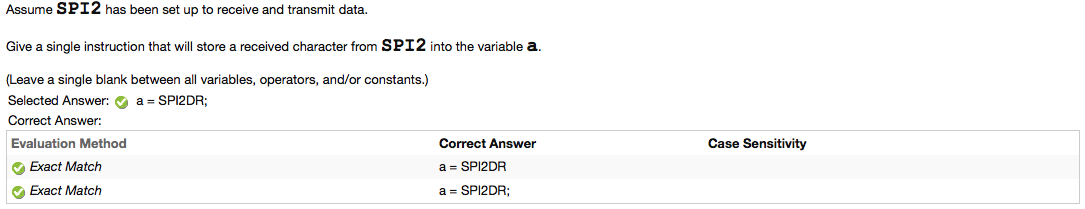
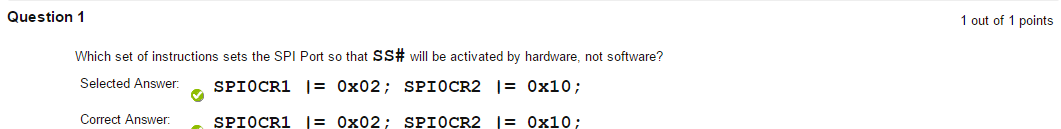
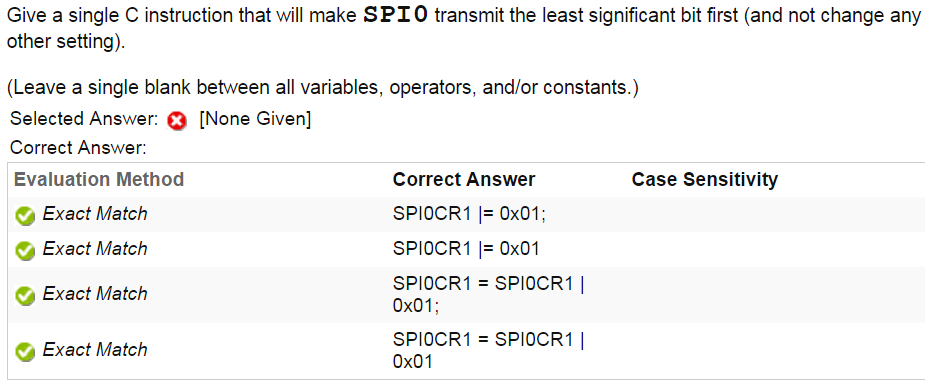
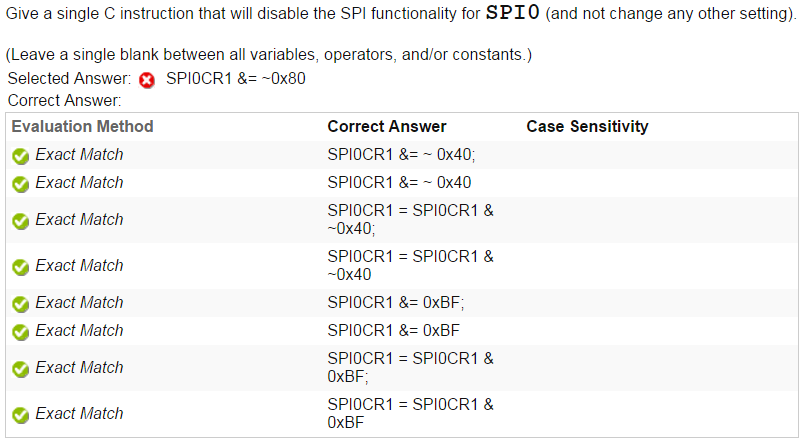
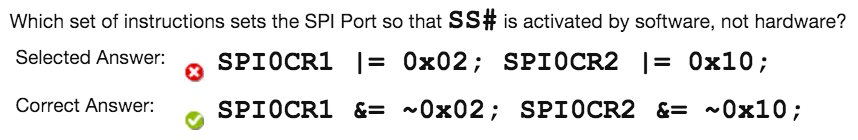
**HW Test 15: SPI-General**

