```
%8:
 %9 = shl i64 %5, 5
 %cmp218.i = icmp sgt i32 %3, 0
 %wide.trip.count.i = zext i32 %3 to i64
 %10 = add nsw i64 %wide.trip.count.i, -1
 %xtraiter.i = and i64 %wide.trip.count.i, 3
 %11 = icmp ult i64 %10, 3
 %unroll_iter.i = sub nuw nsw i64 %wide.trip.count.i, %xtraiter.i
 %lcmp.mod.i = icmp eq i64 %xtraiter.i, 0
 br i1 %cmp218.i, label %pregion_for_entry.entry.i.us.preheader, label
... %mvt_kernel1.exit
```

```
pregion_for_entry.entry.i.us.preheader:
 br label %pregion_for_entry.entry.i.us
```

```
pregion_for_entry.entry.i.us:
 %_local_id_x.0.us = phi i64 [ %38, %if.end.i.us ], [ 0,
... %pregion_for_entry.entry.i.us.preheader ]
 %12 = add nuw nsw i64 %_local_id_x.0.us, %9
 %conv.i.us = trunc i64 %12 to i32
 %cmp.i.us = icmp slt i32 %conv.i.us, %3
 br i1 %cmp.i.us, label %for.body.lr.ph.i.us, label %if.end.i.us
```

```
for.body.lr.ph.i.us:
 %mul.i.us = mul nsw i32 %conv.i.us, %3
 %sext.i.us = shl i64 %12, 32
 %idxprom7.i.us = ashr exact i64 %sext.i.us, 32
 %arrayidx8.i.us = getelementptr inbounds float, float* %1, i64 %idxprom7.i.us
 %13 = sext i32 %mul.i.us to i64
 %.pre.i1.us15 = load float, float* %arrayidx8.i.us, align 4, !tbaa !12
 br i1 %11, label %if.end.loopexit.unr-lcssa.i.us, label
... %for.body.i.us.preheader
```

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

```
for.body.i.us:
 %niter.nsub.3.i7.us = phi i64 [ %niter.nsub.3.i.us, %for.body.i.us ], [
... %unroll_iter.i, %for.body.i.us.preheader ]
 %indvars.iv.next.3.i4.us = phi i64 [ %indvars.iv.next.3.i.us, %for.body.i.us
... ], [ 0, %for.body.i.us.preheader ]
 %14 = phi float [ %30, %for.body.i.us ], [ %.pre.i1.us15,
... %for.body.i.us.preheader ]
 %15 = add nsw i64 %indvars.iv.next.3.i4.us, %13
 %arrayidx.i.us = getelementptr inbounds float, float* %0, i64 %15
 %16 = load float, float* %arrayidx.i.us, align 4, !tbaa !12
 %arrayidx5.i.us = getelementptr inbounds float, float* %2, i64
... %indvars.iv.next.3.i4.us
 %17 = load float, float* %arrayidx5.i.us, align 4, !tbaa !12
 %18 = tail call float @llvm.fmuladd.f32(float %16, float %17, float %14) #2
 store float %18, float* %arrayidx8.i.us, align 4, !tbaa !12,
... !llvm.access.group !16
 %indvars.iv.next.i.us = or i64 %indvars.iv.next.3.i4.us, 1
 %19 = add nsw i64 %indvars.iv.next.i.us, %13
 %arrayidx.1.i.us = getelementptr inbounds float, float* %0, i64 %19
 %20 = load float, float* %arrayidx.1.i.us, align 4, !tbaa !12
 %arrayidx5.1.i.us = getelementptr inbounds float, float* %2, i64
... %indvars.iv.next.i.us
 %21 = load float, float* %arrayidx5.1.i.us, align 4, !tbaa !12
 %22 = tail call float @llvm.fmuladd.f32(float %20, float %21, float %18) #2
 store float %22, float* %arrayidx8.i.us, align 4, !tbaa !12,
... !llvm.access.group !16
 %indvars.iv.next.1.i.us = or i64 %indvars.iv.next.3.i4.us, 2
 %23 = add nsw i64 %indvars.iv.next.1.i.us, %13
 %arrayidx.2.i.us = getelementptr inbounds float, float* %0, i64 %23
 %24 = load float, float* %arrayidx.2.i.us, align 4, !tbaa !12
 %arrayidx5.2.i.us = getelementptr inbounds float, float* %2, i64
... %indvars.iv.next.1.i.us
 %25 = load float, float* %arrayidx5.2.i.us, align 4, !tbaa !12
 %26 = tail call float @llvm.fmuladd.f32(float %24, float %25, float %22) #2
 store float %26, float* %arrayidx8.i.us, align 4, !tbaa !12,
... !llvm.access.group !16
 %indvars.iv.next.2.i.us = or i64 %indvars.iv.next.3.i4.us, 3
 %27 = add nsw i64 %indvars.iv.next.2.i.us, %13
 %arrayidx.3.i.us = getelementptr inbounds float, float* %0, i64 %27
 %28 = load float, float* %arrayidx.3.i.us, align 4, !tbaa !12
 %arrayidx5.3.i.us = getelementptr inbounds float, float* %2, i64
... %indvars.iv.next.2.i.us
 %29 = load float, float* %arrayidx5.3.i.us, align 4, !tbaa !12
 %30 = tail call float @llvm.fmuladd.f32(float %28, float %29, float %26) #2
 store float %30, float* %arrayidx8.i.us, align 4, !tbaa !12,
... !llvm.access.group !16
 %indvars.iv.next.3.i.us = add nuw nsw i64 %indvars.iv.next.3.i4.us, 4
 %niter.nsub.3.i.us = add i64 %niter.nsub.3.i7.us, -4
 %niter.ncmp.3.i.us = icmp eq i64 %niter.nsub.3.i.us, 0
 br i1 %niter.ncmp.3.i.us, label %if.end.loopexit.unr-lcssa.i.us.loopexit,
... label %for.body.i.us
```

```
if.end.loopexit.unr-lcssa.i.us.loopexit:
 %.lcssa = phi float [ %30, %for.body.i.us ]
 %indvars.iv.next.3.i.us.lcssa = phi i64 [ %indvars.iv.next.3.i.us,
... %for.body.i.us ]
 br label %if.end.loopexit.unr-lcssa.i.us
```

```
if.end.loopexit.unr-lcssa.i.us:
 %31 = phi float [ %.pre.i1.us15, %for.body.lr.ph.i.us ], [ %.lcssa,
... %if.end.loopexit.unr-lcssa.i.us.loopexit ]
 %32 = phi i64 [ 0, %for.body.lr.ph.i.us ], [ %indvars.iv.next.3.i.us.lcssa,
... %if.end.loopexit.unr-lcssa.i.us.loopexit ]
 br i1 %lcmp.mod.i, label %if.end.i.us, label %for.body.epil.i.us.preheader
```

