

[illegible]

```
%conv2.i = true i64 %add62.i.i to i32
%cmp5.i = icmp sgt i32 %sub4.i, %conv2.i
%sr.contr334.i = add i1 %cmp.i, %cmp5.i
%cmp15.i = icmp sgt i32 %conv2.i, 0
%rress.add.i = add i32 %mul189.i, %conv2.i
%rress.mul.i = mul i32 %rress.add.i, %4
%sub22.i = add nsw i32 %conv2.i, -1
%mul23.i = mul nsw i32 %sub22.i, %4
%add.i = add nsw i32 %mul23.i, %mul21.i
%add32.i = add nsw i32 %mul23.i, %mul29.i
%mul97.i = mul nsw i32 %conv2.i, %4
%add107.i = add nsw nsw i32 %conv2.i, 1
%mul108.i = mul nsw i32 %add107.i, %4
br i1 %contr334.i, label %region for entry.entry.i.us.preheader, label
... %vector.csecheck
```

```

, %conv.i.us
6cmp9.i.us
.i, %or.cond.i.us

```

i64 %idxprom.i.us

0, i64

2

0 1

0 1

/

```

%{m37,i3us} #2
%79 = tail call float @llvm.fmuladd.f32(float %76, float 5.000000e+00, float
%80 = tail call float @llvm.fmuladd.f32(float %77, float 7.000000e+00, float
...%79) #2
%81 = tail call float @llvm.fmuladd.f32(float %76, float -8.000000e+00,
...float %80) #2
%82 = tail call float @llvm.fmuladd.f32(float %77, float 1.000000e+01, float
...%81) #2
%add89.1.i.us = add i32 %mul84.i.us, %conv1.us
%add89.i.us = add i32 %mul87.i.us, %mul23.i
%idxprom90.i.us = sext i32 %add89.i.us to i64
%arrayidx91.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom90.i.us
%63 = load float, float* %arrayidx91.i.us, align 4, tbaa 112
%82 = tail call float @llvm.fmuladd.f32(float %63, float 3.000000e+00,
...float %62) #2
%add100.i.us = add i32 %add87.i.us, %mul97.i
%idxprom101.i.us = sext i32 %add100.i.us to i64
%arrayidx102.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom101.i.us
%64 = load float, float* %arrayidx102.i.us, align 4, tbaa 112
%86 = tail call float @llvm.fmuladd.f32(float %65, float 6.000000e+00, float
...%84) #2
%add111.i.us = add i32 %add87.i.us, %mul108.i
%idxprom112.i.us = sext i32 %add111.i.us to i64
%arrayidx113.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom112.i.us
%65 = load float, float* %arrayidx113.i.us, align 4, tbaa 112
%88 = tail call float @llvm.fmuladd.f32(float %67, float -9.000000e+00,
...float %86) #2
%add121.i.us = add nsw nsw i32 %conv1.us, i
%add122.i.us = add nsw i32 %add121.i.us, %add1
%idxprom123.i.us = sext i32 %add122.i.us to i64
%arrayidx124.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom123.i.us
%69 = load float, float* %arrayidx124.i.us, align 4, tbaa 112
%89 = tail call float @llvm.fmuladd.f32(float %69, float 2.000000e+00, float
...%88) #2
%add133.i.us = add nsw i32 %add121.i.us, %add32.i
%idxprom134.i.us = sext i32 %add133.i.us to i64
%arrayidx135.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom134.i.us
%91 = load float, float* %arrayidx135.i.us, align 4, tbaa 112
%92 = tail call float @llvm.fmuladd.f32(float %91, float 4.000000e+00, float
...%90) #2
%add142.i.us = add i32 %add121.i.us, %mul21.i
%add144.i.us = add i32 %add142.i.us to i64
%idxprom145.i.us = sext i32 %add144.i.us to i64
%arrayidx146.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom145.i.us
%93 = load float, float* %arrayidx146.i.us, align 4, tbaa 112
%94 = tail call float @llvm.fmuladd.f32(float %93, float 5.000000e+00, float
...%92) #2
%add153.i.us = add i32 %add121.i.us, %mul29.i
%add155.i.us = add i32 %add153.i.us, %mul97.i
%idxprom156.i.us = sext i32 %add155.i.us to i64
%arrayidx157.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom156.i.us
%95 = load float, float* %arrayidx157.i.us, align 4, tbaa 112
%96 = tail call float @llvm.fmuladd.f32(float %95, float 7.000000e+00, float
...%94) #2
%add166.i.us = add i32 %add142.i.us, %mul108.i
%idxprom167.i.us = sext i32 %add166.i.us to i64
%arrayidx168.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom167.i.us
%97 = load float, float* %arrayidx168.i.us, align 4, tbaa 112
%98 = tail call float @llvm.fmuladd.f32(float %97, float -8.000000e+00,
...float %96) #2
%add177.i.us = add i32 %add153.i.us, %mul108.i
%idxprom178.i.us = sext i32 %add177.i.us to i64
%arrayidx179.i.us = gemitelnetr_inbounds float, float* %60, i64
...%idxprom178.i.us
%99 = load float, float* %arrayidx179.i.us, align 4, tbaa 112
%100 = tail call float @llvm.fmuladd.f32(float %99, float 1.000000e+01,
...float %98) #2
br label %end.r.exit.i.us

```

<pre>%dyprom193.i.us.sink = phi i64 [%dyprom101.i.us, %ifthen.i.us], [%idxprom193.i.us, %false.i.us] %k.sink = phi float [%100, %ifthen.i.us], [0.000000e+00, %false.i.us] %arrayidx194.i.us = getelementptr inbounds float, @float*, i64 %idxprom193.i.us.sink, %k.sink store float %k.sink, float* @arrayidx194.i.us, align 4, !tbaa !12, !llvm.access.group !16 %101 = add nuw nsw i64 %local_id, x.0.us, 1 %exitcond2.not = icmp eq i64 %101, 32 br i1 %exitcond2.not, label %region for end.i.loopexit, label %region for entry.entry.us, !llvm.loop !19</pre>	
T	F

T	F

[illegible]

```

store -<8 x 132> -zeroinitializer, <8 x 132>* %b3, align 4, !tbaa 112,
... !lvm.access.group 116
%64 = add i32 %reass.mul.i, %x20
%65 = sext i32 %64 to i64
%66 = getelementptr inbounds float, float* %1, i64 %65
%67 = bitcast float* %66 to <8 x 132>*
store -<8 x 132> -zeroinitializer, <8 x 132>* %67, align 4, !tbaa 112,
... !lvm.access.group 116
%68 = add i32 %reass.mul.i, %x22
%69 = sext i32 %68 to i64
%70 = getelementptr inbounds float, float* %1, i64 %69
%71 = bitcast float* %70 to <8 x 132>*
store -<8 x 132> -zeroinitializer, <8 x 132>* %71, align 4, !tbaa 112,
... !lvm.access.group 116
%72 = add i32 %reass.mul.i, %x24
%73 = sext i32 %72 to i64
%74 = getelementptr inbounds float, float* %1, i64 %73
%75 = bitcast float* %74 to <8 x 132>*
store -<8 x 132> -zeroinitializer, <8 x 132>* %75, align 4, !tbaa 112,
... !lvm.access.group 116
br label %region for_end.i

```

```
... %idxpron
%119 = bit
```

```
pregon_for end.i:
%134 = add nuw nsw i64 % local_id_y.0, 1
%exitcond3.not = icmp eq i64 %134, 8
br i1 %exitcond3.not, label %Convolution3D_ke
```