

%i11 = sext i32 %6 to i64  
%i12 = sext i32 %6 to i64  
%i13 = icmp slt i64 %i12, 32  
%i14 = select i1 %i13, i64 %i12, i64 32  
%i15 = sext i32 %5 to i64  
%i16 = icmp slt i64 %i15, 8  
%i17 = select i1 %i16, i64 %i15, i64 8  
%mul1.i = shl i64 %8, 5  
%mul3.i.i = shl i64 %9, 3  
%idxprom.i = sext i32 %4 to i64  
%arrayidx.i = getelementptr inbounds float, float\* %0, i64 %idxprom.i  
%i18 = icmp ugt i64 %i14, 1  
%umax = select i1 %i18, i64 %i14, i64 1  
%i19 = icmp ugt i64 %i17, 1  
%umax4 = select i1 %i19, i64 %i17, i64 1  
%20 = add nsw i64 %umax, -1  
%21 = shl i64 %8, 5  
%22 = trunc i64 %9 to i32  
%23 = mul i32 %22, %6  
%24 = shl i32 %23, 3  
%25 = zext i32 %24 to i64  
%26 = add i64 %21, %25  
%27 = zext i32 %6 to i64  
%scevgep = getelementptr float, float\* %0, i64 %idxprom.i  
%scevgep8 = bitcast float\* %scevgep to i8\*  
%uglygep = getelementptr i8, i8\* %scevgep8, i64 1  
%28 = shl i64 %8, 5  
%29 = trunc i64 %9 to i32  
%30 = mul i32 %29, %6  
%31 = shl i32 %30, 3  
%32 = zext i32 %31 to i64  
%33 = add i64 %49, %32  
%34 = zext i32 %6 to i64  
%scevgep11 = getelementptr float, float\* %2, i64 %umax  
%35 = add nsw i64 %umax, -1  
%36 = shl i64 %8, 5  
%37 = trunc i64 %9 to i32  
%38 = mul i32 %37, %6  
%39 = shl i32 %38, 3  
%40 = zext i32 %39 to i64  
%41 = add i64 %36, %40  
%42 = zext i32 %6 to i64  
%43 = shl i32 %37, 3  
%44 = add i32 %43, -1  
%45 = mul i32 %44, %6  
%46 = zext i32 %45 to i64  
%47 = shl i64 %8, 5  
%48 = add i64 %47, %46  
%49 = shl i64 %8, 5  
%50 = trunc i64 %9 to i32  
%51 = mul i32 %50, %6  
%52 = shl i32 %51, 3  
%53 = zext i32 %52 to i64  
%54 = add i64 %49, %53  
%55 = zext i32 %6 to i64  
%scevgep37 = getelementptr float, float\* %2, i64 %umax  
%56 = shl i32 %50, 3  
%57 = add i32 %56, -1  
%58 = mul i32 %57, %6  
%59 = zext i32 %58 to i64  
%60 = shl i64 %8, 5  
%61 = add i64 %60, %59  
%scevgep42 = getelementptr float, float\* %3, i64 %umax  
%scevgep47 = getelementptr float, float\* %3, i64 %umax  
br label %pregon for entry.pregon for init.i

pregon for entry.pregon for init.i:  
%local\_id.y.0 = phi i64 [ 0, %i11 ], [ %i135, %pregon\_for\_end.i ]  
%62 = mul i64 %local\_id.y.0, %55  
%63 = add i64 %54, %62  
%sext = shl i64 %63, 32  
%64 = ashr exact i64 %sext, 32  
%scevgep35 = getelementptr float, float\* %2, i64 %64  
%scevgep38 = getelementptr float, float\* %scevgep37, i64 %64  
%65 = add i64 %61, %62  
%sext65 = shl i64 %65, 32  
%66 = ashr exact i64 %sext65, 32  
%scevgep40 = getelementptr float, float\* %3, i64 %66  
%scevgep43 = getelementptr float, float\* %scevgep42, i64 %66  
%67 = mul i64 %local\_id.y.0, %42  
%68 = add i64 %41, %67  
%69 = trunc i64 %68 to i32  
%70 = add i64 %48, %67  
%71 = trunc i64 %70 to i32  
%72 = mul i64 %local\_id.y.0, %34  
%73 = add i64 %33, %72  
%sext66 = shl i64 %73, 32  
%74 = ashr exact i64 %sext66, 32  
%scevgep9 = getelementptr float, float\* %2, i64 %74  
%scevgep910 = bitcast float\* %scevgep9 to i8\*  
%telementptr float, float\* %scevgep11, i64 %74  
%75 = mul i64 %local\_id.y.0, %27  
%76 = add i64 %26, %75  
%77 = trunc i64 %76 to i32  
%add6.i.i = add i64 %local\_id.y.0, %mul3.i.i  
%conv2.i = trunc i64 %add6.i.i to i32  
%mul1.i = mul nsw i32 %conv2.i, %6  
%cmp6.i = icmp eq i32 %conv2.i, 0  
%sub.i = add nsw i32 %conv2.i, -1  
%mul22.i = mul nsw i32 %sub.i, %6  
br i1 %cmp6.i, label %pregon\_for\_entry.entry.i.us.preheader, label  
... %pregon\_for\_entry.entry.i.preheader