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

```
for.body.epil.i.us:
 %epil.iter.sub.i13.us = phi i64 [ %epil.iter.sub.i.us, %for.body.epil.i.us
... ], [ %xtraiter.i, %for.body.epil.i.us.preheader ]
 %indvars.iv.next.epil.i11.us = phi i64 [ %indvars.iv.next.epil.i.us,
... %for.body.epil.i.us ], [ %32, %for.body.epil.i.us.preheader ]
 %33 = phi float [ %37, %for.body.epil.i.us ], [ %31,
... %for.body.epil.i.us.preheader ]
 %34 = add nsw i64 %indvars.iv.next.epil.i11.us, %13
 %arrayidx.epil.i.us = getelementptr inbounds float, float* %0, i64 %34
 %35 = load float, float* %arrayidx.epil.i.us, align 4, !tbaa !12
 %arrayidx5.epil.i.us = getelementptr inbounds float, float* %2, i64
... %indvars.iv.next.epil.i11.us
 %36 = load float, float* %arrayidx5.epil.i.us, align 4, !tbaa !12
 %37 = tail call float @llvm.fmuladd.f32(float %35, float %36, float %33) #2
 store float %37, float* %arrayidx8.i.us, align 4, !tbaa !12,
... !llvm.access.group !16
 %indvars.iv.next.epil.i.us = add nuw nsw i64 %indvars.iv.next.epil.i11.us, 1
 %epil.iter.sub.i.us = add nsw i64 %epil.iter.sub.i13.us, -1
 %epil.iter.cmp.i.us = icmp eq i64 %epil.iter.sub.i.us, 0
 br i1 %epil.iter.cmp.i.us, label %if.end.i.us.loopexit, label
... %for.body.epil.i.us, !llvm.loop !18
```

```
if.end.i.us.loopexit:
 br label %if.end.i.us
```

```
if.end.i.us:
 %38 = add nuw nsw i64 %_local_id_x.0.us, 1
 %exitcond = icmp eq i64 %38, 32
 br i1 %exitcond, label %mvt_kernel1.exit.loopexit, label
... %pregion_for_entry.entry.i.us, !llvm.loop !20
```

```
mvt_kernel1.exit.loopexit:
 br label %mvt_kernel1.exit
```

```
mvt_kernel1.exit:
 ret void
```

CFG for '_pocl_kernel_mvt_kernel1' function