```
entry:
 %max_v_samp_factor = getelementptr inbounds %struct.jpeg_compress_struct,
... %struct.jpeg_compress_struct* %cinfo, i64 0, i32 39
 %0 = load i32, i32* %max_v_samp_factor, align 4, !tbaa !3
 %image_width = getelementptr inbounds %struct.jpeg_compress_struct,
... %struct.jpeg_compress_struct* %cinfo, i64 0, i32 6
 %1 = load i32, i32* %image_width, align 8, !tbaa !11
 tail call void @jcopy_sample_rows(i8** %input_data, i32 0, i8**
... %output_data, i32 0, i32 %0, i32 %1) #5
 %2 = load i32, i32* %max_v_samp_factor, align 4, !tbaa !3
 %3 = load i32, i32* %image_width, align 8, !tbaa !11
 %width_in_blocks = getelementptr inbounds %struct.jpeg_component_info,
... %struct.jpeg_component_info* %compptr, i64 0, i32 7
 %4 = load i32, i32* %width_in_blocks, align 4, !tbaa !12
 %mul = shl i32 %4, 3
 %sub.i = sub i32 %mul, %3
 %cmp.i = icmp sgt i32 %sub.i, 0
 %cmp121.i = icmp sgt i32 %2, 0
 %or.cond.i = and i1 %cmp121.i, %cmp.i
 br i1 %or.cond.i, label %for.body.us.preheader.i, label
... %expand_right_edge.exit
```

```
for.body.us.preheader.i:
 %idx.ext.i = zext i32 %3 to i64
 %5 = add i32 %3, -1
 %6 = sub i32 %5, %mul
 %7 = icmp sgt i32 %6, -2
 %smax.i = select i1 %7, i32 %6, i32 -2
 %8 = sub i32 1, %3
 %9 = add i32 %8, %mul
 %10 = add i32 %9, %smax.i
 %11 = zext i32 %10 to i64
 %12 = add nuw nsw i64 %11, 1
 %13 = add i32 %2, -1
 %xtraiter = and i32 %2, 3
 %lcmp.mod = icmp eq i32 %xtraiter, 0
 br i1 %lcmp.mod, label %for.body.us.preheader.i.split, label
... %for.body.us.i.prol.preheader
```

```
for.body.us.i.prol.preheader:
 br label %for.body.us.i.prol
```

```
for.body.us.i.prol:
 %indvars.iv.i.prol = phi i64 [ %indvars.iv.next.i.prol, %for.body.us.i.prol
... ], [ 0, %for.body.us.i.prol.preheader ]
 %prol.iter = phi i32 [ %prol.iter.sub, %for.body.us.i.prol ], [ %xtraiter,
... %for.body.us.i.prol.preheader ]
 %arrayidx.us.i.prol = getelementptr inbounds i8*, i8** %output_data, i64
... %indvars.iv.i.prol
 %14 = load i8*, i8** %arrayidx.us.i.prol, align 8, !tbaa !14
 %add.ptr.us.i.prol = getelementptr inbounds i8, i8* %14, i64 %idx.ext.i
 %arrayidx2.us.i.prol = getelementptr inbounds i8, i8* %add.ptr.us.i.prol,
... i64 -1
 %15 = load i8, i8* %arrayidx2.us.i.prol, align 1, !tbaa !15
 tail call void @llvm.memset.p0i8.i64(i8* %add.ptr.us.i.prol, i8 %15, i64
... %12, i32 1, i1 false) #5
 %indvars.iv.next.i.prol = add nuw nsw i64 %indvars.iv.i.prol, 1
 %prol.iter.sub = add i32 %prol.iter, -1
 %prol.iter.cmp = icmp eq i32 %prol.iter.sub, 0
 br i1 %prol.iter.cmp, label %for.body.us.preheader.i.split.loopexit, label
... %for.body.us.i.prol, !llvm.loop !16
```

```
for.body.us.preheader.i.split.loopexit:
 %indvars.iv.next.i.prol.lcssa = phi i64 [ %indvars.iv.next.i.prol,
... %for.body.us.i.prol ]
 br label %for.body.us.preheader.i.split
```

```
for.body.us.preheader.i.split:
 %indvars.iv.i.unr = phi i64 [ 0, %for.body.us.preheader.i ], [
