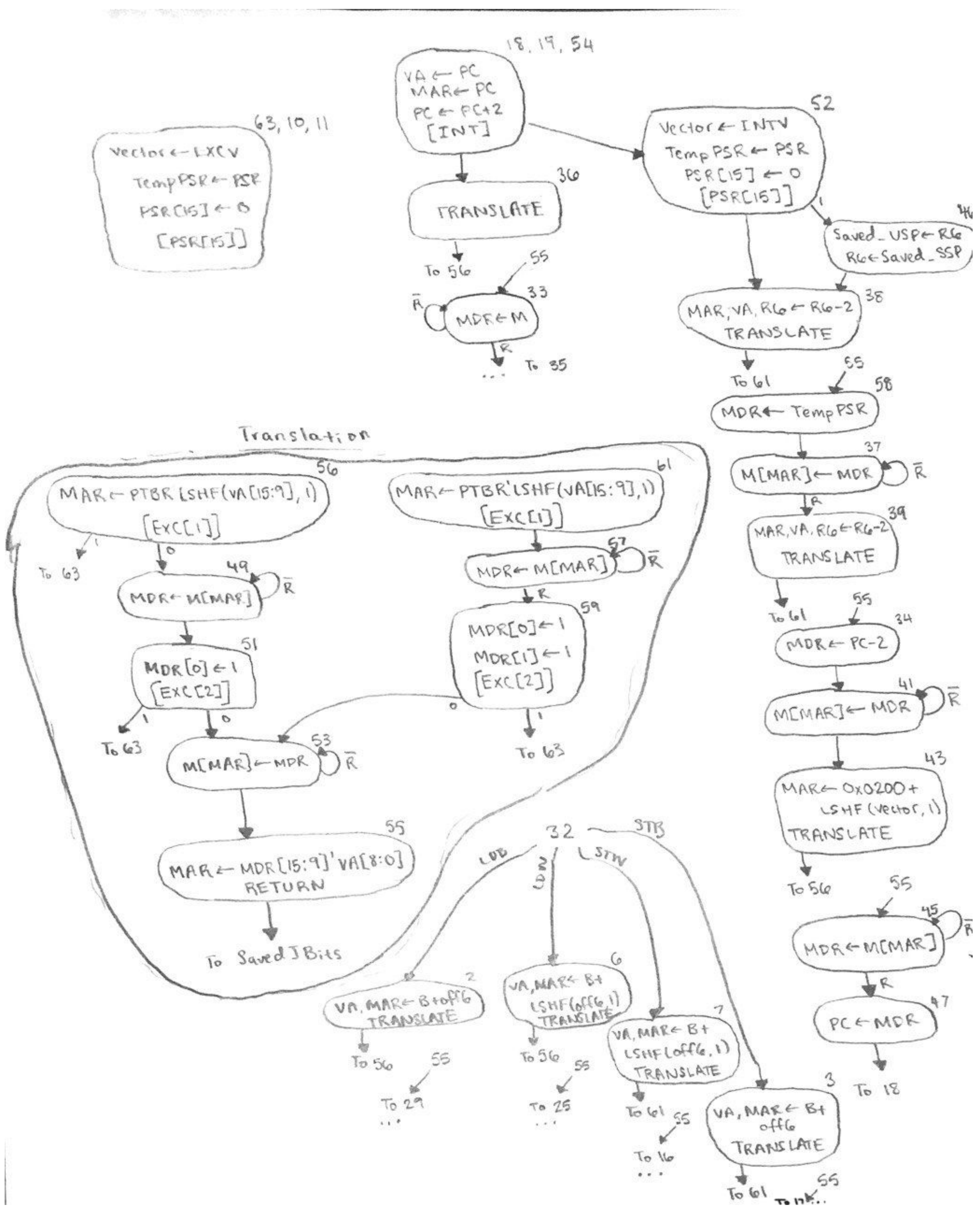
