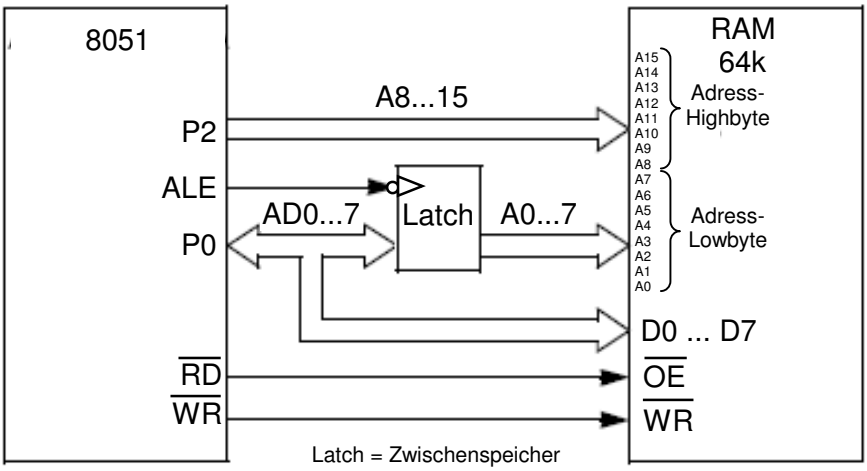
Friedrich-Ebert-Schule Esslingen	MIKROCONTROLLER	Name:
2.5.1.1	Anschluß von externem Arbeitsspeicher	Datum:

Der 8051 kann an seinen externen Anschlüssen 64kByte RAM-Speicher ansteuern. Die Speicheradressen eines RAM-Bausteins werden in Hex-Schreibweise angegeben. Somit ergeben sich bei voller Ausnutzung des **Adressraumes** die Adressen 0000h – FFFFh. Zur Auswahl der Speicherbytes werden 16 Adressleitungen benötigt, denn 2^{16} = 65536 = 64k. Die Adressleitungen sind durchnummeriert von A0 ... A15 und werden als **Adressbus** bezeichnet. Ist eine Speicheradresse angewählt, erfolgt die Übertragung der Daten über den **Datenbus** (8 Bit). Dazu wird noch jeweils eine Steuerleitung für das Lesen der Daten aus dem RAM (RD/) und eine für das Schreiben (WR/) von Daten ins RAM benötigt. Zusammen wären das 26 Leitungen oder 3¼ IO-Ports. Um Portleitungen des 8051 zu sparen, wird deshalb das niederwertige Adressbyte (A0 ... A7) und das Datenbyte (D0 ... D7) zeitlich nacheinander über den Port 0 ausgegeben (**Zeitmultiplex**). Damit wird ein 8-Bit Port eingespart.

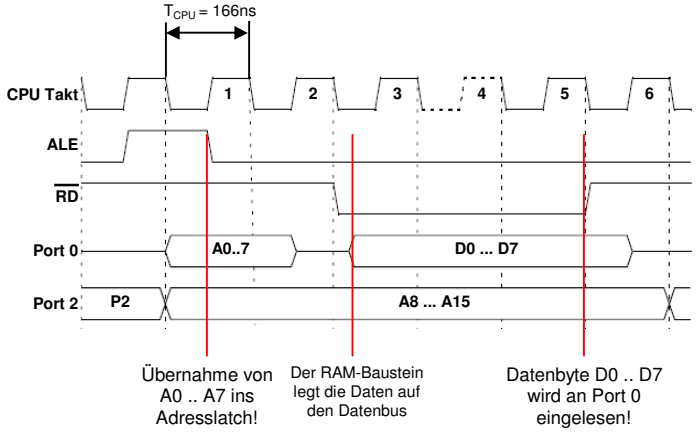
Der gemultiplexte Adress- und Datenbus muss aber außerhalb des 8051 wieder getrennt werden. Vom Controller wird dazu ein Steuersignal (ALE) erzeugt, das anzeigt, wann die Adressdaten A0...A7 gültig sind.
Das ALE-Signal (**A**dress **L**atch **E**nable) steuert einen Zwischenspeicher (Latch), in den das Adress-Lowbyte übernommen wird und dann während des gesamten Schreib-/Lesezugriffs an den Anschlüssen A0...A7 des RAM-Bausteins ansteht. Die Impulsdiagramme zeigen die zeitlichen Abläufe eines lesenden und eines schreibenden Speicherzugriffs:



Ein Speicherzugriff benötigt 6 CPU-Taktzyklen. Dies entspricht 12 Quarztakten (Taktteilung). Bei 12 MHz ergibt sich eine Zugriffszeit von 1µs.

