DBL embedded systems group 16

Made by:

Wigger Boelens

Rolf Verschuuren

Stefan van den Berg

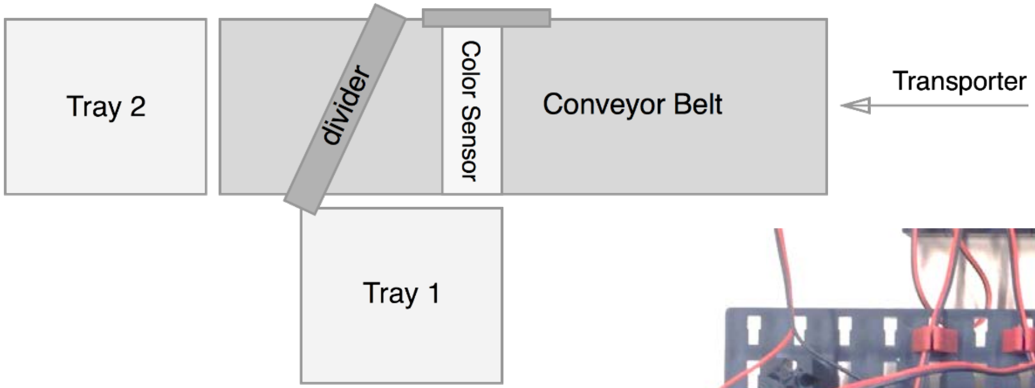
TU/e Technische Universiteit Eindhoven University of Technology

