ,vmn
c
      data ttold /0./
c
      rpk=prad
c
      if (t .ge. 0.15) then
c
        rpk=rpk*1.e-5
        r=r*1.e-5
c
        if (t .ne. ttold) then
          rz=rzv
          rmn=rz-9.31e3*t**0.12115
          rz=rz*1.e-5
          rmn=rmn*1.e-5
          vmn=650./(t*sqrt(t))-0.5*vpk
          rneg=rz-rmn
          rpls=rpk-rz
          rnp=rpk-rmn
          aln=vpk/rpls
          bln=vpk-aln*rpk
          ovmnl=aln*rmn+bln
          fmlt=abs(vmn/vpk)
          ovmhy=vmn+fmlt*(ovmnl-vmn)
          alpha=(rnp/(ovmhy-vpk)+rpls/vpk)/rneg
          beta=-rpls*(alpha+1./vpk)
          fngz=rnp/rpls
          dnom=alpha*rnp+beta
          bcrmn=1.-vmn/(rnp/dnom+vpk)
          crmnlb=log(bcrmn)
          cgz1=(beta*(1./bcrmn-1.)/dnom)/((fngz*crmnlb
c                *rnp** (fngz-1.))*(rnp+vpk*dnom))
          cgz=exp(cgz1)
          bgz1=crmnlb/cgz** (rnp**fngz)
          bgz=exp(bgz1)
        endif
c
        rbr=rpk-r
        gr=1.-bgz** (cgz** (rbr**fngz))
        if (r .gt. rz) gr=(rpk-r)/rpls*gr+(r-rz)/rpls
        hr=rbr/(alpha*rbr+beta)+vpk
        vr=gr*hr
        rpk=rpk*1.e5
        r=r*1.e5
      endif
c
      if (t .le. 0.2) then
c

```

```
        rtio=r/prad
        wflt=vpk*rtio*sqrt(rtio)
c
        if (t .lt. 0.15) then
            vr=wflt
        else
            vr=(wflt*(0.20-t)+vr*(t-0.15))*20.
        endif
    endif
c
    ttold=t
c
    return
end
```

```

function wfpkop (r)
c
c   save
c
c   Calculates the waveform peak overpressure (wfpkop) at the specified
c   radius (r), i.e., the overpressure at the shock front having the
c   specified radius (r).
c
c   r           - Radius(cm)
c   wfpkop      - Peak overpressure (dynes/cm**2)
c
c   This routine is part of the AFWL 1KT Standard by Needham, et al.
c
c   common /wfprt/ prad,oppk,odpk,vpk,opr,odr,vr,rzp,rzd,rzv,opmn,odmn
c           ,vmn
c
c   data ac /3.04e18/, aq /1.13e14/, astar /7.9e9/, rstar /4.454e4/
c
c   rr=1./r
c   rtio=2.24517e-5*r
c   cf=sqrt(alog(rtio+3.*exp(-(sqrt(rtio)/3.))))
c   wfpkop=((ac*rr+aq)*rr+astar/cf)*rr
c
c   return
c   end

```

```

function speed (t)
c
  save
c
c  This routine is the time derivative of wfpr2.  It returns the
c  shock front speed as a function of time.
c
c  This routine is part of the AFWL 1KT Standard by Needham, et al.
c
  r1=24210.*t**0.371*(1.+(1.23*t+0.123)*(1.-exp(-26.25*t**0.79)))
  r2=(1.-0.03291*t**(-1.086))*(33897.*t+8490.)+8.36e3+2.5e3*
c  alog(t)+800.*t**(-0.21)
  v1=10086.685*t**(-0.629)+40826.049*t**0.371+exp(-26.25*t**0.79)*
c  (617527.5*t**1.161-40826.049*t**0.371-1104.7749*t**(-0.629)+
c  61752.75*t**0.161)
  v2=33897.+95.937326*t**(-1.086)+303.43481*t**(-2.086)+2.5e3/t-
c  168.*t**(-1.21)
  v3=(v2*(t-0.21)+v1*(0.28-t))/0.07
c
  if (t .lt. 0.21) then
    speed=v1
  else if (t .gt. 0.28) then
    speed=v2
  else
    speed=v3
  endif
c
  return
end

```

```

subroutine air (eee,rrr,gmone,p,temp)
c
  e=eee*1.0e-10
  rhoIn=aalog(773.39520495*rrr)
  e1=(8.5-e)*1.0256410256
  if (abs(e1) .lt. 5.) go to 20
  if (e1 .le. 0.) go to 10
  fo=exp(-0.22421524664*e)
  fon=0.
  ws=1.
  go to 30
c
10 fo=0.
  fon=exp(-0.15082956259*e)
  ws=0.