... %indvars.iv.next.i.prol.lcssa, %for.body.us.preheader.i.split.loopexit ]
 %16 = icmp ult i32 %13, 3
 br i1 %16, label %expand_right_edge.exit.loopexit, label
... %for.body.us.preheader.i.split.split
```

```
for.body.us.preheader.i.split.split:
 br label %for.body.us.i
```

```
for.body.us.i:
 %indvars.iv.i = phi i64 [ %indvars.iv.i.unr,
... %for.body.us.preheader.i.split.split ], [ %indvars.iv.next.i.3,
... %for.body.us.i ]
 %arrayidx.us.i = getelementptr inbounds i8*, i8** %output_data, i64
... %indvars.iv.i
 %17 = load i8*, i8** %arrayidx.us.i, align 8, !tbaa !14
 %add.ptr.us.i = getelementptr inbounds i8, i8* %17, i64 %idx.ext.i
 %arrayidx2.us.i = getelementptr inbounds i8, i8* %add.ptr.us.i, i64 -1
 %18 = load i8, i8* %arrayidx2.us.i, align 1, !tbaa !15
 tail call void @llvm.memset.p0i8.i64(i8* %add.ptr.us.i, i8 %18, i64 %12, i32
... 1, i1 false) #5
 %indvars.iv.next.i = add nuw nsw i64 %indvars.iv.i, 1
 %arrayidx.us.i.1 = getelementptr inbounds i8*, i8** %output_data, i64
... %indvars.iv.next.i
 %19 = load i8*, i8** %arrayidx.us.i.1, align 8, !tbaa !14
 %add.ptr.us.i.1 = getelementptr inbounds i8, i8* %19, i64 %idx.ext.i
 %arrayidx2.us.i.1 = getelementptr inbounds i8, i8* %add.ptr.us.i.1, i64 -1
 %20 = load i8, i8* %arrayidx2.us.i.1, align 1, !tbaa !15
 tail call void @llvm.memset.p0i8.i64(i8* %add.ptr.us.i.1, i8 %20, i64 %12,
... i32 1, i1 false) #5
 %indvars.iv.next.i.1 = add nsw i64 %indvars.iv.i, 2
 %arrayidx.us.i.2 = getelementptr inbounds i8*, i8** %output_data, i64
... %indvars.iv.next.i.1
 %21 = load i8*, i8** %arrayidx.us.i.2, align 8, !tbaa !14
 %add.ptr.us.i.2 = getelementptr inbounds i8, i8* %21, i64 %idx.ext.i
 %arrayidx2.us.i.2 = getelementptr inbounds i8, i8* %add.ptr.us.i.2, i64 -1
 %22 = load i8, i8* %arrayidx2.us.i.2, align 1, !tbaa !15
 tail call void @llvm.memset.p0i8.i64(i8* %add.ptr.us.i.2, i8 %22, i64 %12,
... i32 1, i1 false) #5
 %indvars.iv.next.i.2 = add nsw i64 %indvars.iv.i, 3
 %arrayidx.us.i.3 = getelementptr inbounds i8*, i8** %output_data, i64
... %indvars.iv.next.i.2
 %23 = load i8*, i8** %arrayidx.us.i.3, align 8, !tbaa !14
 %add.ptr.us.i.3 = getelementptr inbounds i8, i8* %23, i64 %idx.ext.i
 %arrayidx2.us.i.3 = getelementptr inbounds i8, i8* %add.ptr.us.i.3, i64 -1
 %24 = load i8, i8* %arrayidx2.us.i.3, align 1, !tbaa !15
 tail call void @llvm.memset.p0i8.i64(i8* %add.ptr.us.i.3, i8 %24, i64 %12,
... i32 1, i1 false) #5
 %indvars.iv.next.i.3 = add nsw i64 %indvars.iv.i, 4
 %lftr.wideiv.3 = trunc i64 %indvars.iv.next.i.3 to i32
 %exitcond.3 = icmp eq i32 %lftr.wideiv.3, %2
 br i1 %exitcond.3, label %expand_right_edge.exit.loopexit.unr-lcssa, label
... %for.body.us.i
```

```
expand_right_edge.exit.loopexit.unr-lcssa:
 br label %expand_right_edge.exit.loopexit
```

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

```
expand_right_edge.exit:
 ret void
```

CFG for 'fullsize_downsample' function