

```
%11:
%12 = sext i32 %2 to i64
%13 = icmp slt i64 %12, 32
%14 = select i1 %13, i64 %12, i64 32
%15 = sext i32 %1 to i64
%16 = icmp slt i64 %15, 8
%17 = select i1 %16, i64 %15, i64 8
%mul3.i.i = shl i64 %8, 5
%mul3.i.i = shl i64 %9, 3
%mul6.i = mul i32 %6, %1
%cmp970.i = icmp sgt i32 %2, 0
%18 = zext i32 %2 to i64
%19 = icmp ugt i64 %14, 1
%umax = select i1 %19, i64 %14, i64 1
%20 = icmp ugt i64 %17, 1
%umax27 = select i1 %20, i64 %17, i64 1
br i1 %cmp970.i, label %preregion_for_entry.preregion_for_init.i.us.preheader,
... label %preregion_for_entry.preregion_for_init.i.us.preheader
```

```
preregion_for_entry.preregion_for_init.i.us.preheader:
br label %preregion_for_entry.preregion_for_init.i.us
```

```
preregion_for_entry.preregion_for_init.i.us:
% local_id y.0.us = phi i64 [ %40, %preregion_for_end.i.us-lcssa.us.us ], [ 0,
... %preregion_for_entry.preregion_for_init.i.us.preheader ]
%add6.i.i.us = add i64 % local_id y.0.us.us, %mul3.i.i
%conv2.i.us = trunc i64 %add6.i.i.us to i32
%reass.add.i.us = add i32 %mul6.i, %conv2.i.us
%reass.mul.i.us = mul i32 %reass.add.i.us, %2
%30 = sext i32 %reass.mul.i.us to i64
br label %preregion_for_entry.entry.i.us.us
```

```
preregion_for_entry.entry.i.us.us:
% local_id x.0.us.us = phi i64 [ 0, %preregion_for_entry.preregion_for_init.i.us
... ], [ %32, %if.end.r_exit.i.loopexit.us.us ]
%add1.i.i.us.us = add i64 % local_id x.0.us.us.us, %mul.i.i
%conv1.i.us.us = trunc i64 %add1.i.i.us.us to i32
%add8.i.us.us = add nsw i32 %reass.mul.i.us, %conv1.i.us.us
%idxprom.i.us.us = sext i32 %add8.i.us.us to i64
%arrayidx.i.us.us = getelementptr inbounds float, float* %5, i64
... %idxprom.i.us.us
store float 0.000000e+00, float* %arrayidx.i.us.us, align 4, !tbaa !12,
... !llvm.access.group !16
%sext.i.us.us = shl i64 %add1.i.i.us.us, 32
%31 = ashr exact i64 %sext.i.us.us, 32
br label %for.body.i.us.us
```

```
for.body.i.us.us:
%indvars.iv.next.i2.us.us = phi i64 [ %indvars.iv.next.i.us.us,
... %for.body.i.us.us ], [ 0, %preregion_for_entry.entry.i.us.us ]
%33 = phi float [ %39, %for.body.i.us.us ], [ 0.0000000e+00,
... %preregion_for_entry.entry.i.us.us ]
%34 = add nsw i64 %indvars.iv.next.i2.us.us, %30
%arrayidx24.i.us.us = getelementptr inbounds float, float* %3, i64 %34
%35 = load float, float* %arrayidx24.i.us.us, align 4, !tbaa !12
%36 = mul nuw nsw i64 %indvars.iv.next.i2.us.us, %18
%37 = add nsw i64 %36, %31
%arrayidx28.i.us.us = getelementptr inbounds float, float* %4, i64 %37
%38 = load float, float* %arrayidx28.i.us.us, align 4, !tbaa !12
%39 = tail call @llvm.fmuladd.f32(float %35, float %38, float %33) #3
store float %39, float* %arrayidx.i.us.us, align 4, !tbaa !12,
... !llvm.access.group !16
%indvars.iv.next.i.us.us = add nuw nsw i64 %indvars.iv.next.i2.us.us, 1
%exitcond.not.i.us.us = icmp eq i64 %indvars.iv.next.i.us.us, %18
br i1 %exitcond.not.i.us.us, label %if.end.r_exit.i.loopexit.us.us, label
... %for.body.i.us.us, !llvm.loop !21
```

```
if.end.r_exit.i.loopexit.us.us:
%32 = add nuw i64 % local_id x.0.us.us, 1
%exitcond.not = icmp eq i64 %32, %umax
br i1 %exitcond.not, label %preregion_for_end.i.us-lcssa.us.us, label
... %preregion_for_entry.entry.i.us.us, !llvm.loop !19
```

```
preregion_for_end.i.us-lcssa.us.us:
%40 = add nuw i64 % local_id y.0.us, 1
%exitcond28.not = icmp eq i64 %40, %umax27
br i1 %exitcond28.not, label %doitgen_kernel1.exit.loopexit, label
... %preregion_for_entry.preregion_for_init.i.us, !llvm.loop !23
```

```
doitgen_kernel1.exit.loopexit:
br label %doitgen_kernel1.exit
```

```
preregion_for_entry.entry.i.preheader:
% local_id x.0.ph = phi i64 [ 0, %vector.scevcheck ], [ 0,
... %preregion_for_entry.preregion_for_init.i ], [ %n.vec, %middle.block ]
br label %preregion_for_entry.entry.i
```

```
preregion_for_entry.entry.i:
% local_id x.0 = phi i64 [ %62, %preregion_for_entry.entry.i ], [
... % local_id x.0.ph, %preregion_for_entry.entry.i.preheader ]
%add1.i.i = add i64 % local_id x.0, %mul.i.i
%conv1 = trunc i64 %add1.i.i to i32
%add8.i = add nsw i32 %reass.mul.i, %conv1
%idxprom.i = sext i32 %add8.i to i64
%arrayidx.i = getelementptr inbounds float, float* %5, i64 %idxprom.i
store float 0.000000e+00, float* %arrayidx.i, align 4, !tbaa !12,
... !llvm.access.group !16
%62 = add nuw i64 % local_id x.0, 1
%exitcond30.not = icmp eq i64 %62, %umax
br i1 %exitcond30.not, label %preregion_for_end.i.loopexit, label
... %preregion_for_entry.entry.i, !llvm.loop !27
```