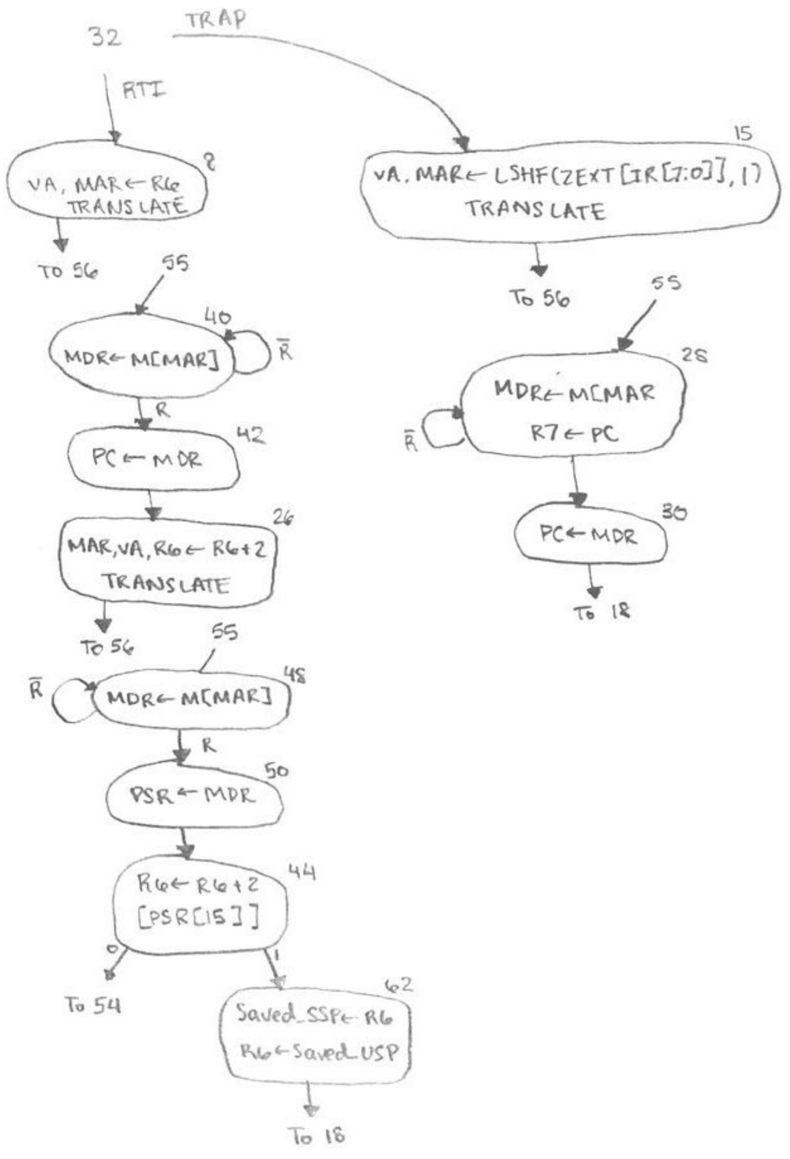
Sidharth Nair - sn25377



