®

Editor: John Kessenich, LunarG

Version 1.1 Authors: John Kessenich, Dave Baldwin, Randi Rost

!

" # ! $ #% # ! # &# # # '

).+ 5recision and 5recision >uali'iers

2hDs)doOcu "5ent 53ec0'ies onl. version).)* o' the 03enGL (hadin! Lan!ua!e. It re<uires ??VER(I08?? to su:stitute))*, and re<uires #version to acce3t onl.))*. I' #version is declared with a s"aller nu":er, the lan!ua!e acce3ted is a 3revious version o' the shadin! lan!ua!e, which will :e su33orted de3endin! on the version and t.3e o' conte4t in the 03enGL A5I. (ee the 03enGL Gra3hics (.ste" (3eci'ication, Version).), 'or details on what lan!ua!e versions are su33orted.

5revious versions o' the 03enGL (h,ð'3 ¶0`nG 'o50on and tl50sions ondla `sions 1 a s%))iisJie7¼ 1c5revds EV5)-

1.1 Acknowledgments

2his s3eci'ication is :ased on the wor o' those who contri:uted to 3ast versions o' the 03enGL Lan!ua!e (3eci'ication, the 03enGL E(#.* Lan!ua!e (3eci'ication, and the 'ollowin! contri:utors to this version:

5at Brown, 8VIDIA
Je'' Bol=, 8VIDIA

- **ä**

- new offset and align la.out <uali'iers 'or control over :u''er :loc la.outs
- add location

- Bu! 11#//: 6i4ed textureOffset 'or sampler2DArray hadow to ta e a ivec2 Dnot a vec2E 'or the o''set.
- Bu! 1*))*: $lari'. that a na"e collision :etween "e":ers o' two anon."ous :loc s or a varia:le and a "e":er o' an anon."ous :loc is an error.
- Bu! 1*,--: $lari'ication that o3a<ue t.3es De.!., sa"3lersE can :e a uni'or" De.!., "e":er in a structE, not Fust a non9a!!re!ate uni'or" varia:le.
- Bu! 1*,-/: Be even "ore clear that :loc s !enerall. cannot :e redeclared as a wa. to si=e an unsi=ed arra. contained in the :loc .
- Bu! 1*,&#: $lari'. that :uilt9in 'unctions with void return or out ar!u"ents are not included in in the set o' constant e43ressions.
-

- Bu! 1*-/%: $lari'. that within a declaration, i' inout is used, neither in nor out "a. :e used, and none o' these can :e re3eated.
-

)

2he 03enGL (hadin! Lan!ua!e is actuall. several closel. related lan!ua!es. 2hese lan!ua!es are used to create shaders 'or each o' the 3ro!ra" "a:le 3rocessors contained in the 03enGL 3rocessin! 3i3eline.

A co"3ute shader has access to "an. o' the sa"e resources as 'ra!"ent and other shader 3rocessors,

Details that 'ull. de'ine source strin!s, co" "ents, line nu":erin!, new line eli"ination, and 3re3rocessin! are all discussed in u3co"in! sections. (ections :e.ond those descri:e GL(L 3rocessin!.

"

! " "

```
""#$%&""
""'$#&""
""(&)*$+%""
```

(34 will su:stitute a deci"al inte!er constant that is one "ore than the nu":er o'3recedin! new lines in the current source strin!.

+(4 will su:stitute a deci"al inte!er constant that sa.s which source strin! nu":er is currentl. :ein!3rocessed.

4 $(&3 will su:stitute a deci"al inte!er re'lecÈÀno:uill su:siAă

! " "

2he defined

! " "

s3eci'. #version

B. de'ault, co"3ilers o' this lan!ua!e "ust issue co"3ile9ti"e le4ical and !ra""atical errors 'or shaders that do not con'or" to this s3eci'ication. An. e4tended :ehavior "ust 'irst :e ena:led. Directives to

2he initial state o' the co " 3iler is as i' the directive

! " "

" , 4

lowp mediump highp precision

sampler1D sampler2D samplerHD sampler"u#e

" 6

A shader contains a o' Dor toE a varia:le i', a'ter 3re3rocessin!, the shader contains a state " ent that would read Dor writeE

(%

$% " &' "

%

%À%À%À%À%À%À%À%À

6loatin!95oint 03a<ue 2.3es

% **9**

%	9
usampler**J**uffer uimage**J**uffer	a handle 'or accessin! an unsi!ned inte!er :u''er te4ture
usampler2D**K** uimage2D**K**	a handle 'or accessin! an unsi!ned inte!er #D "ulti9sa"3le te4ture
usampler2D**K** Array uimage2D**K** Array	a handle 'or accessin! an unsi!ned inte!er #D "ulti9sa"3le te4ture arra.
usampler"u#eArray uimage"u#eArray	a handle 'or accessin! an unsi!ned inte!er cu:e "a3 arra. te4ture

In addition, a shader can a!!re!ate these :asic t.3es usin! arra.s and structures to :uild "ore co"3le4 t.3es.

