

```

entry:
%rescale1 = getelementptr inbounds %struct.cjpeg_source_struct,
... %struct.cjpeg_source_struct* %sinfo, i64 1, i32 3
%0 = bitcast %struct._IO_FILE** %rescale1 to i8**
%1 = load i8*, i8** %0, align 8, !tbaa !3
%iobuffer = getelementptr inbounds %struct.cjpeg_source_struct,
... %struct.cjpeg_source_struct* %sinfo, i64 1
%2 = bitcast %struct.cjpeg_source_struct* %iobuffer to i8**
%3 = load i8*, i8** %2, align 8, !tbaa !11
%buffer_width = getelementptr inbounds %struct.cjpeg_source_struct,
... %struct.cjpeg_source_struct* %sinfo, i64 1, i32 2
%4 = bitcast void (%struct.jpeg_compress_struct*,
... %struct.cjpeg_source_struct** %buffer_width to i64*
%5 = load i64, i64* %4, align 8, !tbaa !12
%input_file = getelementptr inbounds %struct.cjpeg_source_struct,
... %struct.cjpeg_source_struct* %sinfo, i64 0, i32 3
%6 = load %struct._IO_FILE*, %struct._IO_FILE** %input_file, align 8, !tbaa
... !13
%call = tail call i64 @fread(i8* %3, i64 1, i64 %5, %struct._IO_FILE* %6)
%7 = load i64, i64* %4, align 8, !tbaa !12
%cmp = icmp eq i64 %call, %7
br i1 %cmp, label %if.end, label %if.then

```

T

F

```

if.then:
%err = getelementptr inbounds %struct.jpeg_compress_struct,
... %struct.jpeg_compress_struct* %cinfo, i64 0, i32 0
%8 = load %struct.jpeg_error_mgr*, %struct.jpeg_error_mgr** %err, align 8,
... !tbaa !14
%msg_code = getelementptr inbounds %struct.jpeg_error_mgr,
... %struct.jpeg_error_mgr* %8, i64 0, i32 5
store i32 42, i32* %msg_code, align 8, !tbaa !18
%error_exit = getelementptr inbounds %struct.jpeg_error_mgr,
... %struct.jpeg_error_mgr* %8, i64 0, i32 0
%9 = load void (%struct.jpeg_common_struct*)*, void
... (%struct.jpeg_common_struct** %error_exit, align 8, !tbaa !20
%10 = bitcast %struct.jpeg_compress_struct* %cinfo to
... %struct.jpeg_common_struct*
tail call void %9(%struct.jpeg_common_struct* %10) #3
br label %if.end

```

```

if.end:
%image_width = getelementptr inbounds %struct.jpeg_compress_struct,
... %struct.jpeg_compress_struct* %cinfo, i64 0, i32 6
%11 = load i32, i32* %image_width, align 8, !tbaa !21
%cmp640 = icmp eq i32 %11, 0
br i1 %cmp640, label %for.end, label %for.body.preheader

```

T

F

```

for.body.preheader:
%12 = load i8*, i8** %2, align 8, !tbaa !11
%buffer = getelementptr inbounds %struct.cjpeg_source_struct,
... %struct.cjpeg_source_struct* %sinfo, i64 0, i32 4
%13 = load i8**, i8*** %buffer, align 8, !tbaa !22
%14 = load i8*, i8** %13, align 8, !tbaa !23
%xtraiter = and i32 %11, 1
%lcmp.mod = icmp eq i32 %xtraiter, 0
br i1 %lcmp.mod, label %for.body.preheader.split, label %for.body.prol

```

T

F

```

for.body.prol:
%incdec.ptr.prol = getelementptr inbounds i8, i8* %12, i64 1
%15 = load i8, i8* %12, align 1, !tbaa !24
%idxprom.prol = zext i8 %15 to i64
%arrayidx7.prol = getelementptr inbounds i8, i8* %1, i64 %idxprom.prol
%16 = load i8, i8* %arrayidx7.prol, align 1, !tbaa !24
%incdec.ptr8.prol = getelementptr inbounds i8, i8* %14, i64 1
store i8 %16, i8* %14, align 1, !tbaa !24
%incdec.ptr9.prol = getelementptr inbounds i8, i8* %12, i64 2
%17 = load i8, i8* %incdec.ptr.prol, align 1, !tbaa !24
%idxprom11.prol = zext i8 %17 to i64
%arrayidx12.prol = getelementptr inbounds i8, i8* %1, i64 %idxprom11.prol
%18 = load i8, i8* %arrayidx12.prol, align 1, !tbaa !24
%incdec.ptr13.prol = getelementptr inbounds i8, i8* %14, i64 2
store i8 %18, i8* %incdec.ptr8.prol, align 1, !tbaa !24
%incdec.ptr14.prol = getelementptr inbounds i8, i8* %12, i64 3
%19 = load i8, i8* %incdec.ptr9.prol, align 1, !tbaa !24
%idxprom16.prol = zext i8 %19 to i64
%arrayidx17.prol = getelementptr inbounds i8, i8* %1, i64 %idxprom16.prol
%20 = load i8, i8* %arrayidx17.prol, align 1, !tbaa !24
%incdec.ptr18.prol = getelementptr inbounds i8, i8* %14, i64 3
store i8 %20, i8* %incdec.ptr13.prol, align 1, !tbaa !24
%dec.prol = add i32 %11, -1
br label %for.body.preheader.split