  go to 30
c
20 ws=(8.5+0.15504314*rhoIn-e)*exp(-0.05*rhoIn+0.02531780798)
  ws=1./(exp(-ws)+1.)
  fo=exp(-0.22421524664*e)*ws
  fon=exp(-0.15082956259*e)*(1.-ws)
c
30 beta=0.
  if (e .gt. 1.) beta=(6.9487e-3*ws+1.38974e-2)*aalog(e)
  e2=(e-40.)*0.33333333333333
  if (abs(e2) .lt. 5.) go to 50
  if (e2 .gt. 0.) go to 40
  fn=0.
  ws=0.
  go to 60
c
40 fn=exp(-0.039215686275*e)
  ws=1.
  go to 60
c
50 ws=(e-exp(0.0157*rhoIn+3.806662489))*exp(-0.085*rhoIn-1.38629436)
  ws=1.0/(exp(-ws)+1.0)
  fn=exp(-0.039215686275*e)*ws
c
60 beta=beta+ws*(0.045-beta)
  ws=(e-160.)/amax1(30.+3.474356*rhoIn,6.)
  fe=0.
  if (ws .gt. -5.) fe=1./(exp(-ws)+1.)
  gm=(0.161+0.255*fo+0.28*fon+0.137*fn+0.05*fe)*exp(beta*rhoIn)
  gmone=gm
  p=gmone*eee*rrr
  rhols=rhoIn*rhoIn
  if (e .gt. 120.) go to 111
  f=9.72e-1-2.71434e-3*rhoIn+6.582549e-5*rhols
  g=2.645e-2-6.418873e-4*rhoIn-2.338785e-5*rhols
  h=-9.21e-5+5.971549e-6*rhoIn+3.923123e-7*rhols
  con1=3480.*gm*e
  con2=f+g*e+h*e*e
  temp=con1/con2
  go to 222
c

```

```
111 alr=log10(rrr)
    c1=1.69081e-5+2.99265e-7*alr
    c2=7.69813e-1+3.8618e-3*alr
    temp=c1*eee**c2+102.6275735
c
222 continue
    beta=(e-3.)*1.5151515151
    if (beta .gt. 10.) return
    swit=1./(exp(beta)+1.)
    temp=con1/(con2*(1.-swit)+swit)
    if (temp .gt. 0.) return
    alr=log10(rrr)
    c1=1.69081e-5+2.99265e-7*alr
    c2=7.69813e-1+3.8618e-3*alr
    temp=c1*eee**c2+102.6275735
c
    return
end
```

```

      subroutine matm62 (ty,wsp,cs,wsr,wst)
c
      save
c
c   This routine supplies internal specific energy and density.
c
c   This routine is part of the DNA 1-Kt Standard by Needham, et al.
c
c   calculate atmosphere
c
c   tabat(1)=r, the gas constant in ergs/mole/degree
c   tabat(2)=radius of the earth in cm.
c   tabat(3)=acceleration due to gravity at sea level in cm/sec**2
c   tabat(4)=molecular weight of air at sea level
c
c   tabz is altitude in cm.
c   tabt is molecular scale temperature in degrees kelvin.
c   tabl is molecular scale temperature gradient in deg./cm.
c   tabp is pressure in dynes/cm**2.
c
c   nz is the number of altitudes.  tabz, tabt, and tabp are
c   dimensioned nz.  tabl is dimensioned nz-1.
