

```

vector.ph:
%sub.i = add nsw i32 %2, -1, !llvm.access.group !12
%mul.i.i = shl i64 %4, 8
%broadcast.splatinsert = insertelement <8 x i64> undef, i64 %mul.i.i, i32 0
%broadcast.splat = shufflevector <8 x i64> %broadcast.splatinsert, <8 x i64>
... undef, <8 x i32> zeroinitializer
%broadcast.splatinsert1 = insertelement <8 x i32> undef, i32 %sub.i, i32 0
%broadcast.splat2 = shufflevector <8 x i32> %broadcast.splatinsert1, <8 x
... i32> undef, <8 x i32> zeroinitializer
br label %vector.body

```

```

vector.body:
%index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]
%vec.ind = phi <8 x i64> [ <i64 0, i64 1, i64 2, i64 3, i64 4, i64 5, i64 6,
... i64 7>, %vector.ph ], [ %vec.ind.next, %vector.body ]
%7 = add nuw nsw <8 x i64> %vec.ind, %broadcast.splat, !llvm.access.group !12
%8 = trunc <8 x i64> %7 to <8 x i32>, !llvm.access.group !12
%9 = icmp sgt <8 x i32> %8, zeroinitializer, !llvm.access.group !12
%10 = icmp sgt <8 x i32> %broadcast.splat2, %8, !llvm.access.group !12
%11 = and <8 x i1> %9, %10, !llvm.access.group !12
%12 = extractelement <8 x i64> %7, i32 0
%13 = shl i64 %12, 32, !llvm.access.group !12
%14 = ashr exact i64 %13, 32, !llvm.access.group !12
%15 = getelementptr inbounds float, float* %1, i64 %14, !llvm.access.group
... !12
%16 = bitcast float* %15 to <8 x float>*
%wide.load = load <8 x float>, <8 x float>* %16, align 4, !tbaa !14,
... !llvm.access.group !12
%17 = getelementptr inbounds float, float* %0, i64 %14, !llvm.access.group
... !12
%18 = bitcast float* %17 to <8 x float>*
call void @llvm.masked.store.v8f32.p0v8f32(<8 x float> %wide.load, <8 x
... float>* %18, i32 4, <8 x i1> %11), !tbaa !14, !llvm.access.group !12
%index.next = add i64 %index, 8
%vec.ind.next = add <8 x i64> %vec.ind, <i64 8, i64 8, i64 8, i64 8, i64 8,
... i64 8, i64 8, i64 8>
%19 = icmp eq i64 %index.next, 256
br i1 %19, label %runJacobi1D_kernel2.exit, label %vector.body, !llvm.loop
... !18

```

T

F

```

runJacobi1D_kernel2.exit:
ret void

```

CFG for '_pocl_kernel_runJacobi1D_kernel2' function