

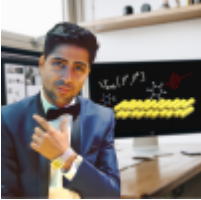


Mem. Addr.	Mnemonics	Hex Codes	Comments
2000	XRA, A	AF	Clear register A
2001	MVI B, 32	06	Load in Reg. B the no. whose parity is to be checked(32)
2002		32	
2003	MVI C,00	0E	Initialize Reg. C
2004		00	
2005	ADD B	80	Perform an operation so that the flags are affected
2006	JPO 200A	E2	Jump if Parity is Odd to Address 200A
2007		0A	
2008		20	
2009	INR C	0C	Increment C if Parity is Even
200A	MOV A, C	79	Move the content of C in Accumulator
200B	STA 2050	32	Store the result in Address 2050H
200C		50	
200D		20	
200E	HLT	76	

The above Program would store 1 at 2050H mem. Addr. If the Parity is Even and 0 if the parity is odd. In the above case we have taken the no. to be 32 which is (100000) in binary, hence has odd parity. Note: Parity is the number of 1's in the circuit.

Input: $(52)_{10} = (110100)_2$

Output: 0 (as it has odd no. of 1's i.e. Odd Parity)



Manas Sharma

I'm a physicist specializing in computational material science with a PhD in Physics from Friedrich-Schiller University Jena, Germany. I write efficient codes for simulating light-matter interactions at atomic scales. I like to develop Physics, DFT, and Machine Learning related apps and software from time to time. Can code in most of the popular languages. I like to share my knowledge in Physics and applications using this Blog and a YouTube channel.

manas.bragitoff.com/



Share this:

[Click to share on Facebook \(Opens in new window\)](#)

[Click to share on Twitter \(Opens in new window\)](#)

[Click to share on WhatsApp \(Opens in new window\)](#)

[Click to share on Pinterest \(Opens in new window\)](#)

[Click to share on Reddit \(Opens in new window\)](#)

[Click to share on LinkedIn \(Opens in new window\)](#)

[Click to email a link to a friend \(Opens in new window\)](#)

[Click to print \(Opens in new window\)](#)

[Click to share on Tumblr \(Opens in new window\)](#)

[Click to share on Pocket \(Opens in new window\)](#)

[Click to share on Telegram \(Opens in new window\)](#)

[wpedon id="7041" align="center"]