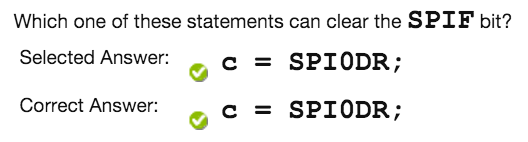


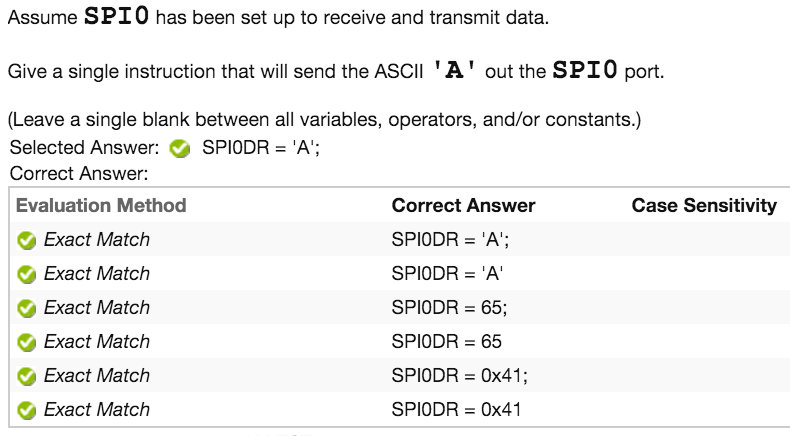


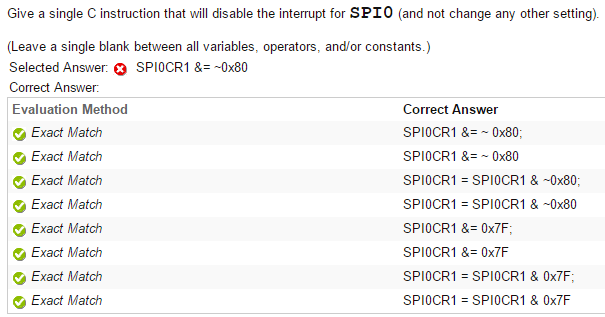


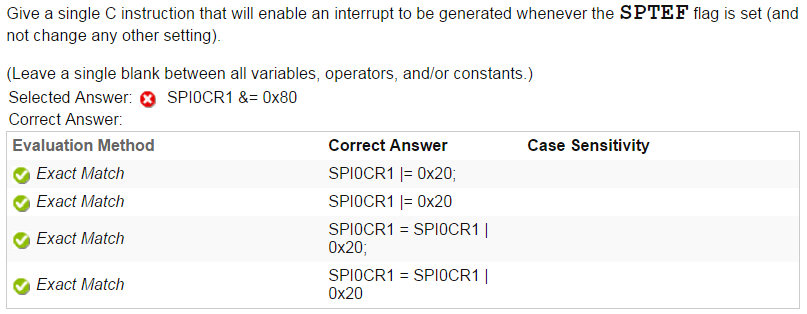


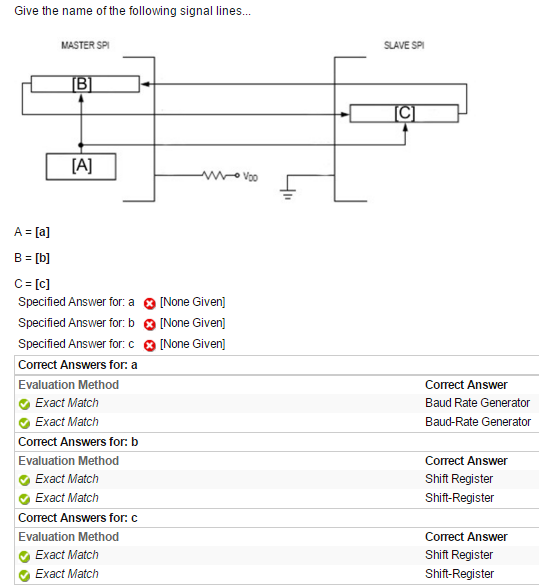


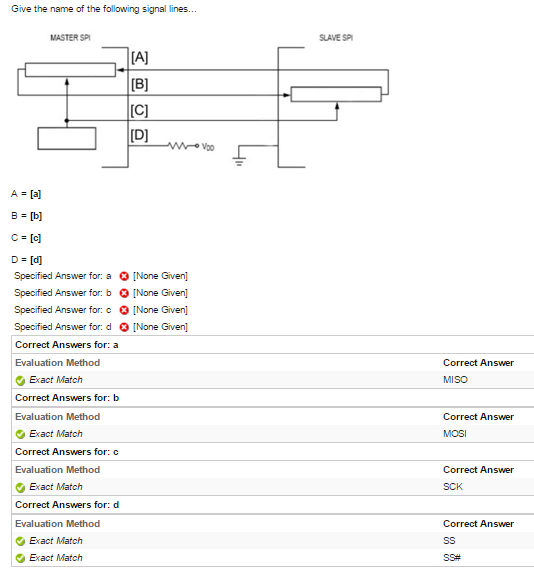


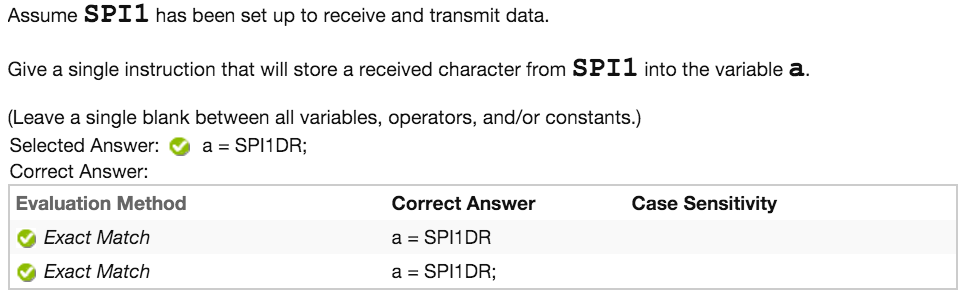




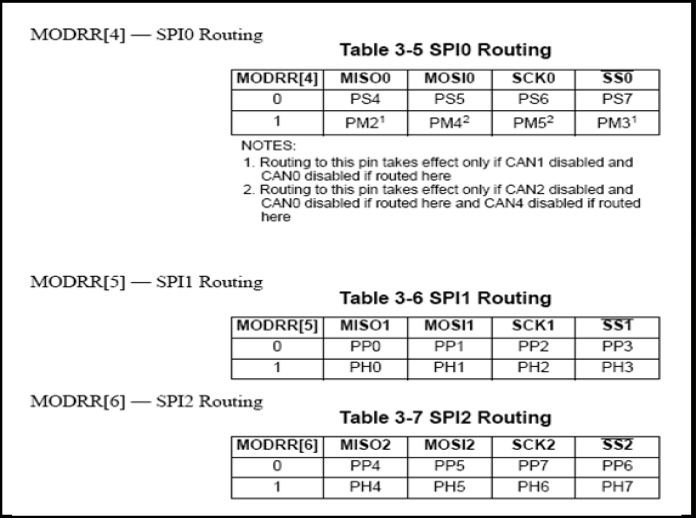








**HW Test 15: SPI-MODRR**



**Write out bits exactly like this if Port H or Port M:**

**0??? 0000**

**To fill in Question Marks, put a 1 where your corresponding SPI is.**

**Other Question Marks are 0’s**

**Use an OR in your answer.**

**MODRR |=**

**If Port P or Port S:**

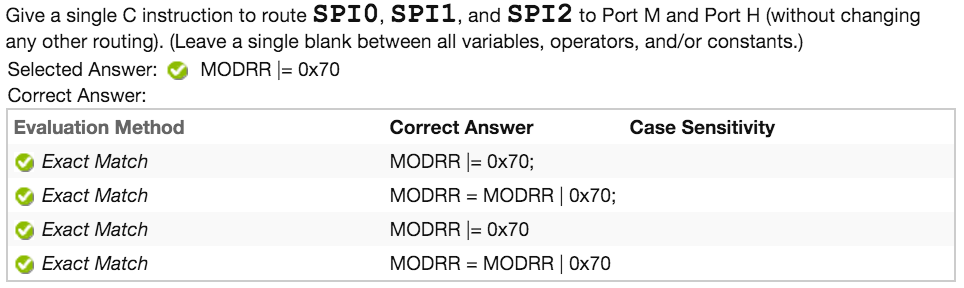
**1??? 1111**

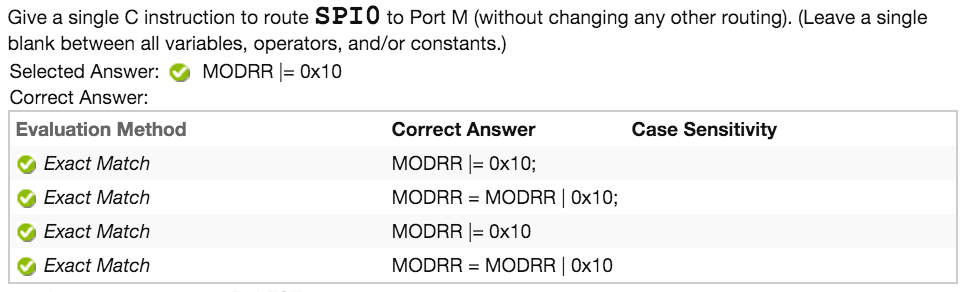
**Fill in Question Marks with a 0 where your corresponding SPI is.**

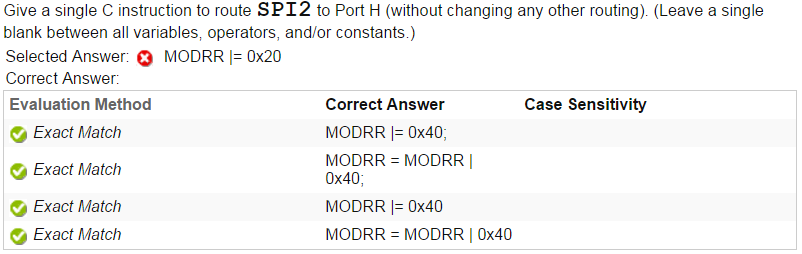
**Other Question Marks are 1’s**

**Use an AND in your answer.**

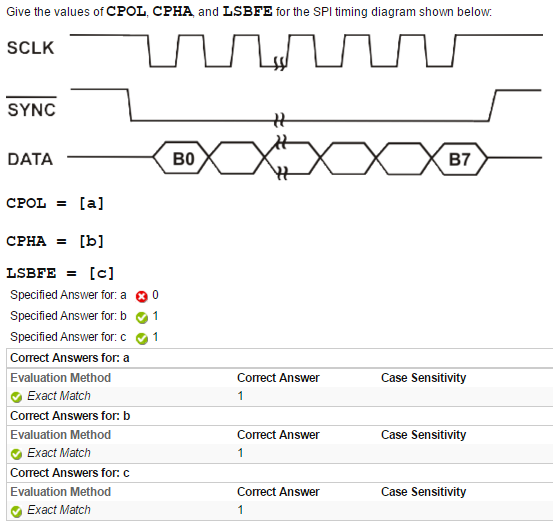
**MODRR &=**







**HW Test 15: SPI Clock Timing**



**CPOL is 1 if SCLK starts HIGH**

**CPOL is 0 if SCLK starts LOW**

**If SYNC and DATA beginnings line up CPHA 0**

**If SYNC and DATA beginnings do not line up CPHA 1**

**LSBFE is 0 if DATA starts at B7**

**LSBFE is 1 if DATA starts at B0**

**HW Test 15: What Is Baud Rate**

**BusClock = CoreClock/2**

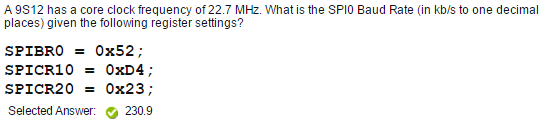
**BaudRateDivisor = (SPIBR[6:4]+1) \* 2^(SPIBR[2:0]+1)**

**BaudRate = BusClock / BaudRateDivisor**

**BaudRate(in kb) = BusClock/ (BaudRateDivisor \* 2^10)**

**MHz = 10^6**

**Kb = 2^10**



**HW Test 15: What Is SPIBR**

**BusClock = CoreClock / 2**

**Convert kb to b (BaudRate \* 2^10)**

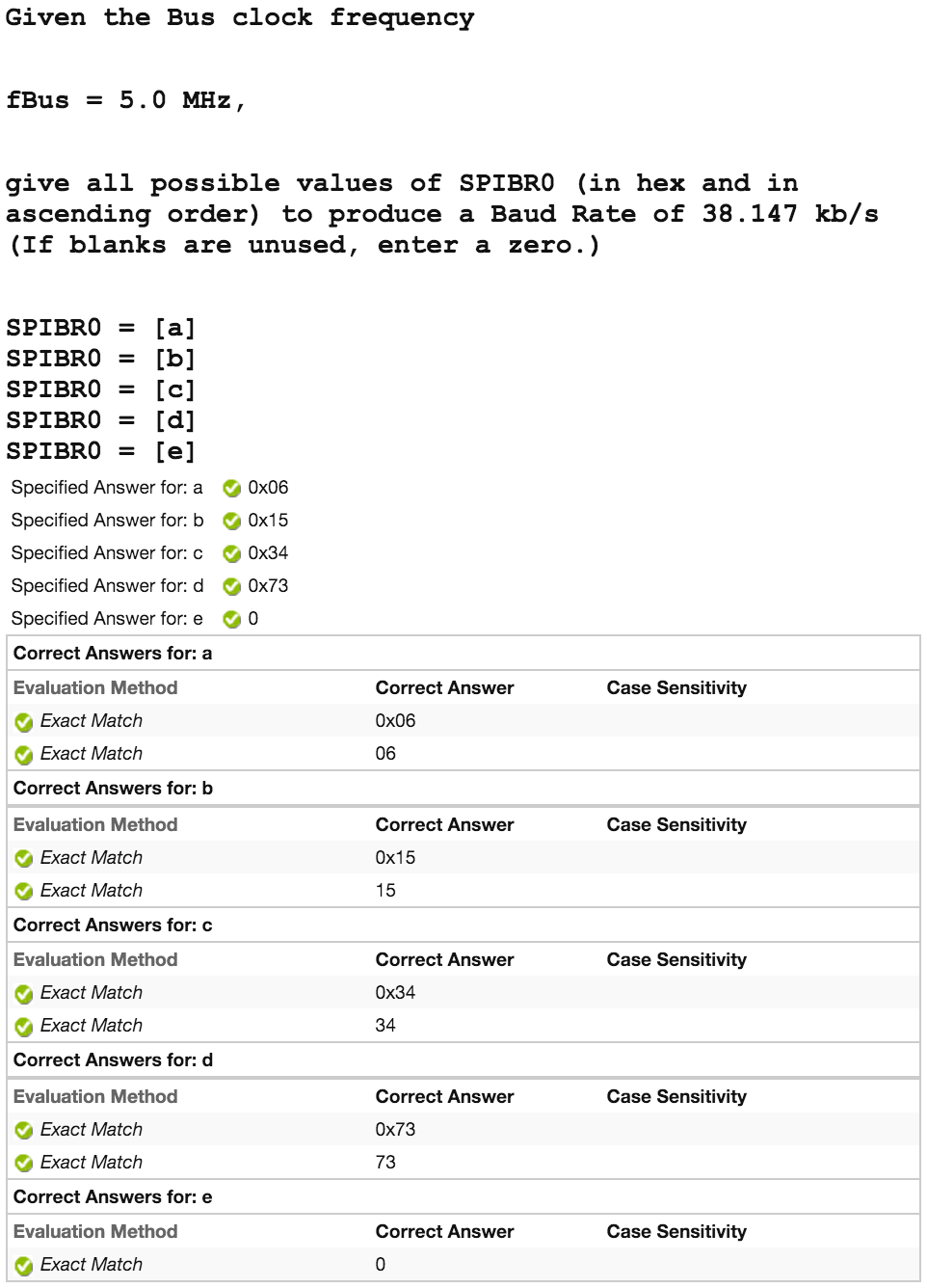
**BaudRateDivisor = BusClock / BaudRate**

**Round BaudRateDivisor to nearest whole number**

**BaudRateDivisor = (#one + 1) \* 2^(#two + 1)**

**Choose sets of numbers that equal BaudRateDivisor**

**#one and #two can’t be above 7**

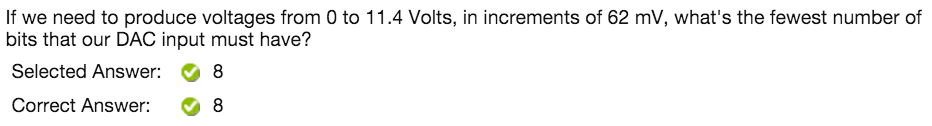


**HW Test 16: DAC-General**



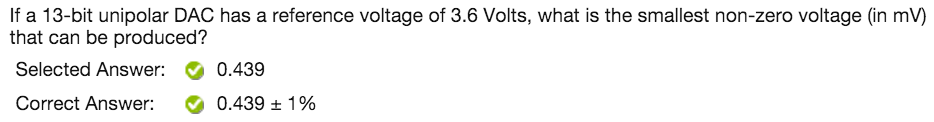


**HW Test 16: DAC-Levels**



**Bits = Logbase2(VoltageSpan(V) / Increments(V))**

**\*\*Always round up\*\***



**For Unipolar:**

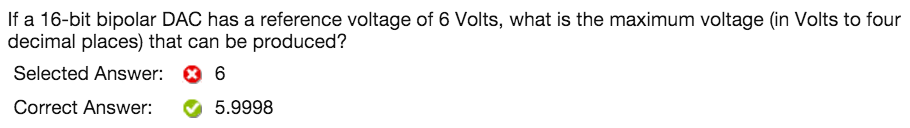
**SmallestVoltage(V) = RefVolt / (2^Bits)**

**Pay attention to mV**

**For Bipolar:**

**SmallestVoltage(V) = RefVolt / (2^(Bits - 1))**

**Pay attention to mV**



**For Unipolar:**

**MaxVolts(V) = RefVolt - (RefVolt / 2^Bits)**

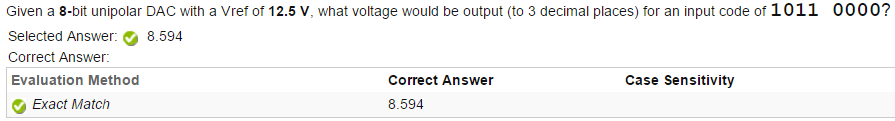
**Pay attention to decimal places**

**For Bipolar:**

**MaxVolts(V) = RefVolt - (RefVolt / 2^(Bits - 1)**

**Pay attention to decimal places**

**HW Test 16: DAC-Values**



**For Unipolar:**

**Decimal = Input Code in Decimal**

**Voltage = (Vref / 2^Bits) \* Decimal**

**Pay attention to decimal points**

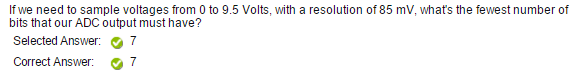
**For Bipolar:**

**Decimal = Input Code in Decimal**

**Voltage = (Vref / 2^(Bits - 1)) \* Decimal**

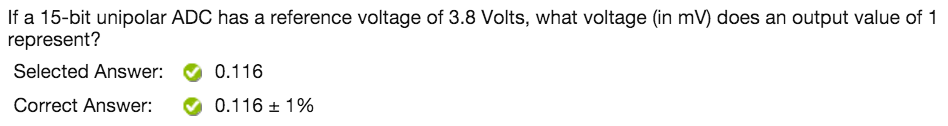
**Pay attention to decimal points**

**HW Test 17: ADC-Levels**



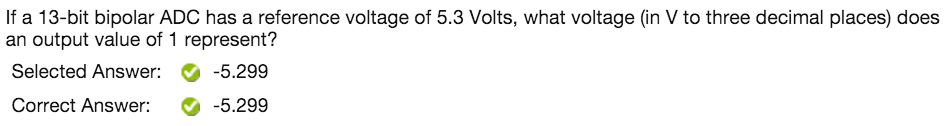
**Bits = Logbase2(VoltageSpan / resolution(V))**

