

# ESP32-deurbel via Telegram

“de postbode belt maar één keer”



**Luc Lemmens** (Elektor Labs)

Het overkomt ons allemaal wel eens: u bent wel thuis als er iemand aan de deur staat, maar toevallig hoort u de bel niet! Dat is geen probleem als degene die voor de deur staat uw telefoonnummer weet, dan kan hij u altijd nog bellen. Zo niet, dan kan het heel vervelend worden; misschien zat u wel met smart te wachten op een pakketje met spullen die u meteen nodig hebt. Maar hier is een project om aan deze ellende een eind te maken.

Bezorgers en postbodes hebben vaak geen tijd om even te wachten tot u naar de voordeur komt. Voor u het weet

zijn ze alweer vertrokken met achterlating van een hatelijk briefje dat u “niet thuis” was en een mededeling dat ze

nog wel een keer langs komen of dat u het pakketje zelf kunt gaan ophalen bij een afhaalpunt.

PROJECT-INFO

ESP32

Arduino IoT

M5Stack

beginners

→ gevorderden

experts

ca. 3 uur

Arduino IDE, labgereedschap

ca. € 85

## Telegram voor u!

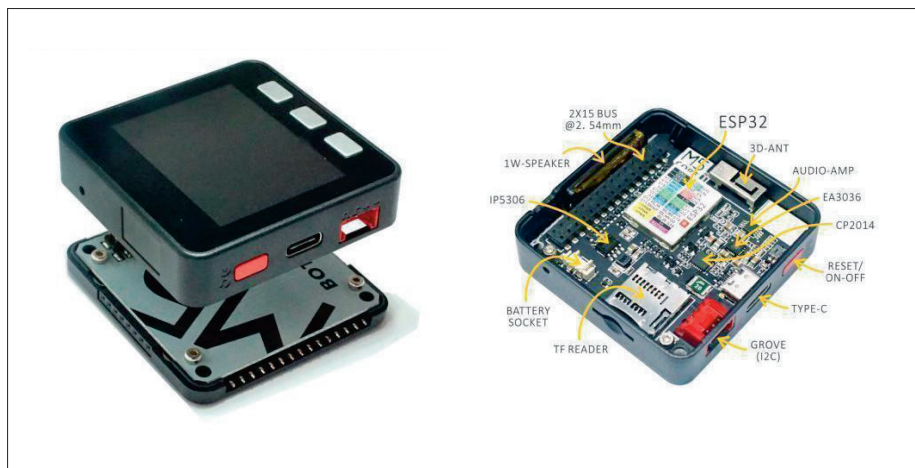
Zodra de bel gaat, stuurt deze schakeling een berichtje naar uw smartphone of computer waar u de app *Telegram* op draait. Telegram is, volgens de makers, "een cloud-based messaging-app die geoptimaliseerd is voor veiligheid en snelheid." Om precies te zijn, de volledige naam is *Telegram Messenger*. Net als veel andere volledig internet-gebaseerde communicatietools, draait Telegram op verschillende platformen. Er zijn mobiele apps voor

- iPhone/iPad;
- Android;
- Windows Phone.

En er zijn desktop-apps voor

- PC/Mac/Linux;
- MacOS;
- webbrowser.

Telegram is een instant messaging-dienst die qua functionaliteit veel lijkt op WhatsApp. Beide diensten zijn gekoppeld aan een mobiel telefoonnummer. Daarnaast maakt Telegram het mogelijk om een gebruikersnaam te registreren, zodat u uw account mee kunt nemen als u een nieuw mobiel nummer krijgt. Er zijn meer grote en kleine verschillen tussen de twee diensten en het zal niemand verbazen dat het internet vol



Figuur 1. De M5Stack ESP32 Basic Core-module uit de Elektor-shop.

is van vergelijkingen, goede en slechte meningen, fans en haters, discussie en debat.

