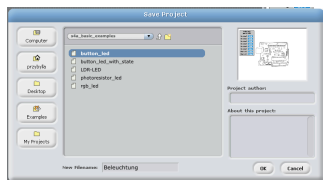
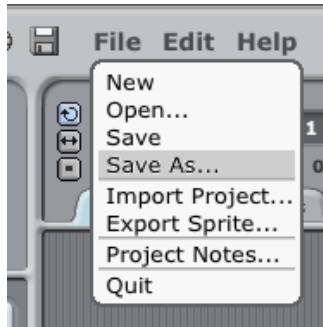


Einführung in Physical Computing mit S4A und My Interactive Garden

Projekt 1: Gartenbeleuchtung



Zu allererst...

- Datei – neu
- Datei – speichern unter...
- Speicherort auswählen
- Dateiname angeben (Beleuchtung)
- Ok klicken

→ ein neues Projekt wurde erstellt und unter dem Namen Beleuchtung gespeichert

Themen:

- Neues Projekt erstellen
- Projekte speichern
- Projekte öffnen

Projekt öffnen in diesem Zusammenhang thematisieren, dabei wichtig: unbedingt vorher speichern, da aktuelles Projekt beim Öffnen eines neuen Projektes geschlossen wird und nichtgespeicherte Änderungen unwiderruflich verloren gehen!

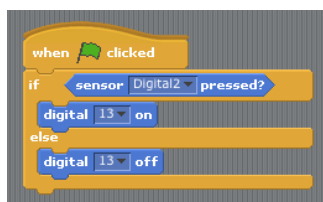


LED an Ausgang „digital13“ anschließen, Skript erstellen. Zweite LED an Ausgang „analog5“ anschließen, zweites Skript erstellen

Themen:

- Kontrollblöcke „When green flag clicked“, „stop all“
- LED – ein Aktor. Was sind Aktoren (im Baukasten)?
- Aktoren anschließen: analog (0..255) oder digital(0,1)? Wann nutze ich was? Was passiert, wenn man die Werte am analogen Ausgang verändert (Testwerte z.B. 0, 20, 150, 255)?

Weiterführendes Problem: Wir wollen unsere Lampe mit einem Taster ein- und ausschalten können und nicht dauerhaft leuchten lassen. Wie können wir einen Taster anschließen und verwenden?

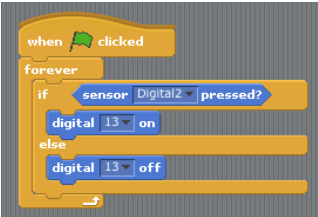
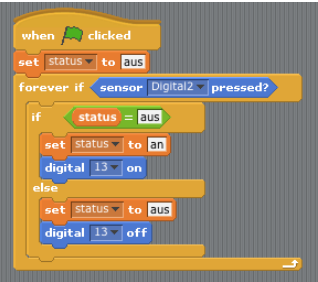
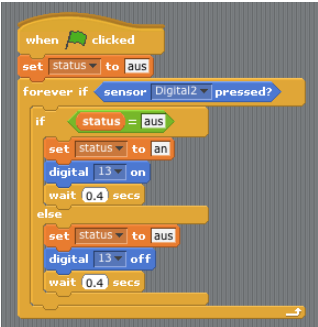


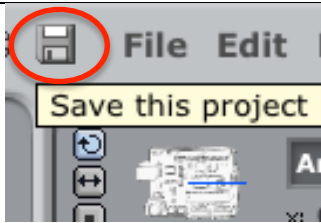
Taster an Eingang „digital2“ anschließen.

Wenn Taster gehalten wird, leuchtet die LED, sonst (= wenn Taster losgelassen wird), leuchtet LED nicht

Themen:

- Kontrollblock „if-else“: Entscheidung (vgl. mit Kontrollblock „if“)
- Taster – ein Sensor. Was sind Sensoren (im Baukas-

	<p>ten)?</p> <ul style="list-style-type: none"> - Sensoren anschließen: analog (0..1023)/digital(0,1)? Wann nutze ich was? Welche analogen Werte werden beim Taster gelesen? <p>Weiterführendes Problem: Wenn ich meine Lampe mit Programmstart ein- oder ausschalten kann, ist das ja immerhin schon etwas. Aber ich würde gern auch zwischendurch das Licht im Garten an- bzw. ausschalten, ohne jedes Mal das Programm neustarten zu müssen. Wie geht das?</p>
	<p>Solange das Programm läuft gilt: Wenn Taster gehalten wird, leuchtet die LED, ansonsten (= wenn Taster losgelassen wird), leuchtet LED nicht</p> <p>Themen:</p> <ul style="list-style-type: none"> - Kontrollblock „forever“ - Schleife <p>Weiterführendes Problem: Wenn ich den garten betrete, kann ich nicht die ganze Zeit am Lichtschalter stehen bleiben. Wie kann erreicht werden, dass die LED auf Tastendruck an- und ausgeschaltet wird und in ihrem Zustand verharrt, bis das nächste Mal die Taste gedrückt wird?</p>
	<p>Solange das Programm läuft gilt: Wenn Taster gedrückt, dann ...wenn der Wert der Variablen status „aus“ ist, wird er auf „an“ gesetzt und die LED eingeschaltet ...sonst (wenn der Wert der Variablen status „an“ ist) wird er auf „aus“ gesetzt und die LED ausgeschaltet</p> <p>Themen:</p> <ul style="list-style-type: none"> - Kontrollblock „forever-if“ - Variablen - Vergleiche <p>Weiterführendes Problem: Was passiert, wenn man den Taster gedrückt hält? Wie kann man das Problem beheben?</p>
	<p>Solange das Programm läuft gilt: Wenn Taster gedrückt, dann ...wenn der Wert der Variablen status „aus“ ist, wird er auf „an“ gesetzt, die LED eingeschaltet und 0.4 Sekunden gewartet ...sonst (wenn der Wert der Variablen status „an“ ist), wird er auf „aus“ gesetzt, die LED ausgeschaltet und 0.4 Sekunden gewartet</p> <p>Themen:</p> <ul style="list-style-type: none"> - Kontrollblock „wait“ <p>Erstes, richtig sinnvolles Programm ist fertig. Nun wollen wir es speichern, um in Zukunft darauf zugreifen zu können.</p> <p>Weiterführendes Problem: Ich hätte gern automatische Beleuchtung, sobald es dunkel ist.</p>

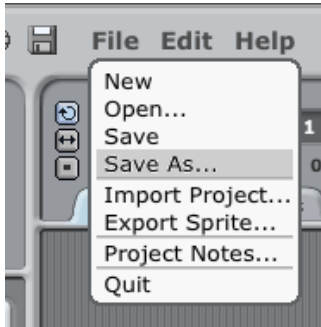


- Datei – speichern
- Diskettensymbol

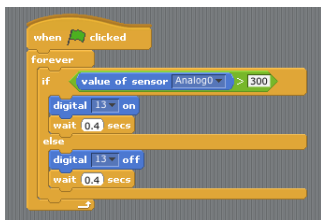
Themen:

- zwischenspeichern

-> Erinnerung an Wichtigkeit des Speicherns



Bei wichtigen Zwischenständen ruhig auch als neues Projekt speichern, um später zurück zu können (Es gibt in S4A kein UNDO!)

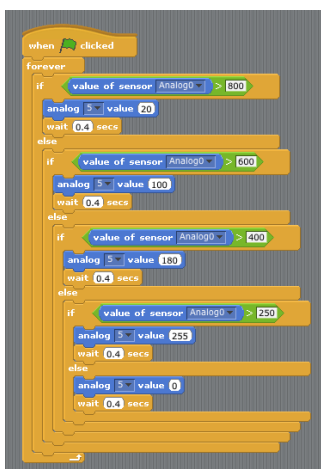


Solange das Programm läuft gilt: Wenn der Wert des Sensors am Eingang analog0 300 überschreitet, wird die LED angeschaltet und 0.4 Sekunden gewartet, sonst wird die LED ausgeschaltet und 0.4 Sekunden gewartet

Themen:

- Schwellwert ermitteln
- LDR: ein analoger Sensor

Weiterführendes Problem: die Helligkeit der LED sollte sich an die Umgebungshelligkeit anpassen. Wenn es im Garten schon sehr dunkel ist, kann das Licht der LED ruhig gedimmt sein, während es bei heller Umgebung noch stärker leuchten sollte, um es überhaupt wahrnehmen zu können.



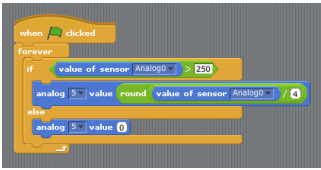
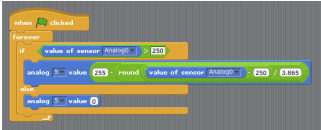
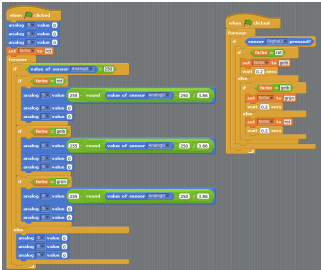
Solange das Programm läuft gilt:

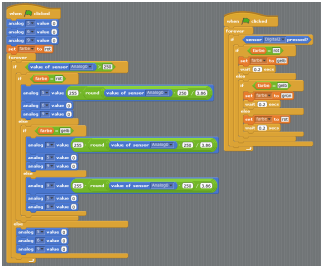
- wenn der Wert des Sensors am Eingang analog0 800 überschreitet, wird die LED sehr gedimmt (Wert: 20) angeschaltet und 0.4 Sekunden gewartet, sonst:
- wenn der Wert des Sensors am Eingang analog0 600 überschreitet, wird die LED gedimmt (Wert 100) angeschaltet und 0.4 Sekunden gewartet, sonst:
- wenn der Wert des Sensors am Eingang analog0 400 überschreitet, wird die LED leicht gedimmt (Wert 180) angeschaltet und 0.4 Sekunden gewartet, sonst:
- wenn der Wert des Sensors am Eingang analog0 250 überschreitet, wird die LED bei maximaler Helligkeit angeschaltet und 0.4 Sekunden gewartet, sonst wird die LED ausgeschaltet und 0.4 Sekunden gewartet

Themen:

- Verschachtelung von Anweisungen

Weiterführendes Problem: die Helligkeit regelt sich in stark sichtbaren Stufen. Ich würde mir eine Lampe wünsche, die sich stufenlos an die Umgebungshelligkeit anpasst. Geht das?

	<p>Solange das Programm läuft, gilt: Wenn der Wert des Sensors am Eingang analog0 250 überschreitet, wird die LED entsprechend der Berechnung gedimmt eingeschaltet.</p> <p>Themen:</p> <ul style="list-style-type: none"> - arithmetische Ausdrücke <p>Weiterführendes Problem: je höher der analoge Eingangswert, desto höher auch der Ausgangswert, desto heller also die LED. Wir wollen jedoch den umgekehrten Effekt haben. Wie geht das?</p>
	<p>Solange das Programm läuft, gilt: Wenn der Wert des Sensors am Eingang analog0 250 überschreitet, wird die LED entsprechend der Berechnung $\text{round}(255 - ((\text{analog0} - 250) : 3,86))$ gedimmt eingeschaltet.</p> <p>Berechnung: Die Ausgangswerte sollen ab einem Eingangswert von über 250 gleichmäßig verteilt werden und zwischen 55 und 255 liegen. Es gibt also 200 Werte, die sich auf 773 Eingangssignale verteilen, das heißt ein Eingangswertunterschied von $(773/200) = 3,86$ entspricht einem Ausgangswertunterschied von 1. 255 ist der Maximalwert, der ausgegeben werden kann, dies geschieht beim Eingangssignal von 251.</p> <p>Themen:</p> <ul style="list-style-type: none"> - verschachtelte arithmetische Ausdrücke <p>Weiterführendes Problem: Wie müssen das Programm und die Lampenkonstruktion geändert werden, damit sich auf Knopfdruck die Farbe des Lichts ändert? (Stimmungsleuchten, glühende Hecke, Partylichter...)</p>
	<p>LEDs zusätzlich an analog6 und analog9 anschließen.</p> <p>Wenn die grüne Flagge angeklickt wurde, werden die analogen Ausgangswerte genullt. Die Variable farbe bekommt einen Anfangswert (rot) zugewiesen.</p> <p>Solange das Programm läuft, gilt: Wenn der Wert des Sensors am Eingang analog0 250 überschreitet, wird die LED entsprechend der in der Variablen Farbe gespeicherten Farbe und entsprechend der Berechnung gedimmt eingeschaltet.</p> <p>Wenn der Taster am Eingang digital2 gedrückt wird, wird die Variable farbe mit einem neuen Wert (rot – gelb – grün) belegt und 0.2 Sekunden gewartet.</p> <p>Themen:</p> <ul style="list-style-type: none"> - komplexes Projekt mit Schleifen, Entscheidungen, Variablen, verschachtelten arithmetischen Ausdrücken,

	<p>mehreren Sensoren und mehreren Aktoren</p> <ul style="list-style-type: none"> - parallele Abarbeitung mehrerer Skripte (single stepping zeigt: in Realität nacheinander) <p>Weiterführendes Problem: wenn man „single stepping“ anschaltet, wird deutlich, dass das Projekt noch einige unnötige Abarbeitungsschritte enthält. So wird beispielsweise in einem Schleifendurchlauf dreimal der Wert der Variablen abgefragt und verglichen, auch wenn schon der erste Vergleich erfolgreich war. Wie kann man es optimieren?</p>
	<p>Wenn die grüne Flagge angeklickt wurde, werden die analogen Ausgangswerte genullt. Die Variable farbe bekommt einen Anfangswert (rot) zugewiesen. Solange das Programm läuft, gilt: Wenn der Wert des Sensors am Eingang analog0 250 überschreitet, wird die LED entsprechend der in der Variablen farbe gespeicherten Farbe und entsprechend der Berechnung gedimmt eingeschaltet.</p> <p>Wenn der Taster am Eingang digital2 gedrückt wird, wird die Variable farbe mit einem neuen Wert (rot – gelb – grün) belegt und 0.2 Sekunden gewartet.</p> <p>Die Vergleiche der Variablen mit den möglichen Farben erfolgen nun nicht mehr nacheinander mit „if“-, sondern ineinander geschachtelt mit „if-else“-Blöcken, so dass nicht jedes Mal alle Vergleiche durchgeführt werden müssen.</p> <p>Themen:</p> <ul style="list-style-type: none"> - Optimierung (Effizienz: Laufzeit) <p>Weiterführendes Problem: Bisher ist die Lampensteuerung nicht besonders stromsparend, denn sobald es dunkel wird, geht das Licht an. Dies soll aber nur der Fall sein, wenn ich zu Hause bin. Wie lässt sich das Problem lösen?</p>
<p>...</p>	<p>Idee 1: Wenn Tür auf, dann Licht an. Idee 2: Globaler Ein-/Aus-Schalter.</p> <p>...</p> <p>Weiterführendes Problem: Ich habe im ganzen Garten Beleuchtung verbaut. Um aber nicht permanent alle Winkel zu beleuchten und somit sinnlos Energie zu verschwenden, soll jeweils nur der Teil des Gartens beleuchtet sein, in dem ich mich gerade befinde. Wie kann das realisiert werden?</p>
<p>...</p>	<p>...</p> <p>Weiterführendes Problem: Mehrere Bewohner...</p>