# Assembly Language Conversion: GCPU to TI DSC

| GCPU Instructions | Description | DSC Instructions | Description |
|---|---|---|---|
| LDAA #data<br>LDAB #data | Load A or B with immediate data | MOV reg, #data<br>Ex.: MOV AR5, #0xF<br>Note: Use regs AR0-AR7 or AH/AL for the reg | Load the immediate value 'data' into the reg |
| LDAA addr<br>LDAB addr | Load A or B with value at addr | MOV reg1, *(16-bit address)<br>Ex.: MOV AL, *(0xB371)<br>Note: Use AH/AL or AR0-AR7 for reg1 | Load the value at the location in specified into reg1. The specified address is 16-bits. |
| LDAA addr<br>LDAB addr | Load A or B with value at addr | MOV reg1, *reg2<br>Ex.: MOV AL, *AR1<br>Note: Use AH,AL,AR6,AR7 for reg1 and use AR0-AR5 for reg2 | Load the value at the location in reg2 into reg1. The value in reg2 is the addr. |
| STAA addr<br>STAB addr | Store data in B or A to memory location addr | MOV *reg1, reg2<br>Ex: MOV *AR1, AR2<br>Note: Use AR0-AR5 for reg1 and use AL/AH or AR0-AR7 for reg2 | Store the value in reg2 at the location in reg1. The value in reg1 is the addr |
| STAA addr<br>STAB addr | Store data in B or A to memory location addr | MOV *(16-bit address), reg1<br>Ex: MOV *(0xB371), AR1<br>Note: Use AL/AH or AR0-AR7 for reg1 | Store the value in reg1 at the location in the specified 16-bit address. |
| SUM_BA<br>SUM_AB | Add the two registers together and store in A or B | ADD reg1, reg2<br>Ex: ADD AH, AR0<br>Note: Use AH/AL for reg1 and use AH/AL or AR0-AR7 for reg2 | Add reg2 to reg1.<br>reg1 = reg1 + reg2 |
| AND_BA<br>AND_AB | AND/OR the two regs together and store result in A or B | AND reg1, reg2<br>OR reg1, reg2<br>Ex: AND AL, AR1<br>Note: Use AH/AL for reg1 and use AH/AL or AR0-AR7 for reg2 | AND/OR reg2 with reg1. Store result in reg1.<br>reg1 = reg1 AND/OR reg2 |
| COMA<br>COMB | Complement the contents in register A or B | NOT reg<br>Ex: NOT AR6<br>Note: Use AH/AL or AR0-AR7 for reg. | Complement the contents in the reg |
| SHFA_L<br>SHFA_R | Shift the contents in register A left or right. | LSL reg, X<br>LSR reg, X<br>Ex: LSL AH, 3<br>Note: Use AH/AL for reg | Shift the contents in the reg X times |
| INX<br>INY | Increment the contents in register X or Y | INC reg<br>Ex: INC AR5<br>Note: Use AH/AL or AR0-AR7 for reg. | Increments the contents in the reg |