Als u nieuwsgierig bent geworden, bekijk dan nu eerst eens de tamelijk kale website van Telegram met zijn monochrome cartoons [1].

Terug naar onze deurbel. Als uw Telegram-app actief is en is verbonden met de Telegram-server (via WiFi of een mobiel netwerk), dan laat hij u weten dat er iemand aan de deur is – waar u ook bent. Als u heel zeker weet wie het is, en die persoon volkomen vertrouwt, kunt u zelfs een 'bericht' terug sturen waarmee een elektrische deur opener wordt aangestuurd, zodat u niet eens naar de voordeur of naar het trapportaal hoeft te rennen om de deur handmatig te openen.

## Telegram en ESP32 – met een schoonheidsfoutje

We zullen nog zien dat Telegram gemakkelijk geïnstalleerd kan worden en verbonden kan worden met een 'bot' voor automatische berichtverwerking. We willen graag communicatie op afstand en besturing met een ESP32. Om precies te zijn willen we een bericht sturen als er op een knop wordt gedrukt en een bericht (eventueel) vertalen naar het activeren van een relais. Natuurlijk zijn er veel meer mogelijkheden dan deze simpele toepassing, het idee kan gemakkelijk worden uitgebreid en aangepast aan eigen ideeën.

De ESP32 (en zijn voorganger, de ESP8266), worden nog steeds vaak gebruikt in Elektor-projecten en -boeken. We gebruiken de ESP32 hier als een slimme module die met de

buitenwereld communiceert via WiFi. In combinatie met de bekende Arduino-ontwikkelomgeving en de bot-library *UniversalTelegram* zou het ontwikkelen van 'eenvoudige' toepassingen zoals onze deurbel geen probleem mogen zijn. Maar dat bleek tegen te vallen, want bij het werken aan dit project werden we geconfronteerd met een wel heel verrassend probleem dat ons een hoop tijd en moeite kostte om op te lossen, zie het **kader** 'Close Encounters'.

## Hardware: M5Stack ESP32 Basic Core

Met wat kleine aanpassingen in de software kan dit project worden gebruikt met elke ESP32- of ESP8266-module. Wij hebben gekozen voor de mooie *M5Stack ESP32 Basic Core*-module, die verkrijgbaar is in de Elektor-shop en te zien is in **figuur 1**. Het is een compacte module met een fraaie behuizing, plus:

- een grafisch display;
- een aantal drukknoppen;
- een luidspreker;
- een ingebouwde accu.

De I/O-lijnen van de M5Stack ESP32 Basic Core zijn beschikbaar op connectoren aan de buitenkant. Maar als u de achterkant van het kastje open maakt en de insteekkaart verwijdert, vindt u een 30-polige uitbreidingsconnector voor uw eigen hardware.

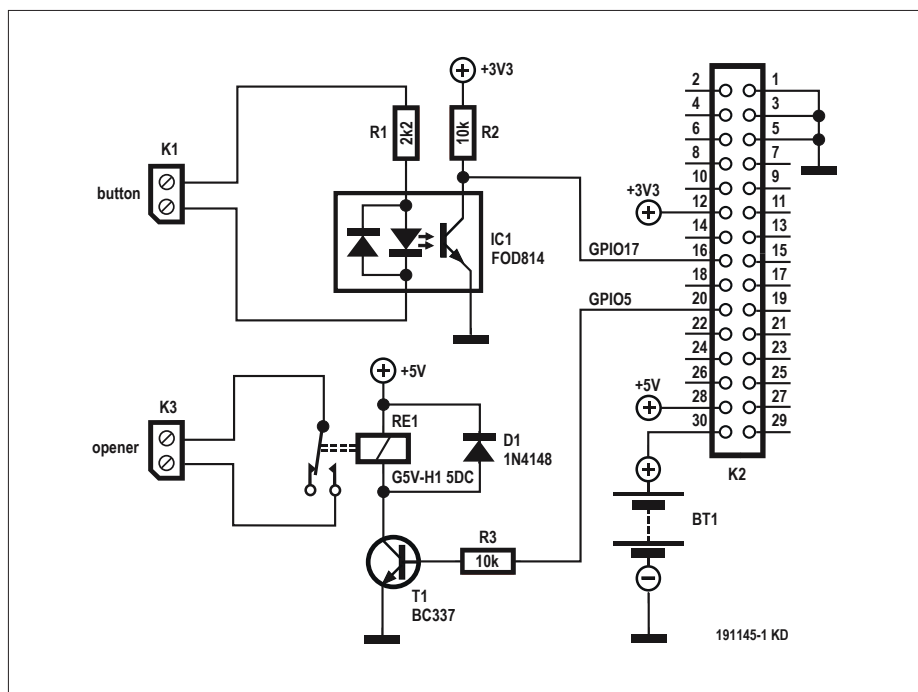
Voor een klein project als dit past de extra hardware op een print met dezelfde afmetingen als de M5Stack-module zelf. Met een aangepaste behuizing, waarvoor een 3D-ontwerp van het web [2] kan worden gedownload, ontstaat een keurig

