

```

%7:
%8 = shl i64 %4, 8
%add.i = add nsw i32 %1, 1
%mul.i = mul nsw i32 %2, %1
%add6.i = add nsw i32 %mul.i, %1
%idxprom7.i = sext i32 %add6.i to i64
%arrayidx8.i = getelementptr inbounds float, float* %0, i64 %idxprom7.i
br label %preion_for_entry.entry.i

```

```

preion_for_entry.entry.i:
%_local_id_x.0 = phi i64 [ 0, %7 ], [ %13, %if.end.r_exit.i ]
%9 = add nuw nsw i64 %_local_id_x.0, %8
%10 = trunc i64 %9 to i32
%conv2.i = add i32 %add.i, %10
%cmp.i = icmp slt i32 %conv2.i, %2
br i1 %cmp.i, label %if.then.i, label %if.end.r_exit.i

```

T

F

```

if.then.i:
%add4.i = add nsw i32 %conv2.i, %mul.i
%idxprom.i = sext i32 %add4.i to i64
%arrayidx.i = getelementptr inbounds float, float* %0, i64 %idxprom.i
%11 = load float, float* %arrayidx.i, align 4, !tbaa !12
%12 = load float, float* %arrayidx8.i, align 4, !tbaa !12
%div.i = fdiv float %11, %12, !fpmath !16
store float %div.i, float* %arrayidx.i, align 4, !tbaa !12,
... !llvm.access.group !17
br label %if.end.r_exit.i

```

```

if.end.r_exit.i:
%13 = add nuw nsw i64 %_local_id_x.0, 1
%exitcond = icmp eq i64 %13, 256
br i1 %exitcond, label %lu_kernel1.exit, label %preion_for_entry.entry.i,
... !llvm.loop !19

```

T

F

```

lu_kernel1.exit:
ret void

```

CFG for '_pocl_kernel_lu_kernel1' function