T	F
---	---

pregon\_for\_entry.entry.i.us.preheader:  
%min.ifers.check = icmp ult i64 %umax, 32  
br i1 %min.ifers.check, label %pregon\_for\_entry.entry.i.us.preheader67,  
... label %vector.scevcheck

T	F
---	---

pregon\_for\_entry.entry.i.preheader:  
%min.ifers.check23 = icmp ult i64 %umax, 8  
br i1 %min.ifers.check23, label %pregon\_for\_entry.entry.i.preheader68,  
... label %vector.scevcheck31

T	F
---	---

vector.scevcheck:  
%i108 = trunc i64 %20 to i32  
%i109 = add i32 %77, %i108  
%i110 = icmp slt i32 %i109, %77  
%i111 = icmp ugt i64 %20, 4294967295  
%i112 = or i1 %i110, %i111  
br i1 %i112, label %pregon\_for\_entry.entry.i.us.preheader67, label  
... %vector.memcheck

T	F
---	---

vector.scevcheck31:  
%78 = trunc i64 %35 to i32  
%79 = add i32 %69, %78  
%80 = icmp slt i32 %79, %69  
%81 = icmp ugt i64 %35, 4294967295  
%82 = or i1 %80, %81  
%83 = trunc i64 %35 to i32  
%84 = add i32 %71, %83  
%85 = icmp slt i32 %84, %71  
%86 = icmp ugt i64 %35, 4294967295  
%87 = or i1 %85, %86  
%88 = or i1 %82, %87  
br i1 %88, label %pregon\_for\_entry.entry.i.preheader68, label  
... %vector.memcheck33

T	F
---	---

vector.memcheck:  
%bound0 = icmp ult float\* %arrayidx.i, %scevgep12  
%bound1 = icmp ugt i8\* %uglygep, %scevgep910  
%found.conflict = and i1 %bound0, %bound1  
br i1 %found.conflict, label %pregon\_for\_entry.entry.i.us.preheader67,  
... label %vector.ph

T	F
---	---

vector.memcheck33:  
%bound050 = icmp ult float\* %scevgep35, %scevgep43  
%bound151 = icmp ult float\* %scevgep40, %scevgep38  
%found.conflict52 = and i1 %bound050, %bound151  
%bound053 = icmp ugt float\* %scevgep47, %2  
%bound154 = icmp ugt float\* %scevgep37, %3  
%found.conflict55 = and i1 %bound053, %bound154  
%conflict.rdx = or i1 %found.conflict52, %found.conflict55  
br i1 %conflict.rdx, label %pregon\_for\_entry.entry.i.preheader68, label  
... %vector.ph34