apparaatje dat niet hoeft te misstaan in de huiskamer. Om kort te gaan: deze M5Stack-module vormt een prima basis voor veel ESP32-apparaten, van ontwikkelplatforms tot consumententoepassingen (*start-ups, hoort u ons?*). In principe kan de M5Stack met onze uitbreidingskaart een complete deurbel-installatie vormen. De ingebouwde luidspreker kan de concurrentie met een ‘echte’ gemakkelijk aan, en er kan een aparte beldrukker parallel aan of in plaats van de transistor van de optocoupler worden aangesloten. Het belsignaal moet dan worden toegevoegd aan de Arduino-sketch, en het logische niveau op GPIO17 moet worden geïnverteerd als de knop op die manier is aangesloten. Maar dat gaan we hier verder niet behandelen, wij beperken ons tot een uitbreiding die met een bestaande installatie wordt verbonden.

### Het schema

In **figuur 2** ziet u het schema van onze ‘Telegram’-deurbel; simpeler kan niet! We gaan ervan uit dat de M5Stack-module extern wordt gevoed via USB, zodat de extra hardware alleen zorgt voor het aansturen van de deuropener via T1 en RE1 en voor de koppeling met de beldrukker via optocoupler IC1. Onze deurbeluitbreiding kan dus rechtstreeks worden aangesloten in een bestaand systeem; de twee draden van de beldrukker worden verbonden met K1 in onze schakeling. Een elektrische deurbel wordt meestal gevoed met wisselstroom, maar de spanning kan van installatie tot installatie verschillen, en kan variëren van 6 V tot 24 V – en er zijn ongetwijfeld ook exotische installaties die zelfs daar nog van afwijken. Zolang er niet op de knop wordt gedrukt, staat de wisselspanning over de beldrukker en dus ook op K1. De optocoupler vertaalt de aanwezigheid van die spanning naar een logisch 0-niveau op GPIO17 van de ESP32, dat is dus de rustsituatie; als er wordt aangebeld, gaat het niveau naar “1”. Verder is er de aansluiting BT1 om eventueel de lithium-accu, die standaard in de M5Stack-module zit, (weer) aan te sluiten. Let op de polariteit: de rode draad gaat naar de positieve aansluiting en de zwarte naar de negatieve.

