

Einen wunderschönen guten Morgen und herzlich willkommen zur nächsten Vorlesung bei den Rechnerstrukturen. Um ganz ehrlich mit Ihnen zu sein, bei mir ist es nicht morgen, bei mir ist es ungefähr 13.30 Uhr. Gespräch vor der Sendung aufgezeichnet. Ich habe leider dringende Termingründe zur Zeit, wo Sie sich befinden, also Freitagvormittag und musste deshalb dieses Stück Vorlesung aufzeichnen. Allerdings soll für Sie kein Nachteil entstehen. Das heißt, dieser Stream wird ganz normal von den Mitarbeitern mitbetreut. Sie können wie gewohnt Ihre Fragen stellen. Sie können auch im Anschluss an die Vorlesung Fragen stellen. Dafür sind dann die Mitarbeiter Wirklich live. Okay, also wir sind stehen geblieben im Kapitel Alu-Design und wir wollen einfach uns eine schöne 32-Bit-Alu bauen und wollen uns dann Möglichkeiten betrachten, wie man schnelle, also Alus mit großer Performance bauen kann. Wir wollen einen Blick werfen auf eine schnelle Addition und wir wollen einen Blick werfen auf die schnelle Multiplikation. bit alu und sind also stehen geblieben hier wir haben aufgebaut bis jetzt also das sind die zwei eingänge 2 1 bit eingänge a und b das sind carry in eingang von der stufe vorher wenn wir dann eine größere alu bauen also eine 32 bit alu ist ja unser ziel das ist der carry out das ist also der übertrag in die nächste stufe und hier haben wir einen multiplexer 4 Operationen implementiert und zwar ist die Operation 0 das logische UND, die Operation 1 ist das logische ODER, die Operation 2 ist das Ergebnis der 1-Bit-Addition und die Operation 3 ist das Setzen von der 0. Dieses Ergebnis können Sie hier am Result ablesen und angesteuert welches Ergebnis durchgeschaltet wird, wird eben über diese Kontrollleitungen, die hier Wir erweitern jetzt diese 1-Bit-Alu zu einer 32-Bit-Alu. Das ist eigentlich nicht allzu viel zu tun, weil diese einzelnen kleinen Alus, der Alu 0 bis Alu 31, die sind im Wesentlichen identisch. Das heißt, die haben alle als Eingänge im Carry-In 2-1-Bit-Eingänge A0 und B0. Sie haben natürlich auch die Steuerleitungen anliegen, die Operation. die Result-Leitung und es wird ja in der ALU intern der Carry verarbeitet und dieser Carry-In wird als der verarbeitete Carry-In, wird dann als Carry-Out in die nächste Stufe weitergegeben und wird dann hier wieder als Carry-In anliegen. Also von links nach rechts läuft alles erstmal schön parallel, aber von oben nach unten läuft es sequenziell, das heißt ich muss erst den Carry-In in der Stufe 0 berechnen, damit ich dann den nächsten Carry-Out berechnen in der Stufe 1, damit ich ihn in Stufe 2 eingehen lassen kann und so weiter.