c
c
c   dimension tabat(4), tabz(22), tabl(21), tabt(22), tabp(22)
c
c   logical init
c
c   data init/.false./
c
c   Data for temperate atmosphere
c
c   data nz/21/,rhoz/1.22500000e-03/,tabat/8.31440000e+07,6.35670000e+
108,9.80665000e+02,2.89644000e+01/
c
c   data tabz/
1  0. , 1.10190000e+06, 2.00630000e+06, 3.21620000e+06,
2  4.73500000e+06, 5.24290000e+06, 6.15910000e+06, 7.99940000e+06,
3  9.00000000e+06, 1.00000000e+07, 1.10000000e+07, 1.20000000e+07,
4  1.50000000e+07, 1.60000000e+07, 1.70000000e+07, 1.90000000e+07,
5  2.20000000e+07, 3.00000000e+07, 4.00000000e+07, 5.00000000e+07,
6  6.00000000e+07, 7.00000000e+07/
c
c   data tabl/
1-6.49291769e-05, 9.28018576e-08, 9.86255062e-06, 2.77080373e-05,
2-1.72248474e-07, -1.96000240e-05, -3.91696897e-05, 1.60821507e-07,
3  2.98166740e-05, 5.02020080e-05, 9.97762300e-05, 2.00108809e-04,
4  1.49589024e-04, 1.00407490e-04, 6.97598500e-05, 6.68801467e-05,
5  3.49035000e-05, 3.31099360e-05, 2.58868500e-05, 1.71252960e-05,
6  1.09162420e-05/
c
c   data tabt/
1  2.88150000e+02, 2.16604540e+02, 2.16688470e+02, 2.28621170e+02,
2  2.70704137e+02, 2.70616652e+02, 2.52659110e+02, 1.80575130e+02,
3  1.80736048e+02, 2.10552722e+02, 2.60754730e+02, 3.60530960e+02,
4  9.60857386e+02, 1.11044641e+03, 1.21085390e+03, 1.35037360e+03,

```



```

5 1.55101404e+03, 1.83024204e+03, 2.16134140e+03, 2.42020990e+03,
6 2.59146286e+03, 2.70062528e+03/
c
    data tabp/
1 1.01325000e+06, 2.26320000e+05, 5.47486994e+04, 8.68013979e+03,
2 1.10904998e+03, 5.90004987e+02, 1.82098959e+02, 1.03769924e+01,
3 1.64379881e+00, 3.00749781e-01, 7.35439451e-02, 2.52169805e-02,
4 5.06169601e-03, 3.69429709e-03, 2.79259780e-03, 1.68519867e-03,
5 8.67381898e-04, 1.94317430e-04, 4.15743157e-05, 1.13023464e-05,
6 3.55894454e-06, 1.22936355e-06/
c
c Check if initialized.
c
1 if (init) go to 2
c
c Initialize.
c
    re=tabat(2)
    cons=re**2*tabat(3)*tabat(4)/tabat(1)
    trho=tabt(1)*rhoz*1.e3
    init=.true.
c
2 tty=ty*100.
  j=1
  if (tty .le. 0.) go to 5
c
c Find the altitude and index of point of interest.
c
    do 3 j=2,nz
      if (tty-tabz(j)) 4,5,3
3 continue
c
  j=nz+1
c
4 j=j-1
c
5 dum2=(tabz(j)-tty)/((re+tty)*(re+tabz(j)))
  dum3=(re+tabz(j))/(re+tty)
  var1=(re+tabz(j))*tabl(j)-tabt(j)
  rvar1=1./var1
  var2=((tty-tabz(j))*tabl(j)+tabt(j))/tabt(j)
  fs=-cons*(((tabl(j)*alog(dum3*var2))*rvar1+dum2)*rvar1)
  wsp=tabp(j)*exp(fs)
  wst=tabt(j)+tabl(j)*(tty-tabz(j))
  wsr=trho*wsp/(wst*tabp(1))
  wsp=0.1*wsp
  cs=sqrt(1.4*wsp/wsr)
c
  return
end

```

```

      function enrgyl (rho,pr)
c
      gm1=.404574
      dgamma=gm1
c
      10 e=pr/(gm1*rho)
         call air (e,rho,gmon1,p,temp)
         dp=p-pr
         dgamma=0.5*dgamma
         if (dp) 30,50,20
c
      20 gm1=gm1+dgamma
         go to 40
c
      30 gm1=gm1-dgamma
c
      40 if (abs(dp)/pr .gt. 1.e-3 .and. abs(dgamma) .gt. 1.e-3) go to 10
c
      50 enrgyl=e
c
      return
      end

```

Index

A

Acceleration, 7, 46, 60, 342, 384
 drag, 140, 141
 gravity, 46, 192
 pressure, 142
 radial, 105
 shock, 141
Active cases, 91–94
Active gauge(s), 170
Adiabatic, 198
Adiabatically, 6
Algorithm, 32, 89, 108, 109, 268, 373, 379, 380
Aluminum, 143, 169, 299, 334, 349, 351, 354, 355, 357, 360, 361
 burning, 58, 68, 93, 143, 275, 349, 352
 case, 76, 93, 95, 311
 foil, 169
 fragments, 93
 heating, 68, 357
 particles, 360