De uitbreidingskaart bevat alleen bedrade componenten en is gemakkelijk op te bouwen. Let wel op K2, want die moet op een wat ongewone manier geplaatst worden (**figuur 3**). Normaal gesproken

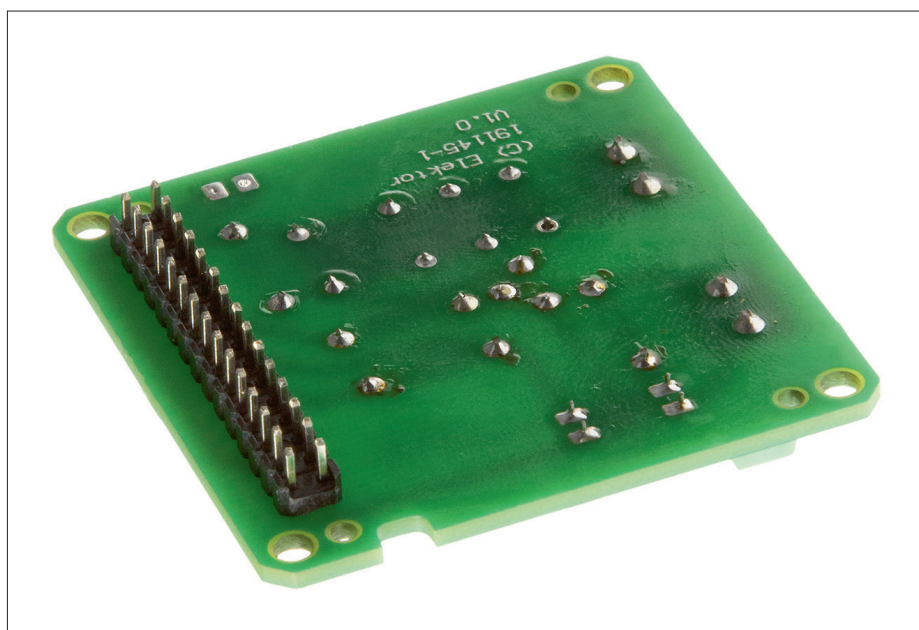


Figuur 2. Schema van de deurbelverlener met de krachtige combinatie van Arduino, ESP32 en de gratis online messaging-app ‘Telegram’.

steken we de korte kant van de pennen in de print vanaf de componentenzijde en solderen ze dan vast. Maar hier doen we het andersom: we steken de lange pennen in de print vanaf de koperzijde en solderen ze vast aan de componentenzijde. De korte pennen gaan in de voetjes op de M5Stack-module en het plastic gedeelte van K2 zorgt voor voldoende afstand tussen de twee printen.

### Telegram installeren

De volgende stap is om met de mobiele telefoon een account aan te maken op Telegram. Download eerste de Telegram-app uit de App Store of van Google Play. Het installeren is een fluitje van een cent. Het account wordt gekoppeld aan het mobiele nummer van die telefoon. Daarna is het verstandig om de Telegram-app ook te installeren op



Figuur 3. Aanbevolen manier om K2 te monteren: de **lange** pennen worden vanaf de koperzijde (de ‘onderkant’ van de print) door de gaten gestoken en aan de componentenzijde gesoldeerd.



## Close Encounters of The ESP32 Kind

Stelt u zich eens voor: uw zorgvuldig gecompileerde broncode compileert foutloos, wordt overgedragen naar de ESP32, en alles lijkt goed te werken. Zo lijkt het. Maar dan blijkt dat de microcontroller telkens wordt gereset als er een Telegram-bericht binnenkomt! We controleren alles nog eens, maar er is geen fout te vinden. Dan maar eens compileren en testen met een andere computer, en dan blijkt alles opeens vlekkeloos te werken! Beide computers hebben dezelfde versie van de Arduino-IDE en de bibliotheken; geen verschil te vinden. Dan maar eens de hele Arduino-IDE verwijderen van de eerste PC. Dat is geen gewone uninstall, er blijft in ieder geval van alles achter in verborgen mappen. Arduino opnieuw geïnstalleerd en alle problemen zijn verdwenen.

Wat was er nu precies aan de hand? We weten het nog steeds niet, en we hebben geen zin meer om het uit te zoeken. We hebben geleerd dat de Arduino-IDE zich `_soms_` vreemd gedraagt. Dit was de eerste keer dat dit gebeurde bij Elektor Labs en hopelijk ook de laatste!

uw PC of Mac; die gaat anders werken als de verbinding tussen Telegram en de ESP32-sketch is gemaakt.