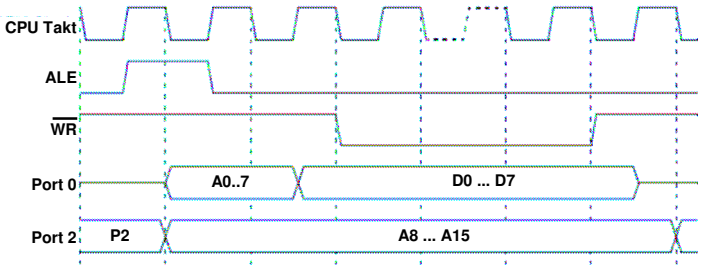
Bei der fallenden Flanke des ALE-Signal ist das Adress-Lowbyte auf Port 0 gültig. Der Controller legt anschließend Port 0 hochohmig. Die fallende Flanke des RD/-Signals veranlasst dann den Speicher-Baustein die Daten auf den Datenbus zu legen. Mit steigender Flanke werden die Daten in den Controller eingelesen.

Lesen von Daten aus dem RAM



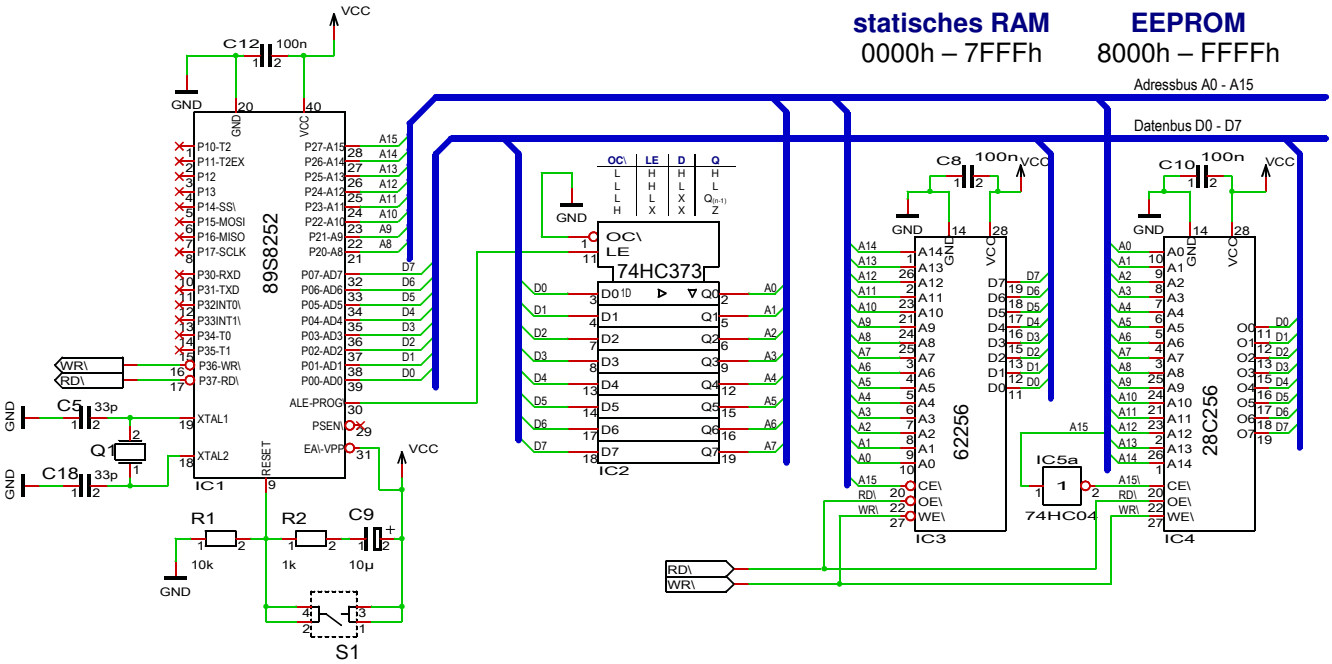
Das Schreiben von Daten in den Speicher-Baustein läuft ähnlich. Die Daten liegen jedoch während der kompletten Dauer des WR/-Signals an Port 0 an. Das Speichern auf der gewählten Adresse im RAM kann somit bei fallender oder steigender Flanke oder bei 0-Pegel des WR/-Signals erfolgen.

Schreiben von Daten ins RAM



Friedrich-Ebert-Schule Esslingen	**MIKROCONTROLLER**	Name:
2.5.1.2	**Anschluß von externem Arbeitsspeicher**	Datum:

Die Schaltung zeigt den Anschluß von zwei Speicherbausteinen an einen 8051-Controller. Als Adresslatch wird häufig der TTL-Baustein 74LS373 oder 74LS573 mit 8 D-Flipflops verwendet.