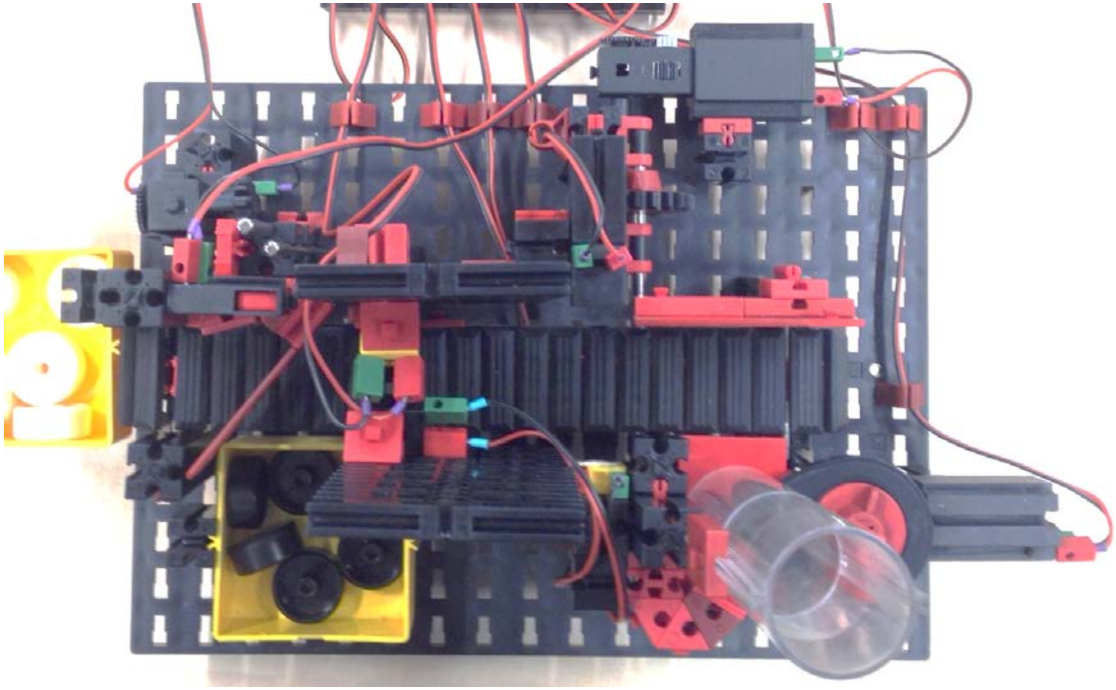
Where innovation starts

Machine Design

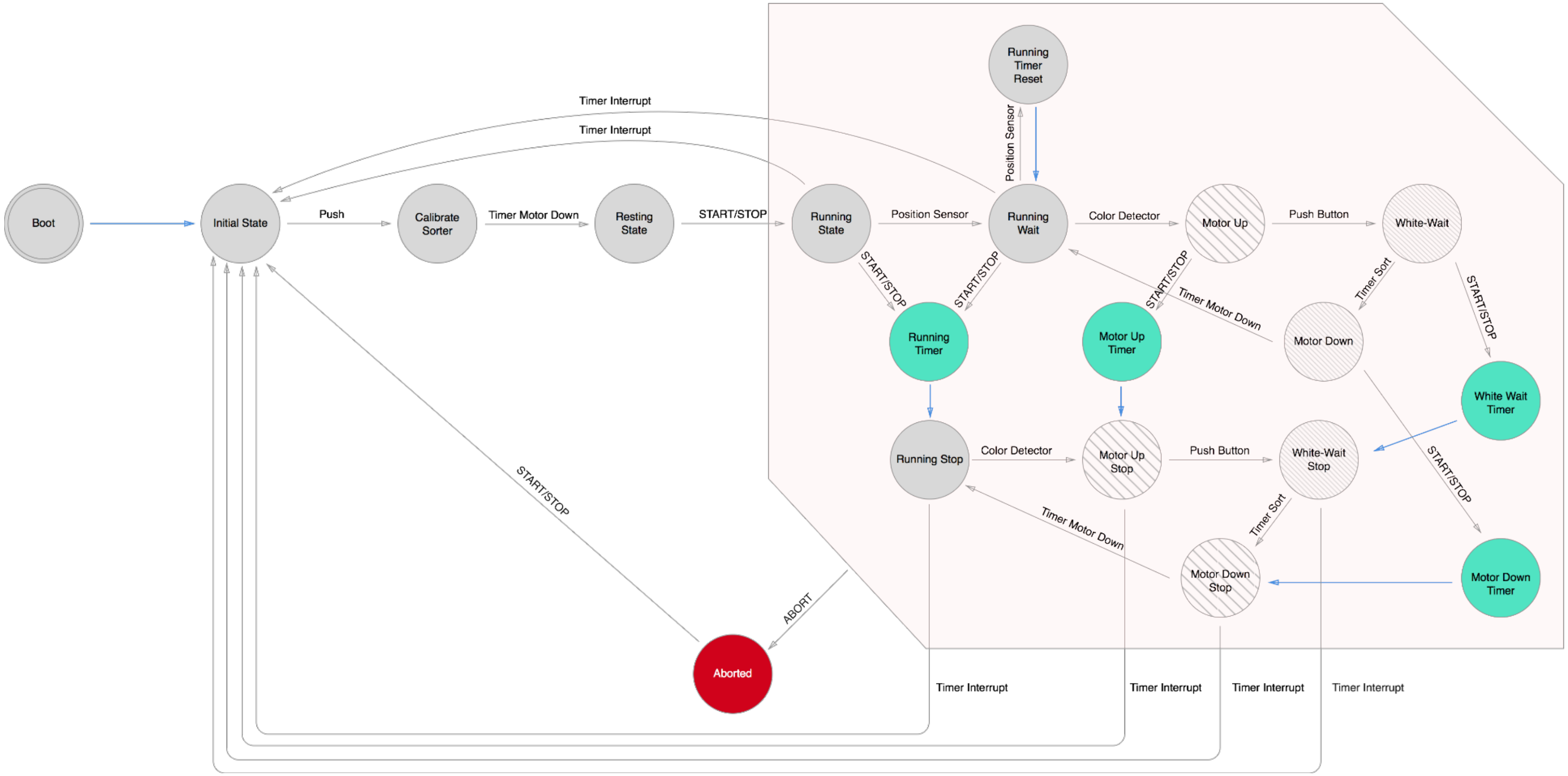
- Reliability
- Speed
- Robustness
- Accessibility
- Floor space
- Complexity
- Amount of parts