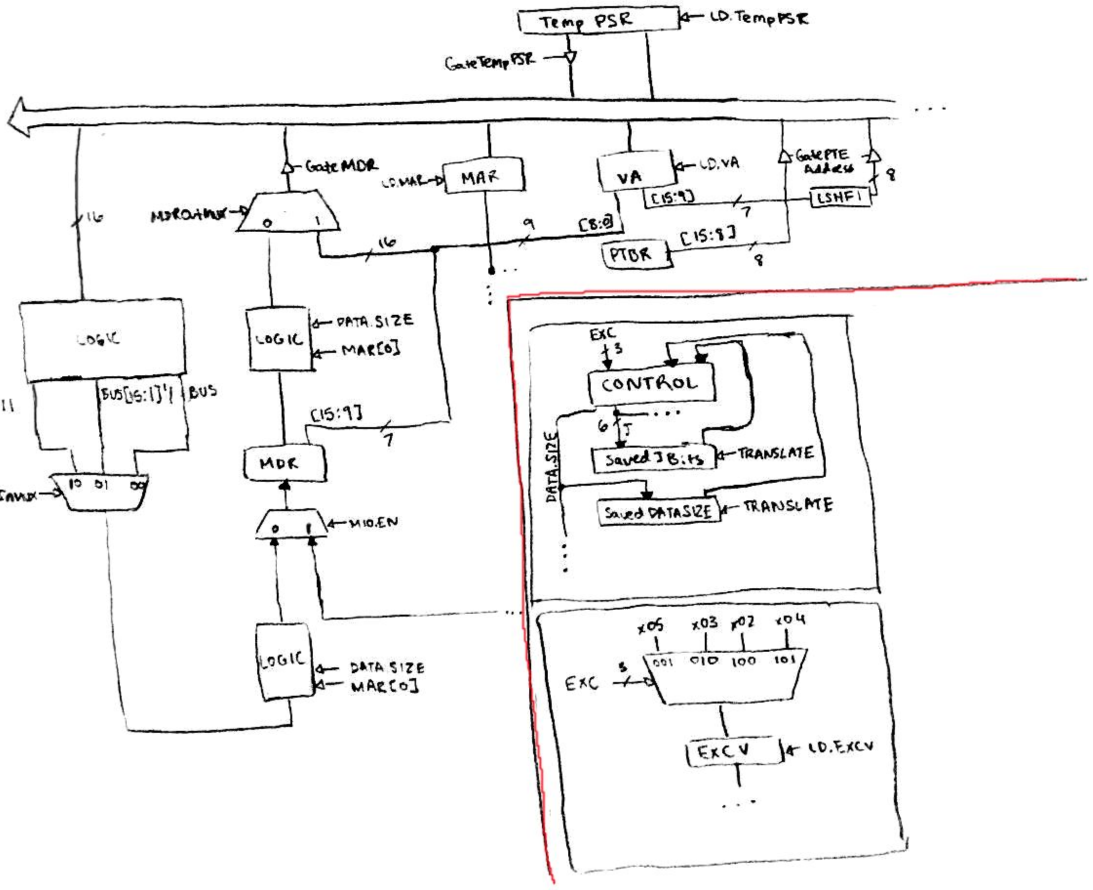
Sidharth Nair - sn25377



Sidharth Nair - sn25377

Additions to State Machine: In order to accommodate virtual memory, we need to add states that carry out the translation of virtual addresses to physical addresses. In order to do this, I added a set of states (56, 49, 51, 61, 57, 59, 53, and 55) that takes the virtual address loaded into the MAR/VA registers (now the VA will always be loaded with the MAR outside of the address translation process) to generate the PTE of the physical address of that virtual address, loads the PTE into the MDR, sets its reference bit and (if the memory access that requires the translation is a store) its modified bit, stores the PTE back into the page table, and determines the physical address using the PFN from the PTE and the offset from the VA. Since we need to differentiate between load and store memory accesses for whether we need to set the modified bit of the PTE, there are two states that start the translation process (56 for loads and 61 for stores), both of which join together in state 53 to store the PTE back to memory. Before every memory access on the state machine we must enter and perform this translation process which is shown by states labeled with "TRANSLATE". In these states, the J bits of the next address are stored into a SavedJBits register which will be used to return to the state (after state 55; after the virtual address translation process is done) that actually performs the memory access. The DATA.SIZE bit is also saved in a SavedDATASIZE register which will be used for unaligned exception checking in state 56. This covers the main changes to the state machine but a few modifications had to be made to account for the fact that the translation process will change what is in the MDR before translation. One change was that in states 10, 11, 52, and 63 the PSR no longer is put into the MDR but rather it is put into a TempPSR register so that it can be loaded into the MDR later in state 58. Also, the operations in state 39 and 34 were swapped from lab 4 for the same reason.

Sidharth Nair - sn25377



Additions to the Data Path: As mentioned in the modifications to the state machine section, additions to the data path include a TempPSR register, a VA register, a PTBR register, a SavedJBits register, and a SavedDATASIZE register. There is now a MDRInMUX and MDROutMUX which are added to enable virtual address translation. VA[15:9] and PTBR[15:8]<<1 are gated onto the bus together (since they are both used to generate the address to a PTE) using the GatePTEAddress signal. This PTE address is loaded into the MAR and then the PTE is read from memory into the MDR. Based on whether the access requiring the translation is a load or a store, the MDR gets loaded with itself ORed with x0001 (valid bit set) or x0003 (valid and modified bits set) respectively. The updated PTE gets stored back to the page table and then the MAR is loaded with MDR[15:9]'VA[8:0] (the physical address of the initial virtual address). Since we have an additional exception now, the exception vector mux has been updated with an additional input and the EXC signal has been made into 3 bits (the logic for which is explained later in this document).

