

A. Appendix for as6811 Frankenstein Assembler

A.1 Pseudo Operations

A.1.1 *Standard_Pseudo_Operation_Mnemonics*

End	END
File Inclusion	INCL INCLUDE
If	IF
Else	ELSE
End If	ENDI
Equate	EQU
Set	SET
Org	ORG
Reserve Memory	RESERVE RMB
Define Byte Data	BYTE DB FCB
Define Word Data	DW FDB WORD
Define String Data	FCC STRING
Define Character Set Translation	CHARSET
Define Character Value	CHARDEF CHD
Use Character Translation	CHARUSE

A.1.2 *Machine_Dependent_Pseudo_Operations*

A.1.2.1 *Instruction_Set_Selection*

CPU string

The instruction set can be specified in the source file with the CPU pseudooperation. The string, delimited by quotes or apostrophes, is scanned for a substring which selects which instruction set is used. When the program is invoked, this operation is performed on the name of the program, then the -p optional argument, if any, and then any CPU statements. The last one selects which subset of the instructions the assembler will accept. The instruction set can be changed at any place in the source file.

Instruction Set	Substrings
68hc11	11
6801	01 03
6800	00 02 08

A.2 Instructions

A.2.1 *Instruction_List*

Opcode	Syntax	Selection Criteria
--------	--------	--------------------

ABA

Opcode	Syntax	Selection Criteria
ABX		TS6801PLUS
ABY		TS6811
ADC	ACCUM '#' expr	REGA
ADC	ACCUM '#' expr	REGB
ADC	ACCUM expr	REGA+DIRECT
ADC	ACCUM expr	REGA+EXTENDED
ADC	ACCUM expr	REGB+DIRECT
ADC	ACCUM expr	REGB+EXTENDED
ADC	ACCUM indexed	REGA+REGX
ADC	ACCUM indexed	REGA+REGY+TS6811
ADC	ACCUM indexed	REGB+REGX
ADC	ACCUM indexed	REGB+REGY+TS6811
ADCA	'#' expr	
ADCA	expr	DIRECT
ADCA	expr	EXTENDED
ADCA	indexed	REGX
ADCA	indexed	REGY+TS6811
ADCB	'#' expr	
ADCB	expr	DIRECT
ADCB	expr	EXTENDED
ADCB	indexed	REGX
ADCB	indexed	REGY+TS6811
ADD	ACCUM '#' expr	REGA
ADD	ACCUM '#' expr	REGB
ADD	ACCUM expr	REGA+DIRECT
ADD	ACCUM expr	REGA+EXTENDED
ADD	ACCUM expr	REGB+DIRECT
ADD	ACCUM expr	REGB+EXTENDED
ADD	ACCUM indexed	REGA+REGX
ADD	ACCUM indexed	REGA+REGY+TS6811
ADD	ACCUM indexed	REGB+REGX
ADD	ACCUM indexed	REGB+REGY+TS6811
ADDA	'#' expr	
ADDA	expr	DIRECT
ADDA	expr	EXTENDED
ADDA	indexed	REGX
ADDA	indexed	REGY+TS6811
ADDB	'#' expr	
ADDB	expr	DIRECT
ADDB	expr	EXTENDED
ADDB	indexed	REGX

Opcode	Syntax	Selection Criteria
ADDB	indexed	REGY+TS6811
ADDD	'#' expr	0+TS6801PLUS
ADDD	expr	DIRECT+TS6801PLUS
ADDD	expr	EXTENDED+TS6801PLUS
ADDD	indexed	REGX+TS6801PLUS
ADDD	indexed	REGY+TS6811
AND	ACCUM '#' expr	REGA
AND	ACCUM '#' expr	REGB
AND	ACCUM expr	REGA+DIRECT
AND	ACCUM expr	REGA+EXTENDED
AND	ACCUM expr	REGB+DIRECT
AND	ACCUM expr	REGB+EXTENDED
AND	ACCUM indexed	REGA+REGX
AND	ACCUM indexed	REGA+REGY+TS6811
AND	ACCUM indexed	REGB+REGX
AND	ACCUM indexed	REGB+REGY+TS6811
ANDA	'#' expr	
ANDA	expr	DIRECT
ANDA	expr	EXTENDED
ANDA	indexed	REGX
ANDA	indexed	REGY+TS6811
ANDB	'#' expr	
ANDB	expr	DIRECT
ANDB	expr	EXTENDED
ANDB	indexed	REGX
ANDB	indexed	REGY+TS6811
ASL	ACCUM	REGA
ASL	ACCUM	REGB
ASL	expr	
ASL	indexed	REGX
ASL	indexed	REGY+TS6811
ASLA		
ASLB		
ASLD		TS6801PLUS
ASR	ACCUM	REGA
ASR	ACCUM	REGB
ASR	expr	
ASR	indexed	REGX
ASR	indexed	REGY+TS6811

Opcode	Syntax	Selection Criteria
ASRA		
ASRB		
BCC	expr	
BCLR	expr ',' expr	TS6811
BCLR	indexed ',' expr	REGX+TS6811
BCLR	indexed ',' expr	REGY+TS6811
BCS	expr	
BEQ	expr	
BGE	expr	
BGT	expr	
BHI	expr	
BHS	expr	
BIT	ACCUM '#' expr	REGA
BIT	ACCUM '#' expr	REGB
BIT	ACCUM expr	REGA+DIRECT
BIT	ACCUM expr	REGA+EXTENDED
BIT	ACCUM expr	REGB+DIRECT
BIT	ACCUM expr	REGB+EXTENDED
BIT	ACCUM indexed	REGA+REGX
BIT	ACCUM indexed	REGA+REGY+TS6811
BIT	ACCUM indexed	REGB+REGX
BIT	ACCUM indexed	REGB+REGY+TS6811
BITA	'#' expr	
BITA	expr	DIRECT
BITA	expr	EXTENDED
BITA	indexed	REGX
BITA	indexed	REGY+TS6811
BITB	'#' expr	
BITB	expr	DIRECT
BITB	expr	EXTENDED
BITB	indexed	REGX
BITB	indexed	REGY+TS6811
BLE	expr	
BLO	expr	

Opcode	Syntax	Selection Criteria
BLS	expr	
BLT	expr	
BMI	expr	
BNE	expr	
BPL	expr	
BRA	expr	
BRCLR	expr ',' expr ',' expr	TS6811
BRCLR	indexed ',' expr ',' expr	REGX+TS6811
BRCLR	indexed ',' expr ',' expr	REGY+TS6811
BRN	expr	TS6801PLUS
BRSET	expr ',' expr ',' expr	TS6811
BRSET	indexed ',' expr ',' expr	REGX+TS6811
BRSET	indexed ',' expr ',' expr	REGY+TS6811
BSET	expr ',' expr	TS6811
BSET	indexed ',' expr	REGX+TS6811
BSET	indexed ',' expr	REGY+TS6811
BSR	expr	
BVC	expr	
BVS	expr	
CBA		
CLC		
CLI		
CLR	ACCUM	REGA
CLR	ACCUM	REGB
CLR	expr	
CLR	indexed	REGX
CLR	indexed	REGY+TS6811
CLRA		
CLRB		
CLV		

Opcode	Syntax	Selection Criteria
CMP	ACCUM '#' expr	REGA
CMP	ACCUM '#' expr	REGB
CMP	ACCUM expr	REGA+DIRECT
CMP	ACCUM expr	REGA+EXTENDED
CMP	ACCUM expr	REGB+DIRECT
CMP	ACCUM expr	REGB+EXTENDED
CMP	ACCUM indexed	REGA+REGX
CMP	ACCUM indexed	REGA+REGY+TS6811
CMP	ACCUM indexed	REGB+REGX
CMP	ACCUM indexed	REGB+REGY+TS6811
COMPA	'#' expr	
COMPA	expr	DIRECT
COMPA	expr	EXTENDED
COMPA	indexed	REGX
COMPA	indexed	REGY+TS6811
CMPB	'#' expr	
CMPB	expr	DIRECT
CMPB	expr	EXTENDED
CMPB	indexed	REGX
CMPB	indexed	REGY+TS6811
COM	ACCUM	REGA
COM	ACCUM	REGB
COM	expr	
COM	indexed	REGX
COM	indexed	REGY+TS6811
COMA		
COMB		
CPD	'#' expr	TS6811
CPD	expr	DIRECT+TS6811
CPD	expr	EXTENDED+TS6811
CPD	indexed	REGX+TS6811
CPD	indexed	REGY+TS6811
CPX	'#' expr	
CPX	expr	DIRECT
CPX	expr	EXTENDED
CPX	indexed	REGX
CPX	indexed	REGY+TS6811
CPY	'#' expr	TS6811
CPY	expr	DIRECT+TS6811
CPY	expr	EXTENDED+TS6811

Opcode	Syntax	Selection Criteria
CPY	indexed	REGX+TS6811
CPY	indexed	REGY+TS6811
DAA		
DEC	ACCUM	REGA
DEC	ACCUM	REGB
DEC	expr	
DEC	indexed	REGX
DEC	indexed	REGY+TS6811
DECA		
DECB		
DES		
DEX		
DEY		TS6811
EOR	ACCUM '#' expr	REGA
EOR	ACCUM '#' expr	REGB
EOR	ACCUM expr	REGA+DIRECT
EOR	ACCUM expr	REGA+EXTENDED
EOR	ACCUM expr	REGB+DIRECT
EOR	ACCUM expr	REGB+EXTENDED
EOR	ACCUM indexed	REGA+REGX
EOR	ACCUM indexed	REGA+REGY+TS6811
EOR	ACCUM indexed	REGB+REGX
EOR	ACCUM indexed	REGB+REGY+TS6811
EORA	'#' expr	
EORA	expr	DIRECT
EORA	expr	EXTENDED
EORA	indexed	REGX
EORA	indexed	REGY+TS6811
EORB	'#' expr	
EORB	expr	DIRECT
EORB	expr	EXTENDED
EORB	indexed	REGX
EORB	indexed	REGY+TS6811
FDIV		TS6811
IDIV		TS6811
INC	ACCUM	REGA

Opcode	Syntax	Selection Criteria
INC	ACCUM	REGB
INC	expr	
INC	indexed	REGX
INC	indexed	REGY+TS6811
INCA		
INCB		
INS		
INX		
INY		TS6811
JMP	expr	
JMP	indexed	REGX
JMP	indexed	REGY+TS6811
JSR	expr	DIRECT+CPU6800
JSR	expr	DIRECT+TS6801PLUS
JSR	expr	EXTENDED
JSR	indexed	REGX
JSR	indexed	REGY+TS6811
LDA	ACCUM '#' expr	REGA
LDA	ACCUM '#' expr	REGB
LDA	ACCUM expr	REGA+DIRECT
LDA	ACCUM expr	REGA+EXTENDED
LDA	ACCUM expr	REGB+DIRECT
LDA	ACCUM expr	REGB+EXTENDED
LDA	ACCUM indexed	REGA+REGX
LDA	ACCUM indexed	REGA+REGY+TS6811
LDA	ACCUM indexed	REGB+REGX
LDA	ACCUM indexed	REGB+REGY+TS6811
LDAA	'#' expr	
LDAA	expr	DIRECT
LDAA	expr	EXTENDED
LDAA	indexed	REGX
LDAA	indexed	REGY+TS6811
LDAB	'#' expr	
LDAB	expr	DIRECT
LDAB	expr	EXTENDED
LDAB	indexed	REGX
LDAB	indexed	REGY+TS6811
LDD	'#' expr	TS6801PLUS

Opcode	Syntax	Selection Criteria
LDD	expr	DIRECT+TS6801PLUS
LDD	expr	EXTENDED+TS6801PLUS
LDD	indexed	REGX+TS6801PLUS
LDD	indexed	REGY+TS6811
LDS	'#' expr	
LDS	expr	DIRECT
LDS	expr	EXTENDED
LDS	indexed	REGX
LDS	indexed	REGY+TS6811
LDX	'#' expr	
LDX	expr	DIRECT
LDX	expr	EXTENDED
LDX	indexed	REGX
LDX	indexed	REGY+TS6811
LDY	'#' expr	TS6811
LDY	expr	DIRECT+TS6811
LDY	expr	EXTENDED+TS6811
LDY	indexed	REGX+TS6811
LDY	indexed	REGY+TS6811
LSL	ACCUM	REGA
LSL	ACCUM	REGB
LSL	expr	
LSL	indexed	REGX
LSL	indexed	REGY+TS6811
LSLA		
LSLB		
LSLD		TS6801PLUS
LSR	ACCUM	REGA
LSR	ACCUM	REGB
LSR	expr	
LSR	indexed	REGX
LSR	indexed	REGY+TS6811
LSRA		
LSRB		
LSRD		TS6801PLUS
MUL		TS6801PLUS

Opcode	Syntax	Selection Criteria
NEG	ACCUM	REGA
NEG	ACCUM	REGB
NEG	expr	
NEG	indexed	REGX
NEG	indexed	REGY+TS6811
NEGA		
NEGB		
NOP		
ORA	ACCUM '#' expr	REGA
ORA	ACCUM '#' expr	REGB
ORA	ACCUM expr	REGA+DIRECT
ORA	ACCUM expr	REGA+EXTENDED
ORA	ACCUM expr	REGB+DIRECT
ORA	ACCUM expr	REGB+EXTENDED
ORA	ACCUM indexed	REGA+REGX
ORA	ACCUM indexed	REGA+REGY+TS6811
ORA	ACCUM indexed	REGB+REGX
ORA	ACCUM indexed	REGB+REGY+TS6811
ORAA	'#' expr	
ORAA	expr	DIRECT
ORAA	expr	EXTENDED
ORAA	indexed	REGX
ORAA	indexed	REGY+TS6811
ORAB	'#' expr	
ORAB	expr	DIRECT
ORAB	expr	EXTENDED
ORAB	indexed	REGX
ORAB	indexed	REGY+TS6811
PSH	ACCUM	REGA
PSH	ACCUM	REGB
PSH	INDEX	REGX+TS6801PLUS
PSH	INDEX	REGY+TS6811
PSHA		
PSHB		
PSHX		TS6801PLUS
PSHY		TS6811
PUL	ACCUM	REGA

Opcode	Syntax	Selection Criteria
PUL	ACCUM	REGB
PUL	INDEX	REGX+TS6801PLUS
PUL	INDEX	REGY+TS6811
PULA		
PULB		
PULX		TS6801PLUS
PULY		TS6811
ROL	ACCUM	REGA
ROL	ACCUM	REGB
ROL	expr	
ROL	indexed	REGX
ROL	indexed	REGY+TS6811
ROLA		
ROLB		
ROR	ACCUM	REGA
ROR	ACCUM	REGB
ROR	expr	
ROR	indexed	REGX
ROR	indexed	REGY+TS6811
RORA		
RORB		
RTI		
RTS		
SBA		
SBC	ACCUM '#' expr	REGA
SBC	ACCUM '#' expr	REGB
SBC	ACCUM expr	REGA+DIRECT
SBC	ACCUM expr	REGA+EXTENDED
SBC	ACCUM expr	REGB+DIRECT
SBC	ACCUM expr	REGB+EXTENDED
SBC	ACCUM indexed	REGA+REGX
SBC	ACCUM indexed	REGA+REGY+TS6811
SBC	ACCUM indexed	REGB+REGX
SBC	ACCUM indexed	REGB+REGY+TS6811

Opcode	Syntax	Selection Criteria
SBCA	'#' expr	
SBCA	expr	DIRECT
SBCA	expr	EXTENDED
SBCA	indexed	REGX
SBCA	indexed	REGY+TS6811
SBCB	'#' expr	
SBCB	expr	DIRECT
SBCB	expr	EXTENDED
SBCB	indexed	REGX
SBCB	indexed	REGY+TS6811
SEC		
SEI		
SEV		
STA	ACCUM expr	REGA+DIRECT
STA	ACCUM expr	REGA+EXTENDED
STA	ACCUM expr	REGB+DIRECT
STA	ACCUM expr	REGB+EXTENDED
STA	ACCUM indexed	REGA+REGX
STA	ACCUM indexed	REGA+REGY+TS6811
STA	ACCUM indexed	REGB+REGX
STA	ACCUM indexed	REGB+REGY+TS6811
STAA	expr	DIRECT
STAA	expr	EXTENDED
STAA	indexed	REGX
STAA	indexed	REGY+TS6811
STAB	expr	DIRECT
STAB	expr	EXTENDED
STAB	indexed	REGX
STAB	indexed	REGY+TS6811
STD	expr	DIRECT+TS6801PLUS
STD	expr	EXTENDED+TS6801PLUS
STD	indexed	REGX+TS6801PLUS
STD	indexed	REGY+TS6811
STOP		TS6811
STS	expr	DIRECT
STS	expr	EXTENDED
STS	indexed	REGX
STS	indexed	REGY+TS6811

Opcode	Syntax	Selection Criteria
STX	expr	DIRECT
STX	expr	EXTENDED
STX	indexed	REGX
STX	indexed	REGY+TS6811
STY	expr	DIRECT+TS6811
STY	expr	EXTENDED+TS6811
STY	indexed	REGX+TS6811
STY	indexed	REGY+TS6811
SUB	ACCUM '#' expr	REGA
SUB	ACCUM '#' expr	REGB
SUB	ACCUM expr	REGA+DIRECT
SUB	ACCUM expr	REGA+EXTENDED
SUB	ACCUM expr	REGB+DIRECT
SUB	ACCUM expr	REGB+EXTENDED
SUB	ACCUM indexed	REGA+REGX
SUB	ACCUM indexed	REGA+REGY+TS6811
SUB	ACCUM indexed	REGB+REGX
SUB	ACCUM indexed	REGB+REGY+TS6811
SUBA	'#' expr	
SUBA	expr	DIRECT
SUBA	expr	EXTENDED
SUBA	indexed	REGX
SUBA	indexed	REGY+TS6811
SUBB	'#' expr	
SUBB	expr	DIRECT
SUBB	expr	EXTENDED
SUBB	indexed	REGX
SUBB	indexed	REGY+TS6811
SUBD	'#' expr	TS6801PLUS
SUBD	expr	DIRECT+TS6801PLUS
SUBD	expr	EXTENDED+TS6801PLUS
SUBD	indexed	REGX+TS6801PLUS
SUBD	indexed	REGY+TS6811
SWI		
TAB		
TAP		
TBA		
TEST		

Opcode	Syntax	Selection Criteria
TPA		
TST	ACCUM	REGA
TST	ACCUM	REGB
TST	expr	
TST	indexed	REGX
TST	indexed	REGY+TS6811
TSTA		
TSTB		
TSX		
TSY		TS6811
TXS		
TYS		TS6811
WAI		
XGDX		TS6811
XGDY		TS6811

A.2.2 ***Selection_Criteria_Keywords***

CPU6800	The instruction is only valid for the 6800 instruction set.
CPU6801	The instruction is only valid for the 6801 instruction set.
TS6801PLUS	The instruction is valid for the 6801 and 68hc11.
TS6811	The instruction is only valid for the 68hc11 instruction set.
REGA	The instruction can use the A accumulator.
REGB	The instruction can use the B accumulator.
REGX	The instruction can use the X index register.

REGY	The instruction can use the Y index register.
DIRECT	The instruction can use the direct addressing mode. The expression must have a value between 0 and 255 that is defined when the instruction is processed in the input pass.
EXTENDED	The instruction can use the extended addressing mode.

A.2.3 **Notes**

A.2.3.1 **Apostrophes** The apostrophes in the syntax field are a notation used for the parser generator and are not put in the assembler source statements.

A.2.3.2 **Indexed Addressing** The indexed addressing is represented in two ways. Either "expression, index register" or "index register, expression" is accepted by the assembler. Index register is X for all instruction sets and X or Y for the 6811.

A.3 **Reserved Symbols**

A.3.1 **Machine_Dependent_Reserved_Symbols** A B X Y a b x y

A.3.2 **Standard_Reserved_Symbols** AND DEFINED EQ GE GT HIGH LE LOW LT MOD NE NOT OR SHL SHR XOR and defined eq ge gt high le low lt mod ne not or shl shr xor

CONTENTS

A.	Appendix for as6811 Frankenstein Assembler.....	1
A.1	Pseudo Operations.....	1
A.2	Instructions.....	1
A.3	Reserved Symbols.....	15