

KL1言語系

KL1-u (user)

- ・モジュール化機能

KL1-c (core)

- ・Flat GHC
- ・Metacall

KL1-p (pragma)

- ・プロセッサ割付け
- ・優先順位付け

KL1-b (base)

- ・抽象機械の命令
- ・ハードウェア操作等の組込述語

KL1-b

設計思想

- (1) 当面Multi-PSIを強く意識
 - ・処理単位をKL1-cのゴール・レベルに設定
 - ・PE内の逐次性を利用した最適化
- (2) 実用性
 - ・デバッグ機能
 - ・高い信頼性
- (3) システム・プログラムの記述を考慮
 - ・ハードウェア操作組込述語

PSI上の逐次処理系

1. 目的

- Multi-PSIにおける機械語(KL1-b)の設計及び部分試作
- モジュール化機能(KL1-u)の検討及び試作
- デバッグ機能の検討及び試作

2. 現状

- KL1-u/cのKL1-bへのコンパイラ及びKL1-bのエミュレータをESPで記述 → 約0.9kRPS
- Multi-PSI用の処理系に拡張中

汎用計算機上の逐次処理系

目的

- (1) 汎用機上に真に実用的な処理系を実現する。
- (2) それによってGHCの普及をはかる。
- (3) 逐次計算機むきのコンパイル・最適化技法を研究・開発する。
そのを並列計算機上の実現にも役立てる。

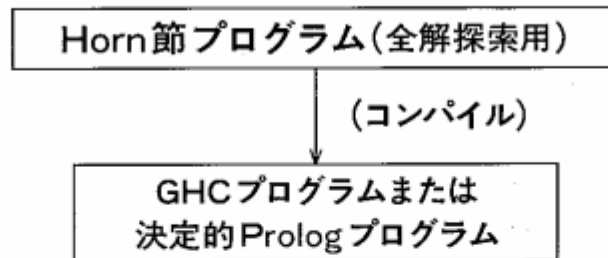
現況

- 目的コードの設計, 性能予測

予測性能: “append”: $32 \times \alpha$ kRPS
クイックソート: $19 \times \alpha$ kRPS
素数生成: $15 \times \alpha$ kRPS

(α : 閉じたサブルーチン化による速度低下)

GHCによる全解探索処理



意 義

- (1) 並列探索に一般に必要な多重環境を、コンパイルによって消去する。
このときextralogicalな機能を用いない。
- (2) 問題によっては、多重環境を用いた探索より効率が向上する。
リストの分解で30倍，順列生成で6倍(Prolog処理系での比較)
- (3) 探索の制御が可能になる。

GHCプログラムの アルゴリズムック・デバッグ

プログラムの動作とは独立な抽象的性質だけを用いる。
バグの存在する範囲を機械的に絞り込む。

- GHCプログラムの抽象的意味の導入
- 対処可能なエラー
 - 間違った答を返して停止する
 - デッドロック
- Divide & Query戦略
 - Query総数 $\sim O(\log_2 N)$ N : 手続き呼出し総数
- GHCで実現されている。