Log met de computer in op Telegram met het mobiele nummer. De app op de telefoon laat dan een verificatiecode zien, die op de computer moet worden ingetypt, daarna is Telegram op de PC geactiveerd. De sketch op de ESP32 werkt met een 'bot', te vergelijken met een soort automatisch antwoordapparaat. Hij kan reageren op commando's die binnenkomen als een Telegram-chatbericht en hij kan berichten versturen via Telegram als de ESP iets te melden heeft, bijvoorbeeld het indrukken van de belknop of de waarde op

een ADC-ingang. In dit project besturen de commando's een deuropener. De bot moet eerst worden aangemaakt in Telegram. Doe dat door eerst de 'BotFather' te zoeken in de contactenlijst. BotFather is zelf ook een bot die u helpt om uw deurbel-bot aan te maken via een chat-dialoog. Na invoeren van het commando `/newbot` begint de dialoog, U moet nu eerst een naam invoeren (die komt straks in de contactenlijst in Telegram) en een gebruikersnaam voor de bot. Die gebruikersnaam moet altijd eindigen met de letters 'bot'. Let op de slash ( / ) aan het begin van elk bot-commando. Als het aanmaken is gelukt, stuurt de BotFather u een code

('token'), waarmee u de nieuwe bot kunt autoriseren om requests naar de Bot-API te sturen. Bewaar dit token op een veilige plaats, want iedereen kan het gebruiken om uw bot aan te sturen. We hebben het token dus ook onleesbaar gemaakt in onze eigen dialoog in **figuur 4**.

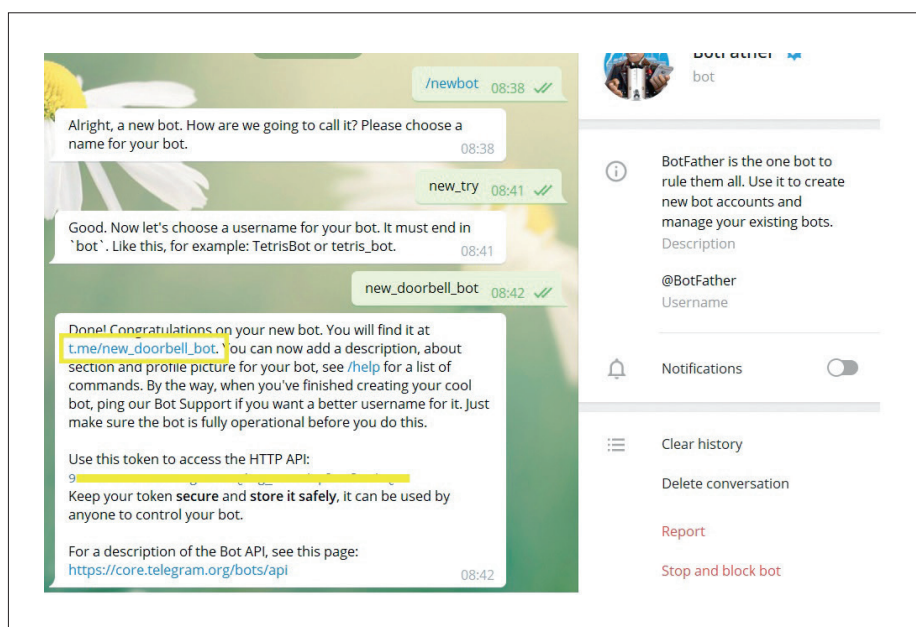
Klik in de dialoog met BotFather, op de omkaderde link ('t.me/...') om een chatvenster te openen voor de nieuwe bot, die wordt toegevoegd aan uw contactenlijst. De bot is nu klaar voor gebruik, maar we moeten hem nog wel koppelen met de Arduino-sketch voor onze deurbelverlenger.