New Control Signals:
TRANSLATE - Used to load the SavedJBits and SavedDATASize registers and in the microsequencer to set the next state number to 56 or 61 based on whether the W signal (R.W = 1) is asserted.

Sidharth Nair - sn25377

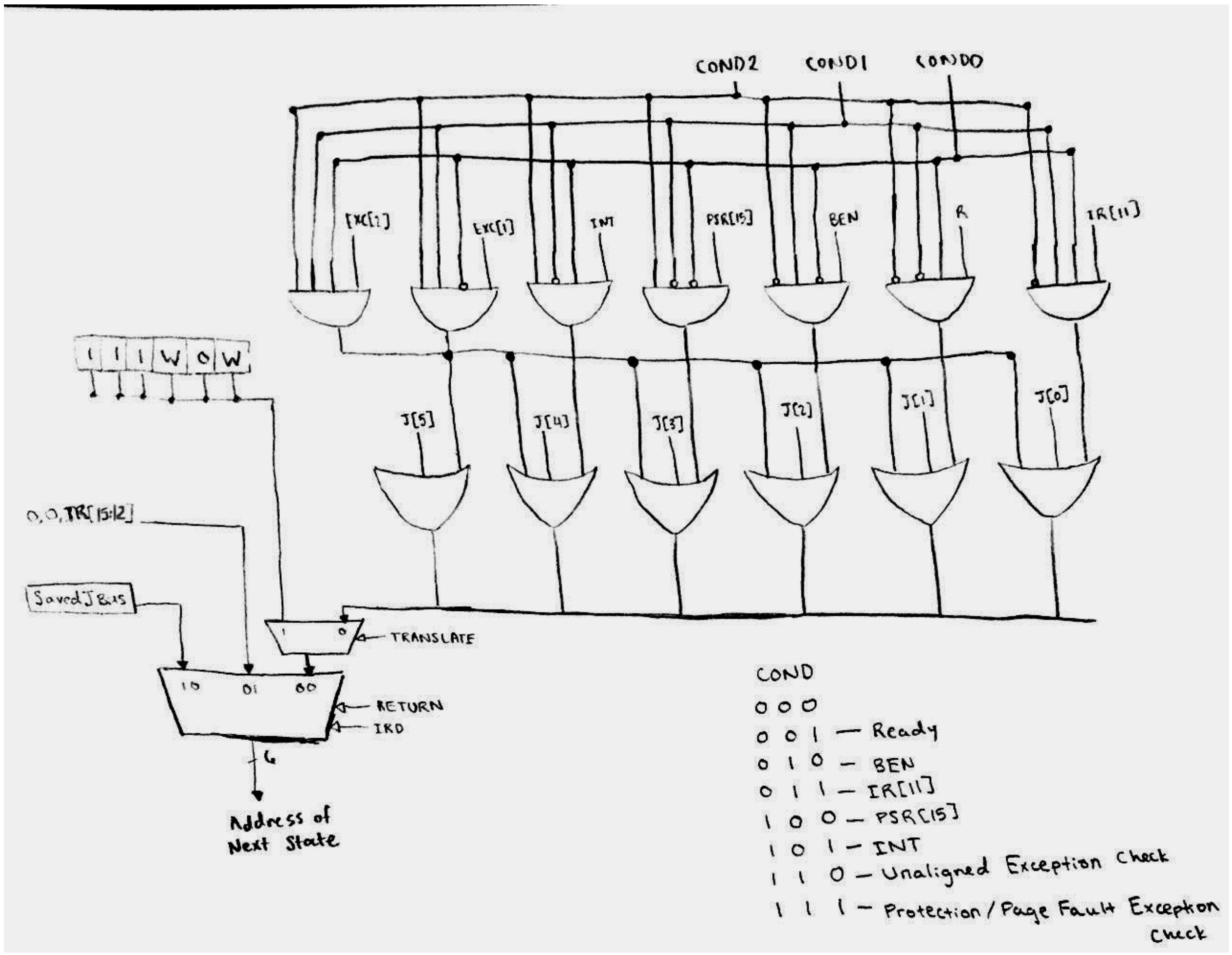
RETURN - Used in the microsequencer to return to perform the memory access after address translation. If asserted it will use the SavedJBits register to get the next state.
LD.VA - Used to load the VA register
LD.TempPSR - Used to load the TempPSR register
GatePTEAddress - Used to gate VA[15:9] and PTBR[15:8]<<1 (the address of the PTE) onto the bus
GateTempPSR - Used to gate the TempPSR register onto the bus
MDRInMux - Used to select an MDR input from the BUS:

 00 = BUS, 01 = BUS[15:1]’1, 10 = BUS[15:2]’11

MDROutMUX - Used to select the MDR output to be gated on the BUS:

 0 = Output of logic block, 1 = MDR[15:9]’VA[8:0] (PA of given VA)

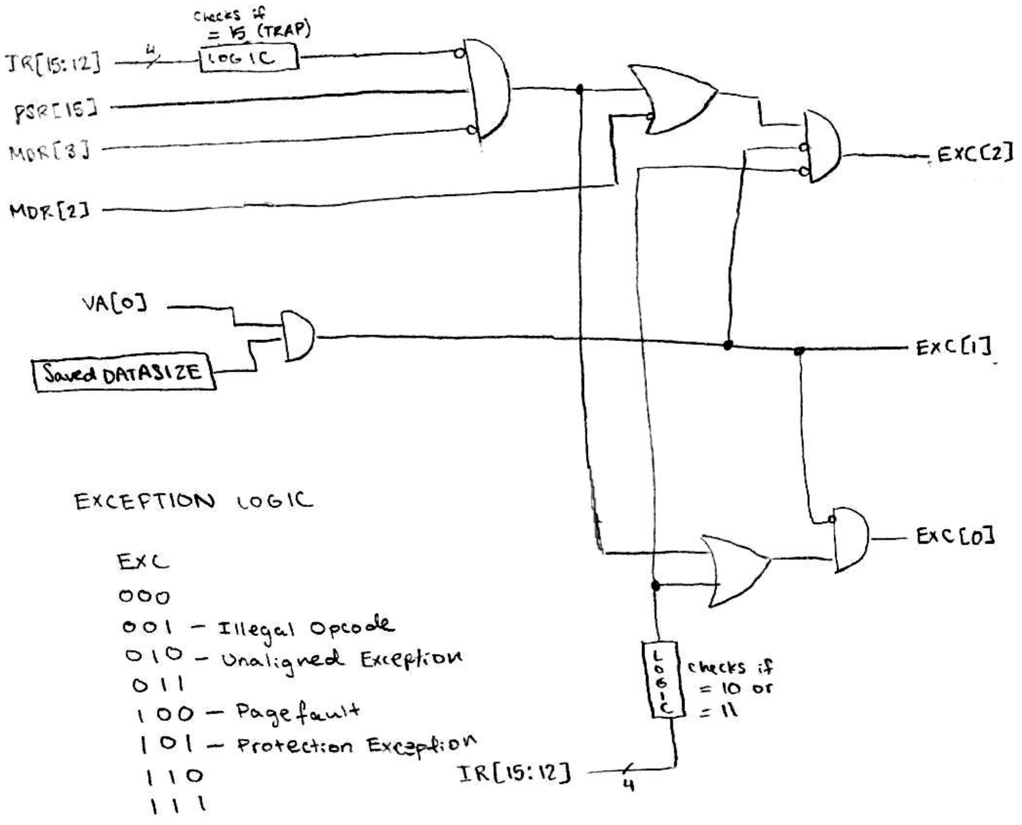
Sidharth Nair - sn25377



COND	
0 0 0	
0 0 1	Ready
0 1 0	BEN
0 1 1	IR[11]
1 0 0	PSR[15]
1 0 1	INT
1 1 0	Unaligned Exception Check
1 1 1	Protection / Page Fault Exception Check

Changes to the Microsequencer: A new COND state was added which differentiates the unaligned exception check (COND6) from the protection/page fault exception check (COND7). If the TRANSLATE signal is asserted, the new MUX selects state 51 or 56 based on whether W is enabled (R.W = 1), otherwise it selects the output of the 6 OR gates. The final MUX before the new state is outputted has been modified to be a 3 input MUX controlled by RETURN’IRD. This allows for the next state to be determined by the SavedJBits register as well which is important for returning to the memory access state after address translation.

Sidharth Nair - sn25377



Exception Logic: This image shows how the EXC signals are determined from the data path to be used for micro branches in the state machine and to load the EXCV register for the exception handler context switch.