Microcontroller Documentation

Charles Glasspool

October 14, 2022

1 Architecture

2 Instruction Set Summary

FLAGS

S	Z	P	C	CA			

Instruction	Description	Cycles (T/F)	Opcode	Flags
	Arithmatic			
`ADD [r1]`	Adds the contents of register `r1` to the value in the accumulator, storing the result in the accumulator.	1		1,2,3,4
`ADI [data]`	Adds the immediate value `data` to the accumulator value, storing the result in the accumulator	2		1,2,3,4
`SUB [r1]`	Subtrcts the contents of register `r1` from the value in the accumulator, storing the result in the accumulator.	1		1,2,3,4
`SBI [data]`	Subtracts the immediate value `data` fro the the accumulator value, storing the result in the accumulator	2		1,2,3,4
	Data Transfer			
`MOV [r1] [r2]`	Copies the contents from the `r1` register to the `r2` register	1		none
`LDA [addr]`	Loads the content of `addr` to the accumulator	4		none
`STA [addr]`	Stores the content of the accumulator to address `addr`	4		none
	Logic			
	Branch Control			
`JMP [addr]/[label]`	Unconditionally sets the program pointer to the address specified, either as a literal value or a label of another location in the program	3		none
`J[...] [addr]/[label]`	Conditional jump depending on flag status: `SP` sign positive (`S` = 1), `SN` sign negative (`S` = 0), `Z` zero value (`Z` = 1), `NZ` non-zero value (`Z` = 0), `PO` parity odd (`P` = 1), `PE` parity even (`P` = 0), `C` carry/borrow (`C` = 1), `NC` no carry/borrow (`C` = 0)	3/2		none
`CALL [addr]/[label]`	Unconditionally calls a subroutine, pushing the current program pointer the stack and setting the new program pointer to the specified address/label	5		none
`C[...] [addr]/[label]`	Conditional call depending on flag status (see `J[...]|`)	5/2		none
`RET`	Unconditionally return from a subroutine, pops the address called from from the stack and sets the program pointer to it.	3		none
`R[...] [addr]/[label]`	Conditional return depending on flag status (see `J[...]|`)	3/1		none

3 Instruction Set Details

Register inputs can be provided as either a numerical index of the register or using a keyword specific to the register.

opcode goes to the instruction register

8 bit instruction register

BX	B	C	
DX	D	E	
FX	G	H	
Z	X	Y	Designated for most 16 bit use, such as storing adresses to memory locations
	–	–	Possible future use
	Stack Pointer		
	Program Counter		
	Flag	–	

All values given are the hexadecimal values, but can be provided in any of the accepted formats
A is the accumulator

3.1 Data Transfer

3.1.1 `MOV`