**\*\*Always round up\*\***



**Voltage(V) = Vref / 2^Bits**

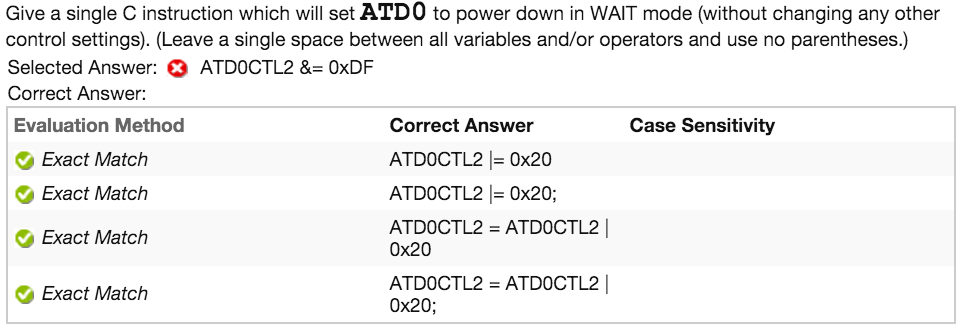
**Pay attention to mV**

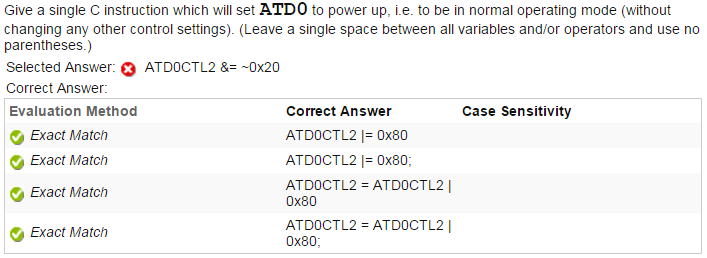


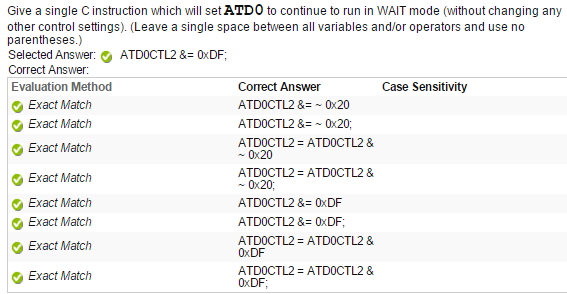
**Voltage = (Vref / 2^Bits) - Vref**

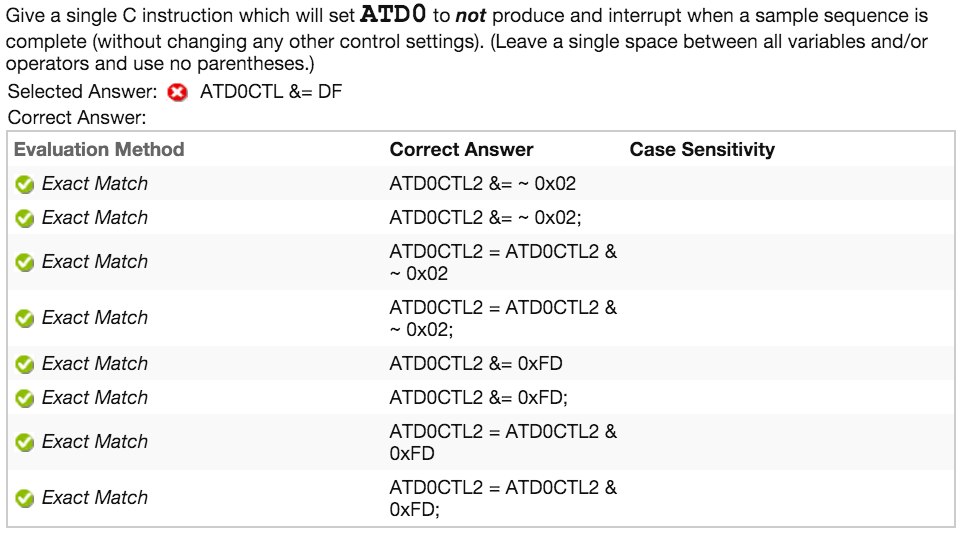
**Pay attention to decimal points**

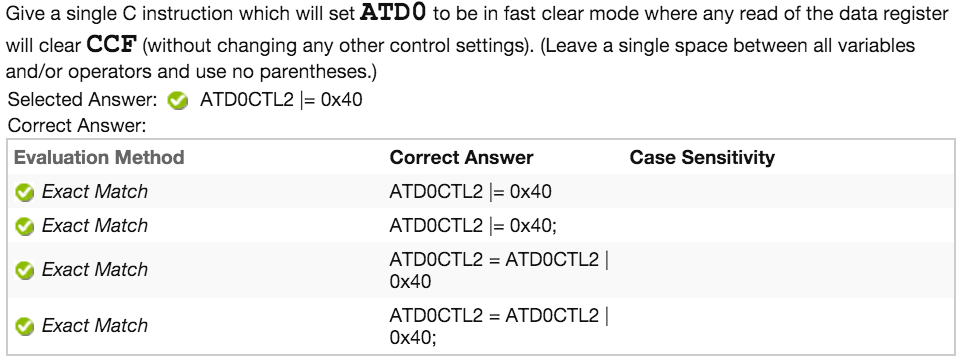
**HW Test 18: ADC-CTRL2**

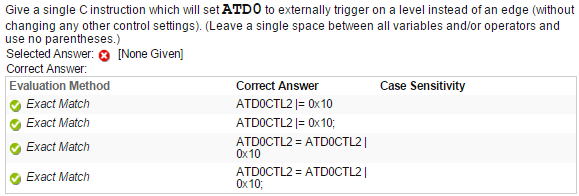


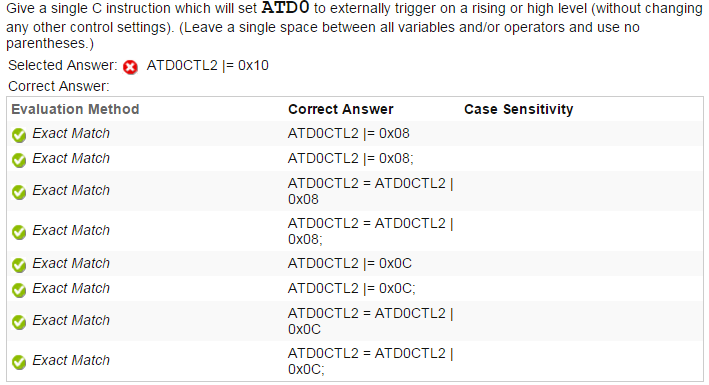


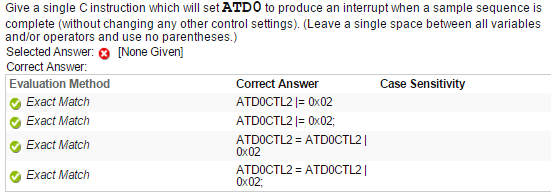


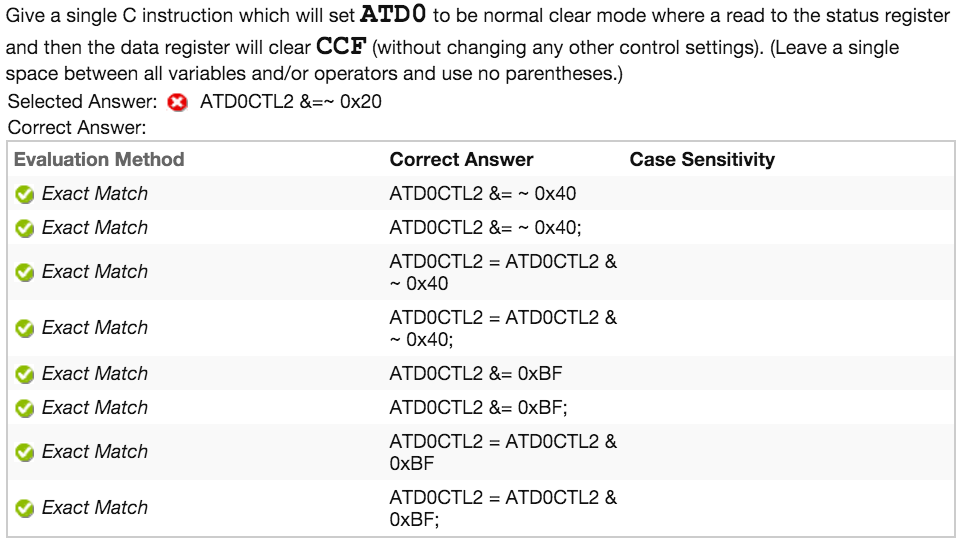


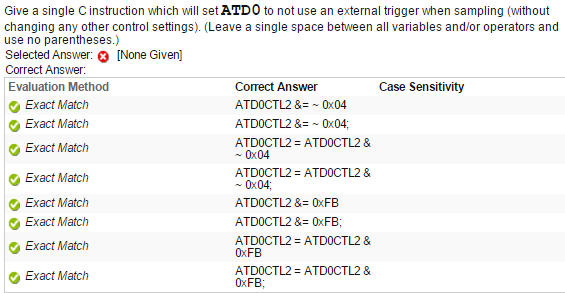


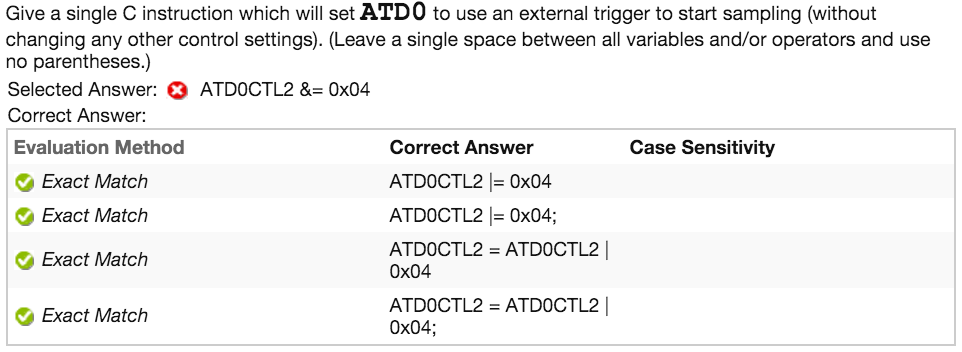




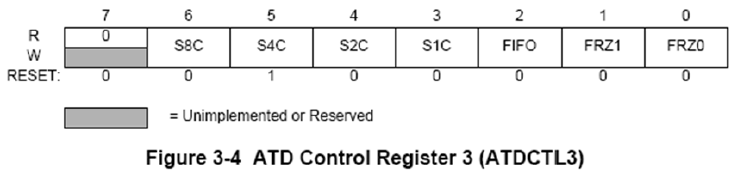


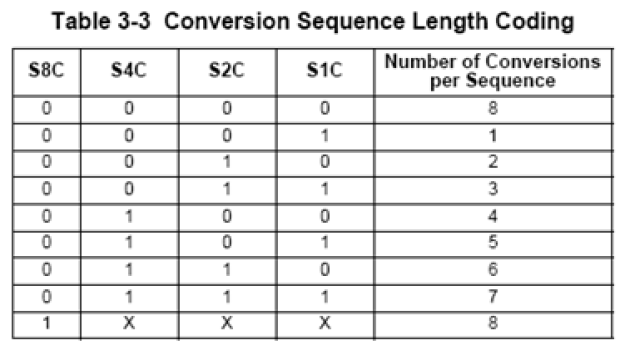






**HW Test 18: ADC-CTRL3**

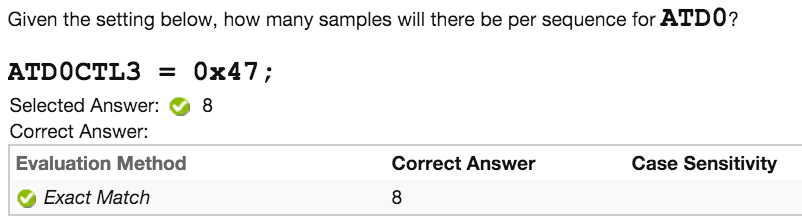




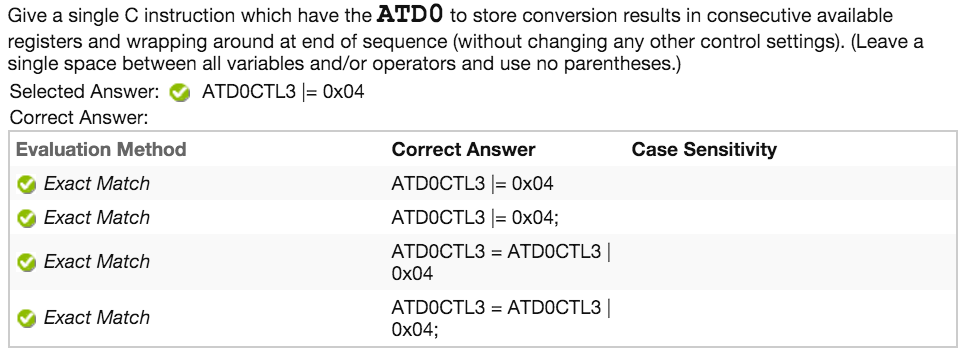
**If given ATDCTL3:**

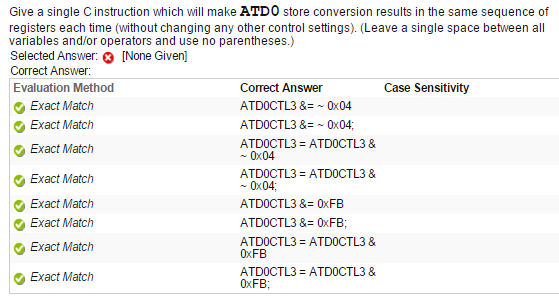