Amplitude, 6, 35, 99, 117, 160, 174
Anemometer, 174
Arena test, 87, 95
Arrival, 75, 115, 127, 168
 time, 192, 195, 240, 265, 288

B

Backdrop, 152, 166, 167, 242
Baffle(s), 325–328
Ballistic
 Lab Army, 91, 241, 365
 pendulum, 181
Blast
 generator, 64

interaction, 52, 294–296, 363–370
loading, 1, 279, 312, 314, 391
measurement, 54, 74, 246, 384
parameter, 99, 165, 181–196, 240, 251, 268
pressure, 10, 53, 146, 245, 250
propagation, 99, 122, 301, 329–335, 345, 391
 standard, 26–32, 54–58, 119, 232, 363
Boundary layer, 121–133, 137, 138, 144, 145, 163, 178, 180, 201, 217, 241, 250, 331
Breakaway, 26, 27, 37

C

Calculation, 4, 6, 7, 32, 44, 45, 49, 53, 54, 59, 60, 73, 76, 89–91, 103, 105–115, 125, 126, 133, 147, 148, 150, 152, 154, 155, 157–160, 170, 173, 182, 190, 192, 197, 204, 205, 209, 210, 215, 216, 221, 222, 232, 236, 247, 249, 250, 252–254, 259, 268–271, 274, 281, 284, 285, 287, 290, 292, 298, 299, 301–303, 305, 306, 323, 325, 328, 329, 345, 352, 355, 356, 360, 364, 370, 372, 378, 380, 388, 390
Cantilever gauge, 125, 180
Cased
 explosive, 73–95, 326
 heavily cased, 77–89, 350
Casing, 73–76, 350, 351
 light, 73–77
CGS, 3, 5, 11, 33
Charge
 array(s), 340–342
 bare, 73, 74, 91, 93, 351, 352
 cylindrical, 53, 76, 78, 84, 311

Charge (*cont.*)

- hemispherical, 53, 339
- spherical, 54, 73
- TNT, 349, 350, 363, 365

Collision(s), 6, 215**Combustion**, 58, 59, 349–357**Compression**, 6, 8, 27, 48, 79, 80, 108, 124, 137, 184, 185, 210, 247, 248, 251, 261, 337, 360**Computational fluid dynamics (CFD)**, 21, 32, 42, 44, 59, 76, 80, 89, 98, 110–113, 125, 133, 192, 204, 210, 216, 217, 219, 221, 252, 253, 270, 274, 284, 285, 290, 292, 299, 301, 323, 328, 342, 370, 373, 374, 378, 379**Conservation**, 1, 9, 10, 15, 16, 27, 37, 41, 42, 45, 101, 102, 111, 112, 114, 119, 138, 205, 244, 251, 259, 262, 329, 363, 364, 389**Cubes**, 55, 62, 104, 125, 178, 179, 181, 183–188, 194, 196, 229, 237, 246, 263, 280, 338, 390**D****Decay(s)**, 3, 6, 7, 19, 27, 29, 33, 35, 36, 45, 63, 65, 66, 69, 97–100, 102, 103, 105, 115–117, 123, 125, 126, 131, 138, 141, 168, 183, 186, 190–192, 196, 203, 213, 214, 221, 224, 225, 227, 228, 242, 254, 262, 263, 265, 267, 276, 280, 285, 287, 288, 296, 306, 307, 315, 316, 325, 330, 331, 337, 338, 340, 342, 345, 370, 385, 386, 390**Decomposition**, 6, 360**Decursor**, 279**Density**

- ambient, 80, 82, 98
- atmospheric, 119, 188, 196
- loading, 262, 279, 285
- over density, 4, 44, 62

Deposition, 8, 21, 26–31, 189, 262**Detonable**

- gasses, 9, 13, 22, 23, 44, 49
- limits, 50

Detonation

- front, 55
- internal, 55, 83
- nuclear, 138, 169
- TNT, 4, 378
- wave, 25, 41–70, 78, 79, 90, 151, 339, 356

Diaphragm, 22–25, 169, 170, 337, 338**Diffusion**, 26, 108, 277**Dimension(s)**

- three, 368, 370, 378
- two, 1, 98, 102–104, 109

Dissociation

- nitrogen, 355
- oxygen, 143, 191, 215, 216, 275, 334

Distant plain, 44, 275, 276**Drag**

- coefficient, 106
- force, 140, 141, 174
- gauge, 174

Duration

- positive, 292, 317
- precursor, 124, 343
- pressure, 178, 204

Dust

- acceleration, 142, 146, 244
- entrainment, 251
- momentum, 252, 253, 323

E**Energy**

- conservation, 9, 259, 262, 329, 363, 389
- internal, 3, 4, 11, 21, 43, 44, 46–48, 55, 81, 94, 111, 114, 126, 144, 197, 271, 331
- kinetic, 47, 48, 55, 77, 79, 83, 87, 91, 93–95, 137, 140, 145, 181, 221, 260, 315, 318
- rotational, 10, 204, 261, 264, 293