Machine Design

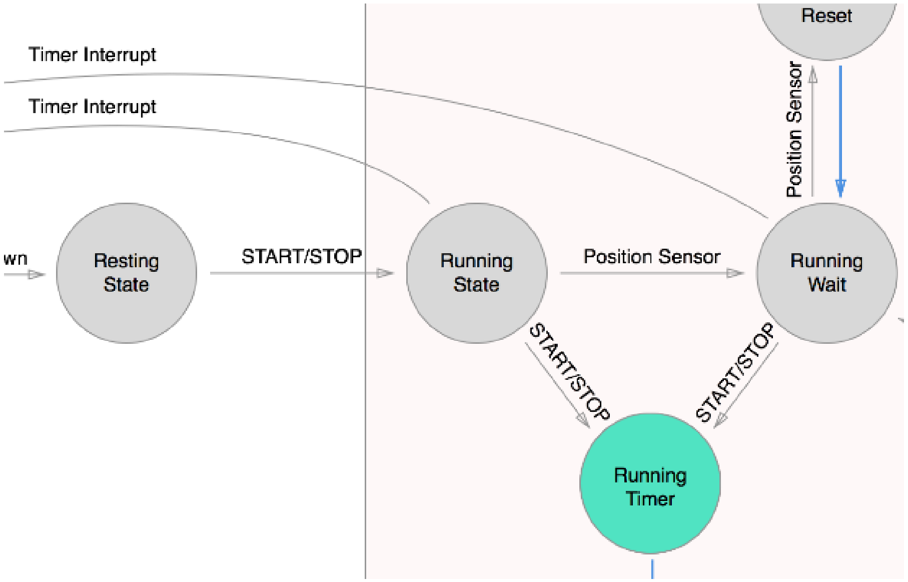


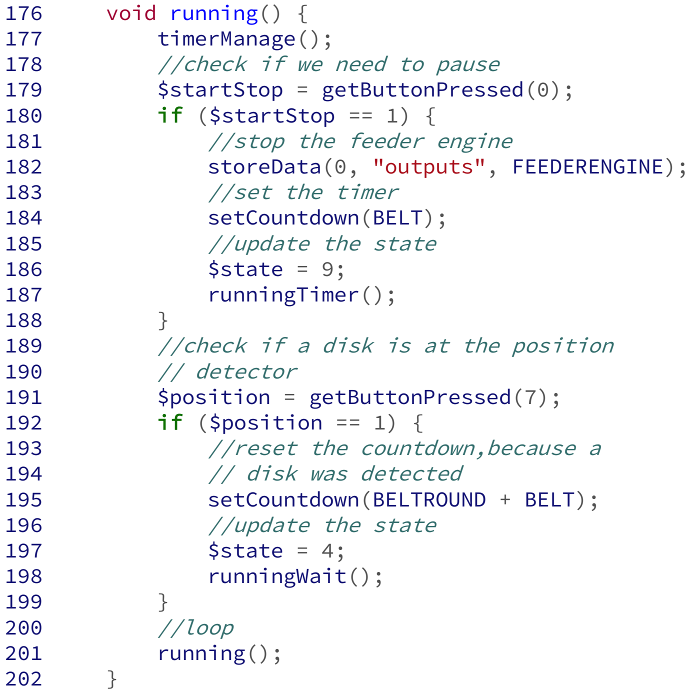


Software Design



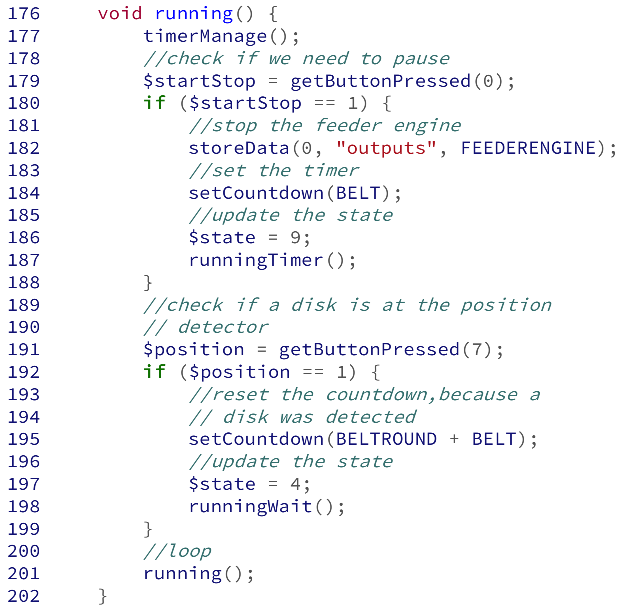
Software Design



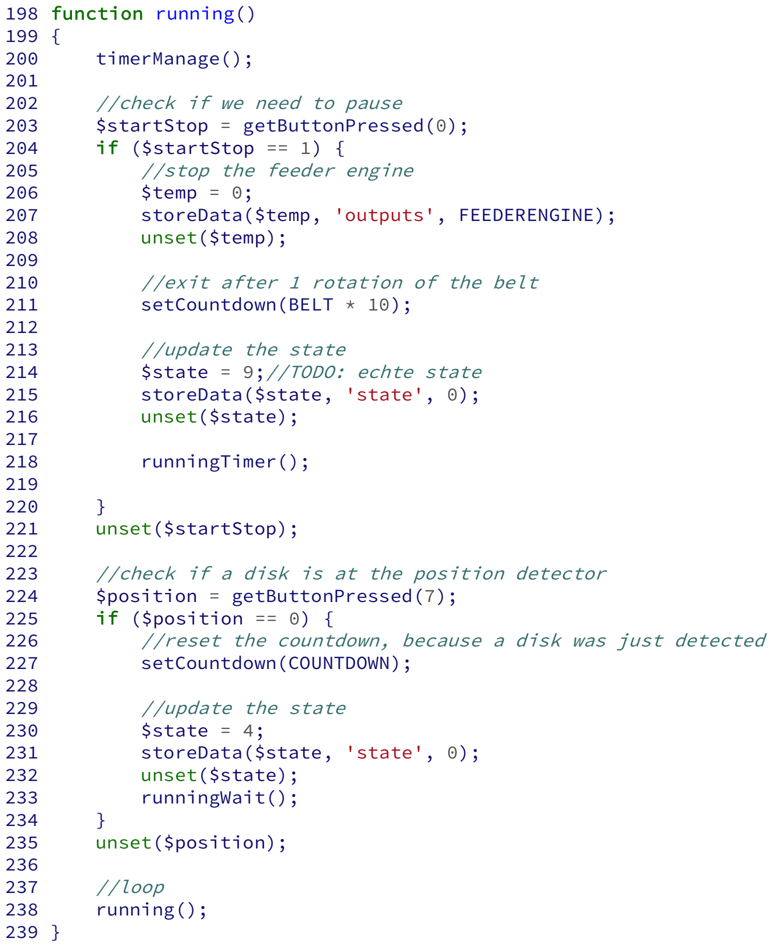


```
void running() {
    timerManage();
    //check if we need to pause
    $startStop = getButtonPressed(0);
    if ($startStop == 1) {
        //stop the feeder engine
        storeData(0, "outputs", FEEDERENGINE);
        //set the timer
        setCountdown(BELT);
        //update the state
        $state = 9;
        runningTimer();
    }
    //check if a disk is at the position
    // detector
    $position = getButtonPressed(7);
    if ($position == 1) {
        //reset the countdown,because a
        // disk was detected
        setCountdown(BELTROUND + BELT);
        //update the state
        $state = 4;
        runningWait();
    }
    //loop
    running();
}
```

Software Design



```
void running() {
    timerManage();
    //check if we need to pause
    $startStop = getButtonPressed(0);
    if ($startStop == 1) {
        //stop the feeder engine
        storeData(0, "outputs", FEEDERENGINE);
        //set the timer
        setCountdown(BELT);
        //update the state
        $state = 9;
        runningTimer();
    }
    //check if a disk is at the position
    // detector
    $position = getButtonPressed(7);
    if ($position == 1) {
        //reset the countdown,because a
        // disk was detected
        setCountdown(BELTROUND + BELT);
        //update the state
        $state = 4;
        runningWait();
    }
    //loop
    running();
}
```



```
function running()
{
    timerManage();

    //check if we need to pause
    $startStop = getButtonPressed(0);
    if ($startStop == 1) {
        //stop the feeder engine
        $temp = 0;
        storeData($temp, 'outputs', FEEDERENGINE);
        unset($temp);