2here are no 3ointer t.3es.

(

6unctions that do not return a value "ust :e declared as void. 2here is no de'ault 'unction return t.3e. 2he e.word void

```
  4 ?              >=     8                 31
  4 ?         @    &))+)2
  4 ?              >=     8                 ? ''''''''
  4 ?         @    >=     8                 ? ''''''''
  4 31                          1 8                  !        8
                                       3        >=3
                          !          ? ''''''''             8
                                    316
  4 31                           1  8                  !        8
                                       3        >=3
                          !          ? ''''''''             8
                                    ? ''''''''6
  4 >?????????                                        >=     8
                                          8         31=A<ABC=AB
  4 ? D???????          08 >=3
  4 E?????????      &))+)2                 >=
; 4 ? '''''''''      &))+)2                 >=
  4 ? F???????              3=1<C<F>B<F 44 ? F???????
  4 =1<C<F>B<F              3=1<C<F>B<F
```

Des3ite all these e4a"3les initiali=in! varia:les, literals are reco!ni=ed and !iven values and t.3es inde3endentl. o' their co cni

6loatin!93oint constants are de'ined as 'ollows.

```
0
    0                    '     '   , 0        00   ,
    #     7        '     '   0        00   ,
0
    #     7      #     7
    #     7
     #     7
  '       '
          , #     7
          , #     7
```

$ " &' "

$ " &' "

$ " &' "

$

$ " &' "

$ " &' "

$ " &' "

$ " &' "

$ " &' "

" ;

2 . % ;	9
centroid	centroid9:ased inter3olation
sample	3er9sa " 3le inter3olation
patch	

$ " &' "

$ " &' "

$ " &' "

$ " &' "

$ " &' "

3oint, or

- the value was written :. "ulti3le shader invocations since the 3revious s.nchroni=ation 3oint,

: % 5 0

0! 8 0666

666 666

An inter'ace :loc is started : . an in, out, uniform, or #uffer

$ " &' "

I' an instance na"e D E is not used, the na"es declared inside the :loc are sco3ed at the !lo:al level and accessed as i' the. were declared outside the :loc . I' an instance na"e D E

```
    (      5
       < H             DH$                              (     6H
       = ,
9 M                                        M     6H

    (     = 5
       < M             DH$           M
          M    =
9

           !          (     =      2
    (     > 5
           $       0
       < M             &))+)8                     M        (     =
9
      M    =           &))+)8               M    =    (     =
```

6or :loc s declared as arra.s, the arra. inde4 "ust also :e included when accessin! "e":ers, as in this e4a"3le

```
          ,         5     DH$       ,        I=J                        =
          <          G    (   G
          <          G    (   H  ;      G
```

$ " &' "

Fayout Uualifier	Uualifier Only	Pndividual ' aria#le	Jlock	Jlock Kem#er	Allowed Pnterfaces
shared packed std1G,					

attri:ute 'ro" which in3ut values are ta en. 6or in3uts o' all other shader t.3es, the location s3eci'ies a

$ " &' "

$ %

Additional la.out <uali'ier identi'iers 'or !eo"etr. shader in3uts include ' identi'iers and an identi'ier:

- 7 0 #

2he intrinsicall. declared in3ut arra. ;< will also :e si=ed :. an. in3ut 3ri"itive9la.out declaration. 1ence, the e43ression

" 6

will return the value 'ro" the ta:le a:ove.

Redeclarations are done as 'ollows

< "' M

D

```
0          4 >          <
```

will esta:lish that the 'ra!"ent shader out3ut is assi!ned to 'ra!"ent color % as the 'irst Dinde4 =eroE in3ut to the :lend e<uation. And,

```
0          4 >8       4 1          <
```

will esta:lish that the 'ra!"ent shader out3ut 0 is assi!ned to 'ra!"ent color % as the second Dinde4 oneE in3ut to the :lend e<uation.

$ " &' "

$ " &' "

0 " 4=8 " 4? 5 =

$ " &' "

verte4, a !eo " etr. shader should write to all out3uts associated with the strea " to which the verte4 will

Redeclarations o' gl$NragDepth

uni'or" s without an e43licit location, it assu" es 'or all uni'or" s with an e43licit location all their arra.

