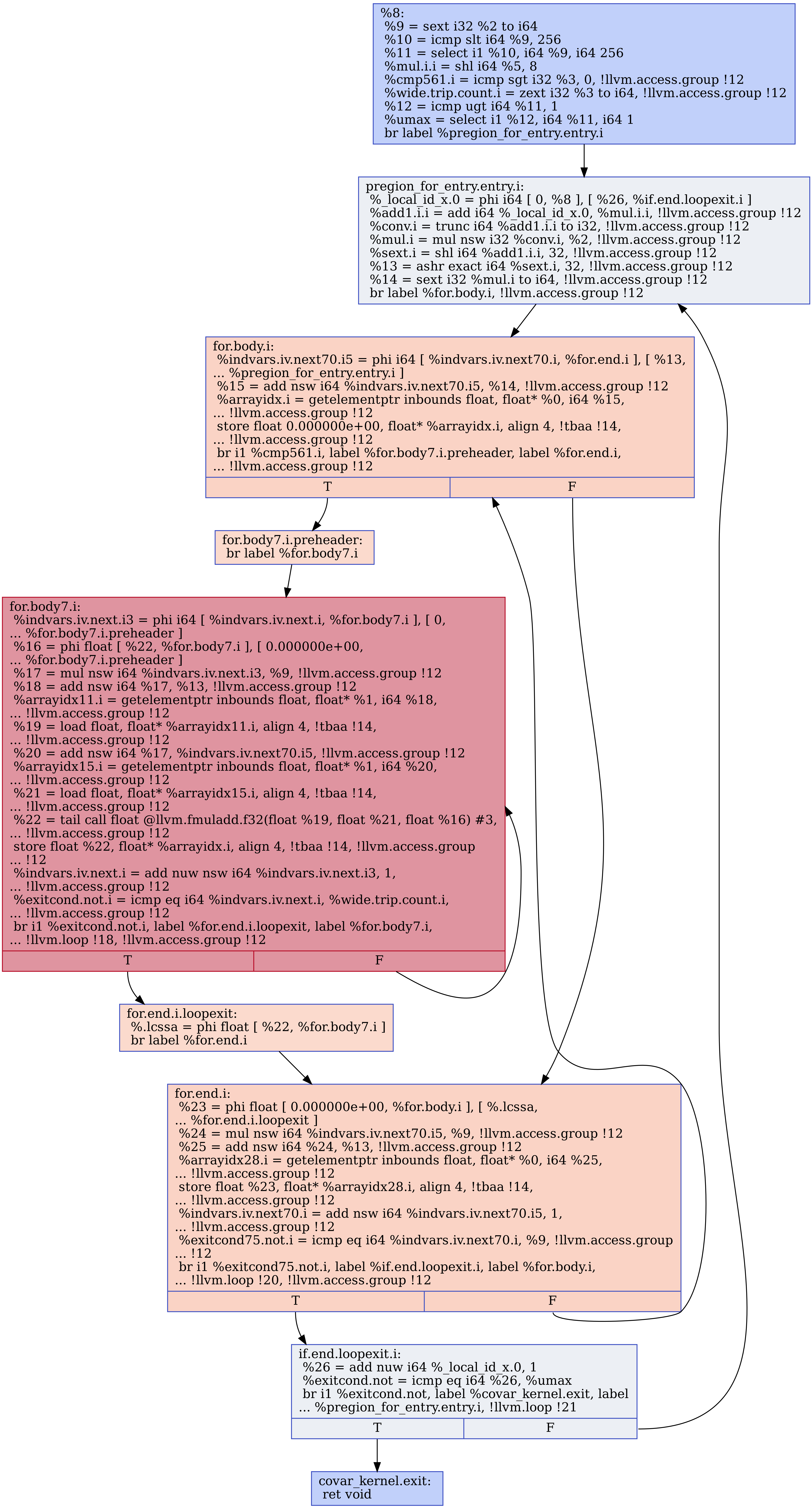

```
%8:
 %9 = sext i32 %2 to i64
 %10 = icmp slt i64 %9, 256
 %11 = select i1 %10, i64 %9, i64 256
 %mul.i.i = shl i64 %5, 8
 %cmp561.i = icmp sgt i32 %3, 0, !llvm.access.group !12
 %wide.trip.count.i = zext i32 %3 to i64, !llvm.access.group !12
 %12 = icmp ugt i64 %11, 1
 %umax = select i1 %12, i64 %11, i64 1
 br label %pregion_for_entry.entry.i
```

```
pregion_for_entry.entry.i:
 %_local_id_x.0 = phi i64 [ 0, %8 ], [ %26, %if.end.loopexit.i ]
 %add1.i.i = add i64 %_local_id_x.0, %mul.i.i, !llvm.access.group !12
 %conv.i = trunc i64 %add1.i.i to i32, !llvm.access.group !12
 %mul.i = mul nsw i32 %conv.i, %2, !llvm.access.group !12
 %sext.i = shl i64 %add1.i.i, 32, !llvm.access.group !12
 %13 = ashr exact i64 %sext.i, 32, !llvm.access.group !12
 %14 = sext i32 %mul.i to i64, !llvm.access.group !12
 br label %for.body.i, !llvm.access.group !12
```

```
for.body.i:
 %indvars.iv.next70.i5 = phi i64 [ %indvars.iv.next70.i, %for.end.i ], [ %13,
... %pregion_for_entry.entry.i ]
 %15 = add nsw i64 %indvars.iv.next70.i5, %14, !llvm.access.group !12
 %arrayidx.i = getelementptr inbounds float, float* %0, i64 %15,
... !llvm.access.group !12
 store float 0.000000e+00, float* %arrayidx.i, align 4, !tbaa !14,
... !llvm.access.group !12
 br i1 %cmp561.i, label %for.body7.i.preheader, label %for.end.i,
... !llvm.access.group !12
```

```
for.body7.i.preheader:
 br label %for.body7.i
```

```
for.body7.i:
 %indvars.iv.next.i3 = phi i64 [ %indvars.iv.next.i, %for.body7.i ], [ 0,
... %for.body7.i.preheader ]
 %16 = phi float [ %22, %for.body7.i ], [ 0.000000e+00,
... %for.body7.i.preheader ]
 %17 = mul nsw i64 %indvars.iv.next.i3, %9, !llvm.access.group !12
 %18 = add nsw i64 %17, %13, !llvm.access.group !12
 %arrayidx11.i = getelementptr inbounds float, float* %1, i64 %18,
... !llvm.access.group !12