        //exit after 1 rotation of the belt
        setCountdown(BELT * 10);

        //update the state
        $state = 9;//TODO: echte state
        storeData($state, 'state', 0);
        unset($state);

        runningTimer();

    }
    unset($startStop);

    //check if a disk is at the position detector
    $position = getButtonPressed(7);
    if ($position == 0) {
        //reset the countdown, because a disk was just detected
        setCountdown(COUNTDOWN);

        //update the state
        $state = 4;
        storeData($state, 'state', 0);
        unset($state);
        runningWait();
    }
    unset($position);

    //loop
    running();
}
```

Software Design

PHP	Assembly
197 //state 3	
198 function running()	
199 {	
200 timerManage();	265 running: BRS timerManage
201	
202 //check if we need to pause	
203 $startStop = getButtonPressed(0);	266 PUSH R3
	267 LOAD R3 0
	268 BRS _pressed
	269 PULL R3
	270 SUB SP 5
	271 PULL R3
	272 ADD SP 4
204 if ($startStop == 1) {	273 CMP R3 1
	274 BEQ conditional3
205 //stop the feeder engine	
206 $temp = 0;	289 conditional3: LOAD R4 0
207 storeData($temp, 'outputs', FEEDERENGINE);	290 STOR R4 [GB +outputs + FEEDERENGINE]
208 unset($temp);	291
209	
210 //exit after 1 rotation of the belt	
	292 PUSH R5 ;reset timer
	293 PUSH R4
	294 LOAD R5 -16
	295 LOAD R4 0
	296 SUB R4 [R5+13]
	297 STOR R4 [R5+13] ;set timer
	298 LOAD R4 BELT * 10
	299 STOR R4 [R5+13]
	300 PULL R4
211 setCountdown(BELT * 10);	301 PULL R5
212	
213 //update the state	
214 $state = 9;//TODO: echte state	302 LOAD R4 9 ;$state = 9
215 storeData($state, 'state', 0);	303 STOR R4 [GB +state + 0]
216 unset($state);	304
217	
218 runningTimer();	305 BRA runningTimer
219	
220 }	
221 unset($startStop);	275 return3:
…	…

Machine Design

- Reliability
- Speed
- Robustness
- Accessibility
- Floor space
- Complexity
- Amount of parts

Summary