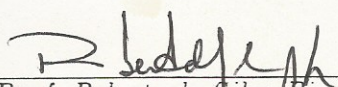


FOLHA DE APROVAÇÃO

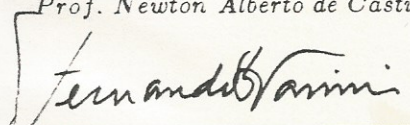
GERAÇÃO DE CÓDIGO
EM UM COMPILADOR CHILL

CARLOS CAMARÃO DE FIGUEIREDO

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:


Prof. Roberto da Silva Bigonha

Prof. Newton Alberto de Castilho Lages


Prof. Fernando Antônio Vanini

Belo Horizonte, 01 de julho de 1988.

GERAÇÃO DE CÓDIGO EM UM COMPILADOR CHILL

Carlos Camarão de Figueiredo

Dissertação apresentada ao Instituto
de Ciências Exatas da Universidade
Federal de Minas Gerais, como
requisito parcial para a obtenção do
grau de Mestre em Ciência da
Computação.

Universidade Federal de Minas Gerais
3 de agosto de 1988

Agradecimentos

Agradeço a Roberto de Silva Higashida e Fernando Antônio Varizal pelo apoio, pela orientação e pela amizade.

Agradeço a Antônio Vitor de Andrade Farias Neto, que implementou junto comigo a geração de código, e a todos os membros do Projeto CHILL, que ajudaram e tornaram possível de cumprir um trabalho tão valioso para diversas pesquisas no CPqD-Feltrina.

Agradeço aos meus pais, pelo apoio e incentivo.

Agradeço, ainda, a todos que de alguma forma contribuíram para a realização deste trabalho.

À Dani

Agradecimentos

Agradeço a Roberto da Silva Bigonha e Fernando Antônio Vanini pelo apoio, pela orientação e pela amizade.

Agradeço a Antônio Vieira de Andrade Palma Neto, que implementou junto comigo o gerador de código, e a todos os membros do Projeto CHILL, que ajudaram a tornar o gerador de código um trabalho útil e valioso para diversos projetos no CPqD-Telebrás.

Agradeço aos meus pais, pelo apoio e incentivo.

Agradeço, enfim, a todos que de alguma forma contribuíram para a realização desse trabalho.

Sumário

Este trabalho apresenta um gerador de código para os micro-processadores iAPX 80286 e 8086/88, que implementa as construções e ações da linguagem CHILL [1], destacando aquelas não existentes em outras linguagens como, por exemplo, PASCAL[29] e MODULA-2[28].

A geração de código é feita a partir de uma árvore sintática *abstrata* do programa fonte.

Abstract

This work presents a code generator for the iAPX 80286 and 8086/88 micro-processors that implements the constructions and actions of the CHILL language[1], emphasizing those not found in other languages such as, for example, PASCAL[29] and MODULA-2[28].

The code generation is made from an *abstract* syntax tree of the source program.

CONTÉUDO DA DISSERTAÇÃO	2
1 INTRODUÇÃO	4
1.1 Estrutura geral da linguagem de descrição de código	4
1.1.1 Estrutura geral da descrição de código	4
1.1.2 Estrutura da descrição de código	10
1.1.3 Modificadores e atributos de parâmetros de código	11
1.2 Tabela de símbolos	12
1.3 Ações de programação	13
1.3.1 Representação linear	14
1.3.2 Representação bidimensional	15
1.4 Descrição de Operações de Máquina Alvo	16
2 Estrutura Geral da Programação CHILL na Máquina Alvo	18
2.1 Introdução	18
2.1.1 Estrutura geral do programa de geração de código	18
2.2 Processos	19
2.3 Regiões	19
2.4 Procedimentos	20
2.5 Computação e manipulação de dados	21
2.6 Limitações de Tabela de símbolos na máquina alvo	22
2.7 Alocação dinâmica de memória	23
2.8 Tratamento de Erros	24
2.8.1 Descrição de Erros	24
2.8.2 Comentários e Erros	25
2.8.3 Implementação	26
3 Representação de Dados	28
3.1 Tipos simples	28
3.1.1 Tipos básicos	28
3.1.2 Tipo <i>integer</i>	29
3.1.3 Tipos relacionados a <i>float</i> e <i>double</i> distintos	30

Conteúdo

1	CONTEÚDO DA DISSERTAÇÃO	2
2	INTRODUÇÃO	6
2.1	Estrutura geral e Estratégia de Geração de Código	8
2.1.1	Estrutura geral do Gerador de Código	8
2.1.2	Estratégia de Geração de Código	10
2.1.3	Mais sobre a estrutura do gerador de código	11
2.2	Tabela de Símbolos	13
2.3	Árvore de Programa	14
2.3.1	Representação Linear	14
2.3.2	Representação Bidimensional	15
2.4	Descritor de Objetos na Máquina Alvo	16
3	Estrutura Geral de Programas CHILL na máquina objeto	18
3.1	Introdução	18
3.1.1	Uma idéia geral do que contém cada segmento	18
3.2	Processos	19
3.3	Regiões	22
3.4	Procedimentos	22
3.5	Comunicação e sincronização entre processos	23
3.6	Limitações de Tamanhos de segmentos na máquina alvo	28
3.7	Alocação dinâmica de memória	30
3.8	Tratamento de Exceções	30
3.8.1	Descrição do Mecanismo	30
3.8.2	Comentários e discussões	33
3.8.3	Implementação	36
4	Representação de Dados	44
4.1	Tipos simples	44
4.1.1	Tipos discretos	45
4.1.2	Tipo conjunto	45
4.1.3	Tipos referência estática e referência dinâmica	45