- total, 15, 20, 57, 78, 89, 100, 101, 112, 183, 185, 358
- vibrational, 4

Equation of state (EOS), 9, 12, 16, 32, 41–45, 53, 81, 82, 101, 111, 114, 154, 216, 358**Eulerian**, 54, 108, 110–113, 154**Evaporation**, 143, 252**Exit jet**, 342–347, 387–389, 392**Expansion**

- cylindrical, 66, 77, 80, 97, 103, 179
- free air, 52
- spherical, 97–99

Explosive

- fuel air explosive (FAE), 63–67
- solid fuel air explosive(SFAE), 67–69, 349, 351, 356, 357

External detonation, 311–319, 330, 370**F****Fano equation**, 87–89, 91**Fireball**, 26–29, 33, 37, 50, 53, 55, 60, 62, 70, 91, 138, 139, 144, 146, 148, 154, 164,

185, 186, 190, 236, 245, 252–254, 292, 301, 330, 350, 351
 Flux, 108, 111, 112, 186, 187, 189, 252, 262, 264, 268, 274
 radiation, 186, 263
 thermal, 186, 187, 189, 263, 264, 269
 Foam, 147, 250
 Foil meter, 169
 Fragment, 76–78, 80, 81, 83–91, 93–95, 180, 262, 326, 328, 329, 332, 334, 335, 350
 Frequency, 4, 6, 114, 117, 170–176, 180, 186, 191, 192, 204, 215, 239, 285

G

Gamma, 5, 9–11, 13, 16, 20, 21, 23, 32, 33, 41, 44, 82, 83, 111, 185, 197, 198, 204, 225, 260, 279, 338, 365
 Gauge
 electronic, 125, 168, 170, 171
 greg gauge, 175–177, 180
 passive, 125, 168, 170, 178, 179
 snob, 145, 175–177, 180

H

Heating, 5, 6, 48, 53, 59, 65, 67–69, 80, 93, 137, 140, 152, 174, 232, 252, 253, 264, 275, 335, 349, 351, 352, 356, 357, 360
 Height of burst (HOB), 60, 74, 161, 193, 227–281, 283, 284, 286, 287, 291, 292, 305, 323, 365, 368, 369
 Helicopter, 8
 High explosive, 8, 26, 42, 44, 52, 54, 57, 63, 70, 78, 82, 113, 119, 120, 151, 153, 154, 162, 171, 183, 187, 232–244, 246, 275, 277, 279, 288, 292, 337–340, 342, 349, 363
 Hiroshima, 286, 287

I

Ideal surface, 227–244, 251, 274
 Image burst, 364–369, 373, 374, 380
 Impulse, 54, 55, 69, 100, 119, 125–127, 131, 134, 147, 161, 177
 dynamic pressure, 262
 loads, 147, 180, 284
 over pressure, 55
 total impulse, 180, 181, 345
 Infrared (IR), 93, 94, 175, 185
 Instabilities, 55, 60, 76, 151–162, 236, 262, 269, 352

Kelvin–Helmholtz, 110, 156–159, 203, 249, 294
 Raleigh–Taylor, 49, 151–157, 235
 Richtmeyer–Meshkov, 159–160
 Instrumentation, 124, 125, 168, 170, 337
 Interferogram, 171, 172, 200, 204, 216–219, 224
 Interior loads, 314
 Ionization, 11, 82, 191, 192

J

Jeep, 177, 178, 343
 Jones–Wilkins–Lee (JWL), 43, 53, 82, 358

L

Lagrangian, 44–46, 49, 54, 108–113, 152
 Lamb
 addition rules, 363–366, 368, 369, 373, 380
 Landau, Stanyukovich, Zeldovich and Kompaneets (LSZK), 43–45, 53, 81, 82
 Large Blast and Thermal Simulator (LB/TS), 16, 99, 100, 158, 159, 278, 338, 340–343, 388
 Laser, 8, 26, 152, 154, 171–173, 204, 216, 218, 224, 241, 242, 260
 Liquid natural gas (LNG), 64
 Loads, 283, 284, 287, 290, 291, 299, 301, 307, 309, 311, 312

M

Mach, 284–286, 298, 305, 307
 complex Mach reflection (CMR), 201–203, 207, 244
 double Mach reflection (DMR), 109, 201–208, 210, 213, 214, 216, 219, 225, 228, 232, 242–244
 number, 3, 5, 7, 15, 32, 192, 206–210, 214, 219, 221
 reflection (MR), 198–211, 214, 217, 219, 221, 224, 228, 229, 232, 235–237, 240, 241, 244, 245, 251, 254, 307, 366, 369
 stem, 109, 198–206, 213–217, 221, 222, 224, 229, 230, 236, 241, 242, 246, 253–255, 259, 264, 276, 286, 287, 305, 368, 370
 transition, 222, 230, 240
 Mean free path, 26, 41, 175, 186, 215–218
 Measurement, 15, 54, 55, 60, 74, 116, 119, 125, 142, 145, 154, 162–182, 188, 190, 191, 204, 209, 214, 232, 246, 248, 260, 262,

