

```
%10:
%11 = bitcast float* %3 to i8*
%12 = bitcast float* %4 to i8*
%13 = sext i32 %5 to i64
%14 = icmp slt i64 %13, 32
%15 = select i1 %14, i64 %13, i64 32
%16 = icmp slt i64 %13, 8
%17 = select i1 %16, i64 %13, i64 8
%mul.i.i = shl i64 %7, 5
%mul3.i.i = shl i64 %8, 3
%18 = icmp ugt i64 %15, 1
%umax = select i1 %18, i64 %15, i64 1
%19 = icmp ugt i64 %17, 1
%umax1 = select i1 %19, i64 %17, i64 1
%20 = add nsw i64 %umax, -1
%21 = trunc i64 %8 to i32
%22 = mul i32 %21, %5
%23 = shl i32 %22, 3
%24 = zext i32 %23 to i64
%25 = trunc i64 %7 to i32
%26 = shl i32 %25, 5
%27 = zext i32 %26 to i64
%28 = add nuw nsw i64 %24, %27
%29 = zext i32 %5 to i64
%uglygep = getelementptr i8, i8* %11, i64 1
%30 = shl i64 %8, 3
%31 = trunc i64 %8 to i32
%32 = mul i32 %31, %5
%33 = shl i32 %32, 3
%34 = zext i32 %33 to i64
%35 = trunc i64 %7 to i32
%36 = shl i32 %35, 5
%37 = sext i32 %36 to i64
%38 = add nsw i64 %34, %37
%39 = zext i32 %5 to i64
%scevgep8 = getelementptr float, float* %0, i64 %umax
%uglygep11 = getelementptr i8, i8* %12, i64 1
%scevgep13 = getelementptr float, float* %1, i64 %37
%40 = add nsw i64 %umax, %37
%scevgep15 = getelementptr float, float* %1, i64 %40
%scevgep17 = getelementptr float, float* %2, i64 %37
%scevgep19 = getelementptr float, float* %2, i64 %40
br label %pregon_for_entry.pregon_for_init.i
```

```
pregon_for_entry.pregon_for_init.i:
% local_id_y.0 = phi i64 [ 0, %10 ], [ %96, %pregon_for_end.i ]
%41 = add i64 %30, % local_id_y.0
%sext = shl i64 %41, 32
%42 = ashr exact i64 %sext, 30
%scevgep = getelementptr i8, i8* %uglygep, i64 %42
%43 = mul i64 % local_id_y.0, %39
%44 = add i64 %38, %43
%sext44 = shl i64 %44, 32
%45 = ashr exact i64 %sext44, 32
%scevgep6 = getelementptr float, float* %0, i64 %45
%scevgep67 = bitcast float* %scevgep6 to i8*
%scevgep9 = getelementptr float, float* %scevgep8, i64 %45
%scevgep12 = getelementptr i8, i8* %uglygep11, i64 %42
%46 = mul i64 % local_id_y.0, %29
%47 = add i64 %28, %46
%48 = trunc i64 %47 to i32
%add6.i.i = add i64 % local_id_y.0, %mul3.i.i
%conv2.i = trunc i64 %add6.i.i to i32
%sext.i = shl i64 %add6.i.i, 32
%idxprom.i = ashr exact i64 %sext.i, 32
%arrayidx.i = getelementptr inbounds float, float* %3, i64 %idxprom.i
%arrayidx9.i = getelementptr inbounds float, float* %4, i64 %idxprom.i
%mul.i = mul nsw i32 %conv2.i, %5
%min.iters.check = icmp ult i64 %umax, 16
br i1 %min.iters.check, label %pregon_for_entry.entry.i.preheader, label
... %vector.scevcheck
```

```
vector.scevcheck:
%49 = trunc i64 %20 to i32
%50 = add i32 %48, %49
%51 = icmp slt i32 %50, %48
%52 = icmp ugt i64 %20, 4294967295
%53 = or i1 %51, %52
%54 = trunc i64 %20 to i32
%55 = add i32 %26, %54
%56 = icmp slt i32 %55, %26
%57 = icmp ugt i64 %20, 4294967295
%58 = or i1 %56, %57
%59 = or i1 %53, %58
br i1 %59, label %pregon_for_entry.entry.i.preheader, label %vector.memcheck
```

```
vector.memcheck:
%bound0 = icmp ult float* %arrayidx.i, %scevgep9
%bound1 = icmp ugt i8* %scevgep, %scevgep67
%found.conflict = and i1 %bound0, %bound1
%bound022 = icmp ult float* %arrayidx9.i, %scevgep9