**Write out Hex number in Binary**

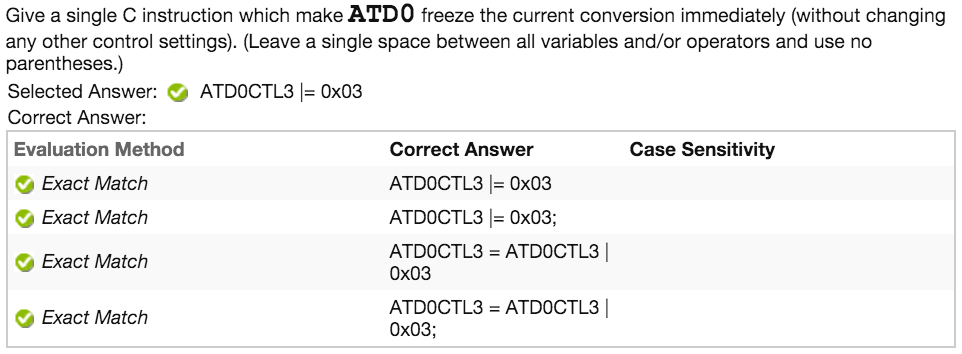
**Match bits 6:3 to table**



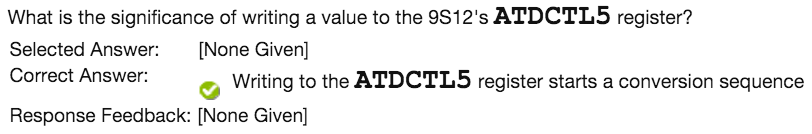
**Otherwise:**

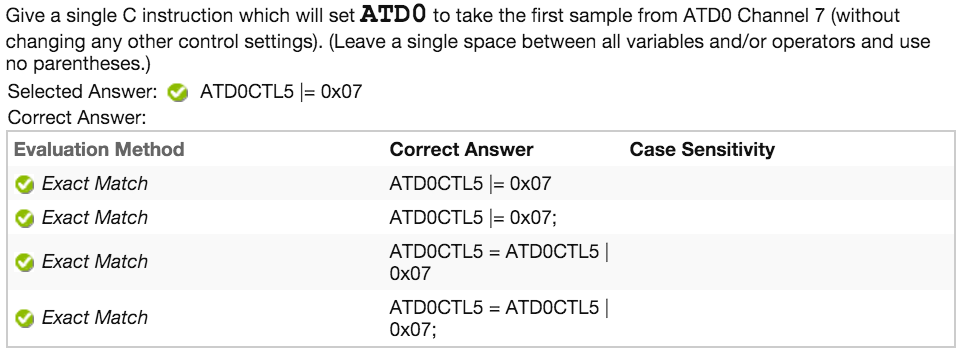


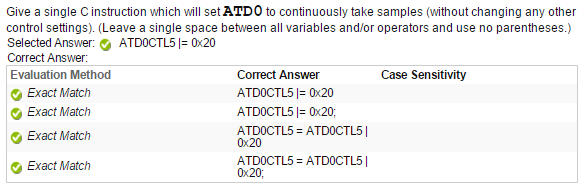


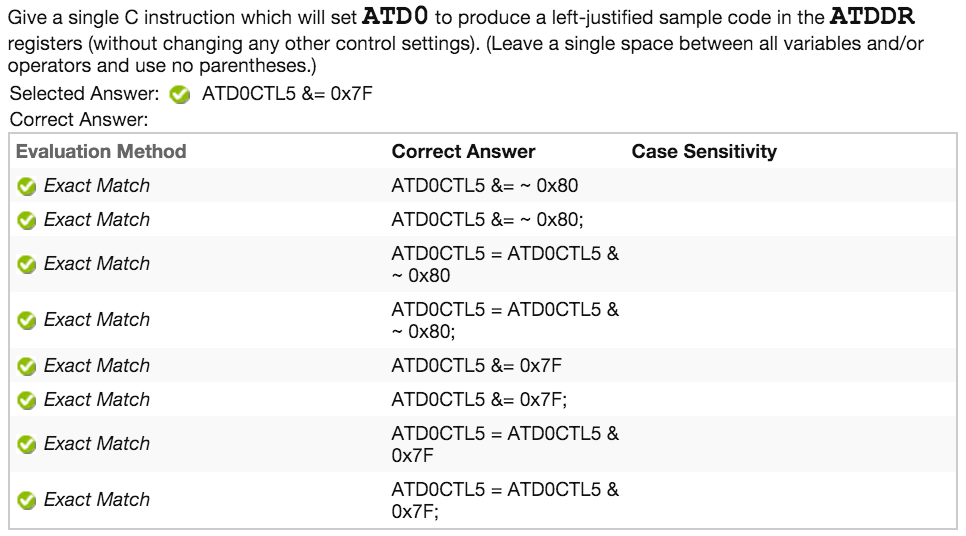


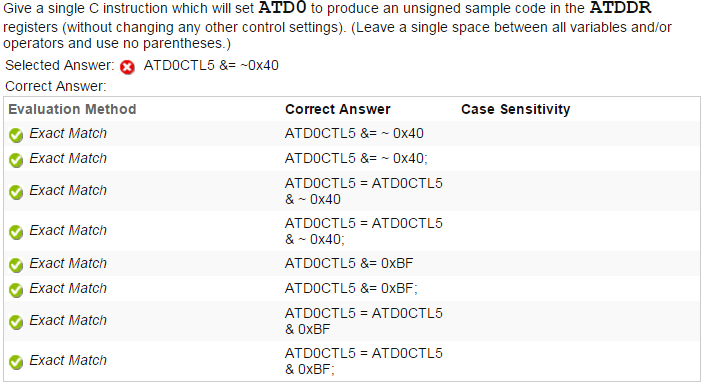
**HW Test 18: ADC-CTRL5**

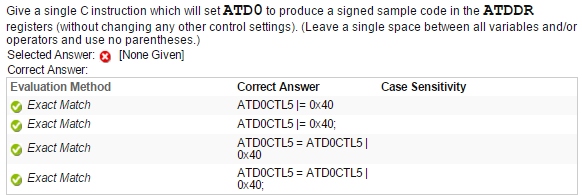


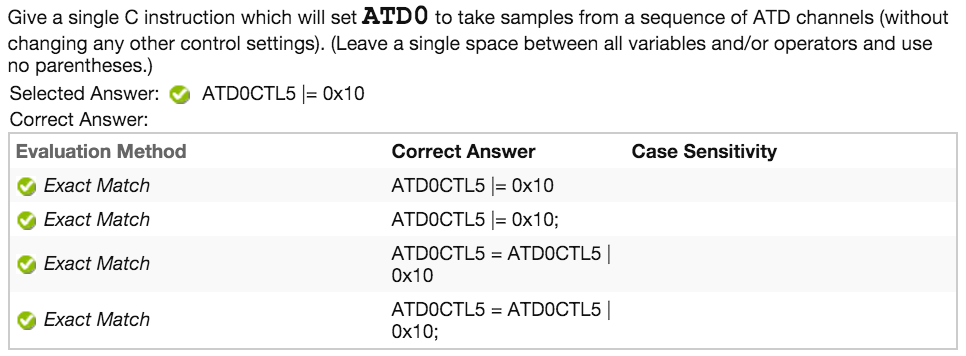


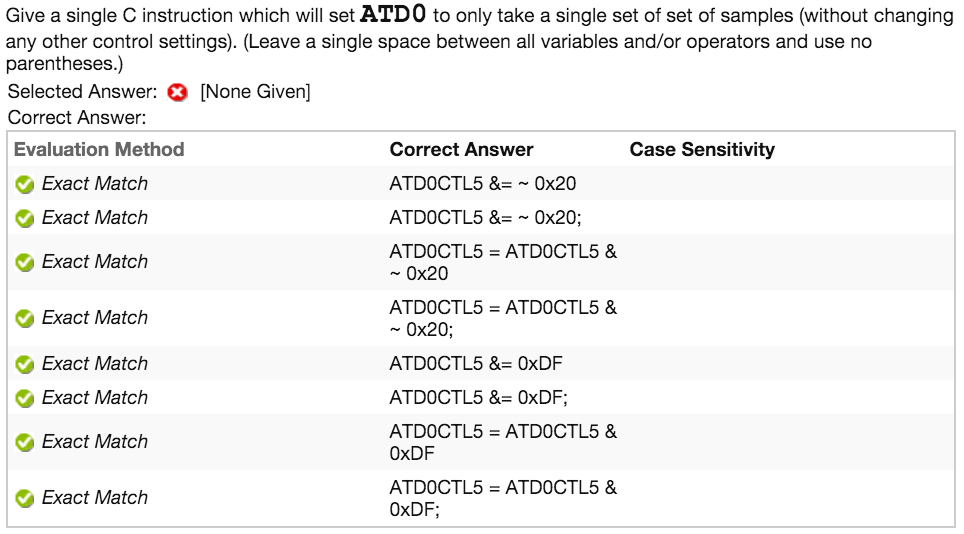




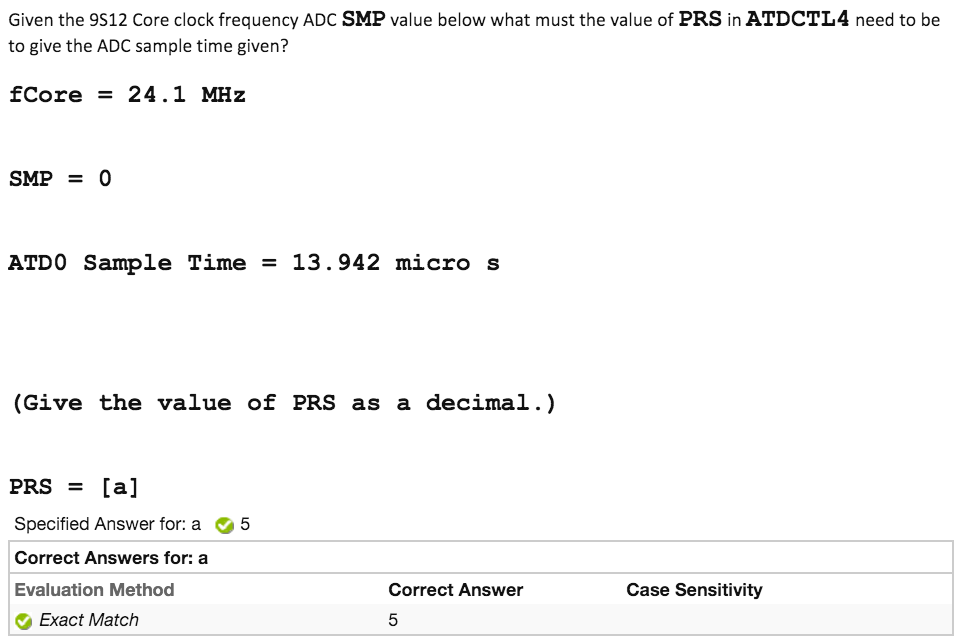








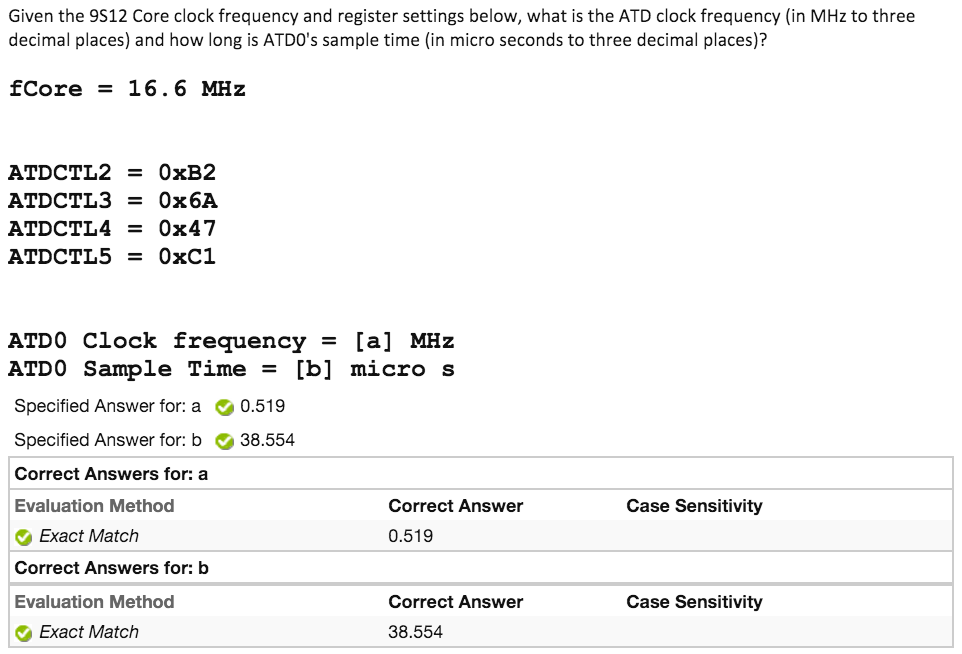
**HW Test 18: ADC-PRS**



**BusClock = CoreClock / 2**

**PRS = (((SampleTime \* BusClock) / (2^(SMP + 1) + 12)) / 2) - 1**

**HW Test 18: ADC-Sampling Time**



**BusClock = CoreClock / 2**

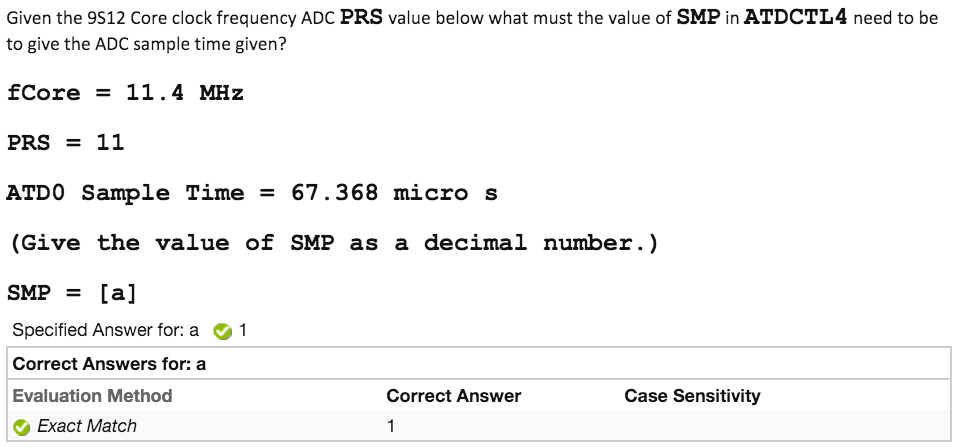
**PRS = ATD\_CTL4[4:0] in Decimal <--- Pay attention, 5 bits**