- 263, 265, 269, 270, 279, 284–287, 306,
337, 338, 343, 345, 365, 384, 386
- Methane, 58–65, 82, 350
- MKS, 5, 179
- Model, 32, 54, 68, 133, 138, 147, 151, 152,
179, 252, 253, 259, 268, 269, 274, 299,
301, 302, 304–306, 309, 355, 356, 363,
366, 368, 370, 372–374, 377–380
- Modeling, 363–380
- Motion, 3, 4, 6–8, 10, 15, 16, 27, 43, 44, 47, 53,
64, 80, 99, 101, 102, 104, 106, 109–113,
125, 140, 145, 159, 163–167, 170, 174,
177, 178, 180, 181, 192, 232, 247, 261,
285, 299, 302, 308, 343, 386, 391
- Mott's distribution, 85–87
- N**
- Negative phase, 19, 29–31, 35, 38, 52, 53, 62,
113, 117, 123, 131, 137, 292, 293, 323,
330, 345
- Non-ideal explosive, 70, 88, 91, 95, 335, 349
- Normal reflection, 198
- Nuclear, 232, 233, 237, 240, 244, 245, 251,
259, 263, 267
blast wave, 33, 169, 190, 339
detonation, 8, 20, 26–31, 34
scaling, 185, 190, 339
- P**
- Particle(s), 6, 26, 67–70, 76, 80, 93, 94, 109,
110, 137–143, 145, 163, 173, 176, 185,
251, 334, 349, 356, 357, 360, 361
- Particulates
aluminum, 76, 82, 137, 138, 140
metal, 349, 350
- Photography, 26, 53, 73, 74, 94, 125, 154, 163,
173, 251, 252, 360
- Photon, 93, 175
- Piston, 8, 169, 170, 192, 260, 337
- Point source, 20, 21, 234
- Positive duration, 19, 29, 36, 53, 69, 100, 113,
114, 118, 120, 127–129, 131, 138, 139,
143, 146, 177, 178, 250, 251, 254, 262,
265, 296, 297, 330, 366, 385
- Positive phase, 19, 29, 30, 62, 69, 75, 113, 123,
139, 181, 240, 248, 292, 331, 343
- Power law, 33, 35, 179
- Precursor, 124, 253, 259–280, 291–293, 297,
342–346
- Pressure, 314, 315, 317, 323, 325, 328,
330, 331
- ambient, 6, 9, 12–14, 16, 20, 36, 37, 44, 46,
49, 60, 63, 90, 189, 191, 196, 210, 217,
218, 225, 260, 263, 364
- atmospheric, 4, 37, 49, 189, 191, 196, 292
- dynamic pressure, 3, 5, 8, 14–16, 19, 32–34,
100, 122–134, 140–142, 144–146, 173,
175–181, 205, 230, 240–244, 246, 250,
254, 259, 260, 262, 265–271, 274,
276–279, 283, 284, 291–293, 302, 307,
308, 343, 346, 365, 379, 389
- over pressure, 4, 5, 12–16, 19, 27, 29, 31,
33–38, 44, 50, 52, 54–57, 62, 63, 65,
69, 100, 113, 115–120, 122, 125–129,
132, 147, 168–171, 223, 243, 265–269,
276, 290, 297, 304, 305, 313, 365, 371
- reflected, 4, 14, 54, 141, 142, 175, 197, 209,
212, 221, 222, 224, 225, 228, 245, 246,
284, 287, 290, 296, 302, 324, 365, 384,
385, 389
- stagnation pressure, 5, 15, 124, 125, 145,
173, 175, 177, 180, 181, 204, 276, 284,
288, 293, 384, 385
- total, 175, 177, 276, 384
- Priscilla, 263–271, 279, 343, 345
- Propagate, 6, 22, 23, 66, 89, 97, 99, 100, 104,
115, 123, 124, 191, 192, 219, 269, 290,
296, 325, 330, 337, 340, 351, 378, 386,
389, 391
- Propagation, 1, 6, 16, 22, 30, 32, 33, 35, 41, 50,
53, 73, 84, 85, 89, 94, 97–123, 125, 133,
140, 148, 159, 161, 166, 183, 189–191,
198, 227, 228, 233, 245, 249, 250,
254, 255, 260, 269, 271, 274, 275, 279,
281, 283, 293–297, 301, 305, 309, 315,
325, 329–335, 338, 345, 374–380, 391
- Propane, 58, 64
- R**
- Radiation, 26, 27, 32, 35, 83, 112, 139, 144,
148, 185, 186, 251, 252, 263, 271, 274,
275, 286
- Rankine-Hugoniot, 259, 264
- Rarefaction, 22–25, 46–49, 60, 99, 102, 250,
288–291, 308, 313, 330, 337, 340, 370,
389
- Real, 9–11, 16, 247, 262–275, 338, 343, 349
air, 9–11
surface, 121, 137, 246–259
- Reflection
factor, 208, 210, 221, 228, 229, 232
regular reflection (RR), 206
shock, 225, 227, 228

