Assembly Language Conversion: GCPU to ATxmega128A1U

GCPU Instructions	Description	ATxmega Instructions	Description
LDAA #data LDAB #data	Load A or B with immediate data	LDI Rd, data Ex.: LDI R16, 0xFF Note: Use R16-R31for Rd	Load the immediate value 'data' into the destination register Rd.
LDAA addr LDAB addr	Load A or B with value at addr	LDS Rd, (16-bit address) Ex.: LDS R20, 0x2000 Note: Use R0-R31 for Rd; for data memory only (**not** for program memory)	Load the value at the specified location into Rd. The specified address is 16-bits.
LDAA 0,X LDAB 0,Y	Load A or B with value at addr X or Y	LD Rd, W Ex.: LD R0, X Note: Use R0-R31 for Rd and use X, Y, or Z for W; for data memory only (**not** for program memory)	Load the value at the location in W into Rd. The value in W is the addr.
No equivalent, but similar to above (but for program memory)		LPM Rd, Z Ex.: LPM R0, Z Note: Use R0-R31 for Rd; for program memory only (**not** for data memory)	Load the value at the location in Z into Rd. The value in Z is the addr.
STAA addr STAB addr	Store data in A or B to memory location addr	STS (16-bit address), Rr Ex: STS 0x3FFF, R0 Note: Use R0-R31 for Rr	Store the value in Rr at the location in the specified 16-bit address.
STAA 0.X STAB 0,Y	Store data in A or B to memory location at addr X or Y	ST W, Rd Ex: ST Y, R17 Note: Use X, Y, or Z for Rd and use R0-R31 for Rr; for data memory only (**not** for program memory)	Store the value in Rd to the location pointed to by W.
SUM_BA SUM_AB	Add the two registers together and store in A or B	ADD Rd, Rr Ex: ADD R5, R6 Note: Use R0-R31 for both Rd and Rr	Add Rr to Rd. Rd = Rd + Rr
AND_BA AND_AB	AND/OR the two regs together and store result in A or B	AND Rd, Rr OR Rd, Rr Ex: AND R25, R16 Note: Use R0-R31 for both Rd and Rr	AND/OR Rr with Rd. Store result in Rd. Rd = Rd AND/OR Rr
COMA COMB	Complement the contents in register A or B	COM Rd Ex: COM R18 Note: Use R0-R31 for Rd	Take One's complement of register Rd. Behaves like NOT.
SHFA_L SHFA_R	Shift the contents in register A left or right.	LSL Rd LSR Rd Ex: LSL R16 Note: Use R0-R31 for Rd	Shift the contents in the Rd left or right by 1 bit.

Assembly Language Conversion: GCPU to ATxmega128A1U

GCPU Instructions	Description	ATxmega Instructions	Description
INX INY	Increment the contents in register X or Y	INC Rd DEC Rd Ex: INC R31 Note: Use R0-R31 for Rd	Increments/decrements the content in Rd.