```

```

for.body.preheader.split:
%col.043.unr = phi i32 [ %11, %for.body.preheader ], [ %dec.prol,
... %for.body.prol ]
%ptr.042.unr = phi i8* [ %14, %for.body.preheader ], [ %incdec.ptr18.prol,
... %for.body.prol ]
%bufferptr.041.unr = phi i8* [ %12, %for.body.preheader ], [
... %incdec.ptr14.prol, %for.body.prol ]
%21 = icmp eq i32 %11, 1
br i1 %21, label %for.end.loopexit, label %for.body.preheader.split.split

```

T

F

```

for.body.preheader.split.split:
br label %for.body

```

```

for.body:
%col.043 = phi i32 [ %col.043.unr, %for.body.preheader.split.split ], [
... %dec.1, %for.body ]
%ptr.042 = phi i8* [ %ptr.042.unr, %for.body.preheader.split.split ], [
... %incdec.ptr18.1, %for.body ]
%bufferptr.041 = phi i8* [ %bufferptr.041.unr,
... %for.body.preheader.split.split ], [ %incdec.ptr14.1, %for.body ]
%incdec.ptr = getelementptr inbounds i8, i8* %bufferptr.041, i64 1
%22 = load i8, i8* %bufferptr.041, align 1, !tbaa !24
%idxprom = zext i8 %22 to i64
%arrayidx7 = getelementptr inbounds i8, i8* %1, i64 %idxprom
%23 = load i8, i8* %arrayidx7, align 1, !tbaa !24
%incdec.ptr8 = getelementptr inbounds i8, i8* %ptr.042, i64 1
store i8 %23, i8* %ptr.042, align 1, !tbaa !24
%incdec.ptr9 = getelementptr inbounds i8, i8* %bufferptr.041, i64 2
%24 = load i8, i8* %incdec.ptr, align 1, !tbaa !24
%idxprom11 = zext i8 %24 to i64
%arrayidx12 = getelementptr inbounds i8, i8* %1, i64 %idxprom11
%25 = load i8, i8* %arrayidx12, align 1, !tbaa !24
%incdec.ptr13 = getelementptr inbounds i8, i8* %ptr.042, i64 2
store i8 %25, i8* %incdec.ptr8, align 1, !tbaa !24
%incdec.ptr14 = getelementptr inbounds i8, i8* %bufferptr.041, i64 3
%26 = load i8, i8* %incdec.ptr9, align 1, !tbaa !24
%idxprom16 = zext i8 %26 to i64
%arrayidx17 = getelementptr inbounds i8, i8* %1, i64 %idxprom16
%27 = load i8, i8* %arrayidx17, align 1, !tbaa !24
%incdec.ptr18 = getelementptr inbounds i8, i8* %ptr.042, i64 3
store i8 %27, i8* %incdec.ptr13, align 1, !tbaa !24
%incdec.ptr.1 = getelementptr inbounds i8, i8* %bufferptr.041, i64 4
%28 = load i8, i8* %incdec.ptr14, align 1, !tbaa !24
%idxprom.1 = zext i8 %28 to i64
%arrayidx7.1 = getelementptr inbounds i8, i8* %1, i64 %idxprom.1
%29 = load i8, i8* %arrayidx7.1, align 1, !tbaa !24
%incdec.ptr8.1 = getelementptr inbounds i8, i8* %ptr.042, i64 4
store i8 %29, i8* %incdec.ptr18, align 1, !tbaa !24
%incdec.ptr9.1 = getelementptr inbounds i8, i8* %bufferptr.041, i64 5
%30 = load i8, i8* %incdec.ptr.1, align 1, !tbaa !24
%idxprom11.1 = zext i8 %30 to i64
%arrayidx12.1 = getelementptr inbounds i8, i8* %1, i64 %idxprom11.1
%31 = load i8, i8* %arrayidx12.1, align 1, !tbaa !24
%incdec.ptr13.1 = getelementptr inbounds i8, i8* %ptr.042, i64 5
store i8 %31, i8* %incdec.ptr8.1, align 1, !tbaa !24
%incdec.ptr14.1 = getelementptr inbounds i8, i8* %bufferptr.041, i64 6
%32 = load i8, i8* %incdec.ptr9.1, align 1, !tbaa !24
%idxprom16.1 = zext i8 %32 to i64
%arrayidx17.1 = getelementptr inbounds i8, i8* %1, i64 %idxprom16.1
%33 = load i8, i8* %arrayidx17.1, align 1, !tbaa !24
%incdec.ptr18.1 = getelementptr inbounds i8, i8* %ptr.042, i64 6
store i8 %33, i8* %incdec.ptr13.1, align 1, !tbaa !24
%dec.1 = add i32 %col.043, -2
%cmp6.1 = icmp eq i32 %dec.1, 0
br i1 %cmp6.1, label %for.end.loopexit.unr-lesssa, label %for.body

```

T

F

```

for.end.loopexit.unr-lesssa:
br label %for.end.loopexit

```

```

for.end.loopexit:
br label %for.end

```

```

for.end:
ret i32 1

```

CFG for 'get\_scaled\_rgb\_row' function