entry:
```
%workspace = alloca [64 x i32], align 16
%fdct1 = getelementptr inbounds %struct.jpeg_compress_struct,
... %struct.jpeg_compress_struct* %cinfo, i64 0, i32 58
%0 = bitcast %struct.jpeg_forward_dct** %fdct1 to
... %struct.my_fdct_controller**
%1 = load %struct.my_fdct_controller*, %struct.my_fdct_controller** %0,
... align 8, !tbaa !3
%do_dct2 = getelementptr inbounds %struct.my_fdct_controller,
... %struct.my_fdct_controller* %1, i64 0, i32 1
%2 = load void (i32*)*, void (i32*)** %do_dct2, align 8, !tbaa !11
%quant_tbl_no = getelementptr inbounds %struct.jpeg_component_info,
... %struct.jpeg_component_info* %compptr, i64 0, i32 4
%3 = load i32, i32* %quant_tbl_no, align 8, !tbaa !14
%idxprom = sext i32 %3 to i64
%arrayidx = getelementptr inbounds %struct.my_fdct_controller,
... %struct.my_fdct_controller* %1, i64 0, i32 2, i64 %idxprom
%4 = load i32*, i32** %arrayidx, align 8, !tbaa !16
%5 = bitcast [64 x i32]* %workspace to i8*
call void @llvm.lifetime.start(i64 256, i8* %5) #3
%idx.ext = zext i32 %start_row to i64
%add.ptr = getelementptr inbounds i8*, i8** %sample_data, i64 %idx.ext
%cmp132 = icmp eq i32 %num_blocks, 0
br i1 %cmp132, label %for.end78, label %for.body.lr.ph, !prof !17
```
T | F

for.body.lr.ph:
```
%arraydecay = getelementptr inbounds [64 x i32], [64 x i32]* %workspace, i64
... 0, i64 0
br label %for.body
```

for.body:
```
%indvars.iv138 = phi i64 [ 0, %for.body.lr.ph ], [ %indvars.iv.next139,
... %for.end74 ]
%start_col.addr.0134 = phi i32 [ %start_col, %for.body.lr.ph ], [ %add77,
... %for.end74 ]
%idx.ext9 = zext i32 %start_col.addr.0134 to i64
br label %for.body6
```

for.body6:
```
%indvars.iv = phi i64 [ 0, %for.body ], [ %indvars.iv.next, %for.body6 ]
%workspaceptr.0129 = phi i32* [ %arraydecay, %for.body ], [ %incdec.ptr39,
... %for.body6 ]
%arrayidx8 = getelementptr inbounds i8*, i8** %add.ptr, i64 %indvars.iv
%6 = load i8*, i8** %arrayidx8, align 8, !tbaa !16
%add.ptr10 = getelementptr inbounds i8, i8* %6, i64 %idx.ext9
%incdec.ptr = getelementptr inbounds i8, i8* %add.ptr10, i64 1
%7 = load i8, i8* %add.ptr10, align 1, !tbaa !18
%conv = zext i8 %7 to i32
%sub = add nsw i32 %conv, -128
%incdec.ptr11 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 1
store i32 %sub, i32* %workspaceptr.0129, align 4, !tbaa !19
%incdec.ptr12 = getelementptr inbounds i8, i8* %incdec.ptr, i64 1
%8 = load i8, i8* %incdec.ptr, align 1, !tbaa !18
%conv13 = zext i8 %8 to i32
%sub14 = add nsw i32 %conv13, -128
%incdec.ptr15 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 2
store i32 %sub14, i32* %incdec.ptr11, align 4, !tbaa !19
%incdec.ptr16 = getelementptr inbounds i8, i8* %incdec.ptr12, i64 1
%9 = load i8, i8* %incdec.ptr12, align 1, !tbaa !18
%conv17 = zext i8 %9 to i32
%sub18 = add nsw i32 %conv17, -128
%incdec.ptr19 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 3
store i32 %sub18, i32* %incdec.ptr15, align 4, !tbaa !19
%incdec.ptr20 = getelementptr inbounds i8, i8* %incdec.ptr16, i64 1
%10 = load i8, i8* %incdec.ptr16, align 1, !tbaa !18
%conv21 = zext i8 %10 to i32
%sub22 = add nsw i32 %conv21, -128
%incdec.ptr23 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 4
store i32 %sub22, i32* %incdec.ptr19, align 4, !tbaa !19
%incdec.ptr24 = getelementptr inbounds i8, i8* %incdec.ptr20, i64 1
%11 = load i8, i8* %incdec.ptr20, align 1, !tbaa !18
%conv25 = zext i8 %11 to i32
%sub26 = add nsw i32 %conv25, -128
%incdec.ptr27 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 5
store i32 %sub26, i32* %incdec.ptr23, align 4, !tbaa !19
%incdec.ptr28 = getelementptr inbounds i8, i8* %incdec.ptr24, i64 1
%12 = load i8, i8* %incdec.ptr24, align 1, !tbaa !18
%conv29 = zext i8 %12 to i32
%sub30 = add nsw i32 %conv29, -128
%incdec.ptr31 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 6
store i32 %sub30, i32* %incdec.ptr27, align 4, !tbaa !19
%incdec.ptr32 = getelementptr inbounds i8, i8* %incdec.ptr28, i64 1
%13 = load i8, i8* %incdec.ptr28, align 1, !tbaa !18
%conv33 = zext i8 %13 to i32
%sub34 = add nsw i32 %conv33, -128
%incdec.ptr35 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 7
store i32 %sub34, i32* %incdec.ptr31, align 4, !tbaa !19
%14 = load i8, i8* %incdec.ptr32, align 1, !tbaa !18
%conv37 = zext i8 %14 to i32
%sub38 = add nsw i32 %conv37, -128
%incdec.ptr39 = getelementptr inbounds i32, i32* %workspaceptr.0129, i64 8
store i32 %sub38, i32* %incdec.ptr35, align 4, !tbaa !19
%indvars.iv.next = add nuw nsw i64 %indvars.iv, 1
%exitcond = icmp eq i64 %indvars.iv.next, 8
br i1 %exitcond, label %for.end, label %for.body6, !prof !20
```
T | F