**ClockFrequency = BusClock / (2 \* (PRS + 1))**

**SMP = ATD\_CTL4[6:5] in Decimal**

**SampleTime = (1 / ClockFrequency) \* (12 + 2^(SMP+1))**

**HW Test 18: ADC-SMP**

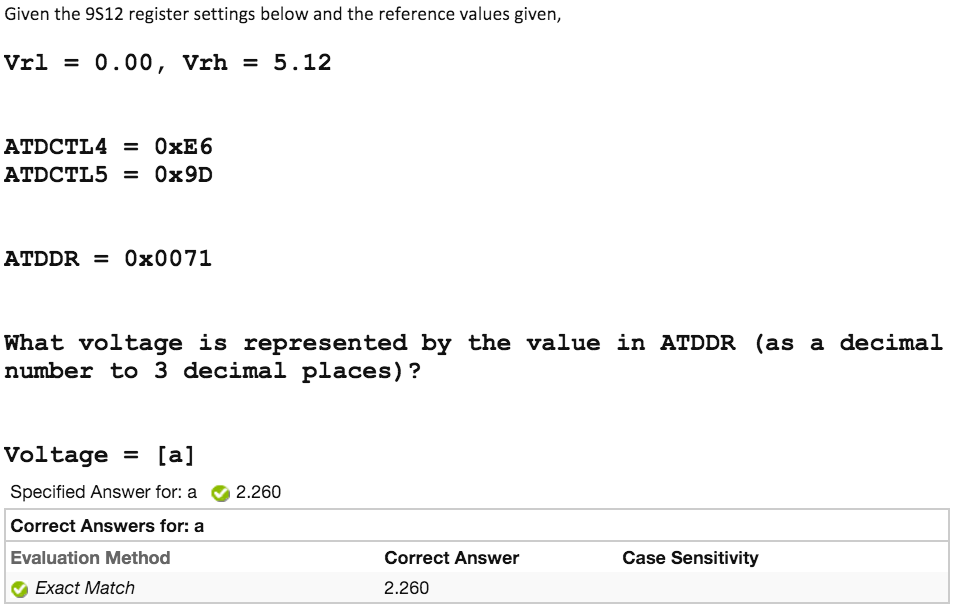


**BusClock = CoreClock / 2**

**SMP = log2[((SampleTime \* BusClock) / (2 \* (PRS + 1))) - 12] - 1**

**Round to nearest whole number**

**HW Test 18: ADC-Value**



**Vref = Span of Vr1 and Vrh**

**Bit Resolution:**

**10 bits if Bit 7 of CTL4 is 0**

**8 bits if Bit 7 of CTL4 is 1**

**ATDDR Bits:**

**Use Left bits if Bit 7 of CTL5 is 0**

**Use Right bits if Bit 7 of CTL5 is 1**

**Unsigned/Signed:**

**Unsigned if Bit 6 of CTL5 is 0**

**Signed if Bit 6 of CTL5 is 1**

**# = (Vref / 2^BitResolution) \* ATDDRBits(Decimal)**

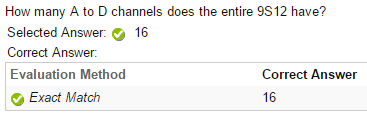
**If Unsigned**

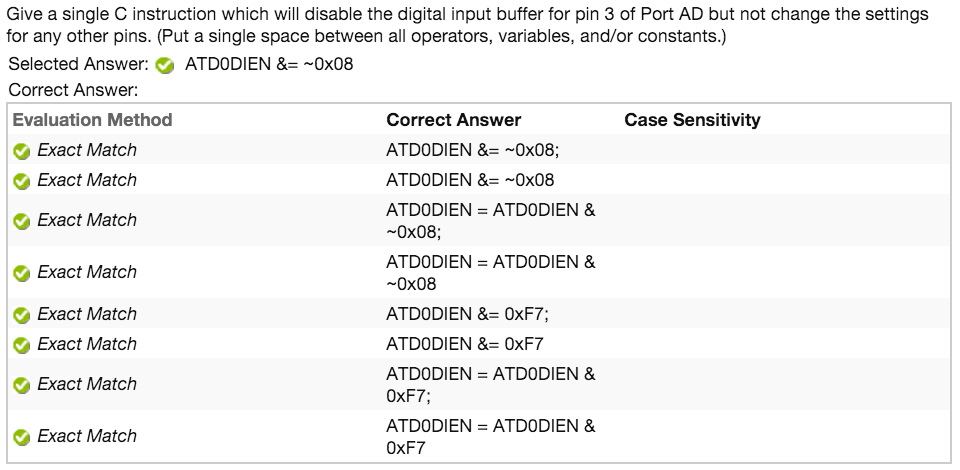
**Voltage = # + Vr1**

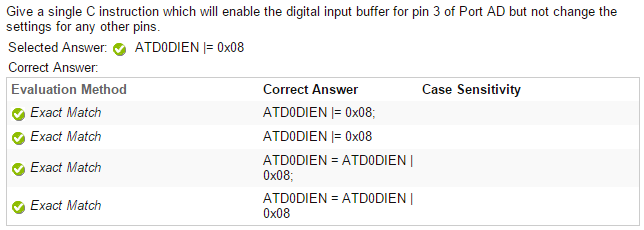
**If Signed**

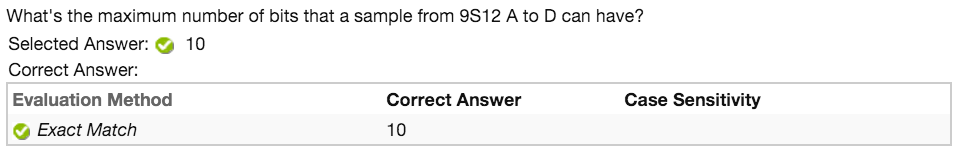
**Voltage = # + ((Vr1 + Vrh) / 2)**

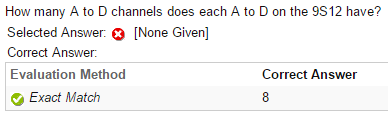
**HW Test 18: ADC-General**

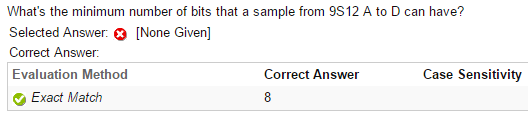


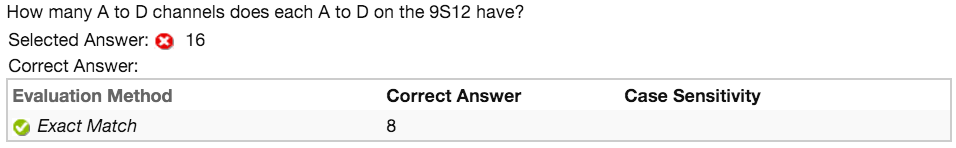


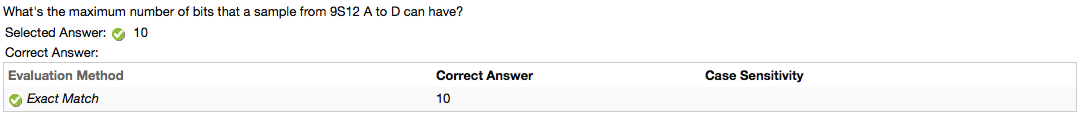




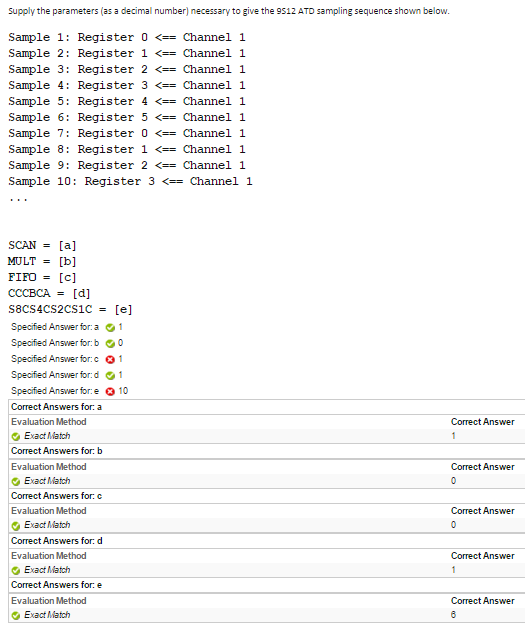








**HW Test 19: ADC-For Sequence**





**SCAN = 0 if “Stop” at bottom**

**SCAN = 1 if “...” at bottom**

**MULT = 0 if all channels are same**

**MULT = 1 if channels are different**

**MULT = 0 or 1 or X if only given 1 channel**

**FIF0 = 1 if registers count up to 7 then back to 0**

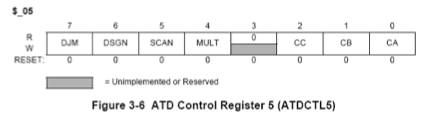
**FIF0 = 0 if registers count up but not to 7**

**FIF0 = 0 or 1 or X if only given 1 register or if registers do not turn back to 0**

**CCBCA = First channel in list**

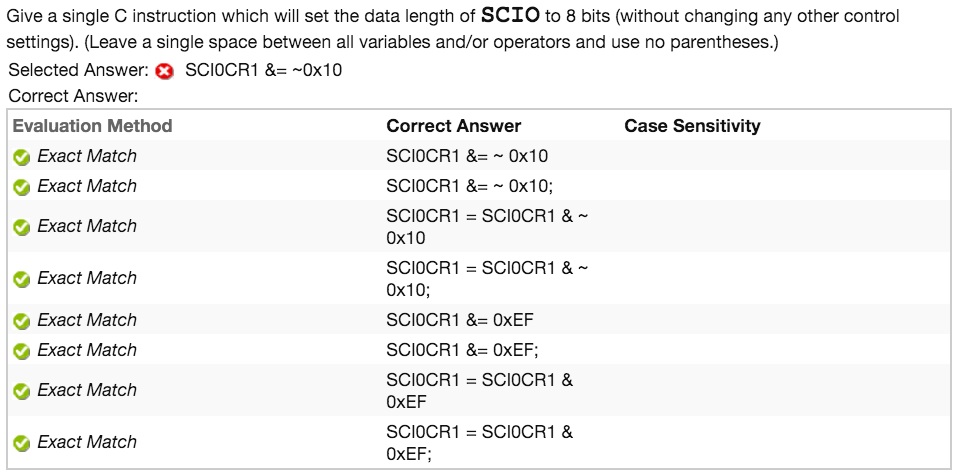
**S8Cs4CS2CS1C = # of unique registers**