### Arduino-sketch

In de software-download voor dit project staat de sketch [DOORBELL.INO](#), dat is de broncode voor de ESP32-firmware [3]. U kunt de sketch naar wens aanpassen; in de huidige vorm demonstreert die hoe Arduino/ESP32 en Telegram kunnen samenwerken. Vooral het aansturen van de deuropener is een veiligheidsrisico: iemand zou de bot kunnen hacken en uw deur openen met niet meer dan een smartphone of een computer. Elektor neemt geen enkele verantwoordelijkheid voor schade die u kunt oplopen door het gebruik van dit project!

In de sketch moet onze M5Stack-module zich aanmelden op uw WiFi-netwerk en verbinding maken met de Telegram-bot die u hebt gemaakt. Voeg de inloggegevens voor uw WiFi-netwerk in op de regels 17 en 18, en het token dat u hebt gekregen van BotFather in regel 21. Daarom raden we aan om de bot aan te maken op uw computer: dan kunt u het token kopiëren en in de sketch plakken. Het is niet bepaald iets dat gemakkelijk overgetypt kan worden.

Als de Arduino-IDE goed is geïnstalleerd, compileert de sketch zonder problemen. Verbind de M5Stack met een USB-poort van de computer en selecteer in het menu Tools het juiste board (M5Stack-Core-ESP32) en de virtuele COM-poort om de sketch te uploaden naar de ESP32. De sketch zal dan opstarten, verbinding maken met het netwerk en het IP-adres van de deurbelverlenger verschijnt op het LCD van de M5Stack. De bot moet elke keer na het opstarten van de sketch worden geactiveerd door het commando `/start` te geven via het chatvenster in Telegram. De bot moet dan antwoord geven met een overzicht van de beschikbare commando's.



Figuur 4. Screenshot van onze dialoog met de BotFather. Het token is natuurlijk onleesbaar gemaakt.



## ONDERDELENLIJST

### Weerstanden:

R1 = 2k2, 5%, 0,25W, 250V  
R2,R3 = 10 k, 5%, 0,25W, 250V

### Halfgeleiders:

D1 = 1N4148  
T1 = BC337  
IC1 = FOD814A, AC-optocoupler met transistor-uitgang, 1 kanaal, DIP, 4 pennen, 50mA, 5kV

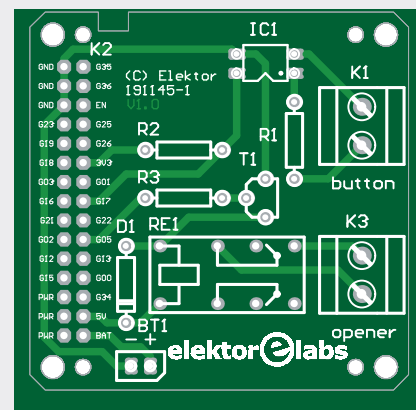
### Diversen:

RE1 = relais G5V-2-H1 5DC, 5VDC, DPDT, 1A (Omron)