$ " &' "

$ " &' "

s3eci'. an offset that is s"aller than the o''set o' the 3revious "e":er in the :loc or that lies within the 3revious "e":er o' the :loc . 2wo :loc s lin ed to!ether in the sa"e 3ro!ra" with the sa"e :loc na"e "ust have the e4act sa"e set o' "e":ers <uali'ied with offset and their

- 7 0 #
#inding > '

2he identi'ier " #

$ " &' "

```
rg#a1!f
rgH2f
rg1!f
```

```
rgH1!a1frg#f
```

/ $;

In addition to 3recision <uali'iers and " e " or . <uali'iers, 3ara " eters can have these 3ara " eter <uali'iers.

; 9

6or e4a " 3le:

$ " &' "

$ " &' "

$ " &' "

; **9**

coherent

$ " &' "

$ " &' "

, *.

,

" ( ) "" "

@hen constructors are used to convert an . 'loatin!93oint t.3e to an inte!er t.3e, the 'ractional 3art o' the

" () "" "

Arra. su:scri3tin! s.nta4 can also :e a33lied to vectors D:ut not to 1 to 3rovide nu"eric inde4in!. (o in

' ; < re'ers to the third ele"ent o' 3os and i (e<uivalent to 6. 2his allow u0ria:le inde4in! into a vector, as well as a !eneric wa. o' acces #n! co"3onents. An. inte!er e43res #on can :e u -d as the su:scri3t. 2he 'irst co"3onent is at inde4 =ero. Readin! 'ro" or writin! to a vector u #n! a constant inte!ral e43res #on with a value that i (ne!ative or !reater than or e<ual to the #=Dies L0 Gia P`stant

- 2he arith"etic :inar. o

" () "" "

" () "" "

6 4 8 I?J I?J
60 4 8 I1J 8 ! ƒ1R à‡ø60 4 8 I1J

$

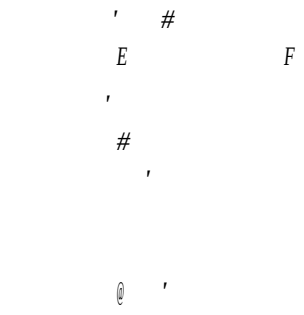
i.vg#(GhE

2he 'unda"ental :uildin! :loc s o' the 03enGL (hadin! Lan!ua!e are:

- state"ents and declarations
- 'unction de'initions
- selection Dif(else and switch(case(defaultE
- iteration•

L :#(Gh n!RgAcià ¥5#(epnLSS4oh

$url. :races are used to !rou3 se<uences o' state "ents into co "3ound state "ents.



* "

* "

0

are uni'or"s, and are assi!ned to s3eci'ic 'unctions onl. throu!h co" "ands D

* "

until the # ' evaluates to 'alse. 2he loo3 is then e4ited, s i33in! its :od. and s i33in! its ' ' Varia:les "odi'ied :. the ' ' "aintain their value a'ter the loo3 is e4ited, 3rovided the. are still in sco3e. Varia:les declared in

derivatives are unde'ined when this e4it is non9uni'or". It would t.3icall. :e used within a conditional state"ent, 'or e4a"3le:

```
0 7 ?6?
```

2he :uilt9in varia:le ((is a co" 3ute9shader in3ut varia:le containin! the t9 di" ensional inde4 o' the local wor !rou3 within the !lo:al wor !rou3 that the current invocation is e4ecutin! in. 2he 3ossi:le values 'or this varia:le ran!e across the local wor !rou3 si=e, i.e., D*,*,*E to D G . , '$6 91, G . , '$6 -91, G . , '$6 6

2he ! ' arra. is 3redeclared as unsi=ed and "ust :e e43licitl. si=ed :. the shader either redeclarin! it with a si=e or i"3licitl. si=ed :. inde4in! it onl. with inte!ral constant e43ressions. 2his

! %+ # $% "

out o' ran!e. I' a 'ra! "ent shader contains a static access to *5' (#* , it will count a!ainst the i"3le"entation de'ined li"it 'or the "a4i"u" nu":er o' in3uts to the 'ra! "ent sta!e.

2he varia:le

inter3olation <uali'iers, and the la.out <uali'iers xf#$offset, xf#$#uffer, and

staticall. read or write :oth *! '* and *! '* Lyme Either *! '*

! %+ # $% "

! % + # $% "

! "

! 0 ! 0

"# * H 5

! 0 ! 0

<]0" 9G $ J

1 8 0

2 $ % 0

6unction 3ara " eters s3eci'ied as are assu " ed to : e in units o' radians. In no case will an . o' these

" $$ 0

2hese all o3erate co " 3onent9wise. 2he descri3tion is 3er co " 3onent.

