```
%mul.i.i = shl i64 %6, 5
                                                                                            mul3.i.i = shl i64 \%7, 3
                                                                                            %sub.i = add nsw i32 %3, -1, !llvm.access.group !12
                                                                                            %sub4.i = add nsw i32 %4, -1, !llvm.access.group !12
                                                                                            br label %pregion for entry.pregion for init.i
                                                                        pregion for entry.pregion for init.i:
                                                                        % local id y.0 = phi i64 [0, \sqrt{8}9], [\%17, \%pregion for end.i]
                                                                        %add6.i.i = add nuw nsw i64 %_local_id_y.0, %mul3.i.i, !llvm.access.group !12
                                                                        %conv2.i = trunc i64 %add6.i.i to i32, !llvm.access.group !12
                                                                        %cmp.i = icmp sgt i32 %sub.i, %conv2.i, !llvm.access.group !12
                                                                        %mul.i = mul nsw i32 %conv2.i, %4
                                                                        %add18.i = add nsw i32 %conv2.i, 1
                                                                         %mul19.i = mul nsw i32 %add18.i, %4
                                                                        br label %pregion for entry.entry.i
                                     pregion for entry.entry.i:
                                      % [\cos \overline{1} \text{ id } x.0 = \text{phi } i64 \text{ [ 0, \%pregion for entry.pregion for init.i ], [ ]}
                                      ... \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \( \) \(
                                      %add1.i.i = add nuw nsw i64 % local id x.0, %mul.i.i, !llvm.access.group !12
                                      %conv.i = trunc i64 %add1.i.i to i32, !llvm.access.group!12
                                      %cmp5.i = icmp sqt i32 %sub4.i, %conv.i, !llvm.access.group !12
                                      %or.cond.i = and i1 %cmp.i, %cmp5.i, !llvm.access.group !12
                                      br i1 %or.cond.i, label %if.then.i, label %if.end.r exit.i,
                                      ...!llvm.access.group!12
                                                                                                                                  F
if.then.i:
%add.i = add i32 %mul.i, %conv.i, !llvm.access.group !12
%idxprom.i = sext i32 %add.i to i64, !llvm.access.group !12
%arrayidx.i = getelementptr inbounds float, float* %2, i64 %idxprom.i,
...!llvm.access.group!12
%10 = load float, float* %arrayidx.i, align 4, !tbaa !15, !llvm.access.group
...!12
%conv7.i = fpext float %10 to double, !llvm.access.group !12
%add10.i = add i32 %add.i, 1, !llvm.access.group !12
%idxprom11.i = sext i32 %add10.i to i64, !llvm.access.group !12
%arrayidx12.i = getelementptr inbounds float, float* %0, i64 %idxprom11.i,
...!llvm.access.group!12
%11 = load float, float* %arrayidx12.i, align 4, !tbaa !15,
...!llvm.access.group!12
%arrayidx16.i = getelementptr inbounds float, float* %0, i64 %idxprom.i,
...!llvm.access.group!12
%12 = load float, float* %arrayidx16.i, align 4, !tbaa !15,
...!llvm.access.group!12
%sub17.i = fsub float %11, %12, !llvm.access.group !12
%add20.i = add nsw i32 %mul19.i, %conv.i, !llvm.access.group !12
%idxprom21.i = sext i32 %add20.i to i64, !llvm.access.group !12
%arrayidx22.i = getelementptr inbounds float, float* %1, i64 %idxprom21.i,
 ..!llvm.access.group!12
%13 = load float, float* %arrayidx22.i, align 4, !tbaa !15,
...!llvm.access.group!12
%add23.i = fadd float %sub17.i, %13, !llvm.access.group !12
%arrayidx27.i = getelementptr inbounds float, float* %1, i64 %idxprom.i,
...!llvm.access.group!12
%14 = load float, float* %arrayidx27.i, align 4, !tbaa !15,
...!llvm.access.group!12
%sub28.i = fsub float %add23.i, %14, !llvm.access.group !12
%conv29.i = fpext float %sub28.i to double, !llvm.access.group !12
%15 = tail call double @llvm.fmuladd.f64(double %conv29.i, double
%conv31.i = fptrunc double %15 to float, !llvm.access.group !12
store float %conv31.i, float* %arrayidx.i, align 4, !tbaa !15,
...!llvm.access.group!12
br label %if.end.r exit.i, !llvm.access.group !12
                                                                         if.end.r exit.i:
                                                                          %16 = add nuw nsw i64 \% local id x.0, 1
                                                                         %exitcond.not = icmp eq \overline{164} %\overline{16}, \overline{32}
                                                                         br i1 %exitcond.not, label %pregion for end.i, label
                                                                          .. %pregion for entry.entry.i, !llvm.loop 19
                                                                            pregion for end.i:
                                                                             \%17 = add nuw nsw i64 % local id y.0, 1
                                                                             %exitcond1.not = icmp eq \overline{i}64 \% \overline{17}, 8
                                                                             br i1 %exitcond1.not, label %fdtd kernel3.exit, label
                                                                            ... %pregion for entry.pregion for init.i, !llvm.loop!22
                                                                                  fdtd kernel3.exit:
                                                                                   ret void
```

CFG for 'pocl kernel fdtd kernel3' function

%9: