	Name :	Klasse : TGI-J2
Fach : ITH-Labor	Projekt1: Ampelsteuerung-Softwarelösung in der C-Sprache	Lehrer. : Paul Fansi StR./Dipl. Ing. M.Sc.
Datum : 20.09.21		Erstelldatum: 03.09.2021

Informationen:

Beim unteren Projekt handelt es sich um den Entwurf und die Simulation des Ampelverkehrs.

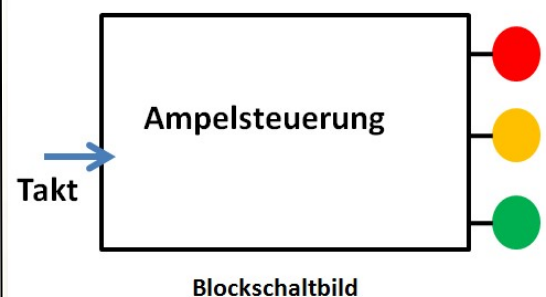
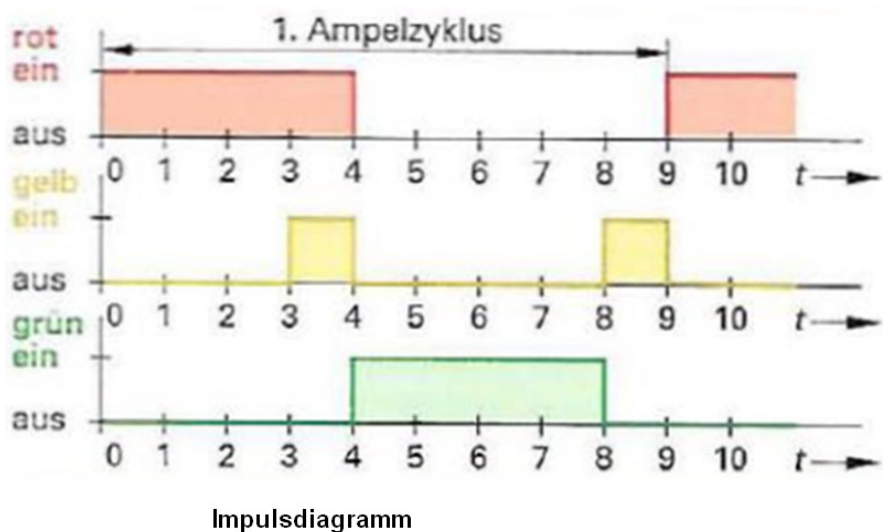
Die Hardwarelösung wurde bereits von mir im Unterricht Schuljahr 20/21 mit Ihnen ausführlich besprochen (siehe Teil I unten). Sie müssen also den Teil I nicht machen; sollten Sie aber Fragen dazu haben, so würde ich empfehlen zunächst den Lösungsvorschlag vom letzten Jahr zu analysieren, und erst wenn Sie nicht weiterkommen, dürfen Sie mich fragen.


Der Teil II ist Pflicht. Hierbei geht es um die **Softwarelösung in der C-Sprache**. (Lösungsvorschlag in der Assemblersprache wurde letztes ausführlich besprochen und simuliert).

Teil I: Hardwarelösung

Entwerfen Sie ein Schaltwerk (SW), das die zeitliche Folge der Ampelphasen einer Verkehrsampel (siehe unteres Bild) simuliert. Die entworfene Schaltung soll mit D Flipflop realisiert und mit Proteus simuliert werden. Gehen Sie dabei folgendermaßen vor:

1. Definieren Sie die Aus- und Eingangsvariablen
2. Bestimmen Sie die Anzahl der Zustände und legen und legen Sie den Anfangszustand fest
3. Bestimmen Sie die Anzahl des Flipflops (wir gehen von einem SW mit vorhandenem Ausgangsschaltnetz aus)
4. Zeichnen Sie den Zustandsgraphen des zu entworfenen Schaltwerks
5. Geben Sie die Zustandsfolgetabelle des Schaltwerks an
6. Erstellen Sie die Funktionstabelle des Ausgangsschaltnetzes sowie die Gleichung fürs Ausgangsschaltnetz in DNF
7. Erstellen Sie die Funktionstabelle des Rückführungsnetz (Übergangsschaltnetz) sowie die Gleichung dieses Schaltnetzes in DNF
8. Bauen Sie die Schaltung mit Proteus auf und simulieren Sie diese.



	Name :	Klasse : TGI-J2
Fach : ITH-Labor Datum : 20.09.21	Projekt1: Ampelsteuerung-Softwarelösung in der C-Sprache	Lehrer. : Paul Fansi StR./Dipl. Ing. M.Sc. Erstelldatum: 03.09.2021

Teil II: Softwarelösung

Nutzen Sie die im Unterricht erworbenen Kenntnisse zum Thema „**Realisierung einer Verzögerung mit dem Timer**“, und Ihre bisher Erfahrungen und Kenntnisse, um die Ampelverkehr-Aufgabe **in der C-Sprache** und **in der Proteus-Umgebung** so zu programmieren, dass die unteren Ampelphasen die erwähnte Dauer bekommen:

- 1) Ampelphase „rot“ : Dauer = 2s.
- 2) Ampelphase „rot_gelb“ : Dauer = 1s.
- 3) Ampelphase „grün“ : Dauer = 4s.
- 4) Ampelphase „gelb“ : Dauer = 1 s

Hinweise:

1. Ihre Lösung muss mindestens auf zwei Teile bestehen (modularer Aufbau):
 - (1) ein Hauptprogramm **main()**, welches die Deklarationen, die Ampelphase steuert und ein Unterprogramm **DELAY()**, welches die Verzögerung zwischen den Ampelphasen erzeugt aufruft.
 - (2) Ein Unterprogramm **DELAY ()** welches die Aufgabe aus (1) erledigt.
 Beim **DELAY ()** müssen zwei Lösungen angeboten werden:
 - ✓ Eine Lösung bei, der die Verzögerung durch Zählschleife realisiert wird.
 - ✓ Eine Lösung bei, der die Verzögerung durch **Polling des Timerflags TF1** realisiert wird.