

# ループ最適化により生成される冗長な prefetch命令の削除

今田俊寛

# バックグラウンド

---

- ▶ **Software Prefetch命令**

- ▶ 命令スロットを消費するのが一般的。

- ▶ ここで、Prefetch命令を含むループを、アンロール or ソフトウェアパイプライン化する事を考える。

- ▶ すると、Prefetch命令は冗長な先読みを行うケースが多くなる。

- ▶ どうすれば、冗長な先読みを減らせるか？



# 前提

---

- ▶ MVE時にprefetch命令を挿入する方法などあるが、ここでは単純なソフトウェアパイプライン以前に、連続アクセスへのPrefetch命令を出力した場合の最適化を考える。



# Prefetch命令

---

- ▶ 連続アクセスなので、一つのPrefetch命令は、
  - ▶ 読み先アドレス～キャッシュライン幅分のデータを読み込む。
- ▶ MVEを行った場合
  - ▶ 一つのPrefetchがカバーする先読み対象とサイズは既知
  - ▶ Prefetch命令の性質から自明
- ▶ とすると、次のスライドの式が得られる。



# 式

---

- ▶  $\{\text{キャッシュライン幅(Byte)}\} \div \{\text{一つのPrefetchがカバーするメモリアクセス幅(Byte)}\} \div \text{<切り捨て>} \{\text{アンロール数}\} + 1$
- ▶ MVEで複製されたPrefetch数は最適では無く、上記の数に間引いたPrefetch数が最適な出力数である。

# 考慮すべき点

---

- ▶ この方式は最適ではない。
  - ▶ 何故か？
- ▶ Prefetchを間引くので、MRT的には詰まっている命令列に空きが出来る事になる。
- ▶ これは、非常に単純な実装(だが、prefetch命令がLSUリソースを使う場合は一定の効果が得られる)。



# 結論

---

- ▶ 古典的な実装でのSoftware Prefetch出力とループ最適化に対して、性能低下をなるべく回避する方法を提案した。



---

終わり