T	F
---	---

vector.ph:  
%n.vec = and i64 %umax, -32  
br label %vector.body

vector.ph34:  
%n.vec58 = and i64 %umax, -8  
br label %vector.body22

vector.body:  
%index = phi i64 [ 0, %vector.ph ], [ %index.next, %vector.body ]  
%i113 = add i64 %index, %mul.i.i  
%i114 = trunc i64 %i113 to i32  
%i115 = load float, float\* %arrayidx.i, align 4, !tbaa !12, !alias.scope !30,  
... !noalias !33  
%broadcast.splatinsert = insertelement <8 x float> undef, float %i115, i32 0  
%broadcast.splat = shufflevector <8 x float> %broadcast.splatinsert, <8 x  
... float> undef, <8 x i32> zeroinitializer  
%broadcast.splatinsert14 = insertelement <8 x float> undef, float %i115, i32 0  
%broadcast.splat15 = shufflevector <8 x float> %broadcast.splatinsert14, <8  
... x float> undef, <8 x i32> zeroinitializer  
%broadcast.splatinsert16 = insertelement <8 x float> undef, float %i115, i32 0  
%broadcast.splat17 = shufflevector <8 x float> %broadcast.splatinsert16, <8  
... x float> undef, <8 x i32> zeroinitializer  
%i116 = load float, float\* %arrayidx.i, align 4, !tbaa !12, !alias.scope !30,  
... !noalias !33  
%broadcast.splatinsert18 = insertelement <8 x float> undef, float %i116, i32 0  
%broadcast.splat19 = shufflevector <8 x float> %broadcast.splatinsert18, <8  
... x float> undef, <8 x i32> zeroinitializer  
%i117 = add nuw nsw i32 %mul.i, %i114  
%i118 = sext i32 %i117 to i64  
%i119 = getelementptr inbounds float, float\* %2, i64 %i118  
%i120 = bitcast float\* %i119 to <8 x float>\*  
store <8 x float> %broadcast.splat, <8 x float>\* %i120, align 4, !tbaa !12,  
... !alias.scope !33, !llvm.access.group !24  
%i121 = getelementptr inbounds float, float\* %i119, i64 8  
%i122 = bitcast float\* %i121 to <8 x float>\*  
store <8 x float> %broadcast.splat15, <8 x float>\* %i122, align 4, !tbaa !12,  
... !alias.scope !33, !llvm.access.group !24  
%i123 = getelementptr inbounds float, float\* %i119, i64 16  
%i124 = bitcast float\* %i123 to <8 x float>\*  
store <8 x float> %broadcast.splat17, <8 x float>\* %i124, align 4, !tbaa !12,  
... !alias.scope !33, !llvm.access.group !24  
%i125 = getelementptr inbounds float, float\* %i119, i64 24  
%i126 = bitcast float\* %i125 to <8 x float>\*  
store <8 x float> %broadcast.splat19, <8 x float>\* %i126, align 4, !tbaa !12,  
... !alias.scope !33, !llvm.access.group !24  
%index.next = add i64 %index, 32  
%i127 = icmp eq i64 %index.next, %n.vec  
br i1 %i127, label %middle.block, label %vector.body, !llvm.loop !35

T	F
---	---

vector.body22:  
%index59 = phi i64 [ 0, %vector.ph34 ], [ %index.next60, %vector.body22 ]  
%89 = add i64 %index59, %mul.i.i  
%90 = trunc i64 %89 to i32  
%91 = add nsw i32 %mul.i, %90  
%92 = sext i32 %91 to i64  
%93 = getelementptr inbounds float, float\* %2, i64 %92  
%94 = bitcast float\* %93 to <8 x float>\*  
%wide.load = load <8 x float>, <8 x float>\* %94, align 4, !tbaa !12,  
... !alias.scope !16, !noalias !19  
%95 = fpxext <8 x float> %wide.load to <8 x double>  
%96 = getelementptr inbounds float, float\* %3, i64 %92  
%97 = bitcast float\* %96 to <8 x float>\*  
%wide.load63 = load <8 x float>, <8 x float>\* %97, align 4, !tbaa !12,  
... !alias.scope !22  
%98 = add nsw i32 %mul22.i, %90  
%99 = sext i32 %98 to i64  
%i100 = getelementptr inbounds float, float\* %3, i64 %99  
%i101 = bitcast float\* %i100 to <8 x float>\*  
%wide.load64 = load <8 x float>, <8 x float>\* %i101, align 4, !tbaa !12,  
... !alias.scope !23  
%i102 = fsub <8 x float> %wide.load63, %wide.load64  
%i103 = fpxext <8 x float> %i102 to <8 x double>  
%i104 = call <8 x double> @llvm.fmuladd.v8f64(<8 x double> %i103, <8 x double>  
... <double -5.000000e-01, double -5.000000e-01, double -5.000000e-01, double  
... -5.000000e-01, double -5.000000e-01, double -5.000000e-01, double  
... -5.000000e-01, double -5.000000e-01>, <8 x double> %95)  
%i105 = fptrunc <8 x double> %i104 to <8 x float>  
%i106 = bitcast float\* %93 to <8 x float>\*  
store <8 x float> %i105, <8 x float>\* %i106, align 4, !tbaa !12, !alias.scope  
... !16, !noalias !19, !llvm.access.group !24  
%index.next60 = add i64 %index59, 8  
%i107 = icmp eq i64 %index.next60, %n.vec58  
br i1 %i107, label %middle.block20, label %vector.body22, !llvm.loop !27

T	F
---	---

middle.block:  
%cmp.n = icmp eq i64 %umax, %n.vec  
br i1 %cmp.n, label %pregon\_for\_end.i.us.preheader67,  
... %pregon\_for\_entry.entry.i.us.preheader67

T	F
---	---

middle.block20:  
%cmp.n62 = icmp eq i64 %umax, %n.vec58  
br i1 %cmp.n62, label %pregon\_for\_end.i.us.preheader68,  
... %pregon\_for\_entry.entry.i.us.preheader68

