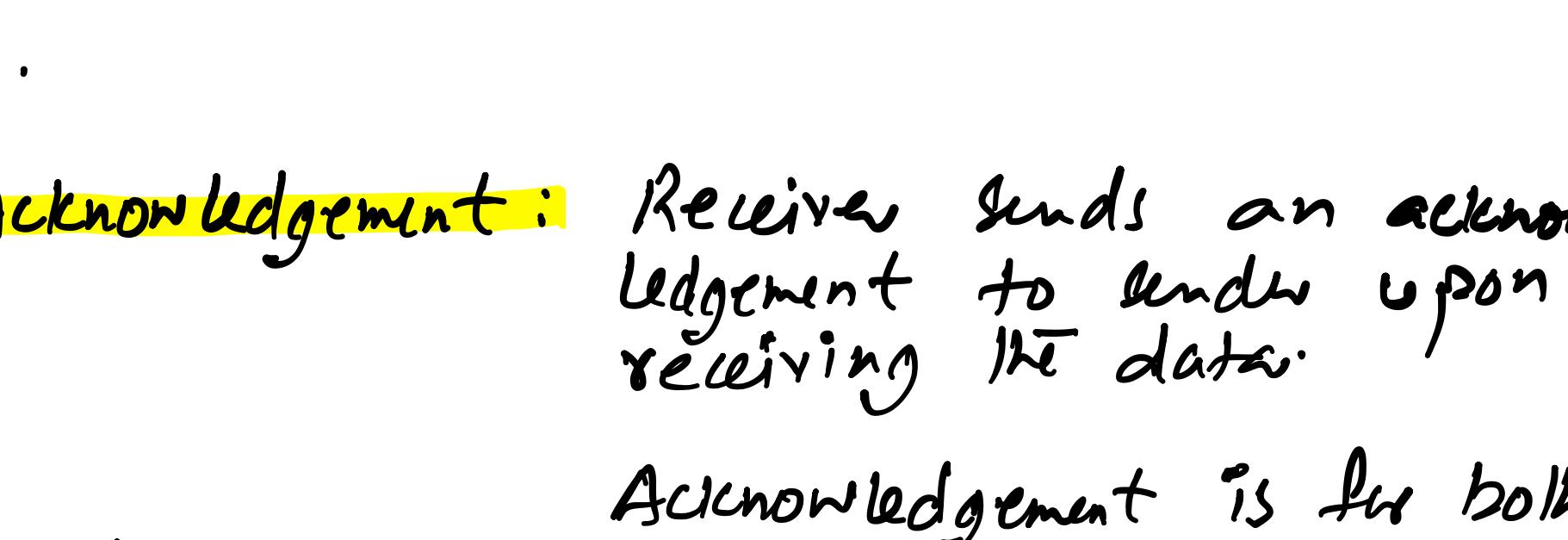
Key terms:

- ARQ ✓
- Parity ✓
- Checksum
- Check digits

Protocols: Rules set forth between devices when some data transfer is to occur. Such as: http, www, SMTP, POP, FTP and ARQ.

Automatic Repeat Request (ARQ):

It is a protocol (method) used to check whether the data is transmitted correctly.

Timeout:

When two devices start data transfer they set a predefined waiting time for the receipt of acknowledgement.

If the ack. is not received during time set then timeout occurs and sender re-sends the data.

This may go on until sender finally receives the ack.

Acknowledgement: Receiver sends an acknowledgement to sender upon receiving the data.

Acknowledgement is for both correct and incorrect data that has reached the receiver.

If the acknowledgement is for "incorrect data" then sender re-sends the data.

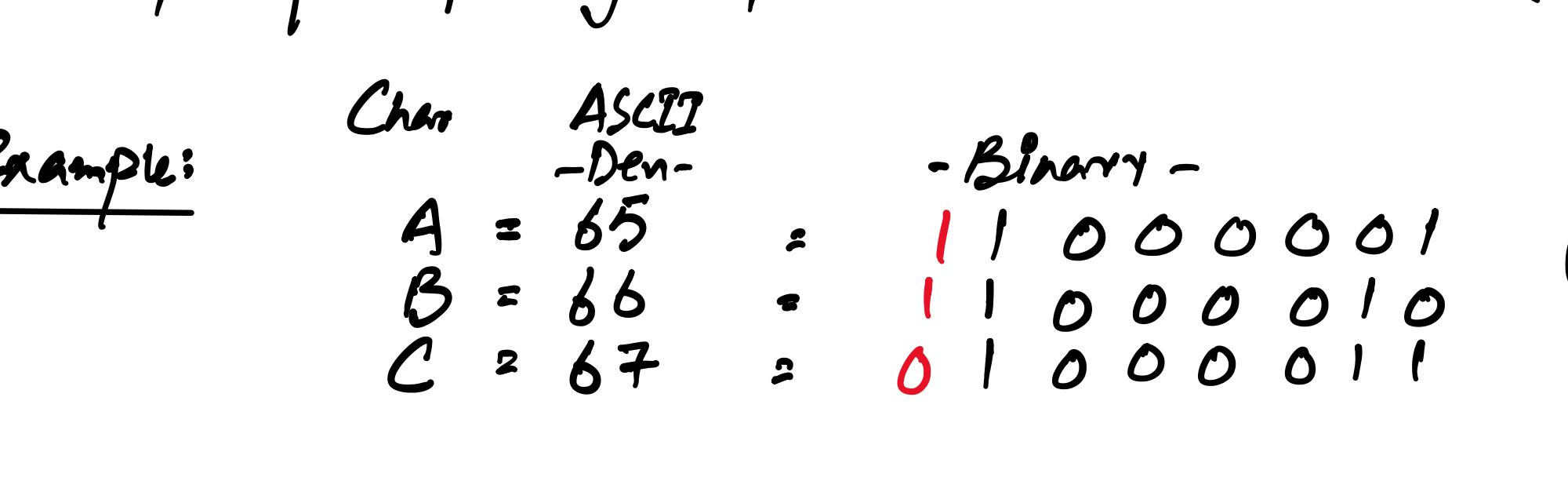
If the acknowledgement is for "correct data" then next data due is sent.

Parity:

It is method that is used to check at receiver's end whether data has been changed or corrupted while being transferred.

It is mainly used with text (ASCII).

It is also used in closed proximity.



There are two parity mechanisms at work.

- Byte parity
- Block parity.

They both use "even" and "odd" parity types.

Example:	Char	ASCII	-Den-	-Binary-	
	A	= 65	=	0 1 0 0 0 0 0 1	
	B	= 66	=	0 1 0 0 0 0 1 0	Even parity.
	C	= 67	=	1 1 0 0 0 0 1 1	

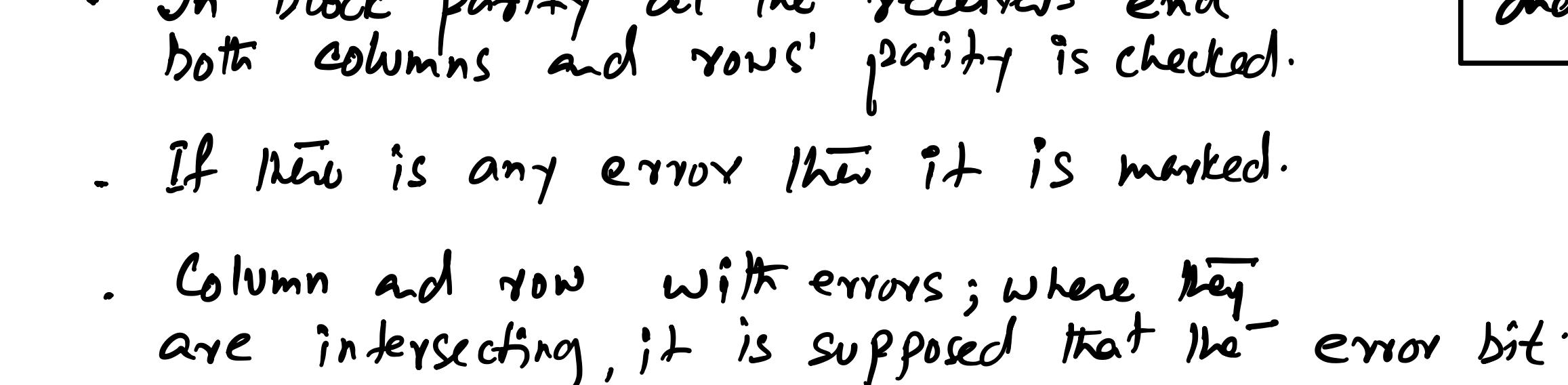
In byte parity single byte is sent and checked every time.