 %19 = load float, float* %arrayidx11.i, align 4, !tbaa !14,
... !llvm.access.group !12
 %20 = add nsw i64 %17, %indvars.iv.next70.i5, !llvm.access.group !12
 %arrayidx15.i = getelementptr inbounds float, float* %1, i64 %20,
... !llvm.access.group !12
 %21 = load float, float* %arrayidx15.i, align 4, !tbaa !14,
... !llvm.access.group !12
 %22 = tail call float @llvm.fmuladd.f32(float %19, float %21, float %16) #3,
... !llvm.access.group !12
 store float %22, float* %arrayidx.i, align 4, !tbaa !14, !llvm.access.group
... !12
 %indvars.iv.next.i = add nuw nsw i64 %indvars.iv.next.i3, 1,
... !llvm.access.group !12
 %exitcond.not.i = icmp eq i64 %indvars.iv.next.i, %wide.trip.count.i,
... !llvm.access.group !12
 br i1 %exitcond.not.i, label %for.end.i.loopexit, label %for.body7.i,
... !llvm.loop !18, !llvm.access.group !12
```

```
for.end.i.loopexit:
 %.lcssa = phi float [ %22, %for.body7.i ]
 br label %for.end.i
```

```
for.end.i:
 %23 = phi float [ 0.000000e+00, %for.body.i ], [ %.lcssa,
... %for.end.i.loopexit ]
 %24 = mul nsw i64 %indvars.iv.next70.i5, %9, !llvm.access.group !12
 %25 = add nsw i64 %24, %13, !llvm.access.group !12
 %arrayidx28.i = getelementptr inbounds float, float* %0, i64 %25,
... !llvm.access.group !12
 store float %23, float* %arrayidx28.i, align 4, !tbaa !14,
... !llvm.access.group !12
 %indvars.iv.next70.i = add nsw i64 %indvars.iv.next70.i5, 1,
... !llvm.access.group !12
 %exitcond75.not.i = icmp eq i64 %indvars.iv.next70.i, %9, !llvm.access.group
... !12
 br i1 %exitcond75.not.i, label %if.end.loopexit.i, label %for.body.i,
... !llvm.loop !20, !llvm.access.group !12
```

```
if.end.loopexit.i:
 %26 = add nuw i64 %_local_id_x.0, 1
 %exitcond.not = icmp eq i64 %26, %umax
 br i1 %exitcond.not, label %covar_kernel.exit, label
... %pregion_for_entry.entry.i, !llvm.loop !21
```

```
covar_kernel.exit:
 ret void
```

CFG for '_pocl_kernel_covar_kernel' function