4.1.4	Tipo Procedimento	46
4.2	Tipos Compostos	46
4.2.1	Estruturas	47
4.2.2	Vetores e Cadeias	52
4.3	Alocação de Valores	57
4.3.1	Repetições de valores em vetores e cadeias	59
4.4	Tuplas com elementos não constantes	65
4.4.1	Exemplos	67
5	Ações	74
5.1	Atribuição	74
5.1.1	Atribuição múltipla	74
5.1.2	Operadores diádicos fechados	75
5.1.3	Objetos com partes sobrepostas	78
5.2	CASE	80
5.2.1	Representação Intermediária	80
5.2.2	Algoritmo de geração de código	83
5.2.3	Exemplos	84
5.2.4	Conclusões	91
5.3	FOR	93
5.3.1	Representação Intermediária	97
5.3.2	Algoritmo de Geração de Código	97
5.3.3	Enumeração em uma faixa	98
5.3.4	Enumeração passo a passo	98
5.3.5	Enumeração em um conjunto	99
5.3.6	Enumeração de variável	99
5.4	WITH	101
5.4.1	Representação Intermediária	103
5.4.2	Algoritmo de Geração de Código	103
5.5	IF	105
5.5.1	Representação Intermediária	105
5.5.2	Algoritmo de Geração de Código	105
5.6	WHILE	107
5.6.1	Representação Intermediária	107
5.6.2	Algoritmo de Geração de Código	107
5.7	RESULT	107
5.8	RETURN	108
5.9	EXIT	108
5.9.1	Representação Intermediária	108
5.9.2	Algoritmo de Geração de Código	108
5.10	CAUSE	109
5.10.1	Representação Intermediária	109
5.10.2	Algoritmo de Geração de Código	109
5.11	ASSERT	110

5.11.1	Representação intermediária	110
5.11.2	Algoritmo de Geração de Código	110
5.12	START, STOP, DELAY, CONTINUE, SEND e RECEIVE CASE	110
6	Avaliação de Expressões	113
6.1	Alocação de Temporários	114
6.2	Operações com Inteiros	115
6.3	Operações Relacionais	115
6.4	Operações Lógicas com Booleanos	116
6.5	Operações Lógicas com Conjuntos e Cadeias de Bits	120
6.6	Operações Relacionais entre Conjuntos	120
6.7	Operação de Pertinência a um Conjunto	120
6.8	Operação de Referenciação	123
6.9	Operação de Derreferenciação	123
6.10	Concatenação de Cadeias de Caracteres	123
7	Implementação de Procedimentos	126
8	Extensões e Facilidades Definidas pela Implementação	133
8.1	Criação de Bibliotecas de Sistema	133
8.2	Tratamento de Interrupções	134
8.3	Acesso ao hardware da máquina alvo	135
8.4	Supervisão de Tempo	135
8.5	Suporte a Multiprocessamento	136
9	Conclusões	137
A	Apresentação da linguagem CHILL	139
A.1	Introdução	139
A.2	Objetos de Dados	139
A.2.1	Tipos existentes em CHILL	140
A.2.2	Definição de Tipos e Conceito de Classe	148
A.2.3	Definição de Sinônimos	150
A.2.4	Declaração de Variáveis	150
A.3	Ações	152
A.3.1	Atribuição	152
A.3.2	IF	152
A.3.3	CASE	153
A.3.4	DO	156
A.4	Estrutura de Programas	161
A.4.1	Procedimentos	161
A.4.2	Módulos e Regiões	163
A.5	Programação Concorrente	165
A.5.1	Processos	165
A.5.2	Regiões e Eventos	167

A.5.3	"Buffers"	172
A.5.4	Sinais	174
A.5.5	SEND	175
A.5.6	RECEIVE SIGNAL CASE	175
A.6	Tratamento de Exceções	178
A.6.1	Como encontrar o tratador de uma exceção	180
B	Terminologia CHILL	182
C	Apresentação do Microprocessador iAPX 286	184
C.1	Modos de Operação	184
C.2	Registradores	185
C.3	Modos de Endereçamento	186
C.4	Conjunto de Instruções	187
C.4.1	Conjunto de Instruções Básicas	188
C.4.2	Conjunto de Instruções Adicionais	191
C.4.3	Conjunto de Instruções de Controle de Sistema	191
D	"Benchmark"	192
D.1	O quebra-cabeças das cestas da floresta	193