Example:	Char	ASCII	-Den-	-Binary-	
	A	= 65	=	1 1 0 0 0 0 0 1	
	B	= 66	=	1 1 0 0 0 0 1 0	Odd parity.
	C	= 67	=	0 1 0 0 0 0 1 1	

Issues with Byte parity: Error will not be detected if:

- two bits swap
- two bits change (drop, gain)

$$1 \rightarrow 0, 0 \rightarrow 1$$



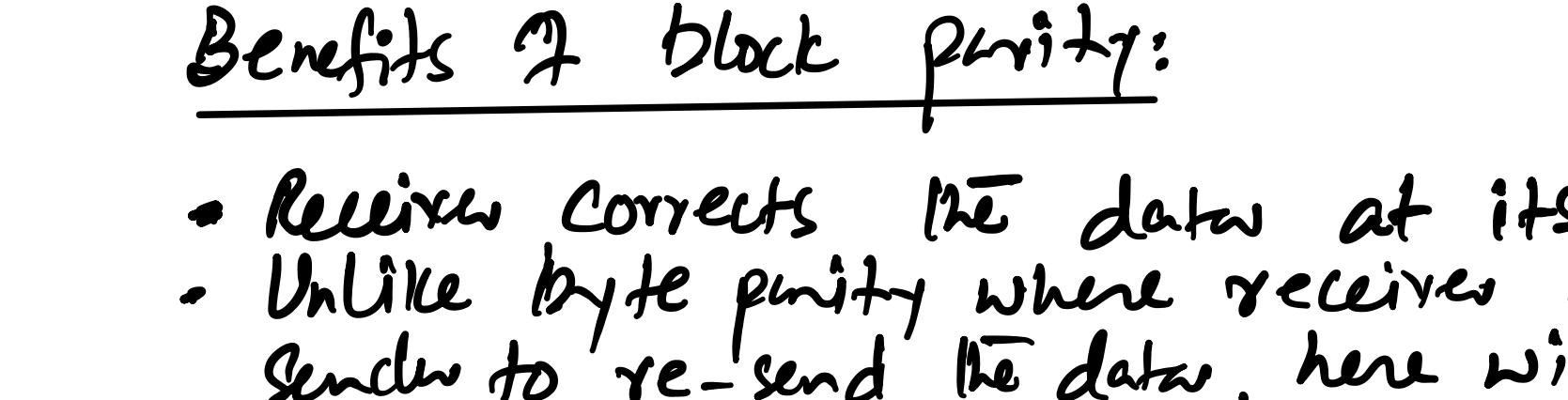
Block Parity: A bunch of bytes is transferred b/w devices. There is an addition of parity byte in a block of data. This is because of the use of column parity.

Data Block ↗

Even parity	A = 65	=	0 1 0 0 0 0 0 1	
	B = 66	=	0 1 0 0 0 0 1 0	
	C = 67	=	1 1 0 0 0 0 1 1	

parity ← Byte.

Note: parity for all 0's in column or row is always 0. This is same for both even and odd parity.



Benefits of block parity:

- Receiver corrects the data at its end.
- Unlike byte parity where receiver asks sender to re-send the data, here with Block parity receiver is able to do it (correct) data itself.

This saves bandwidth and time, only to make communications faster.