%bound123 = icmp ugt i8* %scevgep12, %scevgep67
%found.conflict24 = and i1 %bound022, %bound123
%conflict.rdx = or i1 %found.conflict, %found.conflict24
%bound025 = icmp ult float* %scevgep6, %scevgep15
%bound126 = icmp ult float* %scevgep13, %scevgep9
%found.conflict27 = and i1 %bound025, %bound126
%conflict.rdx28 = or i1 %conflict.rdx, %found.conflict27
%bound029 = icmp ult float* %scevgep6, %scevgep19
%bound130 = icmp ult float* %scevgep17, %scevgep9
%found.conflict31 = and i1 %bound029, %bound130
%conflict.rdx32 = or i1 %conflict.rdx28, %found.conflict31
br i1 %conflict.rdx32, label %pregon_for_entry.entry.i.preheader, label
... %vector.ph
```

```
vector.ph:
%n.vec = and i64 %umax, -16
br label %vector.body
```

```
vector.body:
%index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
%60 = add i64 %index, %mul.i.i
%61 = trunc i64 %60 to i32
%62 = load float, float* %arrayidx.i, align 4, !tbaa !12, !alias.scope !16,
... !noalias !19
%broadcast.splatinsert38 = insertelement <8 x float> undef, float %62, i32 0
%broadcast.splat39 = shufflevector <8 x float> %broadcast.splatinsert38, <8
... x float> undef, <8 x i32> zeroinitializer
%broadcast.splatinsert40 = insertelement <8 x float> undef, float %62, i32 0
%broadcast.splat41 = shufflevector <8 x float> %broadcast.splatinsert40, <8
... x float> undef, <8 x i32> zeroinitializer
%63 = shl i64 %60, 32
%64 = ashr exact i64 %63, 32
%65 = getelementptr inbounds float, float* %1, i64 %64
%66 = bitcast float* %65 to <8 x float>*
%wide.load = load <8 x float>, <8 x float>* %66, align 4, !tbaa !12,
... !alias.scope !21
%67 = getelementptr inbounds float, float* %65, i64 8
%68 = bitcast float* %67 to <8 x float>*
%wide.load33 = load <8 x float>, <8 x float>* %68, align 4, !tbaa !12,
... !alias.scope !21
%69 = load float, float* %arrayidx9.i, align 4, !tbaa !12, !alias.scope !23,
... !noalias !19
%broadcast.splatinsert = insertelement <8 x float> undef, float %69, i32 0
%broadcast.splat37 = shufflevector <8 x float> %broadcast.splatinsert, <8 x
... float> undef, <8 x i32> zeroinitializer
%broadcast.splatinsert36 = insertelement <8 x float> undef, float %69, i32 0
%broadcast.splat37 = shufflevector <8 x float> %broadcast.splatinsert36, <8
... x float> undef, <8 x i32> zeroinitializer
%70 = getelementptr inbounds float, float* %2, i64 %64
%71 = bitcast float* %70 to <8 x float>*
%wide.load34 = load <8 x float>, <8 x float>* %71, align 4, !tbaa !12,
... !alias.scope !25
%72 = getelementptr inbounds float, float* %70, i64 8
%73 = bitcast float* %72 to <8 x float>*
%wide.load35 = load <8 x float>, <8 x float>* %73, align 4, !tbaa !12,
... !alias.scope !25
%74 = fmul <8 x float> %broadcast.splat, %wide.load34
%75 = fmul <8 x float> %broadcast.splat37, %wide.load35
%76 = call <8 x float> @llvm.fmuladd.v8f32(<8 x float> %broadcast.splat39,
... <8 x float> %wide.load, <8 x float> %74)
%77 = call <8 x float> @llvm.fmuladd.v8f32(<8 x float> %broadcast.splat41,
... <8 x float> %wide.load33, <8 x float> %75)
%78 = add nsw i32 %mul.i, %61
