Detailed Block Diagrams

Preferred graphical editors:

draw.io or MS Visio

General rules:

A. avoid overlaps between lines and labels (such as writing labels over lines)
B. do not name basic components with characteristic shapes such as MUXes, registers, XORs, adders, gates, etc. Instead, you can name outputs from these components. Use standard labels (such as ==A, >>, <<<, etc.) for components such as comparators, shifters, rotators, etc.
C. use only one input for a comparator with a constant, as well as arithmetic units where one of the two operands is a constant. Instead, use meaningful names, such as "==A", "<A", "+A", "*A", where A is replaced by a symbol or value of a constant. Sizes of boxes representing these basic components should be preferably the same.
D. in the labels for RAMs and ROMs include the size of memory, expressed using the format <number of words> x <word width>, e.g., 256 x 16
E. you do not need to use an explicit label clk if an input to a sequential component is already marked with a triangle
F. write internal names of ports inside of rectangles representing components
G. use either "solder points" (dots) to indicate that connection exists or flyovers to indicate that wires/buses are not connected (do not use both conventions in the same set of diagrams)
H. do not use "solder points" (dots) for T type connection, when an outgoing bus contains only a subset of lines of the main bus
I. use notation such as $B_{7..4}$, $B_{3..0}$ to indicate a subset of bits of a particular bus
J. identify the most common bus width and denote it using a thicker line rather than labeling this bus width explicitly in all places in your diagram
K. use consistent capitalization for all labels of the same type. You may apply different conventions to two different types of labels, e.g., label signals using small letters and components using capital letters. All labels, other than those representing bus widths, should preferably have the same font size.
L. name and label all data, status, and control signals in your datapath that will appear as signals in your VHDL code
M. do your best to keep your signal names the same or similar to VHDL names and names used in the cipher specification (if necessary, deviate from the names used in the cipher specification, e.g., replace S[2i+1] by S2ip1)
N. use hierarchical block diagrams
O. label lower-level entities inside of the respective symbols
P. clearly state the name of each entity at the top of the respective block diagram
Q. use arrows to indicate the direction of data flow
R. use mostly explicit connections using buses and wires; use connections by name only for connecting signals at the same level of hierarchy when they are located far from each other in your block diagram.