Der Adressraum ist in zwei 32k große Adressbereiche aufgeteilt. Diese Bereiche sind vollständig von je einem statischen RAM (62256) und einem EEPROM (28C256) belegt. Beide Speicherbausteine haben 15 Adressleitungen (A0...A14). Die Leitung A15 wird als Bausteinauswahl-Signal (Chip-Select) verwendet. Da die Chip Enable-Eingänge der Speicherbausteine Nullaktiv sind, muss beim EEPROM die Leitung A15 invertiert werden. Die Tabelle zeigt die Adressbereiche für beide Speicherbausteine.

		A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	Hex
RAM	Startadresse	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000h
	Endadresse	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	7FFFh
EEPROM	Startadresse	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000h
	Endadresse	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	FFFFh

A15: Chip-Select

$2^{15} = 32768$ ☐ 32k Adressen je Baustein

Der Zugriff auf den externen RAM erfolgt mit dem MOVX-Befehl. Dabei kann der komplette 64k-Adressraum mit dem Datapointer (DPTR) adressiert werden. Um den Externen RAM nutzen zu können, muss das interne XRAM deaktiviert und der externe Daten- und Adressbus aktiviert werden. Dazu wird Bit EXTRAM im Spezielle Funktionen Register AUXR auf 1 gesetzt:

7	6	5	4	3	2	1	0	**AUXR** Auxiliary Register (8Eh)
DPU	**-**	**M0**	**XRS2**	**XRS1**	**XRS0**	**EXTRAM**	**AO**	Resetwert: **0X00 1010**b (nicht bitadressierbar)
	Reserve		**XRAM Size**					**ALE Output Bit** 0: ALE-Impulsfrequenz ist konstant 1/6 * f_{CLK} 1: ALE-Impulse nur während MOVX- oder MOVC-Befehlen
						External RAM Bit 0: Zugriff auf internes XRAM mit MOVX und MOVC-Befehlen 1: Zugriff auf externen Speicher aktiviert		
			0	0	0	256 Bytes		
			0	0	1	512 Bytes		
			0	1	0	768 Bytes (default)		
			0	1	1	1024 Bytes		
			1	0	0	1792 Bytes		
		Pulse lenght 0: Pulsdauer des WR/ und RD/-Signals ist 6 * T_{clk} (bei 12MHZ → 500ns) 1: Pulsdauer von WR/ und RD/ wird auf 30 * T_{clk} verlängert! (2,5µs)						
Disable Weak Pull-Up 0: schwache Pull-Ups (20kΩ) von Port 1 bis Port 3 sind immer aktiv. 1: Pull-Ups werden beim Schreiben von 0 in den Port deaktiviert! (Stromsparfunktion)								

Nachfolgend sind Code-Beispiele für das Lesen und Schreiben von Daten im externen RAM/EEPROM gezeigt.

Beispiel für Schreiben von Daten in die externe RAM-Adresse 20FCh

```
            Daten   data    7Fh

init:       mov     auxr,#00000011b       ; Externer Datenbus aktivieren
                                          ; ALE-Signal nur während WR/ und RD/
writeRAM:   mov     a,Daten               ; Daten aus internem RAM holen
            mov     dptr,#20FCh           ; Speicheradresse  für externes RAM
            movx    @dptr,a               ; Daten in externes RAM speichern
```

Beispiel für Lesen von Daten aus der externe EEPROM-Adresse 8000h

```
            Daten   data    7Fh

init:       mov     auxr,#00000011b       ; Externer Datenbus aktivieren
                                          ; ALE-Signal nur während WR/ und RD/
readEEPROM: mov     dptr,#8000h           ; Speicheradresse  für externes EEPROM
            movx    a,@dptr               ; Daten aus externem EEPROM in den Akku holen
            mov     Daten,a               ; Daten im internen RAM speichern
```

Beispiel für Schreiben von Daten in die externe EEPROM-Adresse E000h

```
            Daten   data    7Fh

init:       mov     auxr,#00000011b       ; Externer Datenbus aktivieren
                                          ; ALE-Signal nur während WR/ und RD/
writeEEPROM: mov    a,Daten               ; Daten aus internem RAM holen
            mov     dptr,#E000h           ; Speicheradresse  für externes RAM
            movx    @dptr,a               ; Daten in externes RAM speichern
            lcall   wait_busy             ; EEPROM benötigen in der Regel eine
                                          ; Wartezeit, bevor nach einem Schreibvorgang
                                          ; wieder Daten gelesen oder
                                          ; geschrieben werden dürfen. (Siehe Datenblatt)
```