% .	-
!en2.3e a#s D!en2.3e E !enI2.3e a#s D!enI2.3e E !enD2.3e a#s D!enD2.3e E	Returns i' WN *M otherwise it returns `
!en2.3e sign D!en2.3e E	

! + , "

% .

-

!en2.3e intJits%oNloat D!enI2.3e
!en2.3e uintJits%oNloat D!en72.3e

E Returns a float value corres3ondin! to a si!ned or
E unsi!ned inte!er encodin! o' a float. I' a 8a8 is 3assed in, it will not si!nal, and the resultin! value is

% . -

vec# unpackVnorm2x1! Duint 'E
vec# unpack norm2x1! Duint 'E
vec) unpackVnormGx\ Duint 'E
vec) unpack normGx\ Duint 'E

! %+ , "

% . -

vec# unpack**O**alf2x1**!** Duint E

D ð1 & "c0& bÃ0P 9• Ð

! + , "

!

! + , "

In all 'unctions :elow, the "

2he al!orith" used is !iven :. the 'ollowin! 3seudo9code:

```
        M  !   D       #            !    #
  5
         M   !        !     #+-                          #+-     ! 6
             !    #   7 ,&U,@)&"G$%"#+-     !    #   4 ,&U,@)&"G$%"#+-
             !    #   Q ,&U,@)&"GDU"#+-     !    #   4 ,&U,@)&"GDU"#+-"#+-
```

' . 4 0

% . -

!vec) texture D!sa"3ler1D ' 2'loat C G, 'loat " H E
!vec) texture D!sa"3ler#D ' , vec# C G, 'loat " H E
!vec) texture D!sa"3ler%D ' , vec% C G, 'loat " H E
!vec) texture D!sa"3ler$u:e ' , vec% C G, 'loat " H E
'loat texture Dsa"3ler1D(hadow ' 2 vec% C G, 'loat " H E
'loat

! + , "

! + , "

! + , "

% . -

! % + , "

% . -

!vec) textureTroLEradOffset

! % + , "

% . -

!vec) textureLeather DVVtsatlu:

' $ % / . 0

2he 'ollowin! te4ture 'unctions are onl. in the co " 3ati:ilit. 3ro'ile.

! + , "

inte!er ' 3ara"eter. 2he coordinates and sa"3le nu":er are used to select an individual te4el in

! % + , "

% .

-. 6unctions 'or dNdx should :e evaluated while holdin! . constant. 6unctions 'or dNdy should :e evaluated while holdin! 4 constant. 1owever, "i4ed hi!her order derivatives, li e dNdxDdNdyD-EE and dNdyDdNdxD EE are unde'ined.

,.

! + , "

; ulti3le out3ut strea " s are su33orted onl . i ' the out3ut 3ri " iti, th!Ji p 333341redute "

! + , "

#

```
*,)@M, (+$- TP$#&

$-&%,$'$&) ,VH&"%DG&
'#+D,M+%*,D%, -+@:#&M+%*,D%, $%,M+%*,D%, @$%,M+%*,D%, :++#M+%*,D%,
'$&#-"*&#&M,$+%
#&',"+H )$.P,"+H
$%M"+H -&M"+H #&"+H .&"+H &Y"+H %&"+H
```

$ &&1P
+ *1
3&C4 $C4!1(4

- 7 0
I&)1 4+1 C 43 - 7 0 # (,P1 C* 43

- 7 0 #
- 7 0 #
- 7 0 # !& * - 7 0 #

- 7 0 #
(431(+(4
(431(+(4 4U)* '
$P* 4

' 7 0
C 4!($4

-' 7 0
-' 7 0
ppp P!Đ -' 0 ÿöõ -' Þ 0

4!
/ 4!
/ 4!?
/ 4!
(4!
(4!?

* * - . %

4 () C 4!($(&3
&G C 4!($(&3

' 0

$1)!1(431(+(4 4+1 / *!4 # (,P1 / *!4
$1)!1 4+1 / *!4 # (,P1 / *!4

~~5ó~~÷ 6
#
# #

#
-' ' 0 # $4 (!& &3
-' 7 0 -' ' 0 # $4 (!& &3

#
#
!& *

#
(431(+(4
(431(+(4 - ' 0

6
'
4+ / *!4