

```
%9:
%10 = shl %i64 %6, 8
%icmp221.i1 = icmp sgt i32 %i64, 0
%wsidw.trip.count.i1 = zext i32 %i64 to i64
%11 = add nsw i64 %wsidw.trip.count.i1, -1
%xtrailer.i1 = and i64 %wsidw.trip.count.i1, 3
%12 = icmp ult i64 %i11, 3
%isnull.iter.i1 = sub nws nsw i64 %wsidw.trip.count.i1, %xtrailer.i1
%icmp.iter.i1 = icmp eq i64 %xtrailer.i1, 0
br i1 %icmp221.i1, label %region for_entry_entry.us.preheader, label
... %vectorop.h1

      T      F
      |      |
      |      |
```

[illegible][illegible]

--	--	--	--

```
%idxprom.i.us = ashr exat i64 %sext.i.us, 32
%arrayidx.i.us = getelementptr inbounds float, float* %2, i64 %idxprom.i.us
store float 0.000000e+00, float* %arrayidx.i.us, align 4, !tbaa !12,
... !lvm.access.prior !16
%mulmul.us = mul nsw i32 %convius, %4
%269 = sext i32 %mulmul.us to i64
br i1 %12, label %if.end, loopexit.unl-ssca.i.us, label
... %for.body.i.us.preheader
```

T

F

```

%271 = add nsw i64 %indvars.innext.i.3.3.us, 0 #000000+e0,
%indvars.innext.i.3.3.us, #000000+e0, #tbaa12
%arrayr5x1.us = getelementptr inbounds float, @roast%60, i64 %271
%272 = load float, @roast%arrayr5x1.us, align 4, !tbaa12
%arrayr7.i.us = getelementptr inbounds float, @roast%61, i64
%indvars.innext.i.3.3.us
%273 = load float, @roast%arrayr7.i.us, align 4, !tbaa12
%274 = tail call @llvm.fmuladd.f32(float %272, float %273, float %274)
;#2
store float %274, @roast%arrayr4.i.us, align 4, !tbaa12,
;#llvm.access.group 116
%indvars.innext.i.1.us = or i64 %indvars.innext.i.3.3.us, 1
%275 = add nsw i64 %indvars.innext.i.us, %269
%arrayr5x1.us = getelementptr inbounds float, @roast%60, i64 %275
%276 = load float, @roast%arrayr5x1.us, align 4, !tbaa12
%arrayr7.i.us = getelementptr inbounds float, @roast%61, i64
%indvars.innext.i.us
%277 = load float, @roast%arrayr7.i.us, align 4, !tbaa12
%278 = tail call @llvm.fmuladd.f32(float %276, float %277, float %278)
;#2
store float %278, @roast%arrayr4.i.us, align 4, !tbaa12,
;#llvm.access.group 116
%indvars.innext.i.1.us = or i64 %indvars.innext.i.3.3.us, 2
%279 = add nsw i64 %indvars.innext.i.us, %269
%280 = load float, @roast%arrayr5x1.us, align 4, !tbaa12
%arrayr7.2.us = getelementptr inbounds float, @roast%61, i64
%indvars.innext.i.us
%281 = load float, @roast%arrayr7.2.us, align 4, !tbaa12
%282 = tail call @llvm.fmuladd.f32(float %280, float %281, float %282)
;#2
store float %282, @roast%arrayr4.i.us, align 4, !tbaa12,
;#llvm.access.group 116
%indvars.innext.i.1.us = or i64 %indvars.innext.i.3.3.us, 3
%283 = add nsw i64 %indvars.innext.i.us, %269
%arrayr5x1.us = getelementptr inbounds float, @roast%60, i64 %283
%284 = load float, @roast%arrayr5x1.us, align 4, !tbaa12
%arrayr7.3.us = getelementptr inbounds float, @roast%61, i64
%indvars.innext.i.us
%285 = load float, @roast%arrayr7.3.us, align 4, !tbaa12
%286 = tail call @llvm.fmuladd.f32(float %284, float %285, float %286)
;#2
store float %286, @roast%arrayr4.i.us, align 4, !tbaa12,
;#llvm.access.group 116
%indvars.innext.i.3.us = add nsw nsw i64 %indvars.innext.i.3.3.us, 4
%nteramp.nump.3.us = icmp eq i64 %nteramp.nsum.3.6.us, 4
%nteramp.nump.3.us = icmp eq i64 %nteramp.nsum.3.us, 0
br i1 %nteramp.nump.3.us, label %if.end.loopexit.unroll-issia.us.loopexit,
;#lba0 %nterbody.us

```

```
br label %if.end.loopexit.unr-icssa.i.us
exit.unr-icssa.i.us:
hi float [ 0.0000000e+00, %if.then.i.us ], [ %
loopexit.unr-icssa.i.us [loopexit ]
```

F

```

for.body.epil.us:
... %body.iter.sub.i2.us = phi %i64 (%%body.iter.sub.i.us, %for.body.epil.i.us
... %i2) %strater, %for.body.epil.us, %preheader
... %indvars.iv.next.epil.i10.us = phi %i64 (%indvars.iv.next.epil.us,
... %for.body.epil.i.us) %i280, %for.body.epil.us, %preheader
... %289 = phi float, float, %for.body.epil.i.us, %i287,
... %for.body.epil.us, %preheader
... %290 = add nsw %i64 %indvars.iv.next.epil.i.us, %269
... %arrayidx5.epil.us = getelementptr inbounds float, @body %0, %i64, %290
... %291 = load float, float* %arrayidx5.epil.us, align 4, tbaa11
... %arrayidx7.epil.us = getelementptr inbounds float, @body %1, %i64,
... %indvars.iv.next.epil.i.us, %291
... %292 = load float, float* %arrayidx7.epil.us, align 4, tbaa11
... %293 = tail call @llvm.fmuladd.f32(float %291, float %292, float %289)
store float %293, float* %arrayidx5.epil.us, align 4, tbaa11,
... %llvm.alloc.group 116
... %arrayidx5.next.epil.us = add nsw new %i64 %indvars.iv.next.epil.i10.us, %i
... %epil.iter.sub.us = add nsw %i64 (%epil.iter.sub.i2.us, -1
... %epil.iter.cmp.us = icmp eq %i64 %epil.iter.sub.us, 0
... b.r1 %epil.iter.cmp.us, %loop.end.us, if and us loopexit, label
... %for.body.epil.us, %llvm.loop 118

```

T
F

X

$\lambda = 0.03, 1$

CFG for '_pocl_kernel_bicgKernel1' function