K1,K3 = 2-polige printkroonsteen, steek 0,2" (5,08 mm), 630V

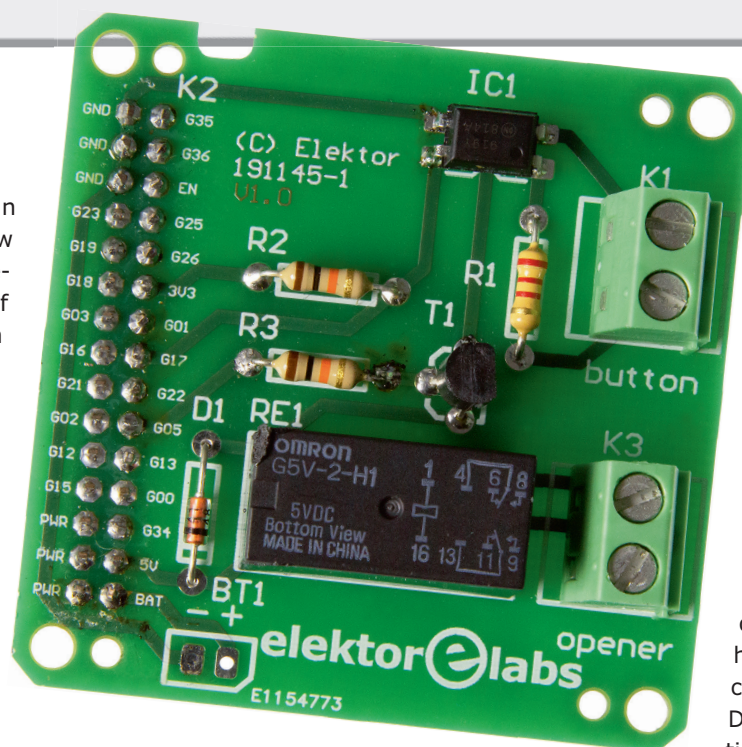
K5 = 30-pens (2x15) pinheader, recht  
M5Stack ESP32 Basic Core (Elektor-shop, of  
equivalente ESP32-module)

BT1 = optionele accu met houder  
Print 191145-1 v1.0 (Elektor-shop)



### Dagelijks gebruik

Als de deurbel gaat, krijgt u een berichtje via de Telegram-app op uw computer of smartphone. U kunt reageren door zelf de deur te openen, of door de /Open-button te gebruiken om de deur opener te activeren (als u die hebt). Wilt u dat niet doen, dan gebruikt u de /Ignore-button, en verschijnt opnieuw de /status-button (die de verbinding met de deurbelverlenger controleert). U kunt die commando's ook zelf intypen in het chatvenster; let op het verschil tussen hoofd- en kleine letters! De bibliotheek *UniversalTelegram bot* doet het meeste werk in dit project. Deze library en de documenta-



tie zijn te vinden op [4]; daar vindt u ook voorbeelden waarmee u gemakkelijk zelf aan de slag kunt gaan. Onze deurbel-sketch is gebaseerd op het voorbeeld 'custom keyboard', dat laat zien hoe u het indrukken van besturingsknoppen kunt doorsturen naar de Telegram-app, zo hoeft u niet zelf allerlei commando's in te typen. De belangrijkste subrou-tine, die het verwerken van binnenkomende

berichten afhandelt, is [handleNewMessages](#). In vier **If**-statements worden de commando's en berichten die onze bot kent afgehandeld:

- start;
- status;
- Ignore;
- Open.

Om berichten te kunnen versturen, heeft de sketch de chat-ID nodig. Die wordt bewaard in de variabele `ThisChat` na het ontvangen van het eerste '/start'-commando. Het indrukken van de beldrucker wordt verwerkt in een interrupt; let op het attribuut `IRAM_ATTR`, dat is noodzakelijk voor de declaratie van een ESP-interruptroutine. ◀

(191145-04)



## IN DE STORE

→ M5Stack ESP32 Basic Core

[www.elektor.nl/m5stack-esp32-basic-core-development-kit](http://www.elektor.nl/m5stack-esp32-basic-core-development-kit)

→ Print 191145-1 V1.0

[www.elektor.nl/191145-1](http://www.elektor.nl/191145-1)

### Weblinks

- [1] Telegram Messenger: <https://telegram.org/>
- [2] 3D printer files for M5Stack behuizing: <https://github.com/m4k3r-net/M5Stack-3DPrintFiles>
- [3] Arduino sketch: [www.elektormagazine.nl/191145-04](http://www.elektormagazine.nl/191145-04)
- [4] UniversalTelegram bot library: <https://Github.com/witnessmenow/Universal-Arduino-Telegram-Bot>