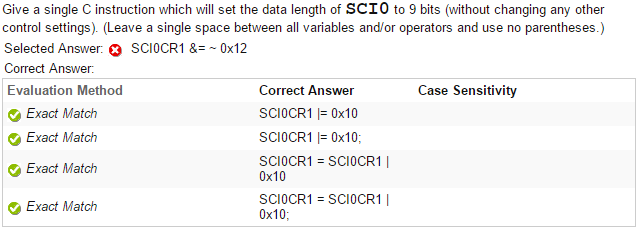
**HW Test 19: ADC-Sequence**

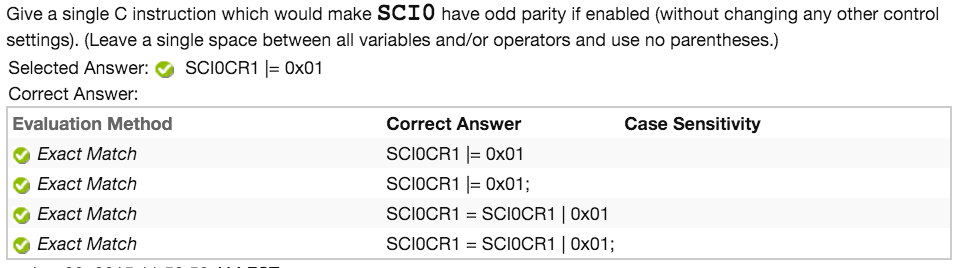


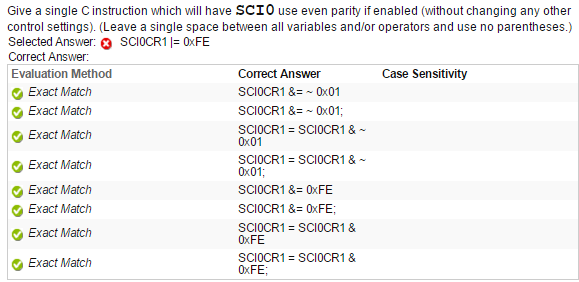
CAN’T DO

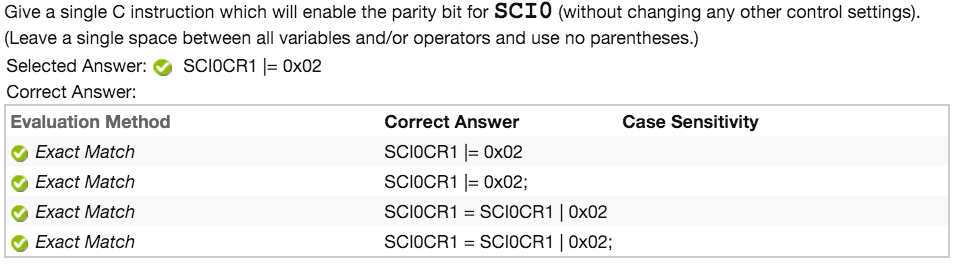
**HW Test 21: SCICR1**

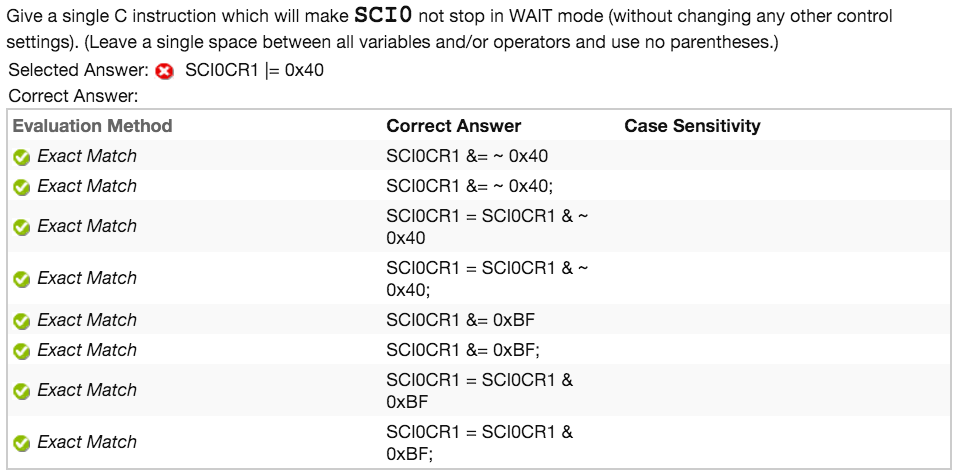


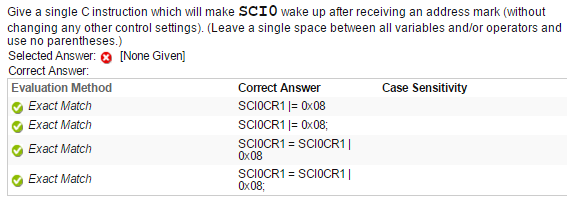


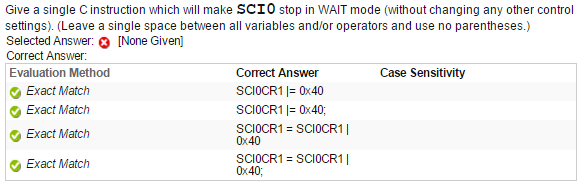


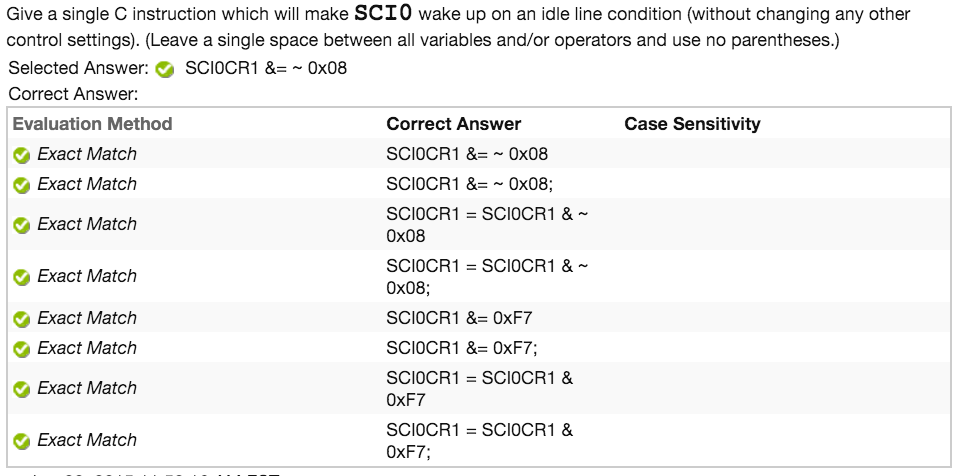


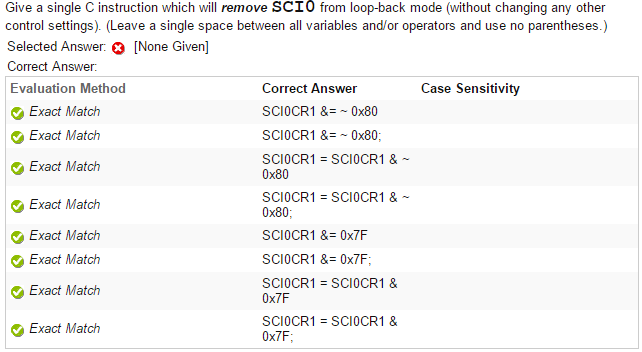




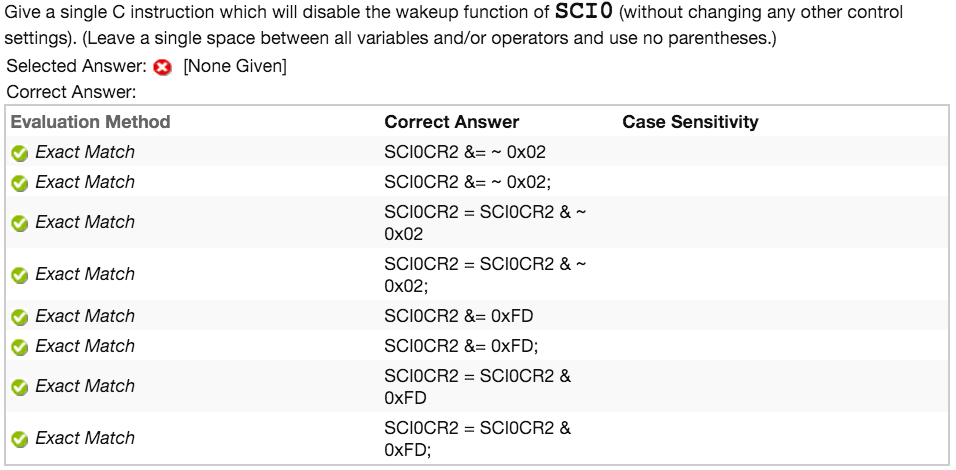


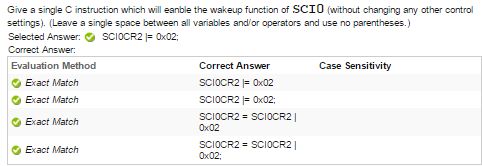


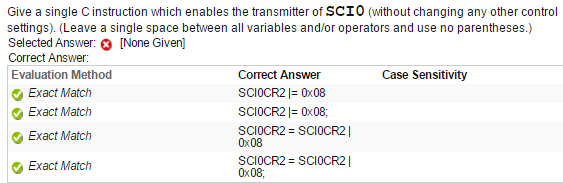


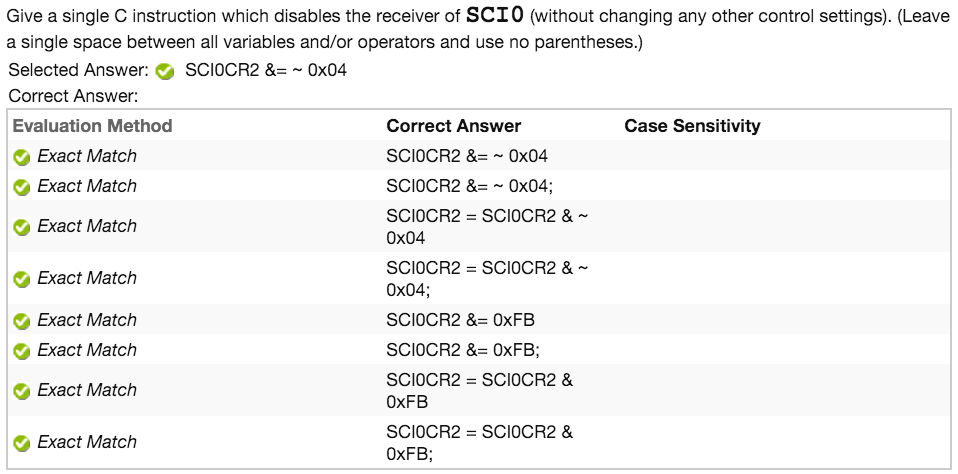


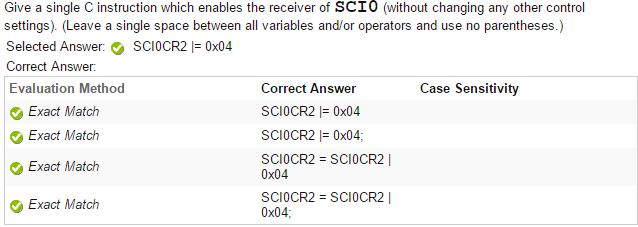
**HW Test 21: SCICR2**

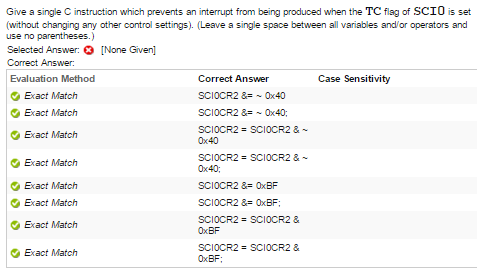


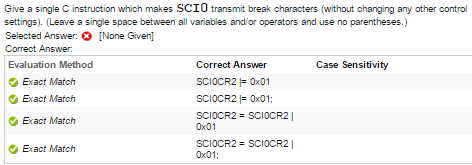


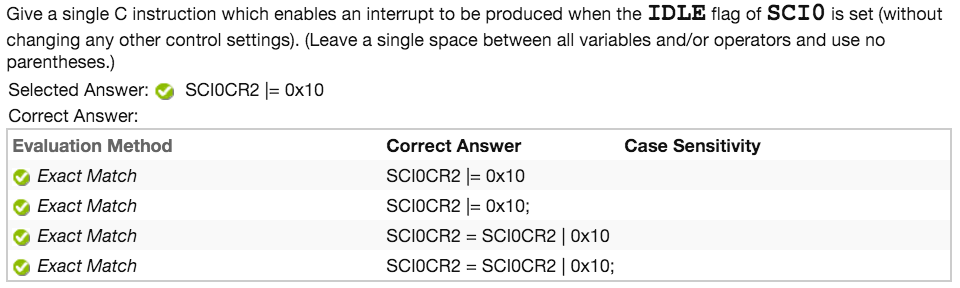


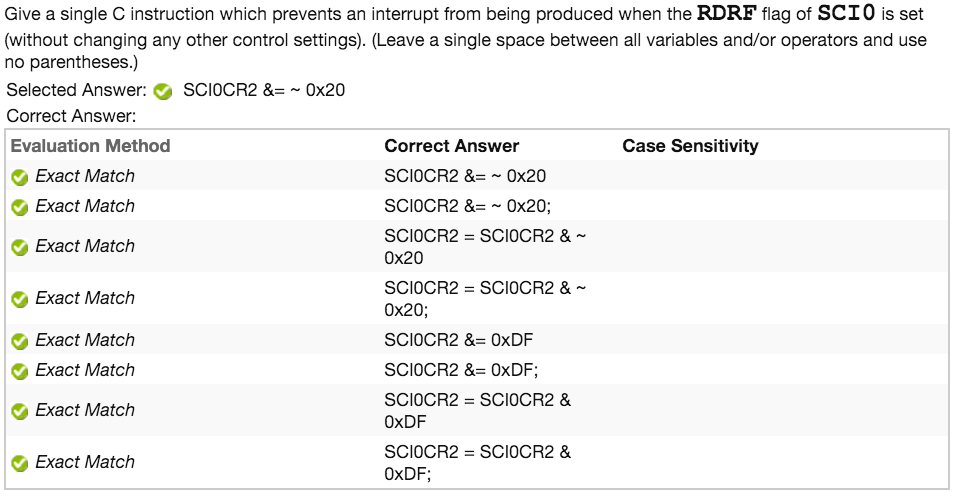




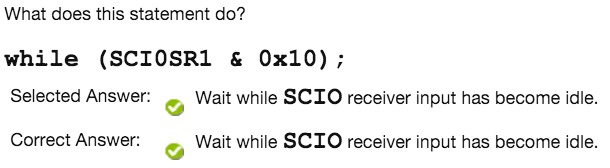


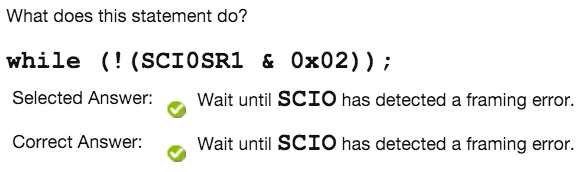


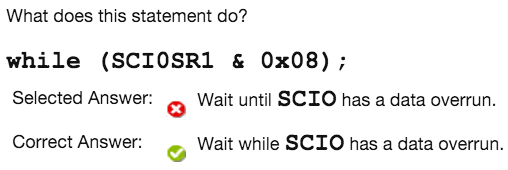


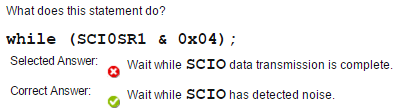


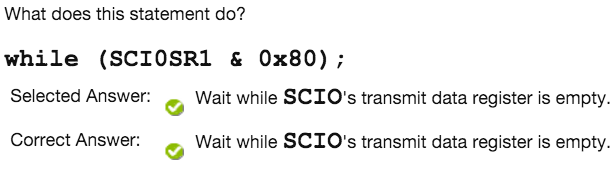
**HW Test 21: SCISR1**

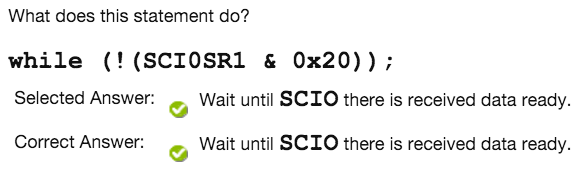


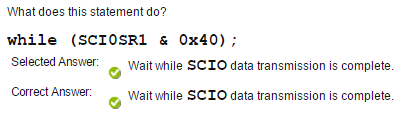


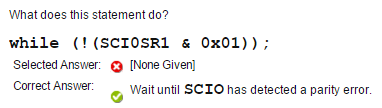








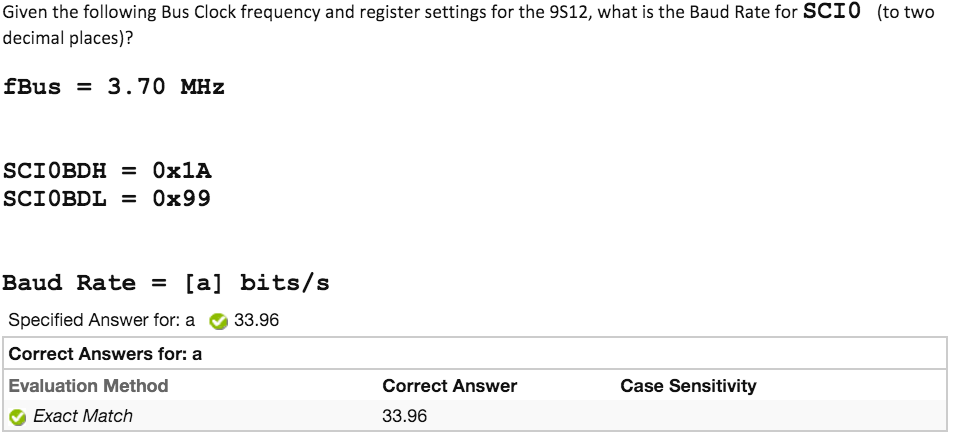




**! Indicates Wait Until**

**No ! Indicates Wait While**

**HW Test 21: SCI-What Is Baud**

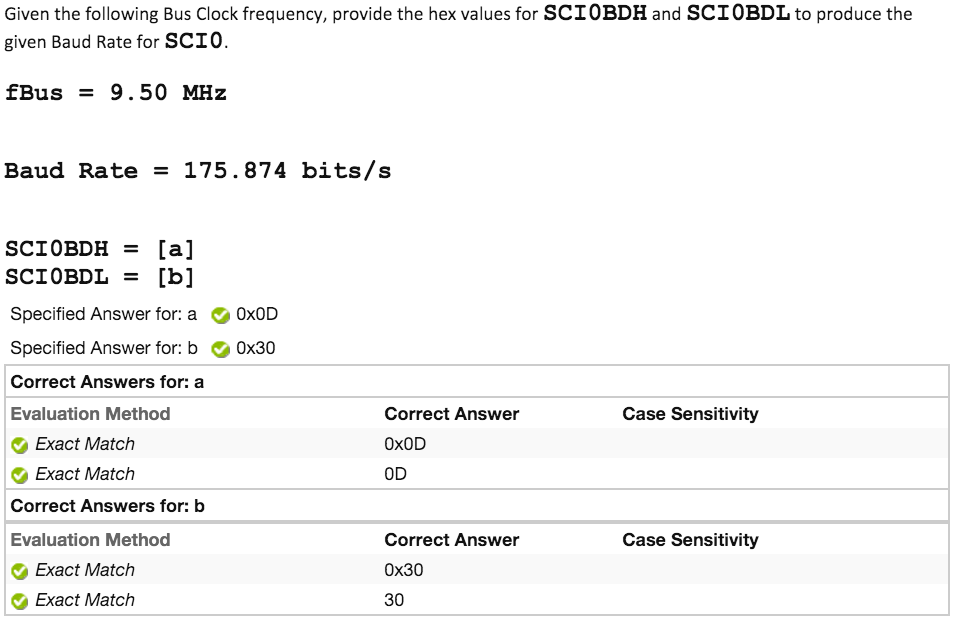


**BusClock = CoreClock / 2**

**BaudRateDivisor = BDH/BDL in Decimal i.e. 1A99 = 6809**

**BaudRate = BusClock / (16 \* BuadRateDivisor)**

**HW Test 21: SCI-What Is SCIBD**



**BusClock = CoreClock / 2**

**BaudRateDivisor = BusClock / (16 \* BaudRate)**

**Round BaudRateDivisor to Nearest Whole Number**

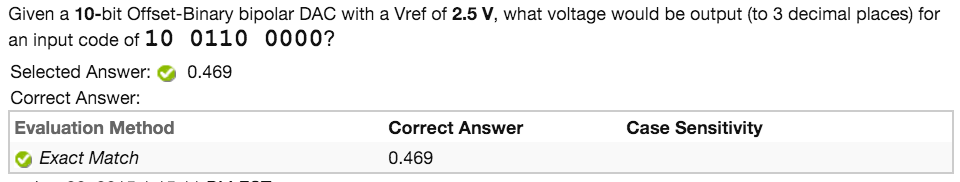
**Convert BaudRateDivisor to Hex**

**i.e. BaudRateDivisor = 0D30**

**BDH = First Two Hex Numbers**

**BDL = Second Two Hex Numbers**

**HW Test 21: SCI-What Is SCIBD**



**For Unipolar: (No known sightings of Unipolar)**

**Decimal = Input Code in Decimal**

**Voltage = (Vref / 2^Bits) \* Decimal**

**Pay attention to decimal points**

**For Bipolar:**

**Decimal = Input Code in Decimal**

**Voltage = ((Vref / 2^(Bits - 1)) \* Decimal) - Vref**

**Pay attention to decimal points**

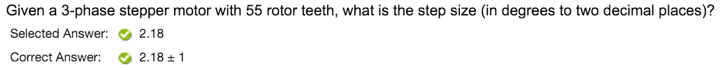
**Note: Negative answers are valid**

**HW Test 22: Stepper Motors - Stator Teeth**



**Rotor Teeth = Stator Teeth**

**HW Test 22: Stepper Motors - Stator Teeth**



**Degrees = 360 / (Phase \* Rotor Teeth)**

**ECE 371 Ethics Notes (Fall 2014)**

**· Professional Ethics**

**o There is an implicit contract between professionals and society.**

**o There is an assumption that engineers will act objectively in the best interest of the client, as long as what is requested is lawful and not harmful to others.**

**o There is an assumption that an engineer will perform tasks only in his or her area of expertise.**

**· Code of Ethics**

**o Statement of a code of ethics is an explicit recognition of the implicit contract between professionals and society**

**o Basis for moral criticism within a profession**

**o Not law, but relevant to tort ($$)**

**· IEEE Code of Ethics**

**1. To accept responsibility in making decisions consistent with the safety, health and welfare of the public, and to disclose promptly factors that might endanger the public or the environment**

**2. To avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;**

**3. To be honest and realistic in stating claims or estimates based on available data**

**4. To reject bribery in all its forms**

**5. To improve the understanding of technology, its appropriate application, and potential consequences**

**6. To maintain and improve our technological competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations**

**7. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others**

**8. To treat fairly all persons regardless of such factors as race, religion, gender, disability, age, or national origin**

**9. To avoid injuring others, their property, reputation, or employment by false or malicious action**

**10. To assist colleagues and co-workers in their professional development and to support them in following this code of ethics**

**· NSPE Code of Ethics**

**o Preamble: Engineering is an important and learned profession. As members of this profession, engineers are expected to exhibit the highest standards of honesty and integrity. Engineering has a direct and vital impact on the quality of life for all people. Accordingly, the services provided by engineers require honesty, impartiality, fairness, and equity, and must be dedicated to the protection of the public health, safety, and welfare. Engineers must perform under a standard of professional behavior that requires adherence to the highest principles of ethical conduct.**

**I. Fundamental Canons**

**1. Hold paramount the safety, health, and welfare of the public.**

**2. Perform services only in areas of their competence.**

**3. Issue public statements only in an objective and truthful manner**

**4. Act for each employer or client as faithful agents or trustees.**

**5. Avoid deceptive acts.**

**6. Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.**

**II. N/A**

**III. Professional Obligations**

**1. Engineers shall be guided in all their relations by the highest standards of honesty and integrity.**

**2. Engineers shall at all times strive to serve the public interest.**

**3. Engineers shall avoid all conduct or practice that deceives the public.**

**4. Engineers shall not disclose, without consent, confidential information concerning the business affairs or technical processes of any present or former client or employer, or public body on which they serve.**

**5. Engineers shall not be influenced in their professional duties by conflicting interests.**

**6. Engineers shall not attempt to obtain employment or advancement or professional engagements by untruthfully criticizing other engineers, or by other improper or questionable methods.**

**7. Engineers shall not attempt to injure, maliciously or falsely, directly or indirectly, the professional reputation, prospects, practice, or employment of other engineers. Engineers who believe others are guilty of unethical or illegal practice shall present such information to the proper authority for action.**

**8. Engineers shall accept personal responsibility for their professional activities, provided, however, that engineers may seek indemnification for services arising out of their practice for other than gross negligence, where the engineer's interests cannot otherwise be protected.**

**9. Engineers shall give credit for engineering work to those to whom credit is due, and will recognize the proprietary interests of others.**