wedge, 222, 224
 Riemann problem, 22, 99
 Rotation, 104, 204, 222

S

Scaling, 1, 33, 65, 132, 185, 187, 188, 190, 192, 195, 244, 337
 Scaling, 55, 183–196
 atmospheric, 194–196
 cube root, 187, 246
 yield, 189, 246
 Sedov solution, 20–22
 Self recording, 60, 168, 170, 181, 265
 Self similar, 21, 43, 183, 184, 214, 225
 Shadowgram, 203, 219, 241, 242, 260, 326
 Shock, 245, 246, 248, 250–254, 259
 Mach number, 3, 5, 15, 192, 206, 208, 210, 214, 221
 tube, 22, 23, 97–100, 123, 132, 158, 199, 216, 222, 227, 260–262, 277, 278, 284, 285, 292, 299
 wave, 1, 3, 275, 299, 326, 331, 360
 Signal, 6, 19, 35, 119, 120, 124–126, 138, 163, 168, 171, 180, 181, 192, 201, 203, 217, 219, 239, 259, 261, 263–265, 274, 278, 298
 Simulation, 1, 106, 261, 275–279, 337–346, 358
 Slip line, 113, 199–204, 206, 207, 213, 214, 216, 219, 222, 224, 230, 242, 244, 246, 278
 Smoke, 163–167, 173, 236, 254, 275
 puff, 165, 167, 171, 173
 smoke rocket, 163–165
 smoke trail, 125, 163, 164, 173
 SMOKY, 254–256, 271
 Snow, 140, 248, 249, 279
 Sound, 3, 5–8, 13, 15, 22, 23, 29, 30, 32, 37, 41, 43, 45, 47, 53, 89, 90, 115, 121, 124, 168, 175, 180, 184, 188, 193, 195, 196, 227, 245, 247, 248, 254, 259–264, 267, 269, 270, 274–279, 288, 291, 308, 330, 351, 389
 Sound wave, 6, 19, 33, 116, 117, 124, 175, 260–262
 Specific heat, 5, 10, 43, 44, 81, 82, 111, 145, 146, 338
 Spectral analysis, 174
 Speed, 6–8, 13, 15, 23, 26, 29, 32, 37, 41, 43, 45, 47
 material, 22, 30, 32, 247
 shock, 299, 302, 306, 312, 315, 316
 Steel can, 169

Structure, 94, 115, 144, 162, 173, 175, 181, 216, 260, 266, 284, 285, 287, 288, 290–296, 378, 384, 388, 391
 interaction, 52, 283–309
 responding structure, 297–306
 rigid structure, 297–306
 Supersonic, 7, 8, 201, 293
 Surface, 4, 14, 25, 33, 35, 46–49, 52, 53, 55, 58, 63–65, 68, 70, 73, 79, 91, 99, 106, 117, 121–133, 137–140, 143–148, 151, 155, 156, 161, 171, 174, 175, 178, 180, 181, 183–186, 191, 192, 197–206, 209, 211, 213, 214, 217–219, 221, 222, 224, 227, 228, 230, 232–237, 239–242, 244–255, 259, 261–264, 267, 269, 270, 274–276, 278, 279, 283–287, 289, 290, 292–294, 299, 301, 312–315, 323, 334, 335, 339, 343, 363, 364, 366, 370, 373, 389
 rough, 217–220
 smooth, 121, 217
 snow, 248, 249
 Sweep up, 137, 138, 147, 252, 268, 280, 343

T

Taylor Wave, 20
 Temperature, 5, 6, 9–12, 14, 15, 26, 27, 32, 37, 38, 43, 55, 58, 59, 63, 65, 67, 69, 83, 91, 93, 94, 114–116, 139, 140, 145, 156, 158, 174, 175, 185, 186, 189, 191, 192, 196, 197, 252, 260, 262, 263, 269, 270, 274, 275, 277, 334, 349–351, 354, 356, 360
 Terrain, 55, 64, 70, 125, 139, 156, 227, 253–259, 271–274, 280, 293
 Thermal flux, 186–190, 263, 264, 269
 Thermal radiation, 139, 144, 148, 185, 251, 252, 254, 263, 264, 271, 274, 275
 Thermobaric(s), 349, 351, 353, 356–358
 Time, 6, 9, 19, 21, 24, 26–32, 36–38, 43, 44, 46–49, 53, 54, 57, 59, 60, 62, 65–69, 75–77, 80, 91, 93, 101, 106–117, 119, 120, 122, 123, 126, 127, 131, 138–143, 145, 151, 152, 154, 155, 158, 160, 163–166, 168–175, 177, 180–182, 184, 186–190, 192–196, 203, 204, 206, 209, 211, 213–216, 221, 222, 236, 240, 242, 244, 251–253, 255, 259, 261, 264–268, 271, 274–276, 284, 285, 288–292, 294, 297–299, 301, 305, 308, 312, 314, 316, 318, 323, 325, 326, 329, 330, 332, 334, 340, 343, 350–352, 356, 357, 360, 361, 363, 364, 366, 368, 370, 372, 373, 377, 379, 383, 386, 388, 389

Time (cont.)

