Proposed snapshot format

1 Word : value of system stack pointer

1 Word : SAM bits (so memory config can be determined).

32K / 64K : memory dump of RAM.

May also need a signature word.

Snapshot method :

Enter via NMI (or simulated NMI)

Get and stack word containing SAM bits.

For Each PIA, get the registers in the following order : CRA,DDRA, PA, CRB, DDRB, PB and stack them.

Create snapshot file.

Write value of S and copy of SAMBITS.

Write a dump of RAM 32K / 64K.

RTI

Restore method :

Open snapshot file & validate. (AVR task?).

Retrieve system stack pointer & SAM bits (to determine memory config).

Set S to be retrieved pointer.

Restore memory being careful about not tramping on working storage ☺

Pull PIA registers from stack, need to restore DDR before port reg so something like

Restore CRA, set ddr bit, restore DDRA, reset ddr bit, restore PA, restore CRA.

We restore CRA twice then we will always set the DDR bit to the same as it was when the snap was created.

Restore SAM bits.

Return (to restored program RTI).

Initially do this via a command that simulates the interrupt, then write a UI that is triggered by snap button on interface.