T	F
---	---

pregon\_for\_entry.entry.i.us.preheader67:  
%local\_id.x.0.us.ph = phi i64 [ 0, %vector.memcheck ], [ 0,  
... %vector.scevcheck ], [ 0, %pregon\_for\_entry.entry.i.us.preheader ], [  
... %n.vec, %middle.block ]  
br label %pregon\_for\_entry.entry.i.us

pregon\_for\_entry.entry.i.preheader68:  
%local\_id.x.0.ph = phi i64 [ 0, %vector.memcheck33 ], [ 0,  
... %vector.scevcheck31 ], [ 0, %pregon\_for\_entry.entry.i.preheader ], [  
... %n.vec58, %middle.block20 ]  
br label %pregon\_for\_entry.entry.i

pregon\_for\_entry.entry.i.us:  
%local\_id.x.0.us = phi i64 [ %i129, %pregon\_for\_entry.entry.i.us ], [  
... %local\_id.x.0.us.ph, %pregon\_for\_entry.entry.i.us.preheader67 ]  
%add1.i.i.us = add i64 %local\_id.x.0.us, %mul.i.i  
%conv.i.us = trunc i64 %add1.i.i.us to i32  
%i128 = load float, float\* %arrayidx.i, align 4, !tbaa !12  
%add10.i.us = add nuw nsw i32 %mul.i, %conv.i.us  
%idxprom11.i.us = sext i32 %add10.i.us to i64  
%arrayidx12.i.us = getelementptr inbounds float, float\* %2, i64  
... %idxprom11.i.us  
store float %i128, float\* %arrayidx12.i.us, align 4, !tbaa !12,  
... !llvm.access.group !24  
%i129 = add nuw i64 %local\_id.x.0.us, 1  
%exitcond3.not = icmp eq i64 %i129, %umax  
br i1 %exitcond3.not, label %pregon\_for\_end.i.us.loopexit, label  
... %pregon\_for\_entry.entry.i.us, !llvm.loop !36

T	F
---	---

pregon\_for\_entry.entry.i:  
%local\_id.x.0 = phi i64 [ %i134, %pregon\_for\_entry.entry.i ], [  
... %local\_id.x.0.ph, %pregon\_for\_entry.entry.i.preheader68 ]  
%add1.i.i = add i64 %local\_id.x.0, %mul.i.i  
%conv1 = trunc i64 %add1.i.i to i32  
%add14.i = add nsw i32 %mul.i, %conv1  
%idxprom15.i = sext i32 %add14.i to i64  
%arrayidx16.i = getelementptr inbounds float, float\* %2, i64 %idxprom15.i  
%i130 = load float, float\* %arrayidx16.i, align 4, !tbaa !12  
%conv17.i = fpxext float %i130 to double  
%arrayidx21.i = getelementptr inbounds float, float\* %3, i64 %idxprom15.i  
%i131 = load float, float\* %arrayidx21.i, align 4, !tbaa !12  
%add23.i = add nsw i32 %mul22.i, %conv1  
%idxprom24.i = sext i32 %add23.i to i64  
%arrayidx25.i = getelementptr inbounds float, float\* %3, i64 %idxprom24.i  
%i132 = load float, float\* %arrayidx25.i, align 4, !tbaa !12  
%sub26.i = fsub float %i131, %i132  
%conv27.i = fpxext float %sub26.i to double  
%i133 = tail call double @llvm.fmuladd.f64(double %conv27.i, double  
... -5.000000e-01, double %conv17.i) #3  
%conv29.i = fptrunc double %i133 to float  
store float %conv29.i, float\* %arrayidx16.i, align 4, !tbaa !12,  
... !llvm.access.group !24  
%i134 = add nuw i64 %local\_id.x.0, 1  
%exitcond.not = icmp eq i64 %i134, %umax  
br i1 %exitcond.not, label %pregon\_for\_end.i.us.loopexit70, label  
... %pregon\_for\_entry.entry.i, !llvm.loop !37

T	F
---	---

pregon\_for\_end.i.us.loopexit:  
br label %pregon\_for\_end.i

pregon\_for\_end.i.us.loopexit70:  
br label %pregon\_for\_end.i

pregon\_for\_end.i:  
%i135 = add nuw i64 %local\_id.y.0, 1  
%exitcond5.not = icmp eq i64 %i135, %umax4  
br i1 %exitcond5.not, label %ftdd\_kernel1.exit, label  
... %pregon\_for\_entry.pregon\_for\_init.i, !llvm.loop !38

T	F
---	---

ftdd\_kernel1.exit:  
ret void

CFG for '\_pool\_kernel\_ftdd\_kernel1' function