for.end:
```
call void %2(i32* %arraydecay) #3
br label %for.body47
```

for.body47:
```
%indvars.iv135 = phi i64 [ 0, %for.end ], [ %indvars.iv.next136, %if.end68 ]
%arrayidx49 = getelementptr inbounds i32, i32* %4, i64 %indvars.iv135
%15 = load i32, i32* %arrayidx49, align 4, !tbaa !19
%arrayidx51 = getelementptr inbounds [64 x i32], [64 x i32]* %workspace, i64
... 0, i64 %indvars.iv135
%16 = load i32, i32* %arrayidx51, align 4, !tbaa !19
%cmp52 = icmp slt i32 %16, 0
%shr = ashr i32 %15, 1
br i1 %cmp52, label %if.then, label %if.else59, !prof !21
```
T | F

if.then:
```
%add = sub i32 %shr, %16
%cmp55 = icmp slt i32 %add, %15
br i1 %cmp55, label %if.end, label %if.then57, !prof !22
```
T | F

if.then57:
```
%div = sdiv i32 %add, %15
br label %if.end
```

if.else59:
```
%add61 = add nsw i32 %16, %shr
%cmp62 = icmp slt i32 %add61, %15
br i1 %cmp62, label %if.end68, label %if.then64, !prof !23
```
T | F

if.end:
```
%temp.0 = phi i32 [ %div, %if.then57 ], [ 0, %if.then ]
%sub58 = sub nsw i32 0, %temp.0
br label %if.end68
```

if.then64:
```
%div65 = sdiv i32 %add61, %15
br label %if.end68
```

if.end68:
```
%temp.1 = phi i32 [ %sub58, %if.end ], [ %div65, %if.then64 ], [ 0,
... %if.else59 ]
%conv69 = trunc i32 %temp.1 to i16
%arrayidx71 = getelementptr inbounds [64 x i16], [64 x i16]* %coef_blocks,
... i64 %indvars.iv138, i64 %indvars.iv135
store i16 %conv69, i16* %arrayidx71, align 2, !tbaa !24
%indvars.iv.next136 = add nuw nsw i64 %indvars.iv135, 1
%exitcond137 = icmp eq i64 %indvars.iv.next136, 64
br i1 %exitcond137, label %for.end74, label %for.body47, !prof !25
```
T | F

for.end74:
```
%indvars.iv.next139 = add nuw nsw i64 %indvars.iv138, 1
%add77 = add i32 %start_col.addr.0134, 8
%lftr.wideiv = trunc i64 %indvars.iv.next139 to i32
%exitcond140 = icmp eq i32 %lftr.wideiv, %num_blocks
br i1 %exitcond140, label %for.end78.loopexit, label %for.body, !prof !17
```
T | F

for.end78.loopexit:
```
br label %for.end78
```

for.end78:
```
call void @llvm.lifetime.end(i64 256, i8* %5) #3
ret void
```

CFG for 'forward_DCT' function