- arrival, 115, 116, 119, 165, 168, 192, 195,
196, 240, 264, 265, 267, 288, 305, 366,
372, 373, 377
- duration, 123

Train(s), 6, 8, 119, 175

Transmitted shock, 62, 326

Triple point, 198, 199, 201–203, 213–216, 219,
221, 222, 230, 241, 242, 244–246, 251,
253, 254, 259, 271, 274, 276, 286, 287,
305, 366, 368–370

Tube, 5, 8, 22, 90, 97, 99, 100, 123, 124,
132, 156, 158, 164, 173, 175, 176,
180, 246, 260, 261, 277, 278, 285,
292, 337, 338, 340, 342, 343, 345,
346, 385–390, 392

Tunnel, 8, 102, 146, 147, 329–335, 351

Turbulence, 101, 133, 294

Turbulent, 8, 122, 138, 148, 203, 242,
253, 294

Two phase flow, 139, 140

U

Urban terrain, 301–304

V

Vector, 3, 5, 14, 19, 98, 101, 115, 181,
197, 198, 279, 318, 323, 364, 373,
375, 376

Velocity, 3, 5–9, 13, 14, 16, 19, 21–33, 37,
38, 41, 43, 45–51, 53, 54, 57, 58, 60,
64, 75, 77, 78, 80, 83, 84, 90, 94, 97–99,
101, 102, 104, 105, 110–116, 121–128,
131, 137–141, 143–145, 147, 148, 156,
158, 160, 162, 163, 165, 168, 173–175,
177, 181, 183–185, 188, 189, 192,
195, 197–200, 203, 205, 215, 216,
222, 224, 228, 230, 244–246, 248–255,
259–264, 268–270, 274, 293, 299,
305, 334, 335, 339, 356, 359, 360,
364, 365

Vibration, 4, 8, 10, 53, 82

Vortex, 113, 139, 147, 173, 178, 219, 235, 242,
244, 246, 254, 261, 262, 266, 270, 278,
284, 285, 292, 293, 313–315, 317, 326,
345, 370, 387, 388, 390

W

Water, 140, 146–149, 151, 156, 192, 249, 250,
252, 274, 354, 355, 359

Wave, 1, 3, 4, 6–8, 14, 19–28, 30, 32, 33,
35–37, 41–43, 46–49, 53, 54, 58–60, 62,
65–67, 69, 73–75, 77–79, 83, 84, 90, 91,
94, 95, 97–100, 102–104, 113, 115–119,
122–127, 130–133, 137–139, 141,
143–149, 151, 154, 155, 158, 159,
161–164, 166, 168–177, 179–181,
183–186, 189–192, 195, 196, 199,
201–204, 209, 212, 214, 215, 218, 219,
221, 224, 225, 227–229, 232–236, 241,
242, 246–254, 259, 260, 262–265,
274–277, 279–281, 283, 284,
286–288, 290–299, 301–303, 305–309,
311–321, 323–326, 329–335, 337–340,
342, 343, 346, 350, 351, 353, 355, 356,
360, 363, 364, 366, 368, 370, 372, 373,
375, 376, 384–387, 389–392

Waveform, 36–38, 117, 118, 147–149, 170,
175, 176, 210, 214, 215, 223, 242–245,
249–251, 262, 263, 265–268, 270, 276,
280, 283, 286, 288, 291, 299, 300,
305–307, 309, 340, 343–346, 352,
355, 363–367, 377, 379, 384–386, 388,
390, 392

Window, 116, 172, 261, 288, 292, 293, 298,
299, 302, 303, 307, 311–318,
352–354

Wolfe–Anderson, 143

Work, 5, 10, 81, 82, 85, 109, 113, 122, 181,
185, 187, 208, 217, 330, 363, 364, 372,
392

X

X-rays, 26, 185, 186, 189