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

```
preregion_for_end.i:
%63 = add nuw i64 % local_id y.0, 1
%exitcond32.not = icmp eq i64 %63, %umax27
br i1 %exitcond32.not, label %doitgen_kernel1.exit.loopexit36, label
... %preregion_for_entry.preregion_for_init.i, !llvm.loop !23
```

```
doitgen_kernel1.exit.loopexit36:
br label %doitgen_kernel1.exit
```

```
doitgen_kernel1.exit:
ret void
```

CFG for '_pocl_kernel_doitgen_kernel1' function

```
preregion_for_entry.preregion_for_init.i.preheader:
%21 = add nsw i64 %umax, -1
%22 = trunc i64 %9 to i32
%23 = shl i32 %22, 3
%24 = add i32 %mul6.i, %23
%25 = mul i32 %24, %2
%26 = zext i32 %25 to i64
%27 = shl i64 %8, 5
%28 = add i64 %27, %26
%29 = zext i32 %2 to i64
br label %preregion_for_entry.preregion_for_init.i
```

```
preregion_for_entry.preregion_for_init.i:
% local_id y.0 = phi i64 [ %63, %preregion_for_end.i ], [ 0,
... %preregion_for_entry.preregion_for_init.i.preheader ]
%41 = mul i64 % local_id y.0, %29
%42 = add i64 %28, %41
%43 = trunc i64 %42 to i32
%add6.i.i = add i64 % local_id y.0, %mul3.i.i
%conv2.i = trunc i64 %add6.i.i to i32
%reass.add.i = add i32 %mul6.i, %conv2.i
%reass.mul.i = mul i32 %reass.add.i, %2
%min.itors.check = icmp ult i64 %umax, 32
br i1 %min.itors.check, label %preregion_for_entry.entry.i.preheader, label
... %vector.scevcheck
```

```
vector.scevcheck:
%44 = trunc i64 %21 to i32
%45 = add i32 %43, %44
%46 = icmp slt i32 %45, %43
%47 = icmp ugt i64 %21, 4294967295
%48 = or i1 %46, %47
br i1 %48, label %preregion_for_entry.entry.i.preheader, label %vector.ph
```

```
vector.ph:
%n.vec = and i64 %umax, -32
br label %vector.body
```

```
vector.body:
%index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
%49 = add i64 %index, %mul.i.i
%50 = trunc i64 %49 to i32
%51 = add nsw i32 %reass.mul.i, %50
%52 = sext i32 %51 to i64
%53 = getelementptr inbounds float, float* %5, i64 %52
%54 = bitcast float* %53 to <8 x float>*
store <8 x float> zeroinitializer, <8 x float>* %54, align 4, !tbaa !12,
... !llvm.access.group !16
%55 = getelementptr inbounds float, float* %53, i64 8
%56 = bitcast float* %55 to <8 x float>*
store <8 x float> zeroinitializer, <8 x float>* %56, align 4, !tbaa !12,
... !llvm.access.group !16
%57 = getelementptr inbounds float, float* %53, i64 16
%58 = bitcast float* %57 to <8 x float>*
store <8 x float> zeroinitializer, <8 x float>* %58, align 4, !tbaa !12,
... !llvm.access.group !16
%59 = getelementptr inbounds float, float* %53, i64 24
%60 = bitcast float* %59 to <8 x float>*
store <8 x float> zeroinitializer, <8 x float>* %60, align 4, !tbaa !12,
... !llvm.access.group !16
%index.next = add i64 %index, 32
%61 = icmp eq i64 %index.next, %n.vec
br i1 %61, label %middle.block, label %vector.body, !llvm.loop !25
```

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