%79 = sext i32 %78 to i64
%80 = getelementptr inbounds float, float* %0, i64 %79
%81 = bitcast float* %80 to <8 x float>*
%wide.load42 = load <8 x float>, <8 x float>* %81, align 4, !tbaa !12,
... !alias.scope !19, !noalias !27
%82 = getelementptr inbounds float, float* %80, i64 8
%83 = bitcast float* %82 to <8 x float>*
%wide.load43 = load <8 x float>, <8 x float>* %83, align 4, !tbaa !12,
... !alias.scope !19, !noalias !27
%84 = fadd <8 x float> %wide.load42, %76
%85 = fadd <8 x float> %wide.load43, %77
%86 = bitcast float* %80 to <8 x float>*
store <8 x float> %84, <8 x float>* %86, align 4, !tbaa !12, !alias.scope
... !19, !noalias !27, !llvm.access.group !28
%87 = bitcast float* %82 to <8 x float>*
store <8 x float> %85, <8 x float>* %87, align 4, !tbaa !12, !alias.scope
... !19, !noalias !27, !llvm.access.group !28
%index.next = add i64 %index, 16
%88 = icmp eq i64 %index.next, %n.vec
br i1 %88, label %middle.block, label %vector.body, !llvm.loop !31
```

```
middle.block:
%cmp.n = icmp eq i64 %umax, %n.vec
br i1 %cmp.n, label %pregon_for_end.i, label
... %pregon_for_entry.entry.i.preheader
```

```
pregon_for_entry.entry.i.preheader:
% local_id_x.0.ph = phi i64 [ 0, %vector.memcheck ], [ 0, %vector.scevcheck
... ], [ 0, %pregon_for_entry.pregon_for_init.i ], [ %n.vec, %middle.block ]
br label %pregon_for_entry.entry.i
```

```
pregon_for_entry.entry.i:
% local_id_x.0 = phi i64 [ %95, %pregon_for_entry.entry.i ], [
... % local_id_x.0.ph, %pregon_for_entry.entry.i.preheader ]
%add1.i.i = add i64 %local_id_x.0, %mul.i.i
%conv.i = trunc i64 %add1.i.i to i32
%89 = load float, float* %arrayidx.i, align 4, !tbaa !12
%sext26.i = shl i64 %add1.i.i, 32
%idxprom6.i = ashr exact i64 %sext26.i, 32
%arrayidx7.i = getelementptr inbounds float, float* %1, i64 %idxprom6.i
%90 = load float, float* %arrayidx7.i, align 4, !tbaa !12
%91 = load float, float* %arrayidx9.i, align 4, !tbaa !12
%arrayidx11.i = getelementptr inbounds float, float* %2, i64 %idxprom6.i
%92 = load float, float* %arrayidx11.i, align 4, !tbaa !12
%mul12.i = fmul float %91, %92
%93 = tail call float @llvm.fmuladd.f32(float %89, float %90, float
... %mul12.i) #3
%add.i = add nsw i32 %mul.i, %conv.i
%idxprom13.i = sext i32 %add.i to i64
%arrayidx14.i = getelementptr inbounds float, float* %0, i64 %idxprom13.i
%94 = load float, float* %arrayidx14.i, align 4, !tbaa !12
%add15.i = fadd float %94, %93
store float %add15.i, float* %arrayidx14.i, align 4, !tbaa !12,
... !llvm.access.group !28
%95 = add nuw i64 % local_id_x.0, 1
%exitcond.not = icmp eq i64 %95, %umax
br i1 %exitcond.not, label %pregon_for_end.i.loopexit, label
... %pregon_for_entry.entry.i, !llvm.loop !34
```

```
pregon_for_end.i.loopexit:
br label %pregon_for_end.i
```

```
pregon_for_end.i:
%96 = add nuw i64 % local_id_y.0, 1
%exitcond2.not = icmp eq i64 %96, %umax1
br i1 %exitcond2.not, label %gemver_kernel1.exit, label
... %pregon_for_entry.pregon_for_init.i, !llvm.loop !35
```

```
gemver_kernel1.exit:
ret void
```

CFG for '\_pocl\_kernel\_gemver\_kernel1' function